if all registers preserved, restart DOS call
---DOS 2+---
CF clear
Return: all registers preserved
return via RETF with CF set or (MS-DOS 1,DR DOS) RETF 2 with CF set
DOS will abort program with errorlevel 0
else (RETF/RETF 2 with CF clear or IRET with CF ignored)
interrupted DOS call is restarted
Notes: this interrupt is invoked whenever DOS detects a ^C or ^Break; it
should never be called directly
MS-DOS 1.25 also invokes INT 23 on a divide overflow (INT 00)
MS-DOS remembers the stack pointer before calling INT 23, and if it is
not the same on return, pops and discards the top word; this is what
permits a return with RETF as well as IRET or RETF 2
MS-DOS 2.1+ ignores the returned CF if SP is the same on return as it
was when DOS called INT 23, so RETF 2 will not terminate the program
Novell DOS 7 always pops a word if CF is set on return, so one should
not return with RETF 2 and CF set or IRET with the stored flags' CF
set
any DOS call may safely be made within the INT 23 handler, although
the handler must check for a recursive invocation if it does
call DOS
SeeAlso: INT 1B,INT 21/AH=92h"PTS-DOS"
--------D-27---------------------------------
INT 27 - DOS 1+ - TERMINATE AND STAY RESIDENT
DX = number of bytes to keep resident (max FFF0h)
CS = segment of PSP
Return: never
Notes: this is an obsolete call
INT 22, INT 23, and INT 24 are restored from the PSP
does not close any open files
the minimum number of bytes which will remain resident is 110h for
DOS 2.x and 60h for DOS 3.0+; there is no minimum for DOS 1.x, which
implements this service in COMMAND.COM rather than the DOS kernel
SeeAlso: INT 21/AH=31h
--------D-28---------------------------------
INT 28 C - DOS 2+ - DOS IDLE INTERRUPT
SS:SP = top of MS-DOS stack for I/O functions
Return: all registers preserved
Desc: This interrupt is invoked each time one of the DOS character input
functions loops while waiting for input. Since a DOS call is in
progress even though DOS is actually idle during such input waits,
hooking this function is necessary to allow a TSR to perform DOS
calls while the foreground program is waiting for user input. The
INT 28h handler may invoke any INT 21h function except functions
00h through 0Ch.
Notes: under DOS 2.x, the critical error flag (the byte immediately after the
InDOS flag) must be set in order to call DOS functions 50h/51h from
the INT 28h handler without destroying the DOS stacks. calls to INT 21/AH=3Fh,40h from within an INT 28 handler may not use a handle which refers to CON at the time of the call, the InDOS flag (see INT 21/AH=34h) is normally set to 01h; if larger, DOS is truly busy and should not be reentered the default handler is an IRET instruction supported in OS/2 compatibility box the _MS-DOS_Programmer's Reference_ for DOS 5.0 incorrectly documents this interrupt as superseded the performance of NetWare Lite servers (and probably other peer-to-peer networks) can be dramatically improved by calling INT 28 frequently from an application's idle loop SeeAlso: INT 21/AH=34h,INT 2A/AH=84h,INT 2F/AX=1680h

--------M-330000-------------------------------------
INT 33 - MS MOUSE - RESET DRIVER AND READ STATUS
AX = 0000h
Return: AX = status
0000h hardware/driver not installed
FFFFh hardware/driver installed
BX = number of buttons
0000h other than two
0002h two buttons (many drivers)
0003h Mouse Systems/Logitech three-button mouse
FFFFh two buttons
Notes: since INT 33 might be uninitialized on old machines, the caller should first check that INT 33 is neither 0000h:0000h nor points at an IRET instruction (BYTE CFh) before calling this API to use mouse on a Hercules-compatible monographics card in graphics mode, you must first set 0040h:0049h to 6 for page 0 or 5 for page 1, and then call this function. Logitech drivers v5.01 and v6.00 reportedly do not correctly use Hercules graphics in dual-monitor systems, while version 4.10 does.
the Logitech mouse driver contains the signature string "LOGITECH" three bytes past the interrupt handler; many of the Logitech mouse utilities check for this signature. Logitech MouseWare v6.30 reportedly does not support CGA video modes if no CGA is present when it is started and the video board is later switched into CGA emulation
SeeAlso: AX=0011h,AX=0021h,AX=002Fh,INT 62/AX=007Ah,INT 74

--------M-330001-------------------------------------
INT 33 - MS MOUSE v1.0+ - SHOW MOUSE CURSOR
AX = 0001h
SeeAlso: AX=0002h,INT 16/AX=FFFFh,INT 62/AX=007Bh,INT 6F/AH=06h"F_TRACK_ON"

--------M-330002-------------------------------------
INT 33 - MS MOUSE v1.0+ - HIDE MOUSE CURSOR
AX = 0002h
Note: multiple calls to hide the cursor will require multiple calls to function 01h to unhide it.
SeeAlso: AX=0001h, AX=0010h, INT 16/AX=FFFFh, INT 62/AX=007Bh
SeeAlso: INT 6F/AH=08h"F_TRACK_OFF"
--------M-330003-----------------------------
INT 33 - MS MOUSE v1.0+ - RETURN POSITION AND BUTTON STATUS
  AX = 0003h
Return: BX = button status (see #03168)
  CX = column
  DX = row
Note: in text modes, all coordinates are specified as multiples of the cell
  size, typically 8x8 pixels
SeeAlso: AX=0004h, AX=000Bh, INT 2F/AX=D000h"ZWmous"

Bitfields for mouse button status:
Bit(s) Description (Table 03168)
  0 left button pressed if 1
  1 right button pressed if 1
  2 middle button pressed if 1 (Mouse Systems/Logitech/Genius)
--------M-330004-----------------------------
INT 33 - MS MOUSE v1.0+ - POSITION MOUSE CURSOR
  AX = 0004h
  CX = column
  DX = row
Note: the row and column are truncated to the next lower multiple of the cell
  size (typically 8x8 in text modes); however, some versions of the
  Microsoft documentation incorrectly state that the coordinates are
  rounded
SeeAlso: AX=0003h, INT 62/AX=0081h, INT 6F/AH=10h"F_PUT_SPRITE"
--------M-330005-----------------------------
INT 33 - MS MOUSE v1.0+ - RETURN BUTTON PRESS DATA
  AX = 0005h
  BX = button number (see #03169)
Return: AX = button states (see #03168)
  BX = number of times specified button has been pressed since last call
  CX = column at time specified button was last pressed
  DX = row at time specified button was last pressed
Note: at least for the Genius mouse driver, the number of button presses
  returned is limited to 7FFFh
SeeAlso: AX=0006h, INT 62/AX=007Ch

(Table 03169)
Values for mouse button number:
  0000h left
  0001h right
  0002h middle (Mouse Systems/Logitech/Genius mouse)
--------M-330006-----------------------------
INT 33 - MS MOUSE v1.0+ - RETURN BUTTON RELEASE DATA
  AX = 0006h
  BX = button number (see #03169)
Return: AX = button states (see #03168)
    BX = number of times specified button has been released since last call
    CX = column at time specified button was last released
    DX = row at time specified button was last released

Note: at least for the Genius mouse driver, the number of button releases returned is limited to 7FFFh

SeeAlso: AX=0005h,INT 62/AX=007Ch

--------M-330007-------------------------------
INT 33 - MS MOUSE v1.0+ - DEFINE HORIZONTAL CURSOR RANGE
    AX = 0007h
    CX = minimum column
    DX = maximum column

Note: in text modes, the minimum and maximum columns are truncated to the next lower multiple of the cell size, typically 8x8 pixels

SeeAlso: AX=0008h,AX=0010h,AX=0031h,INT 62/AX=0080h
SeeAlso: INT 6F/AH=0Ch"F_SET_LIMITS_X"

--------M-330008-------------------------------
INT 33 - MS MOUSE v1.0+ - DEFINE VERTICAL CURSOR RANGE
    AX = 0008h
    CX = minimum row
    DX = maximum row

Note: in text modes, the minimum and maximum rows are truncated to the next lower multiple of the cell size, typically 8x8 pixels

SeeAlso: AX=0007h,AX=0010h,AX=0031h,INT 62/AX=0080h
SeeAlso: INT 6F/AH=0Eh"F_SET_LIMITS_Y"

--------M-330009-------------------------------
INT 33 - MS MOUSE v3.0+ - DEFINE GRAPHICS CURSOR
    AX = 0009h
    BX = column of cursor hot spot in bitmap (-16 to 16)
    CX = row of cursor hot spot (-16 to 16)
    ES:DX -> mask bitmap (see #03170)

Notes: in graphics modes, the screen contents around the current mouse cursor position are ANDed with the screen mask and then XORed with the cursor mask
the Microsoft mouse driver v7.04 and v8.20 uses only BL and CL, so the hot spot row/column should be limited to -128..127
Microsoft KnowledgeBase article Q19850 states that the high bit is right-most, but that statement is contradicted by all other available documentation

SeeAlso: AX=000Ah,AX=0012h,AX=002Ah,INT 62/AX=007Fh,INT 6F/AH=0Ah"F_DEF_MASKS"

Format of mouse mask bitmap:
Offset Size Description (Table 03170)
00h 16 WORDsscreen mask
10h 16 WORDscursor mask
Note: each word defines the sixteen pixels of a row, low bit rightmost

--------M-33000A-------------------------------
INT 33 - MS MOUSE v3.0+ - DEFINE TEXT CURSOR
AX = 000Ah
BX = hardware/software text cursor
    0000h software
    CX = screen mask
    DX = cursor mask
    0001h hardware
    CX = start scan line
    DX = end scan line

Note: when the software cursor is selected, the character/attribute data at
      the current screen position is ANDed with the screen mask and then
      XORed with the cursor mask

SeeAlso: AX=0009h,INT 62/AX=007Eh
----------M-33000B-----------------------------
INT 33 - MS MOUSE v1.0+ - READ MOTION COUNTERS
    AX = 000Bh

Return: CX = number of mickeys mouse moved horizontally since last call
        DX = number of mickeys mouse moved vertically

Notes: a mickey is the smallest increment the mouse can sense
       positive values indicate down/right

SeeAlso: AX=0003h,AX=001Bh,AX=0027h
----------M-33000C-----------------------------
INT 33 - MS MOUSE v1.0+ - DEFINE INTERRUPT SUBROUTINE PARAMETERS
    AX = 000Ch
    CX = call mask (see #03171)
        ES:DX -> FAR routine (see #03172)

SeeAlso: AX=0018h

Bitfields for mouse call mask:
Bit(s) Description (Table 03171)
  0 call if mouse moves
  1 call if left button pressed
  2 call if left button released
  3 call if right button pressed
  4 call if right button released
  5 call if middle button pressed (Mouse Systems/Logitech/Genius mouse)
  6 call if middle button released (Mouse Systems/Logitech/Genius mouse)
  7-15 unused

Note: some versions of the Microsoft documentation incorrectly state that CX
      bit 0 means call if mouse cursor moves

(Table 03172)
Values interrupt routine is called with:
    AX = condition mask (same bit assignments as call mask)
    BX = button state
    CX = cursor column
    DX = cursor row
    SI = horizontal mickey count
    DI = vertical mickey count
Notes: some versions of the Microsoft documentation erroneously swap the meanings of SI and DI
in text modes, the row and column will be reported as a multiple of the character cell size, typically 8x8 pixels

-------M-33000D-------------------------------
INT 33 - MS MOUSE v1.0+ - LIGHT PEN EMULATION ON
AX = 000Dh
SeeAlso: AX=000Eh,INT 10/AH=04h
-------M-33000E-------------------------------
INT 33 - MS MOUSE v1.0+ - LIGHT PEN EMULATION OFF
AX = 000Eh
SeeAlso: AX=000Dh
-------V-FF-------------------------------
INT FF - PC/FORTH - GRAPHICS API
BX = function number
 0001h function REDRAW
 0002h function !PEL
 0003h function @PEL
 0004h function LINE
 0005h function ARC
 0006h function @BLOCK
 0007h function !BLOCK
 0008h function FLOOD
DS:SI -> FORTH program counter
SS:BP -> FORTH parameter stack
SS:SP -> FORTH return stack
details of parameters not available
Return: AX,BX,CX,DX,ES,DI may be destroyed
Note: these functions all display an error message if the graphics routines are not resident

71.3 Port listing

This is only a portion of the port list available with RBIL. For a complete listing please refer CD.

71.3.1 Notations

The port description format is:

PPPPw RW description

where: PPPP is the four-digit hex port number or a plus sign and three hex digits to indicate an offset from a base port address
w is blank for byte-size port, 'w' for word, and 'd' for dword
R is dash (or blank) if not readable, 'r' if sometimes readable,'R' if "always" readable, '?' if readability unknown
W is dash (or blank) if not writable, 'w' if sometimes writable,
'W' if "always" writable, 'C' if write-clear, and '?' if writability unknown

### 71.3.2 Listing

----------P0000001F--------------------------
PORT 0000-001F - DMA 1 - FIRST DIRECT MEMORY ACCESS CONTROLLER (8237)
SeeAlso: PORT 0080h-008Fh"DMA", PORT 00C0h-00DFh

<table>
<thead>
<tr>
<th>Offset</th>
<th>Mode</th>
<th>Access</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 0</td>
<td>current address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0000</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0</td>
<td>base address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>RW</td>
<td>DMA</td>
<td></td>
<td>channel 0</td>
<td>word count</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0002</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 1</td>
<td>current address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0002</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 1</td>
<td>base address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0003</td>
<td>RW</td>
<td>DMA</td>
<td></td>
<td>channel 1</td>
<td>word count</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0004</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 2</td>
<td>current address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0004</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 2</td>
<td>base address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0005</td>
<td>RW</td>
<td>DMA</td>
<td></td>
<td>channel 2</td>
<td>word count</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0006</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 3</td>
<td>current address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0006</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 3</td>
<td>base address</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0007</td>
<td>RW</td>
<td>DMA</td>
<td></td>
<td>channel 3</td>
<td>word count</td>
<td>byte 0, then byte 1</td>
<td></td>
</tr>
<tr>
<td>0008</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>status register</td>
<td>(see #P0001)</td>
<td></td>
</tr>
<tr>
<td>0008</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>command register</td>
<td>(see #P0002)</td>
<td></td>
</tr>
<tr>
<td>0009</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>write request register</td>
<td>(see #P0003)</td>
<td></td>
</tr>
<tr>
<td>000A</td>
<td>RW</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>mask register</td>
<td>(see #P0004)</td>
<td></td>
</tr>
<tr>
<td>000B</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>mode register</td>
<td>(see #P0005)</td>
<td></td>
</tr>
<tr>
<td>000C</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>clear byte pointer flip-flop register</td>
<td>any write clears LSB/MSB flip-flop of address and counter registers</td>
<td></td>
</tr>
<tr>
<td>000D</td>
<td>R-</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>temporary register</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000D</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>master clear register</td>
<td>any write causes reset of 8237</td>
<td></td>
</tr>
<tr>
<td>000E</td>
<td>-W</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>clear mask register</td>
<td>any write clears masks for all channels</td>
<td></td>
</tr>
<tr>
<td>000F</td>
<td>rW</td>
<td>DMA</td>
<td></td>
<td>channel 0-3</td>
<td>write mask register</td>
<td>(see #P0006)</td>
<td></td>
</tr>
</tbody>
</table>

Notes: the temporary register is used as holding register in memory-to-memory DMA transfers; it holds the last transferred byte.
channel 2 is used by the floppy disk controller on the IBM PC/XT channel 0 was used for the memory refresh and channel 3 was used by the hard disk controller on AT and later machines with two DMA controllers, channel 4 is used as a cascade for channels 0-3 command and request registers do not exist on a PS/2 DMA controller.

Bitfields for DMA channel 0-3 status register:

- **Bit(s)**: Description (Table P0001)
  - 7: channel 3 request active
  - 6: channel 2 request active
channel 1 request active
channel 0 request active
channel terminal count on channel 3
channel terminal count on channel 2
channel terminal count on channel 1
channel terminal count on channel 0

SeeAlso: #P0002,#P0481

Bitfields for DMA channel 0-3 command register:
Bit(s) Description (Table P0002)
7 DACK sense active high
6 DREQ sense active high
5 =1 extended write selection
  =0 late write selection
4 rotating priority instead of fixed priority
3 compressed timing (two clocks instead of four per transfer)
  =1 normal timing (default)
  =0 compressed timing
2 =1 enable controller
  =0 enable memory-to-memory
1-0 channel number
SeeAlso: #P0001,#P0004,#P0005,#P0482

Bitfields for DMA channel 0-3 request register:
Bit(s) Description (Table P0003)
7-3 reserved (0)
2 =0 clear request bit
  =1 set request bit
1-0 channel number
  00 channel 0 select
  01 channel 1 select
  10 channel 2 select
  11 channel 3 select
SeeAlso: #P0004

Bitfields for DMA channel 0-3 mask register:
Bit(s) Description (Table P0004)
7-3 reserved (0)
2 =0 clear mask bit
  =1 set mask bit
1-0 channel number
  00 channel 0 select
  01 channel 1 select
  10 channel 2 select
  11 channel 3 select
SeeAlso: #P0001,#P0002,#P0003,#P0484

Bitfields for DMA channel 0-3 mode register:
Bit(s) Description (Table P0005)
7-6 transfer mode
  00 demand mode
  01 single mode
  10 block mode
  11 cascade mode
5 direction
  =0 increment address after each transfer
  =1 decrement address
3-2 operation
  00 verify operation
  01 write to memory
  10 read from memory
  11 reserved
1-0 channel number
  00 channel 0 select
  01 channel 1 select
  10 channel 2 select
  11 channel 3 select

SeeAlso: #P0002,#P0485

Bitfields for DMA channel 0-3 write mask register:

Bit(s) Description (Table P0006)
7-4 reserved
3 channel 3 mask bit
2 channel 2 mask bit
1 channel 1 mask bit
0 channel 0 mask bit

Note: each mask bit is automatically set when the corresponding channel reaches terminal count or an external EOP signal is received

SeeAlso: #P0004,#P0486

----------P0040005F--------------------------
PORT 0040-005F - PIT - PROGRAMMABLE INTERVAL TIMER (8253, 8254)

Notes: XT & AT use ports 40h-43h; PS/2 uses ports 40h, 42h-44h, and 47h
the counter chip is driven with a 1.193 MHz clock (1/4 of the original PC's 4.77 MHz CPU clock)

SeeAlso: PORT 0044h, PORT 0048h

0040 RW PIT counter 0, counter divisor (XT, AT, PS/2)
  Used to keep the system time; the default divisor of (1)0000h produces the 18.2Hz clock tick.

0041 RW PIT counter 1, RAM refresh counter (XT, AT)
  don't set below 3 on PCs (default 12h), and don't mess with this counter at all unless you really know what you're doing....

0042 RW PIT counter 2, cassette & speaker (XT, AT, PS/2)
  During normal operation mode (8253) 40h-42h set the counter values on write and get the current counter value on read. In 16bit modes two consecutive writes/reads must be issued, first with the low byte,
followed by the high byte. In 8254 read back modes, all selected counters and status are latched and must be read out completely before normal operation is valid again. Each counter switches back to normal operation after read out. In 'get status and counter' mode the first byte read is the status, followed by one or two counter values. (see #P0379) Note that 16-bit reads performed without using the "latch" command will get the current high/low portion of the counter at the instant of the port read, so it is possible for the low part of the counter to wrap around before the high part gets read, resulting in a significant measurement error.

0043  RW  PIT  mode port, control word register for counters 0-2 (see #P0380)

Once a control word has been written (43h), it must be followed immediately by performing the corresponding action to the counter registers (40h-42h), else the system may hang!!

Bitfields for 8254 PIT counter status byte:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description                      (Table P0379)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>PIN status of OUTx Pins (1=high, 0=low)</td>
</tr>
<tr>
<td>6</td>
<td>counter start value loaded</td>
</tr>
<tr>
<td></td>
<td>=0: yes, so counter latch is valid to be read</td>
</tr>
<tr>
<td></td>
<td>=1: no, wait for counter latch to be set (may last a while)</td>
</tr>
<tr>
<td>5-0</td>
<td>counter mode, same as bit5-0 at 43h</td>
</tr>
</tbody>
</table>

SeeAlso: #P0380

Bitfields for 8253/8254 PIT mode control word:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description                      (Table P0380)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-6</td>
<td>counter select</td>
</tr>
<tr>
<td>00</td>
<td>counter 0 select</td>
</tr>
<tr>
<td>01</td>
<td>counter 1 select                 (not PS/2)</td>
</tr>
<tr>
<td>10</td>
<td>counter 2 select</td>
</tr>
<tr>
<td>11</td>
<td>(8253) reserved</td>
</tr>
<tr>
<td></td>
<td>(8254) read back counter (see #P0379)</td>
</tr>
<tr>
<td>5-4</td>
<td>counter access</td>
</tr>
<tr>
<td>00</td>
<td>counter latch command</td>
</tr>
<tr>
<td></td>
<td>BUG: Intel Neptune/Mercury/Aries Chipset 8237IB (SIO) needs a short delay after issuing this command, else the latched MSB may be outdated with respect to the LSB, resulting in large measuring errors. Workaround: Check for this condition by comparing results with last results and don't use erroneous results.</td>
</tr>
<tr>
<td>01</td>
<td>read/write counter bits 0-7 only</td>
</tr>
<tr>
<td>10</td>
<td>read/write counter bits 8-15 only</td>
</tr>
<tr>
<td>11</td>
<td>read/write counter bits 0-7 first, then 8-15</td>
</tr>
<tr>
<td>3-1</td>
<td>counter mode</td>
</tr>
<tr>
<td>000</td>
<td>mode 0 select - zero detection interrupt</td>
</tr>
<tr>
<td>001</td>
<td>mode 1 select - programmable one shot</td>
</tr>
</tbody>
</table>
A to Z of C

x10 mode 2 select - rate generator
x11 mode 3 select - square wave generator
counts down twice by two at a time; latch status and check
value of OUT pin to determine which half-cycle is active
divisor factor 3 not allowed!
100 mode 4 select - software triggered strobe
101 mode 5 select - hardware triggered strobe
0 counting style
0 binary counter 16 bits
1 BCD counter (4 decades)
---if read back---
5-4 what to read
00 counter status, then value
01 counter value
10 counter status
11 reserved
3 select counter 2
2 select counter 1
1 select counter 0
0 reserved (0)
Note: after issuing a read back 'get status' command, any new read back
command is ignored until the status is read from all selected
counters.
-------K-P0060006F--------------------------
PORT 0060-006F - KEYBOARD CONTROLLER 804x (8041, 8042) (or PPI (8255) on PC,XT)
Note: XT uses ports 60h-63h, AT uses ports 60h-64h

0060 RW KB controller data port or keyboard input buffer (ISA, EISA)
should only be read from after status port bit0 = 1
should only be written to if status port bit1 = 0
0060 R- KeyBoard or KB controller data output buffer (via PPI on XT)
PC: input from port A of 8255, if bit7 in 61h set (see #P0396)
get scan codes, special codes (in PC: with bit7 in 61h cleared)
(see #P0390)

0061 R- KB controller port B control register (ISA, EISA)
        system control port for compatibility with 8255 (see #P0393)
0061 -W KB controller port B (ISA, EISA) (PS/2 port A is at 0092)
        system control port for compatibility with 8255 (see #P0392)
0061 -W PPI Programmable Peripheral Interface 8255 (XT only)
        system control port (see #P0394)
0062 RW PPI (XT only) data port C (see #P0395)
0063 RW PPI (XT only) command mode register (see #P0397)

0064 R- keyboard controller read status (see #P0398,#P0399,#P0400)
0064 -W keyboard controller input buffer (ISA, EISA) (see #P0401)

0064 -W (Amstrad/Schneider PC1512) set 'DIP switch S1' setting
stored in CMOS RAM that PPI should report for compatibility

0065  -W  (Amstrad/Schneider PC1512) set 'DIP switch S2' RAM size setting
       stored in CMOS RAM, that PPI port C (PORT 0064h) should report for
       compatibility

0066  R- communications port (Olivetti M24)
0066  R? configuration port (Olivetti M24 with model byte 0)
       bit 5 set if 8530 SIO present (see also PORT 0065h"Olivetti")

Bitfields for AT keyboard controller input port:
Bit(s)    Description              (Table P0381)
 7     keyboard enabled
 6     =0   CGA, else MDA
 5     =0   manufacturing jumper installed
 4     =0   system RAM 512K, else 640K
 3-0    reserved
SeeAlso: #P0382,#P0384

Bitfields for AT keyboard controller input port (Compaq):
Bit(s)    Description              (Table P0382)
 7     security lock is unlocked
 6     =0   Compaq dual-scan display, 1=non-Compaq display
 5     system board dip switch 5 is OFF
 4     =0   auto speed selected, 1=high speed selected
 3     =0   slow (4MHz), 1 = fast (8MHz)
 2     no math coprocessor installed
 1-0    reserved
SeeAlso: #P0383

Bitfields for AT keyboard controller output port:
Bit(s)    Description              (Table P0383)
 7     keyboard data output
 6     keyboard clock output
 5     input buffer NOT full
 4     output buffer NOT empty
 3     reserved (see note)
 2     reserved (see note)
 1     gate A20
 0     system reset

Note: bits 2 and 3 are the turbo speed switch or password lock on
       Award/AMI/Phoenix BIOSes. These bits make use of nonstandard
       keyboard controller BIOS functionality to manipulate
       pin 23 (8041 port 22) as turbo switch for AWARD
       pin 35 (8041 port 15) as turbo switch/pw lock for Phoenix

SeeAlso: #P0381,#P0384
----------P0070007F--------------------------
PORT 0070-007F - CMOS RAM/RTC (REAL TIME CLOCK)
Note: the real-time clock may be either a discrete MC146814, MC146818, or
an emulation thereof built into the motherboard chipset
SeeAlso: PORT 00A0h"XT"

0070  -W  CMOS RAM index register port (ISA, EISA)
  bit 7  = 1  NMI disabled from reaching CPU
       = 0  NMI enabled
  bit 6-0  CMOS RAM index
       (64 bytes in early systems, now usually 128 bytes)
Note: any write to PORT 0070h should be followed by an action to
      PORT 0071h or the RTC will be left in an unknown state.

0071  RW  CMOS RAM data port (ISA, EISA) (see #P0409)

(Table P0409)
Values for Real-Time Clock register number (see also CMOS.LST):
  00h-0Dh clock registers
  0Eh  diagnostics status byte
  0Fh  shutdown status byte
  10h  diskette drive type for A: and B:
  11h  reserved / IBM fixed disk / setup options
  12h  fixed disk drive type for drive 0 and drive 1
  13h  reserved / AMI Extended CMOS setup (AMI Hi-Flex BIOS)
  14h  equipment byte
  15h  LSB of system base memory in Kb
  16h  MSB of system base memory in Kb
  17h  LSB of total extended memory in Kb
  18h  MSB of total extended memory in Kb
  19h  drive C extension byte
  1Ah  drive D extension byte
  1Bh-2Dh reserved
  20h-27h commonly used for first user-configurable drive type
  2Eh  CMOS MSB checksum over 10-2D
  2Fh  CMOS LSB checksum over 10-2D
  30h  LSB of extended memory found above 1Mb at POST
  31h  MSB of extended memory found above 1Mb at POST
  32h  date century in BCD
  33h  information flags
  34h-3Fh reserved
  35h-3Ch commonly used for second user-configurable drive type
  3Dh-3Eh word to 82335 MCR memory config register at [22] (Phoenix)
  42h-4Ch AMI 1990 Hyundai super-NB368S notebook
  ???
  54h-57h AMI 1990 Hyundai super-NB368S notebook
  ???
  5Ch-5Dh AMI 1990 Hyundai super-NB368S notebook
  ???
  60h-61h AMI 1990 Hyundai super-NB368S notebook
  ???
--------V-P03C603C9--------------------------
PORT 03C6-03C9 - EGA/VGA/MCGA - DAC REGISTERS
Range: PORT 03C6h or PORT 02C6h (alternate)
SeeAlso: PORT 03C0h, PORT 03C2h, PORT 03C4h, PORT 03CAh, PORT 03CEh "EGA", PORT 03D0h
SeeAlso: PORT 83C6h "Wingine"

03C6 RW (VGA, MCGA) PEL mask register (default FFh)
  VGA: AND mask for color-register address.
  MCGA: Never change from the default FFh.

03C6 RW HiColor ET4000 (Sierra RAMDACs e.g. SC11486, SC11481, SC11488):
  Enable HiColor feature: beside other assignments,
  consecutive read 3C6h 4 times and write magic value 80h to it.

03C7 -W (VGA, MCGA, CEG-VGA) PEL address register (read mode)
  Sets DAC in read mode and assign start of color register
  index (0..255) for following read accesses to 3C9h.
  Don't write to 3C9h while in read mode. Next access to
  03C8h will stop pending mode immediately.

03C7 -W (CEG-Color VGA w/ Edsun Labs RAMDACs)
  Enable and set Countinous Edge Graphics Mode:
  Consecutive write the following three key sequences in read
  mode (!) to 3C9h register DEh : 'CEG', 'EDS', 'UNx' (x see
  below). Current CEG mode can be read from palette register
  BFh 'blue', write access to that register will disable CEG
  features.
  In CEG modes by combining old with new colors and dynamically
  changing palette values, the effective colors displayable
  are enhanced dramatically (in EDP modes up to virtually 32bit
  truecolor) on standard 16/256 color VGA. Also, effective
  resolution enhancement takes effect by anti-aliasing.
  Necessary EDP escape sequences should be moved to image
  border or single colored areas, if possible.
  
REP-mode: if pixel are doubled in current video mode
EDP-mode: pseudo-truecolor with Edsun dynamic palette
(see #P0698, #P0699)

Palette-color-register single-byte-format (each 3 times):
  Mode A:    Mode C:
  bit7-4: mix code    bit3  : 0=color, 1=code
  bit3-0: color code    bit2-0: color / mix code
  Mode B:    Mode D:
  bit7-5: mix code    bit7-0: see mix code table
  bit4  : 0=new, 1=old    Non-CEG modes:
  bit3-0: color code    bit7-0: as usual

In EDP modes, video-memory-palette-changing escape-sequences:
  Mode A:    Mode B:    Mode C:    Mode D:
  7/escape    7/escape    7/escape    0BFh
  red        red    red7-4      red
green       green    red3-0      green
blue        blue    green7-4     blue
address     address    green3-0    address
blue7-4
blue3-0
address

03C7  R- VGA DAC state register
       bit7-2 reserved
       bit1-0: 00b write palette cycle (write mode)
               01h reserved
               10b reserved
               11b read palette cycle (read mode)

03C8  RW (VGA,MCGA) PEL address register (write mode)
       Sets DAC in write mode and assign start of color register
       index (0..255) for following write accesses to 3C9h.
       Don't read from 3C9h while in write mode. Next access to
       03C8h will stop pending mode immediately.

03C8  RW (Genoa SuperEGA) SuperEGA control register (all emulation modes)
       bit7-2: reserved
       bit1 : 0=EGA mode, 1=backward compatibility mode
       bit0 : not used

03C8  R? (S3 Trios32/64) General Input Port (see #P0738)

03C9  RW (VGA,MCGA) PEL data register
       Three consecutive reads (in read mode) or writes (in write
       mode) in the order: red, green, blue. The internal DAC index
       is incremented each 3rd access.
       bit7-6: HiColor VGA DACs only: color-value bit7-6
       bit5-0:          color-value bit5-0

(Table P0698)
Values for EDSUN CEG (Continuous Edge Graphics) modes::

x:  mode:  colors:  mix:  pixel depth:  effective colors:
0 = disabled  256   -   8      256
1 = A          16   16   8   1920
2 = A+REP      16   16   8  dblscn  1920
3 = A+EDP      15   16  truecolor
4 = reserved   -   -   -        -
5 = B          16   8   8      960
6 = B+REP      16   8   8  dblscn  960
7 = B+EDP      15   8  truecolor
8 = reserved   -   -   -        -
9 = C          8   8   4      224
10 = C+REP     8   8   4  dblscn  224
11 = C+EDP     7   8  truecolor
12 = reserved  -   -   -        -
13 = D         223  32   8  792096
14 = D+REP     223  32   8  dblscn  792096
15 = D+EDP     223  32  truecolor
SeeAlso: #P0699

(Table P0699)
Values for EDSUN CEG mixing codes:

<table>
<thead>
<tr>
<th>Mode A: mix: new: old:</th>
<th>Mode C: mix: new: old: colorcode:</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 = 32/32 0/32</td>
<td>0 = - - 0</td>
</tr>
<tr>
<td>1 = 30/32 2/32</td>
<td>1 = - - 1</td>
</tr>
<tr>
<td>2 = 28/32 4/32</td>
<td>2 = - - 2</td>
</tr>
<tr>
<td>3 = 26/32 6/32</td>
<td>3 = - - 3</td>
</tr>
<tr>
<td>4 = 24/32 8/32</td>
<td>4 = - - 4</td>
</tr>
<tr>
<td>5 = 22/32 10/32</td>
<td>5 = - - 5</td>
</tr>
<tr>
<td>6 = 20/32 12/32</td>
<td>6 = - - 6</td>
</tr>
<tr>
<td>7 = 18/32 14/32</td>
<td>7 = - - 7/EDP</td>
</tr>
<tr>
<td>8 = 16/32 16/32</td>
<td>8 = 30/32 2/32</td>
</tr>
<tr>
<td>9 = 14/32 18/32</td>
<td>9 = 28/32 4/32</td>
</tr>
<tr>
<td>10 = 12/32 20/32</td>
<td>10 = 26/32 6/32</td>
</tr>
<tr>
<td>11 = 10/32 22/32</td>
<td>11 = 24/32 8/32</td>
</tr>
<tr>
<td>12 = 8/32 24/32</td>
<td>12 = 22/32 10/32</td>
</tr>
<tr>
<td>13 = 6/32 26/32</td>
<td>13 = 20/32 12/32</td>
</tr>
<tr>
<td>14 = 4/32 28/32</td>
<td>14 = 18/32 14/32</td>
</tr>
<tr>
<td>15 = 2/32 30/32</td>
<td>15 = 16/32 16/32</td>
</tr>
</tbody>
</table>

---Mode B: mix: new: old: | Mode D: mix: new: old: description:
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0 = 30/32 2/32</td>
<td>00h..BEh = - -   normal color</td>
</tr>
<tr>
<td>1 = 26/32 6/32</td>
<td>BFh = - - EDP</td>
</tr>
<tr>
<td>2 = 22/32 10/32</td>
<td>C0h = 32/32 0/32</td>
</tr>
<tr>
<td>3 = 18/32 14/32</td>
<td>C1h = 31/32 1/32</td>
</tr>
<tr>
<td>4 = 14/32 18/32</td>
<td>C2h = 30/32 2/32</td>
</tr>
<tr>
<td>5 = 10/32 22/32</td>
<td>... = ... ...</td>
</tr>
<tr>
<td>6 = 6/32 26/32</td>
<td>DFh = 0/32 32/32</td>
</tr>
<tr>
<td>7 = 2/32 30/32</td>
<td>E0h-FFh = - - normal color</td>
</tr>
</tbody>
</table>

SeeAlso: #P0698

### 71.4 Memory map

DOS, BIOS and other software use certain specific memory address spaces to store important information. So if we know those addresses, we can manipulate the values present there with pointers. For example, the keyboard buffer’s head pointer is found at 0040:001A; we need this address if we want to manipulate the keyboard buffer.

Memory map is one of the wonderful collections present in RBIL. You may want to “play” with pointers. So here I present the full memory map from RBIL.

--------H-M00000000--------------------------
MEM 0000h:0000h R - INTERRUPT VECTOR TABLE
Size: 1024 BYTES
Note: see also the main interrupt list
MEM 0000h:031Dh - 1989 AMI 386sx BIOS - USER-DEFINABLE TYPE 47 HARD DISK PARMS
Size: 16 BYTES
Note: these fields are used if the AMI BIOS setup is set to use the top of the interrupt table for the extended BIOS data area.
SeeAlso: MEM 0000h:032Dh, INT 41

MEM 0000h:032Dh - 1989 AMI 386sx BIOS - USER-DEFINABLE TYPE 48 HARD DISK PARMS
Size: 16 BYTES
Note: these fields are used if the AMI BIOS setup is set to use the top of the interrupt table for the extended BIOS data area.
SeeAlso: MEM 0000h:031Dh, INT 46

MEM 0000h:0400h - BIOS DATA AREA
Size: 256 BYTES
Note: see also the MEM 0040h:xxxxh entries

MEM 0000h:0500h - DATA AREA
Size: 256 BYTES

MEM 0000h:0600h - MS-DOS 1.x LOAD ADDRESS

MEM 0000h:0700h - MS-DOS 2+ LOAD ADDRESS

MEM 0040h:0000h - BASE I/O ADDRESS OF FIRST SERIAL I/O PORT
Size: WORD
Notes: the BIOS sets this word to zero if it is unable to find any serial ports at the addresses it is programmed to check at boot. DOS and BIOS serial device numbers may be redefined by re-assigning these values of the base I/O addresses stored here. Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS COM1=port_address,logical_no[,timeout] directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4, timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0002h, MEM 0040h:0004h, MEM 0040h:0006h, MEM 0040h:0008h

MEM 0040h:0002h - BASE I/O ADDRESS OF SECOND SERIAL I/O PORT
Size: WORD
Notes: the BIOS sets this word to zero if it is unable to find more than one serial port at the addresses it is programmed to check at boot. DOS and BIOS serial device numbers may be redefined by re-assigning these values of the base I/O addresses stored here. Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS COM2=port_address,logical_no[,timeout] directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4, timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0000h, MEM 0040h:0004h, MEM 0040h:0006h, MEM 0040h:000Ah
SeeAlso: MEM 0040h:007Dh, INT 14/AH=00h, PORT 02F8h "SERIAL"
--------S-M00400004--------------------------
MEM 0040h:0004h - BASE I/O ADDRESS OF THIRD SERIAL I/O PORT
Size: WORD
Notes: the BIOS sets this word to zero if is unable to find more than two
serial ports at the addresses it is programmed to check at boot
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS COM3=[port_address|logical_no],[timeout]
directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4,
timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0000h, MEM 0040h:0002h, MEM 0040h:0006h, MEM 0040h:000Ch
SeeAlso: MEM 0040h:007Eh, PORT 03E8h "SERIAL"
--------S-M00400006--------------------------
MEM 0040h:0006h - BASE I/O ADDRESS OF FOURTH SERIAL I/O PORT
Size: WORD
Notes: the BIOS sets this word to zero if is unable to find more than three
serial ports at the addresses it is programmed to check at boot
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS COM4=[port_address|logical_no],[timeout]
directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4,
timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0000h, MEM 0040h:0002h, MEM 0040h:0004h, MEM 0040h:0008h
SeeAlso: MEM 0040h:007Fh, PORT 03E8h "SERIAL"
--------P-M00400008--------------------------
MEM 0040h:0008h - BASE I/O ADDRESS OF FIRST PARALLEL I/O PORT
Size: WORD
Notes: the BIOS POST routine fills in the parallel port address fields in
turn as it finds parallel ports. All fields beyond the last one
for which a valid parallel port was found are set to zero.
the BIOS INT 17 handler uses these fields to address the parallel
ports
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS LPT1=[port_address|logical_no],[timeout]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso: MEM 0040h:0000h, MEM 0040h:000Ah, MEM 0040h:000Ch, INT 17/AH=00h
SeeAlso: PORT 0278h "PRINTER", PORT 03BCh "PRINTER"
--------P-M0040000A--------------------------
MEM 0040h:000Ah - BASE I/O ADDRESS OF SECOND PARALLEL I/O PORT
Size: WORD
Notes: zero if fewer than two parallel ports installed
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS LPT2=[port_address|logical_no],[timeout]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso: MEM 0040h:0002h, MEM 0040h:0008h, MEM 0040h:000Ch, PORT 0278h "PRINTER"
SeeAlso: PORT 0378h "PRINTER", INT 17/AH=00h
--------P-M0040000C--------------------------
MEM 0040h:000Ch - BASE I/O ADDRESS OF THIRD PARALLEL I/O PORT
Size: WORD
Notes: zero if fewer than three parallel ports installed
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS LPT3=[port_address|logical_no][,[timeout]]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso: MEM 0040h:0004h,MEM 0040h:0008h,MEM 0040h:000Ah,MEM 0040h:000Eh
SeeAlso: PORT 0378h"PRINTER",INT 17/AH=00h
--------P-M0040000E--------------------------
MEM 0040h:000Eh - BASE I/O ADDRESS OF FOURTH PARALLEL I/O PORT (pre-PS/2)
Size: WORD
Notes: zero if fewer than four parallel ports installed
Under DR-OpenDOS 7.02+ this setting can be changed with the
undocumented CONFIG.SYS LPT4=(port_address|logical_no)[,[timeout]]
directive, where port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20). To avoid any interference with the PS/2
and later interpretation, this will be rejected if this entry does
not hold 0, which would indicate it is used for different purposes.
SeeAlso: MEM 0040h:0008h,MEM 0040h:000Ah,MEM 0040h:000Eh"BIOS DATA"
SeeAlso: PORT 0378h"PRINTER",INT 17/AH=00h
--------B-M0040000E--------------------------
MEM 0040h:000Eh - SEGMENT OF EXTENDED BIOS DATA SEGMENT (PS/2, newer BIOSes)
Size: WORD
SeeAlso: MEM 0040h:000Eh"PARALLEL",INT 15/AH=C1h

Format of Extended BIOS Data Area (IBM):
Offset | Size | Description (Table M0001)
00h | BYTE | length of EBDA in kilobytes
01h | 15 BYTES | reserved
17h | BYTE | number of entries in POST error log (0-5)
18h | 5 WORDs | POST error log (each word is a POST error number)
22h | DWORD | Pointing Device Driver entry point
26h | BYTE | Pointing Device Flags 1 (see #M0002)
27h | BYTE | Pointing Device Flags 2 (see #M0003)
28h | 8 BYTES | Pointing Device Auxiliary Device Data
30h | DWORD | Vector for INT 07h stored here during 80387 interrupt
34h | DWORD | Vector for INT 01h stored here during INT 07h emulation
38h | BYTE | Scratchpad for 80287/80387 interrupt code
39h | WORD | Timer3: Watchdog timer initial count
3Bh | BYTE | ??? seen non-zero on Model 30
3Ch | BYTE | ???
3Dh | 16 BYTES | Fixed Disk parameter table for drive 0 (for older machines
which don't directly support the installed drive)
4Dh | 16 BYTES | Fixed Disk parameter table for drive 1 (for older machines
which don't directly support the installed drive)
5Dh-67h | ???
68h | BYTE | cache control
bits 7-2 unused (0)
bit 1: CPU cache failed test
bit 0: CPU cache disabled

69h-6Bh  ???
6Ch  BYTE  Fixed disk: (=FFh on ESDI systems)
        bits 7-4: Channel number 00-0Fh
        bits 3-0: DMA arbitration level 00-0Eh

6Dh  BYTE  ???
6Eh  WORD  current typematic setting (see INT 16/AH=03h)
70h  BYTE  number of attached hard drives
71h  BYTE  hard disk 16-bit DMA channel
72h  BYTE  interrupt status for hard disk controller (1Fh on timeout)
73h  BYTE  hard disk operation flags
        bit 7: controller issued operation-complete INT 76h
        bit 6: controller has been reset
        bits 5-0: unused (0)

74h  DWORD  old INT 76h vector
78h  BYTE  hard disk DMA type
        typically 44h for reads and 4Ch for writes
79h  BYTE  status of last hard disk operation
7Ah  BYTE  hard disk timeout counter
7Bh-7Dh
7Eh  8 WORDs storage for hard disk controller status
8Eh-E6h
E7h  BYTE  floppy drive type
        bit 7: drive(s) present
        bits 6-2: unused (0)
        bit 1: drive 1 is 5.25" instead of 3.5"
        bit 0: drive 0 is 5.25"

E8h  4 BYTEs  ???
ECH  BYTE  hard disk parameters flag
        bit 7: parameters loaded into EBDA
        bits 6-0: unused (0)

EDh  BYTE  ???
EEh  BYTE  CPU family ID (03h = 386, 04h = 486, etc.) (see INT 15/AH=C9h)
EFh  BYTE  CPU stepping (see INT 15/AH=C9h)
F0h  39 BYTEs  ???
117h  WORD  keyboard ID (see INT 16/AH=0Ah)
        (most commonly 41ABh)

119h  BYTE  ???
11Ah  BYTE  non-BIOS INT 18h flag
        bits 7-1: unused (0)
        bit 0: set by BIOS before calling user INT 18h at offset 11Dh

11Bh  2 BYTE  ???
11Dh  DWORD  user INT 18h vector if BIOS has re-hooked INT 18h
121h and up:  ??? seen non-zero on Model 60
3F0h  BYTE  Fixed disk buffer (???)
SeeAlso: #M0004
Bitfields for Pointing Device Flags 1:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>command in progress</td>
</tr>
<tr>
<td>6</td>
<td>resend byte (FAh) received</td>
</tr>
<tr>
<td>5</td>
<td>acknowledge byte (FEh) received</td>
</tr>
<tr>
<td>4</td>
<td>error byte (FCh) received</td>
</tr>
<tr>
<td>3</td>
<td>unexpected value received</td>
</tr>
<tr>
<td>2-0</td>
<td>index count for auxiliary device data at 28h</td>
</tr>
</tbody>
</table>

SeeAlso: #M0001,#M0003

Bitfields for Pointing Device Flags 2:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>device driver far call flag</td>
</tr>
<tr>
<td>6-3</td>
<td>reserved</td>
</tr>
<tr>
<td>2-0</td>
<td>package size (number of bytes received) - 1</td>
</tr>
</tbody>
</table>

SeeAlso: #M0001,#M0002

Format of Extended BIOS Data Area (AMI v1.00.12.AX1T):

<table>
<thead>
<tr>
<th>Offset</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>BYTE</td>
<td>length of XBDA in kilobytes</td>
</tr>
<tr>
<td>01h</td>
<td>15 BYTES</td>
<td>reserved</td>
</tr>
<tr>
<td>17h</td>
<td>BYTE</td>
<td>number of entries in POST error log (0-10)</td>
</tr>
<tr>
<td>18h</td>
<td>10 BYTES</td>
<td>unused???</td>
</tr>
<tr>
<td>22h</td>
<td>DWORD</td>
<td>Pointing Device Driver entry point</td>
</tr>
<tr>
<td>26h</td>
<td>BYTE</td>
<td>Pointing Device Flags 1 (see #M0002)</td>
</tr>
<tr>
<td>27h</td>
<td>BYTE</td>
<td>Pointing Device Flags 2 (see #M0003)</td>
</tr>
<tr>
<td>28h</td>
<td>8 BYTES</td>
<td>Pointing Device Auxiliary Device Data</td>
</tr>
<tr>
<td>30h</td>
<td>13 BYTES</td>
<td>??</td>
</tr>
<tr>
<td>3Dh</td>
<td>16 BYTES</td>
<td>Fixed Disk parameter table for drive 0</td>
</tr>
<tr>
<td>4Dh</td>
<td>16 BYTES</td>
<td>Fixed Disk parameter table for drive 1</td>
</tr>
<tr>
<td>5Dh</td>
<td>16 BYTES</td>
<td>parameter table for drive 2??</td>
</tr>
<tr>
<td>6Dh</td>
<td>16 BYTES</td>
<td>parameter table for drive 3??</td>
</tr>
<tr>
<td>80h</td>
<td>56 BYTES</td>
<td>IDE drive 0 manufacturer/model string</td>
</tr>
<tr>
<td>B8h</td>
<td>41 BYTES</td>
<td>AMIBIOS copyright string</td>
</tr>
<tr>
<td>E1h</td>
<td>WORD</td>
<td>unused???</td>
</tr>
<tr>
<td>102h</td>
<td>WORD</td>
<td>??? flags</td>
</tr>
<tr>
<td>bit 15:</td>
<td>???</td>
<td></td>
</tr>
<tr>
<td>108h</td>
<td>WORD</td>
<td>offset of IntelIDECfgTbl (IDE configuration settings) within</td>
</tr>
<tr>
<td></td>
<td></td>
<td>segment F000h</td>
</tr>
<tr>
<td>10Ah</td>
<td>2 BYTES</td>
<td>???</td>
</tr>
<tr>
<td>10Ch</td>
<td>DWORD</td>
<td>pointer to routine to call for language-specific error messages</td>
</tr>
<tr>
<td>110h</td>
<td>WORD</td>
<td>offset in segment F000h of end of currently-loaded optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BIOS subsystems (language, APM, etc.)</td>
</tr>
<tr>
<td>112h</td>
<td>WORD</td>
<td>offset in segment F000h of end of area available for loading</td>
</tr>
<tr>
<td></td>
<td></td>
<td>optional BIOS subsystems</td>
</tr>
<tr>
<td>1F0h</td>
<td>BYTE</td>
<td>APM status flags</td>
</tr>
<tr>
<td>1F1h</td>
<td>8 BYTES</td>
<td>APM power-state data for device classes 01h-06h</td>
</tr>
</tbody>
</table>
bits 0-3: current power state for devices 00h-03h in class
bits 7-4: current engaged state for devices 00h-03h in class

1F9h 4 BYTES APM power-state data for device classes 01h-08h (four devices per class)
1FDh 3 BYTES ???
200h 10 WORDs POST error log
214h ???
SeeAlso: #M0001,#M0005

Format of Extended BIOS Data Area (PhoenixBIOS 4.0):
Offset  Size Description (Table M0005)
00h  BYTE length of XBDA in kilobytes
01h 33 BYTES reserved
22h  DWORD Pointing Device Driver entry point
26h  BYTE Pointing Device Flags 1 (see #M0002)
27h  BYTE Pointing Device Flags 2 (see #M0003)
28h  8 BYTES Pointing Device Auxiliary Device Data
SeeAlso: #M0001,#M0004

--------B-M00400010--------------------------
MEM 0040h:0010h - INSTALLED HARDWARE
Size: WORD
SeeAlso: INT 11

Bitfields for BIOS-detected installed hardware:
Bit(s) Description (Table M0006)
15-14 number of parallel devices
    00 or 11 sometimes used to indicate four LPT ports
    13 (Convertible, PS/2-55LS) internal modem
    12 game port installed
11-9 number of serial devices
    000 or 111 sometimes used to indicate eight COM ports
     reserved
7-6 number of floppy disk drives (minus 1)
  reserved
5-4 initial video mode
    00 EGA,VGA,PGA, or other with on-board video BIOS
    01 40x25 CGA color
    10 80x25 CGA color
    11 80x25 mono text
3-2 (PC only) RAM on motherboard
    00 = 16K, 01 = 32K, 10 = 48K, 11 = 64K
    (some XTs) RAM on motherboard
    00 = 64K, 01 = 128K, 10 = 192K, 11 = 256K
2 (pre-PS/2 except PC) reserved
    (PS/2, some XT clones, newer BIOSes) pointing device installed
  reserved
1 math coprocessor installed
  reserved
0 floppy disk drives are installed
booted from floppy
--------B-M00400012--------------------------
MEM 0040h:0012h - Convertible - POST STATUS
Size: BYTE
--------B-M00400012--------------------------
MEM 0040h:0012h U - AT - MANUFACTURING TEST INITIALIZATION FLAGS
Size: BYTE

Bitfields for AT manufacturing test initialization flags:
Bit(s) Description (Table M0007)
0 start in manufacturing test mode rather than normal operation
1-7 unused
--------b-M00400012--------------------------
MEM 0040h:0012h - MCA - MANUFACTURING TEST
Size: BYTE

Bitfields for MCA manufacturing test flags:
Bit(s) Description (Table M0008)
7 POST flag, ???
6-5 unused
4 POST flag, slot 4 has adapter identifier EDAFh
3 POST flag, 80x25 color video
2 POST flag, ???
1 unused
0 manufacturing test mode rather than normal operation
--------b-M00400012--------------------------
MEM 0040h:0012h - PS/2 Model 25 - POST SYSTEM FLAG
Size: BYTE

Bitfields for PS/2 Model 25 POST system flag:
Bit(s) Description (Table M0009)
0 optional memory failed; memory remapped
1 real-time clock installed
--------B-M00400013--------------------------
MEM 0040h:0013h - BASE MEMORY SIZE IN KBYTES
Size: WORD
SeeAlso: INT 12
--------b-M00400015--------------------------
MEM 0040h:0015h - PC, XT - ADAPTER MEMORY SIZE IN KBYTES
Size: WORD
--------b-M00400015--------------------------
MEM 0040h:0015h U - AT - MANUFACTURING TEST SCRATCH PAD
Size: BYTE
--------K-M00400015--------------------------
MEM 0040h:0015h - Compaq Deskpro 386 - PREVIOUS SCAN CODE
Size: BYTE
--------b-M00400016--------------------------
MEM 0040h:0016h U - AT - MANUFACTURING TEST SCRATCH PAD
Size: BYTE
--------b-M00400016--------------------------
MEM 0040h:0016h - PS/2 Model 30 - BIOS CONTROL FLAGS
Size: BYTE
--------K-M00400016--------------------------

MEM 0040h:0016h - Compaq Deskpro 386 - KEYCLICK VOLUME
Size: BYTE
Range: 00h-7Fh
--------K-M00400017--------------------------

MEM 0040h:0017h - KEYBOARD - STATUS FLAGS 1
Size: BYTE
SeeAlso: MEM 0040h:0018h,INT 16/AH=02h,MEM 0040h:0096h

Bitfields for keyboard status flags 1:

Bit(s)  Description (Table M0010)
7      INSert active
6      Caps Lock active
5      Num Lock active
4      Scroll Lock active
3      either Alt pressed
2      either Ctrl pressed
1      Left Shift pressed
0      Right Shift pressed

SeeAlso: #M0011,#00587
--------K-M00400018--------------------------

MEM 0040h:0018h - KEYBOARD - STATUS FLAGS 2
Size: BYTE
SeeAlso: MEM 0040h:0017h,INT 16/AH=12h

Bitfields for keyboard status flags 2:

Bit(s)  Description (Table M0011)
7      INSert pressed
6      Caps Lock pressed
5      Num Lock pressed
4      Scroll Lock pressed
3      Pause state active
2      Sys Req pressed
1      Left Alt pressed
0      Left Ctrl pressed

SeeAlso: #M0010,#00588
--------K-M00400019--------------------------

MEM 0040h:0019h - KEYBOARD - ALT-NNN KEYPAD WORKSPACE
Size: BYTE
Desc: holds the current value of an Alt-NNN keypad sequence; when Alt is released and this byte is non-zero, the appropriate character is placed in the keyboard buffer
SeeAlso: INT 16/AH=00h,MEM 0040h:001Ah
--------K-M0040001A--------------------------

MEM 0040h:001Ah - KEYBOARD - POINTER TO NEXT CHARACTER IN KEYBOARD BUFFER
Size: WORD
MEM 0040h:001Ch - KEYBOARD - POINTER TO FIRST FREE SLOT IN KEYBOARD BUFFER
Size: WORD
SeeAlso: MEM 0040h:001Ah,MEM 0040h:001Eh,MEM 0040h:0080h,MEM 0040h:0082h
SeeAlso: INT 16/AH=00h

MEM 0040h:001Eh - KEYBOARD - DEFAULT KEYBOARD CIRCULAR BUFFER
Size: 16 WORDs
SeeAlso: MEM 0040h:001Ah,MEM 0040h:001Ch,MEM 0040h:0080h,MEM 0040h:0082h
SeeAlso: INT 16/AH=00h,INT 16/AH=05h

MEM 0040h:003Eh - DISKETTE - RECALIBRATE STATUS
Size: BYTE
SeeAlso: MEM 0040h:003Fh,MEM 0040h:0040h,INT 13/AH=00h,INT 13/AH=11h

Bitfields for diskette recalibrate status:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description (Table M0012)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>diskette hardware interrupt occurred</td>
</tr>
<tr>
<td>6-4</td>
<td>reserved</td>
</tr>
<tr>
<td>3</td>
<td>recalibrate diskette 3 (PC,XT only)</td>
</tr>
<tr>
<td>2</td>
<td>recalibrate diskette 2 (PC,XT only)</td>
</tr>
<tr>
<td>1</td>
<td>recalibrate diskette 1</td>
</tr>
<tr>
<td>0</td>
<td>recalibrate diskette 0</td>
</tr>
</tbody>
</table>

MEM 0040h:003Fh - DISKETTE - MOTOR STATUS
Size: BYTE
SeeAlso: MEM 0040h:003Eh,MEM 0040h:0040h

Bitfields for diskette motor status:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description (Table M0013)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>current operation is write or format, rather than read or verify</td>
</tr>
<tr>
<td>6</td>
<td>reserved (DMA enabled on 82077)</td>
</tr>
<tr>
<td>5-4</td>
<td>diskette drive number selected (0-3)</td>
</tr>
<tr>
<td>3</td>
<td>diskette 3 motor on (PC,XT only)</td>
</tr>
<tr>
<td>2</td>
<td>diskette 2 motor on (PC,XT only)</td>
</tr>
<tr>
<td>1</td>
<td>diskette 1 motor on</td>
</tr>
<tr>
<td>0</td>
<td>diskette 0 motor on</td>
</tr>
</tbody>
</table>

MEM 0040h:0040h - DISKETTE - MOTOR TURN-OFF TIMEOUT COUNT
Size: BYTE
Desc: number of clock ticks until diskette motor is turned off
Note: the typical implementation of the timeout is to have the INT 08 handler decrement this byte on every clock tick, and force the diskette motor off if the result is equal to zero
SeeAlso: MEM 0040h:003Eh,MEM 0040h:003Fh,MEM 0040h:0041h,INT 08"IRQ0"
Size: BYTE
SeeAlso: MEM 0040h:003Eh,MEM 0040h:0042h,INT 13/AH=01h

Bitfields for diskette last operation status:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description (Table M0014)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>drive not ready</td>
</tr>
<tr>
<td>6</td>
<td>seek error</td>
</tr>
<tr>
<td>5</td>
<td>general controller failure</td>
</tr>
<tr>
<td>4-0</td>
<td>error reason</td>
</tr>
<tr>
<td>00h</td>
<td>no error</td>
</tr>
<tr>
<td>01h</td>
<td>invalid request/parameter</td>
</tr>
<tr>
<td>02h</td>
<td>address mark not found</td>
</tr>
<tr>
<td>03h</td>
<td>write-protect error</td>
</tr>
<tr>
<td>04h</td>
<td>sector not found</td>
</tr>
<tr>
<td>06h</td>
<td>diskette change line active</td>
</tr>
<tr>
<td>08h</td>
<td>DMA overrun</td>
</tr>
<tr>
<td>09h</td>
<td>DMA across 64k boundary</td>
</tr>
<tr>
<td>0Ch</td>
<td>media type unknown</td>
</tr>
<tr>
<td>10h</td>
<td>CRC error on read</td>
</tr>
</tbody>
</table>

Note: the following values for this byte differ somewhat from the bitfield definition above:

- 30h drive does not support media sense
- 31h no media in drive
- 32h drive does not support media type
- AAh diskette drive not ready

--------B-M00400042--------------------------

MEM 0040h:0042h - DISK - FLOPPY/HARD DRIVE STATUS/COMMAND BYTES
Size: 7 BYTEs
SeeAlso: MEM 0040h:0041h

42h BYTE  XT: command byte to hard disk controller
          AT: write precompensation cylinder number / 4
43h BYTE  XT: bit 5 = drive number, bits 3-0=head number
          AT: sector count
44h BYTE  XT: bits 6,7 = high bits of track, bits 5-0 = start sector-1
          AT: starting sector
45h BYTE  low byte of track number
46h BYTE  XT: sector count
          AT: high bits of track number
47h BYTE  XT: controlbyte from HD parameters (step rate,...)
          AT: 101DHHHH, D=drive number, HHHH=head number
          bit 7 = ECC mode (1)
          bit 6 = unknown (0)
          bit 5 = 512 byte sectors (1)
          bit 4 = drive number
          bit 3-0 head number
48h BYTE  XT: INT 13h subfunction number
          AT: command byte to hard disk controller
Bitfields for diskette controller status register 0:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-6</td>
<td>interrupt code</td>
</tr>
<tr>
<td>7</td>
<td>normal completion</td>
</tr>
<tr>
<td>6</td>
<td>abnormal termination during execution</td>
</tr>
<tr>
<td>5</td>
<td>invalid command</td>
</tr>
<tr>
<td>4</td>
<td>abnormal termination: ready line on/diskette change</td>
</tr>
<tr>
<td>3</td>
<td>requested seek complete</td>
</tr>
<tr>
<td>2</td>
<td>drive fault</td>
</tr>
<tr>
<td>1-0</td>
<td>head state at time of interrupt</td>
</tr>
<tr>
<td>1</td>
<td>selected drive (drives 2&amp;3 on PC,XT only)</td>
</tr>
</tbody>
</table>

Bitfields for diskette controller status register 1:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>attempted access beyon last cylinder</td>
</tr>
<tr>
<td>6</td>
<td>CRC error on read</td>
</tr>
<tr>
<td>5</td>
<td>DMA overrun</td>
</tr>
<tr>
<td>4</td>
<td>unused</td>
</tr>
<tr>
<td>3</td>
<td>data error</td>
</tr>
<tr>
<td>2</td>
<td>disk write protected</td>
</tr>
<tr>
<td>1</td>
<td>missing address mark</td>
</tr>
<tr>
<td>0</td>
<td>verify equal</td>
</tr>
</tbody>
</table>

Bitfields for diskette controller status register 2:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>unused</td>
</tr>
<tr>
<td>6</td>
<td>found deleted data address mark</td>
</tr>
<tr>
<td>5</td>
<td>CRC error in data field</td>
</tr>
<tr>
<td>4</td>
<td>wrong cylinder number read</td>
</tr>
<tr>
<td>3</td>
<td>verify equal</td>
</tr>
<tr>
<td>2</td>
<td>can't find sector matching verify condition</td>
</tr>
</tbody>
</table>
1 bad cylinder
0 unable to find address mark
SeeAlso: #M0016
--------V-M00400049--------------------------
MEM 0040h:0049h - VIDEO - CURRENT VIDEO MODE
Size: BYTE
SeeAlso: MEM 0040h:004Ah,INT 10/AH=00h
--------V-M0040004A--------------------------
MEM 0040h:004Ah - VIDEO - COLUMNS ON SCREEN
Size: WORD
SeeAlso: MEM 0040h:0049h,MEM 0040h:004Ch,MEM 0040h:004Eh,INT 10/AH=0Fh
--------V-M0040004C--------------------------
MEM 0040h:004Ch - VIDEO - PAGE (REGEN BUFFER) SIZE IN BYTES
Size: WORD
SeeAlso: MEM 0040h:0049h,MEM 0040h:004Eh,MEM 0040h:0050h
--------V-M0040004E--------------------------
MEM 0040h:004Eh - VIDEO - CURRENT PAGE START ADDRESS IN REGEN BUFFER
Size: WORD
SeeAlso: MEM 0040h:004Ch,MEM 0040h:0050h,MEM 0040h:0062h,INT 10/AH=05h
--------V-M00400050--------------------------
MEM 0040h:0050h - VIDEO - CURSOR POSITIONS
Size: 8 WORDs
Desc: contains row and column position for the cursors on each of eight video pages
SeeAlso: MEM 0040h:0049h,MEM 0040h:0060h,INT 10/AH=02h
--------V-M00400060--------------------------
MEM 0040h:0060h - VIDEO - CURSOR TYPE
Size: WORD (big-endian)
Desc: contains cursor start scan line and cursor end scan line
SeeAlso: MEM 0040h:0050h,MEM 0040h:0062h,INT 10/AH=03h
--------V-M00400062--------------------------
MEM 0040h:0062h - VIDEO - CURRENT PAGE NUMBER
Size: BYTE
SeeAlso: MEM 0040h:0049h,MEM 0040h:0063h,INT 10/AH=05h
--------V-M00400063--------------------------
MEM 0040h:0063h - VIDEO - CRT CONTROLLER BASE I/O PORT ADDRESS
Size: WORD
Note: normally 03B4h for mono and 03D4h for color video boards
SeeAlso: MEM 0040h:0063h,MEM 0040h:0066h
--------V-M00400066--------------------------
MEM 0040h:0066h - VIDEO - CURRENT MODE SELECT REGISTER
Size: BYTE
Desc: contains last value written to I/O port 03B8h / 03D8h
SeeAlso: MEM 0040h:0063h,MEM 0040h:0066h

Bitfields for current video mode select register:
Bit(s) Description (Table M0018)
7-6 unused
5 attribute bit 7 controls blinking instead of background
4 mode 6 graphics in monochrome
3 video signal enabled
2 monochrome
1 graphics
0 80x25 text

---V-M00400066--------------------------
MEM 0040h:0066h - VIDEO - CURRENT SETTING OF CGA PALETTE REGISTER
Size: BYTE
Desc: contains the last value written to I/O port 03D9h
SeeAlso: MEM 0040h:0063h,MEM 0040h:0065h,INT 10h/AH=0Bh/BH=01h

Bitfields for CGA palette register:
Bit(s) Description (Table M0019)
7-6 unused
5 palette (0/1)
4 intense background colors in text mode
3 intense border color (40x25) / background color (mode 5)
2 red
1 green
0 blue

---M00400067--------------------------
MEM 0040h:0067h - PC only - CASSETTE TIME COUNT
Size: WORD
SeeAlso: INT 15/AH=00h

---M00400067--------------------------
MEM 0040h:0067h - RESET RESTART ADDRESS
Size: DWORD
Desc: this address stores the address at which to resume execution after a
CPU reset (or jump to F000h:FFF0h) when certain magic values are
stored at 0040h:0072h or in CMOS RAM location 0Fh
SeeAlso: MEM 0040h:0072h,MEM F000h:FFF0h,CMOS 0Fh,INT 19

---M00400069--------------------------
MEM 0040h:0069h - CASSETTE (PC only) - CASSETTE CRC REGISTER
Size: WORD
SeeAlso: MEM 0040h:006Bh"CASSETTE",INT 15/AH=02h

---M00400069--------------------------
MEM 0040h:0069h - V20-XT-BIOS - KEY REPEAT
Size: BYTE

Bitfields for V20-XT-BIOS key repeat flags:
Bit(s) Description (Table M0020)
7 key repeat disabled
6 Ctrl-Alt pressed instead of just Alt

---M0040006B--------------------------
MEM 0040h:006Bh - CASSETTE (PC only) - LAST VALUE READ FROM CASSETTE
Size: BYTE
SeeAlso: MEM 0040h:0069h"CASSETTE",INT 15/AH=02h
MEM 0040h:006Bh - POST LAST UNEXPECTED INTERRUPT (XT and later)
Size: BYTE
Desc: this is a bitmask of IRQs which have occurred while the corresponding interrupt vector points at the default system BIOS handler
(bit 0 = IRQ0 to bit 7 = IRQ7; bit 2 = IRQ8-15 on AT and later)
SeeAlso: INT 0F"IRQ7",INT 70"IRQ8",INT 77"IRQ15"

MEM 0040h:006Ch - TIMER TICKS SINCE MIDNIGHT
Size: DWORD
Desc: updated approximately every 55 milliseconds by the BIOS INT 08 handler
SeeAlso: MEM 0040h:0070h,INT 08"IRQ0",INT 1A/AH=00h

MEM 0040h:0070h - TIMER OVERFLOW
Size: BYTE
Desc: non-zero if timer has counted past midnight since last call to INT 1A/AH=00h
Note: the original IBM BIOS, and thus most other BIOSes, sets this byte to 01h at midnight; a few (such as the Eagle PC-2) increment it each time midnight is passed. The former behavior results in lost days if multiple midnights pass between "get-time" calls while the machine is powered up.
SeeAlso: MEM 0040h:006Ch,INT 1A/AH=00h

MEM 0040h:0071h - Ctrl-Break FLAG
Size: BYTE
Desc: bit 7 is set when Ctrl-Break has been pressed
SeeAlso: INT 1B

MEM 0040h:0072h - POST RESET FLAG
Size: WORD
Desc: specify the action the BIOS should take at the beginning of the power-on self-test when the machine is reset
SeeAlso: INT 19,MEM F000h:FFF0h

(Table M0021)
Values for POST reset flag:
0000h cold boot
0064h Burn-in mode
1234h to bypass memory test (warm boot)
4321h [PS/2 except Mod 25,30] to preserve memory
5678h [Conv] system suspended
9ABCh [Conv] manufacturing test mode
ABCDh [Conv] POST loop mode

MEM 0040h:0074h - FIXED DISK LAST OPERATION STATUS (except ESDI drives)
Size: BYTE
SeeAlso: INT 13/AH=01h,INT 13h/AH=0Ah,MEM 0040h:0041h
Values for fixed disk last operation status:

00h  no error
01h  invalid function request
02h  address mark not found
03h  write protect error
04h  sector not found
05h  reset failed
06h  diskette removed
07h  drive parameter activity failed
08h  DMA overrun
09h  DMA data boundary error
0Ah  bad sector flag detected
0Bh  bad track detected
0Ch  requested diskette media type not found
      (PS/2 or extended BIOS only) unsupported track
0Dh  invalid number of sectors for Format
0 Eh  control data address mark detected
0Fh  DMA arbitration level out of range
10h  uncorrectable ECC or CRC error
11h  ECC corrected data error
20h  general controller failed
40h  seek failed
80h  time out
AAh  drive not ready
B0h  volume not locked in drive (INT 13 extensions)
B1h  volume locked in drive (INT 13 extensions)
B2h  volume not removable (INT 13 extensions)
B3h  volume in use (INT 13 extensions)
B4h  lock count exceeded (INT 13 extensions)
B5h  valid eject request failed (INT 13 extensions)
BBh  undefined error
CCh  write fault on selected drive
E0h  status error/error register is zero
FFh  sense failed

SeeAlso: #00234

MEM 0040h:0074h - WD1002-27X SuperBIOS - TOTAL DRIVES, FIRST CONTROLLER ONLY
Size: BYTE

SeeAlso: MEM 0040h:0075h"SuperBIOS",MEM 0040h:0076h"SuperBIOS"

MEM 0040h:0075h - FIXED DISK - NUMBER OF FIXED DISK DRIVES
Size: BYTE

SeeAlso: MEM 0040h:0076h"FIXED DISK",MEM 0040h:0077h"FIXED DISK"

MEM 0040h:0075h - WD1002-27X SuperBIOS - TOTAL FIXED DRIVES, BOTH CONTROLLERS
Size: BYTE
MEM 0040h:0076h - FIXED DISK - CONTROL BYTE  {IBM documented only for XT}
Size:  BYTE
Desc:  loaded from the disk parameter table control byte (offset 8) during various hard disk operations
SeeAlso:  MEM 0040h:0075h"FIXED DISK",MEM 0040h:0077h"FIXED DISK"

MEM 0040h:0076h - XT: hard disk controller's I/O address (Western Digital)
Size:  BYTE

MEM 0040h:0076h - WD1002-27X SuperBIOS - USED IN TRACK RECALCULATION
Size:  BYTE

MEM 0040h:0077h - FIXED DISK - I/O port offset {IBM documented only for XT}
Size:  BYTE

Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS LPT1=[port_address|logical_no][,[timeout]]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso:  MEM 0040h:0079h,MEM 0040h:007Ah,INT 17/AH=00h

MEM 0040h:0078h - PARALLEL DEVICE 1 TIME-OUT COUNTER
Size:  BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS LPT2=[port_address|logical_no][,[timeout]]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso:  MEM 0040h:0078h,MEM 0040h:0079h,MEM 0040h:007Bh"PARALLEL"

MEM 0040h:007Ah - PARALLEL DEVICE 2 TIME-OUT COUNTER
Size:  BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS LPT3=[port_address|logical_no][,[timeout]]
directive, whereby port_address = 200h..3FCh, logical_no = 0 or 1..3,
timeout=0..255 (default 20).
SeeAlso:  MEM 0040h:0078h,MEM 0040h:0079h,MEM 0040h:007Bh"PARALLEL"
MEM 0040h:007Bh - PARALLEL DEVICE 4 TIME-OUT COUNTER (pre-PS, PS Models 25,30)
Size: BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS LPT4=(port_address|logical_no)[,[timeout]] directive, where port_address = 200h..3FCh, logical_no = 0 or 1..3, timeout=0..255 (default 20). To avoid any interference with the PS/2 and later interpretation, this will be rejected if this entry does not hold 0, which would indicate it is used for different purposes.
SeeAlso: MEM 0040h:0078h,MEM 0040h:007Ah,MEM 0040h:007Bh"INT 4Bh"

--------m-M0040007B--------------------------
MEM 0040h:007Bh - INT 4Bh FLAGS (PS2 and newer)
Size: BYTE
SeeAlso: INT 4B/AX=8102h

Bitfields for INT 4Bh flags:
Bit(s) Description (Table M0023)
7-6 reserved
5 set if Virtual DMA Spec supported [PS] (see INT 4B)
4 reserved
3 set if INT 4Bh intercepted and must be chained
2 reserved
1 set if Generic SCSI CBIOS services available on INT 4Bh
0 reserved

--------B-M0040007C--------------------------
MEM 0040h:007Ch - SERIAL DEVICE 1 TIMEOUT COUNTER
Size: BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS COM1=[port_address|logical_no][,[timeout]] directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4, timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0000h,MEM 0040h:007Dh,MEM 0040h:007Eh,MEM 0040h:007Fh
SeeAlso: INT 14/AH=01h

--------B-M0040007D--------------------------
MEM 0040h:007Dh - SERIAL DEVICE 2 TIMEOUT COUNTER
Size: BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS COM2=[port_address|logical_no][,[timeout]] directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4, timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0002h,MEM 0040h:007Ch,MEM 0040h:007Eh,MEM 0040h:007Fh
SeeAlso: INT 14/AH=01h

--------B-M0040007E--------------------------
MEM 0040h:007Eh - SERIAL DEVICE 3 TIMEOUT COUNTER
Size: BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the undocumented CONFIG.SYS COM3=[port_address|logical_no][,[timeout]] directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4, timeout=0..255 (default 1).
714 A to Z of C

SeeAlso: MEM 0040h:0004h,MEM 0040h:007Ch,MEM 0040h:007Dh,MEM 0040h:007Fh
SeeAlso: INT 14/AH=01h
--------B-M0040007F--------------------------
MEM 0040h:007Fh - SERIAL DEVICE 4 TIMEOUT COUNTER
Size: BYTE
Note: Under DR-OpenDOS 7.02+ this setting can be changed with the
    undocumented CONFIG.SYS COM4=[port_address|logical_no][,[timeout]]
    directive, whereby port_address = 200h..3F8h, logical_no = 0 or 1..4,
    timeout=0..255 (default 1).
SeeAlso: MEM 0040h:0006h,MEM 0040h:007Ch,MEM 0040h:007Dh,MEM 0040h:007Eh
SeeAlso: INT 14/AH=01h
--------K-M00400080--------------------------
MEM 0040h:0080h - KEYBOARD BUFFER START OFFSET FROM SEGMENT 40h (normally 1Eh)
Size: WORD
SeeAlso: MEM 0040h:001Ah,MEM 0040h:001Eh,MEM 0040h:0082h,INT 16/AH=05h
--------K-M00400082--------------------------
MEM 0040h:0082h - KEYBOARD BUFFER END+1 OFFSET FROM SEGMENT 40h (normally 3Eh)
Size: WORD
Note: XT BIOS dated 11/08/82 ends here
SeeAlso: MEM 0040h:001Ch,MEM 0040h:003Eh,MEM 0040h:0080h,INT 16/AH=05h
--------V-M00400084--------------------------
MEM 0040h:0084h - VIDEO (EGA/MCGA/VGA) - ROWS ON SCREEN MINUS ONE
Size: BYTE
SeeAlso: MEM 0040h:0085h,INT 10/AX=1100h
--------V-M00400085--------------------------
MEM 0040h:0085h - VIDEO (EGA/MCGA/VGA) - CHARACTER HEIGHT IN SCAN-LINES
Size: WORD
SeeAlso: MEM 0040h:0084h,INT 10"LIRVGA19"
!!!
--------V-M00400087--------------------------
MEM 0040h:0087h - VIDEO (EGA/VGA) CONTROL: [MCGA: =00h]
Size: BYTE
SeeAlso: MEM 0040h:0084h,MEM 0040h:0085h,MEM 0040h:0088h
Bitfields for EGA/VGA Video control flags:
  Bit(s) Description (Table M0024)
  7 do not to clear RAM on mode set (see INT 10h, AH=00h)
  6-5 RAM on adapter = (this field + 1) * 64K
  4 reserved
  3 EGA/VGA video system INactive
  2 wait for display enable
  1 mono monitor
  0 alphanumeric cursor emulation DISabled
    When enabled, text mode cursor size (INT 10,AH=01h) settings looking
    like CGA ones are translated to equivalent EGA/VGA ones.
--------V-M00400088--------------------------
MEM 0040h:0088h - VIDEO (EGA/VGA) SWITCHES: [MCGA: reserved]
Size: BYTE
SeeAlso: MEM 0040h:0084h,MEM 0040h:0085h,MEM 0040h:0088h
SeeAlso: MEM 0040h:0087h,MEM 0040h:0089h

Bitfields for EGA/VGA Video switches:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
<th>(Table M0025)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-4</td>
<td>power-on state of feature connector bits 3-0</td>
<td></td>
</tr>
<tr>
<td>3-0</td>
<td>configuration switches 4-1 (=0 on, =1 off) (see #M0026)</td>
<td></td>
</tr>
</tbody>
</table>

Note: when bit 4 of 0040h:0089h is 0, VGA emulates 350-line EGA if this byte is x3h or x9h, otherwise emulates 200-line CGA in 400-line double scan. VGA resets this byte to x9h after the mode set.
See also note for 0040h:0089h.

(Table M0026)

Values for EGA/VGA configuration switches:

- 00h Pri MDA, Sec EGA+old color display 40 x 25
- 01h Pri MDA, Sec EGA+old color display 80 x 25
- 02h Pri MDA, Sec EGA+ECD normal mode (CGA emul)
- 03h Pri MDA, Sec EGA+ECD enhanced mode
- 04h Pri CGA 40 x 25, Sec EGA mono display
- 05h Pri CGA 80 x 25, Sec EGA mono display
- 06h Pri EGA+old color display 40 x 25, Sec MDA
- 07h Pri EGA+old color display 80 x 25, Sec MDA
- 08h Pri EGA+ECD normal mode (CGA emul), Sec MDA
- 09h Pri EGA+ECD enhanced mode, Sec MDA
- 0Ah Pri EGA mono display, Sec CGA 40 x 25
- 0Bh Pri EGA mono display, Sec CGA 80 x 25

SeeAlso: #M0025

--------b-M00400088--------------------------
MEM 0040h:0088h - Olivetti EGA capabilities??
Size: BYTE???

Bitfields for Olivetti EGA capabilities flags:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
<th>(Table M0130)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>640x400 mode related???</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>unknown</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>640x400 mode related???</td>
<td></td>
</tr>
<tr>
<td>4-0</td>
<td>unknown</td>
<td></td>
</tr>
</tbody>
</table>

Note: To decide if the 640x400 mode is supported by an Olivetti EGA card (only the Olivetti EGA card 2 supports it), also check that bit 7 and 5 are set.
SeeAlso: C000h:0000h"Olivetti"

--------V-M00400089--------------------------
MEM 0040h:0089h U - VIDEO (MCGA/VGA) - MODE-SET OPTION CONTROL
Size: BYTE
SeeAlso: MEM 0040h:0087h,MEM 0040h:0088h

Bitfields for Video mode-set option control:

<table>
<thead>
<tr>
<th>Bit(s)</th>
<th>Description</th>
<th>(Table M0027)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7,4</td>
<td>requested scan lines</td>
<td></td>
</tr>
</tbody>
</table>
0 0  350-line mode requested
0 1  400-line mode at next mode set
1 0  200-line mode requested
1 1  reserved
Note:  Apparently VGA BIOS mode set disregards bit 7 and uses byte 40h:88h to determine 200/350 selection when bit 4 is zero. Presumably bit 7 is a convenience for other purposes. Bit 7 is reset to zero after the mode set.

6  display switching enabled
5  reserved
4  if set:   use 400-line mode at next mode set
if clear: [VGA] emulate EGA at next mode set
[MCGA] emulate CGA, digital monitor, 200 lines, 8x8 text
Note: this bit is set by the video mode set on VGA, unchanged on MCGA
3  default palette loading DISabled at mode set
2  mono display
1  gray scale summing enabled
0  [VGA] =1 if VGA active, =0 if not
[MCGA] reserved, zero
Note: the Tseng ET4000 BIOS v3.00 uses bits 6-4 of 88h and bits 6-5 of 89h to specify graphics-mode refresh rates as follows
88h/6  640x480:  1 for 72Hz,0 for 60Hz
88h/5+89h/6  800x600:  00  60Hz
               01  56Hz
               10  72Hz

88h/4+89h/5  1024x768:  00  interlaced
               01  60Hz
               10  72Hz??
               11  70Hz

--------V-M0040008A--------------------------
MEM 0040h:008Ah U - VIDEO (MCGA/VGA) - INDEX INTO DISPLAY COMBINATION CODE TBL
Size:  BYTE
SeeAlso: INT 10/AX=1A00h,#M0039
--------*-M0040008B--------------------------
MEM 0040h:008Bh - PC, PCjr, PC/XT 11/8/82, Convertible - RESERVED
Size:  11 BYTES
--------B-M0040008B--------------------------
MEM 0040h:008Bh - DISKETTE MEDIA CONTROL
Size:  BYTE

Bitfields for diskette media control:
Bit(s)  Description  (Table M0028)
7-6  last data rate set by controller
     00=500kbps, 01=300kbps, 10=250kbps, 11=1Mbps
5-4  last diskette drive step rate selected
     00=0Ch, 01=0Dh, 10=0Eh, 11=0Ah
3-2  {data rate at start of operation}
1-0  reserved
Note: EHD BIOS sets this byte to 01h and never reads it back

---B-M0040008C--------------------------
MEM 0040h:008Ch - FIXED DISK - CONTROLLER STATUS [not XT]
Size: BYTE
SeeAlso: MEM 0040h:008Dh,MEM 0040h:008Eh

---B-M0040008D--------------------------
MEM 0040h:008Dh - FIXED DISK - CONTROLLER ERROR STATUS [not XT]
Size: BYTE
SeeAlso: MEM 0040h:008Ch,MEM 0040h:008Eh

---B-M0040008E--------------------------
MEM 0040h:008Eh - FIXED DISK - INTERRUPT CONTROL [not XT]
Size: BYTE
Note: cleared to 00h at start of disk operation, set to FFh by IRQ14 handler when hard disk controller completes command
SeeAlso: MEM 0040h:008Ch,MEM 0040h:008Dh,MEM 0040h:008Fh

---B-M0040008F--------------------------
MEM 0040h:008Fh - DISKETTE CONTROLLER INFORMATION [not XT]
Size: BYTE
SeeAlso: MEM 0040h:008Ch,MEM 0040h:008Dh,MEM 0040h:008Eh

Bitfields for diskette controller information:
Bit(s) Description (Table M0029)
7 reserved
6 =1 drive 1 determined
5 =1 drive 1 is multi-rate, valid if drive determined
4 =1 drive 1 supports 80 tracks, always valid
3 reserved
2 =1 drive 0 determined
1 =1 drive 0 is multi-rate, valid if drive determined
0 =1 drive 0 supports 80 tracks, always valid
Note: EHD BIOS sets this byte to 01h and never alters it again

---B-M00400090--------------------------
MEM 0040h:0090h - DISKETTE DRIVE 0 MEDIA STATE
Size: BYTE
SeeAlso: MEM 0040h:0091h

Bitfields for diskette drive media state:
Bit(s) Description (Table M0030)
7-6 data rate
00=500kbps, 01=300kbps, 10=250kbps, 11=1Mbps
5 double stepping required (e.g. 360kB in 1.2MB)
4 media type established
3 drive capable of supporting 4MB media
2-0 on exit from BIOS, contains
000 trying 360kB in 360kB
001 trying 360kB in 1.2MB
010 trying 1.2MB in 1.2MB
011 360kB in 360kB established
100 360kB in 1.2MB established
101 1.2MB in 1.2MB established
110 reserved
111 all other formats/drives

SeeAlso: #M0031,#M0032

--------B-M00400091--------------------------
MEM 0040h:0091h - DISKETTE DRIVE 1 MEDIA STATE
Size: BYTE
SeeAlso: MEM 0040h:0090h,#M0030

--------B-M00400092--------------------------
MEM 0040h:0092h U - DISKETTE DRIVE 0 MEDIA STATE AT START OF OPERATION
Size: BYTE
Note: officially "Drive 2 media state"
SeeAlso: MEM 0040h:0093h"DRIVE 1"

Bitfields for diskette drive 0 media state at start of operation:
Bit(s) Description (Table M0031)
7-3 (see #M0030)
2 multiple data rate capability determined
1 multiple data rate capability
0 =1 if drive has 80 tracks, =0 if 40 tracks

SeeAlso: #M0030,#M0032

--------d-M00400092--------------------------
MEM 0040h:0092h - Olivetti Quaderno - HARD DISK POWERDOWN COUNTDOWN CLOCK TICKS
Size: BYTE
Note: hard disk is turned off when counter reaches zero

--------B-M00400093--------------------------
MEM 0040h:0093h U - DISKETTE DRIVE 1 MEDIA STATE AT START OF OPERATION
Size: BYTE
Note: officially "Drive 3 media state"
SeeAlso: MEM 0040h:0092h"DRIVE 1"

Bitfields for diskette drive 1 media state at start of operation:
Bit(s) Description (Table M0032)
7-3 (see #M0030)
2 multiple data rate capability determined
1 multiple data rate capability
0 =1 if drive has 80 tracks, =0 if 40 tracks

--HP 100LX/200LX--
display control status
0 =1 if DISPCTL -K
1 =1 if DISPCTL -C

--------B-M00400094--------------------------
MEM 0040h:0094h - DISKETTE DRIVE 0 CURRENT TRACK NUMBER
Size: BYTE
SeeAlso: MEM 0040h:0095h

--------B-M00400095--------------------------
MEM 0040h:0095h - DISKETTE DRIVE 1 CURRENT TRACK NUMBER
Size: BYTE
SeeAlso: MEM 0040h:0094h

--------K-M00400096-------------------------------
MEM 0040h:0096h - KEYBOARD STATUS BYTE 1
Size: BYTE
SeeAlso: MEM 0040h:0097h, INT 16/AH=11h

Bitfields for keyboard status byte 1:
Bit(s)  Description (Table M0033)
7  =1 read-ID in progress
6  =1 last code read was first of two ID codes
5  =1 force Num Lock if read-ID and enhanced keyboard
4  =1 enhanced keyboard installed
3  =1 Right Alt pressed
2  =1 Right Ctrl pressed
1  =1 last code read was E0h
0  =1 last code read was E1h
SeeAlso: #M0034, #M0010

--------K-M00400097-------------------------------
MEM 0040h:0097h - KEYBOARD STATUS BYTE 2
Size: BYTE
SeeAlso: MEM 0040h:0096h, INT 16/AH=11h

Bitfields for keyboard status byte 2:
Bit(s)  Description (Table M0034)
7  =1 keyboard transmit error flag
6  =1 LED update in progress
5  =1 RESEND received from keyboard
4  =1 ACK received from keyboard
3  reserved, must be zero
2  Caps Lock LED
1  Num Lock LED
0  Scroll Lock LED
SeeAlso: #M0033, #M0010

--------B-M00400098-------------------------------
MEM 0040h:0098h - TIMER2 (AT, PS exc Mod 30) - PTR TO USER WAIT-COMPLETE FLAG
Size: DWORD
Note: (see INT 15/AX=8300h)
SeeAlso: MEM 0040h:009Ch, INT 15/AH=83h, INT 15/AH=86h

--------B-M00400099-------------------------------
MEM 0040h:0099h - TIMER2 (AT, PS exc Mod 30) - USER WAIT COUNT IN MICROSECONDS
Size: DWORD
SeeAlso: MEM 0040h:0098h, MEM 0040h:00A0h, INT 15/AH=83h, INT 15/AH=86h

--------V-M0040009F-------------------------------
MEM 0040h:009Fh - HP 100LX/200LX - VIDEO ZOOM MODE
Size: BYTE
(Table M0035)
Values for HP 100LX/200LX video zoom mode:
02h  80x25 mono
03h  80x25 color
80h  64x18 mono
81h  64x18 color
82h  40x25 mono
83h  40x25 color
84h  40x16 mono
85h  40x16 color
SeeAlso: INT 10/AH=D0h

--------B-M004000A0--------------------------
MEM 0040h:00A0h - TIMER2 (AT, PS exc Mod 30) - WAIT ACTIVE FLAG
Size:   BYTE
SeeAlso: MEM 0040h:009Ch,INT 15/AH=83h,INT 15/AH=86h

Bitfields for Timer2 wait active flag:
Bit(s)  Description           (Table M0036)
  7    wait time elapsed
  6-1  reserved
  0    INT 15/AH=86h has occurred

--------N-M004000A1--------------------------
MEM 0040h:00A1h - BIT 5 SET IF LAN SUPPORT PROGRAM INTERRUPT ARBITRATOR PRESENT
Size:   BYTE
Note:   DEVICE=DXMA0MOD.SYS

--------N-M004000A2--------------------------
MEM 0040h:00A2h - RESERVED FOR NETWORK ADAPTERS
Size:   6 BYTES

--------d-M004000A4--------------------------
MEM 0040h:00A4h - PS/2 Mod 30 - SAVED FIXED DISK INTERRUPT VECTOR
Size:   DWORD

--------V-M004000A8--------------------------
MEM 0040h:00A8h - VIDEO (EGA/MCGA/VGA) - POINTER TO VIDEO SAVE POINTER TABLE
Size:   DWORD
SeeAlso: INT 10/AH=1Ch

Format of Video Save Pointer Table [EGA/VGA/MCGA only]:
Offset  Size  Description   (Table M0037)
  00h  DWORD  ptr to Video Parameter Table
  04h  DWORD  ptr to Parameter Dynamic Save Area, else 0 [EGA/VGA only]
  08h  DWORD  ptr to Alphanumeric Character Set Override, else 0
  0Ch  DWORD  ptr to Graphics Character Set Override, else 0
  10h  DWORD  [VGA only] ptr to Secondary Save Pointer Table, must be valid
  14h  DWORD  reserved, zero
  18h  DWORD  reserved, zero
Note: table initially in ROM, copy to RAM to alter, then update 40h:A8h.

Format of Secondary Video Save Pointer Table [VGA only]:
Offset  Size  Description  (Table M0038)
00h  WORD  Length of this table in bytes, including this word (1Ah)
02h  DWORD  ptr to Display Combination Code Table, must be valid
06h  DWORD  ptr to second Alphanumeric Character Set Override, else 0
0Ah  DWORD  ptr to User Palette Profile Table, else 0
0Eh  DWORD  reserved, zero
12h  DWORD  reserved, zero
16h  DWORD  reserved, zero
Note: table initially in ROM, copy to RAM to alter, then alter Save Ptr Table.

Format of Display Combination Code Table [VGA only]:
Offset  Size  Description  (Table M0039)
00h  BYTE  Number of entries in the DCC table at offset 04h
01h  BYTE  Version number
02h  BYTE  Maximum display type code that can appear in DCC table
03h  BYTE  reserved
04h  2N BYTES  Each pair of bytes gives a valid display combination, one
display type per byte (see #M0040)

(Table M0040)
Values for Display Combination display type:
00h  no display
01h  MDA with mono display
02h  CGA with color display
03h  reserved
04h  EGA with color display
05h  EGA with mono display
06h  Professional Graphics Controller
07h  VGA with mono display
08h  VGA with color display
09h  reserved
0Ah  MCGA with digital color display
0Bh  MCGA with analog mono display
0Ch  MCGA with analog color display
FFh  unrecognised video system
SeeAlso: #M0039

Format of Video Parameter Table [EGA, VGA only]:
Offset  Size  Description  (Table M0041)
00h-03h  Modes 00h-03h in 200-line CGA emulation mode
04h-0Eh  Modes 04h-0Eh
0Fh-10h  Modes 0Fh-10h when only 64kB RAM on adapter
11h-12h  Modes 0Fh-10h when >64kB RAM on adapter
13h-16h  Modes 00h-03h in 350-line mode
17h  VGA Modes 00h or 01h in 400-line mode
18h  VGA Modes 02h or 03h in 400-line mode
19h  VGA Mode 07h in 400-line mode
1Ah-1Ch  VGA Modes 11h-13h
A to Z of C

Note: An array of 23 [EGA] or 29 [VGA] elements, each element being 64 bytes long. Elements appear in the above order.

Format of Video Parameter Table element [EGA, VGA only]:
Offset | Size | Description (Table M0042)
---|---|---
00h | BYTE | Columns on screen (see 40h:4Ah)
01h | BYTE | Rows on screen minus one (see 40h:84h)
02h | BYTE | Height of character in scan lines (see 40h:85h)
03h | WORD | Size of video buffer (see 40h:4Ch)
05h | 4 BYTEs | Values for Sequencer Registers 1-4
09h | BYTE | Value for Miscellaneous Output Register
0Ah | 25 BYTEs | Values for CRTC Registers 00h-18h
23h | 20 BYTEs | Values for Attribute Controller Registers 00h-13h
37h | 9 BYTEs | Values for Graphics Controller Registers 00h-08h

Format of Video Parameter Table [MCGA only] {guesswork from inspection}:
Offset | Size | Description (Table M0043)
---|---|---
- 16 triplet BYTEs of R,G,B DAC info for 16 colors;
- An array of 11 elements, each element being 32 bytes long.
Elements appear in the order:
  Modes 00h,01h in 200-line mode for digital displays
  Modes 00h,01h in 400-line mode for analog displays
  Modes 02h,03h in 200-line mode for digital displays
  Modes 02h,03h in 400-line mode for analog displays
  Modes 04h,05h in 200-line mode for digital displays
  Modes 04h,05h in 400-line mode for analog displays
  Mode 06h in 200-line mode for digital displays
  Mode 06h in 400-line mode for analog displays
  Mode 11h
  Mode 13h in 200-line mode for digital displays
  Mode 13h in 400-line mode for analog displays

Format of Video Parameter Table element [MCGA only]:
Offset | Size | Description (Table M0044)
---|---|---
00h | BYTE | Columns on screen (see 40h:4Ah)
01h | BYTE | Rows on screen minus one (see 40h:84h)
02h | BYTE | Height of character in scan lines (see 40h:85h)
03h | WORD | Size of video buffer (see 40h:4Ch)
05h | WORD | ??? always zero
07h | 21 BYTEs | Video data registers 00h-14h to port 3D5h indexed by 3D4h
1Ch | BYTE | PEL Mask to port 3C6h
1Dh | BYTE | CGA Mode Control to port 3D8h
1 Eh | BYTE | CGA Border Control to port 3D9h
1Fh | BYTE | Extended Mode Control to port 3DDh

Format of Video Parameter Dynamic Save Area [EGA, VGA only]:
Offset | Size | Description (Table M0045)
---|---|---
00h | 16 BYTEs | Last data written to Attribute Contr. Palette Registers 0-15
### Format of Alphanumeric Character Set Override:

<table>
<thead>
<tr>
<th>Offset</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>BYTE</td>
<td>Length in bytes of each character in font table</td>
</tr>
<tr>
<td>01h</td>
<td>BYTE</td>
<td>Character generator RAM bank to load, 0=normal</td>
</tr>
<tr>
<td>02h</td>
<td>WORD</td>
<td>Number of characters in font table, normally 256</td>
</tr>
<tr>
<td>04h</td>
<td>WORD</td>
<td>Code of first character in font table, normally 0</td>
</tr>
<tr>
<td>06h</td>
<td>DWORD</td>
<td>ptr to font table</td>
</tr>
<tr>
<td>0Ah</td>
<td>BYTE</td>
<td>Displayable rows (FFh=use maximum calculated value)</td>
</tr>
<tr>
<td>0Bh</td>
<td>BYTES</td>
<td>Array of mode values to which this font is to pertain</td>
</tr>
<tr>
<td></td>
<td>BYTE</td>
<td>FFh end of array</td>
</tr>
</tbody>
</table>

### Format of Second Alphanumeric Character Set Override:

<table>
<thead>
<tr>
<th>Offset</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>BYTE</td>
<td>Length in bytes of each character in font table</td>
</tr>
<tr>
<td>01h</td>
<td>BYTE</td>
<td>Character generator RAM bank to load, normally non-zero</td>
</tr>
<tr>
<td>03h</td>
<td>DWORD</td>
<td>ptr to font table</td>
</tr>
<tr>
<td>07h</td>
<td>BYTES</td>
<td>Array of mode values to which this font is to pertain</td>
</tr>
<tr>
<td></td>
<td>BYTE</td>
<td>FFh end of array</td>
</tr>
</tbody>
</table>

**Note:** Authorities differ, some say same as first override above, but IBM says it is as shown above.

### Format of Graphics Character Set Override:

<table>
<thead>
<tr>
<th>Offset</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>BYTE</td>
<td>Number of displayable character rows</td>
</tr>
<tr>
<td>01h</td>
<td>WORD</td>
<td>Length in bytes of each character in font table</td>
</tr>
<tr>
<td>03h</td>
<td>DWORD</td>
<td>ptr to font table</td>
</tr>
<tr>
<td>07h</td>
<td>BYTES</td>
<td>Array of mode values to which this font is to pertain</td>
</tr>
<tr>
<td></td>
<td>BYTE</td>
<td>FFh end of array</td>
</tr>
</tbody>
</table>

### Format of User Palette Profile Table [VGA only]:

<table>
<thead>
<tr>
<th>Offset</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>BYTE</td>
<td>Underlining: 01h=enable in all alphanumeric modes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>00h=enable in monochrome alphanumeric modes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FFh=disable in all alphanumeric modes</td>
</tr>
<tr>
<td>01h</td>
<td>BYTE</td>
<td>reserved</td>
</tr>
<tr>
<td>02h</td>
<td>WORD</td>
<td>reserved</td>
</tr>
<tr>
<td>04h</td>
<td>WORD</td>
<td>Number (0-17) of Attribute Controller registers in table</td>
</tr>
<tr>
<td>06h</td>
<td>WORD</td>
<td>Index (0-16) of first Attribute Controller register in table</td>
</tr>
<tr>
<td>08h</td>
<td>DWORD</td>
<td>ptr to table of Attribute Controller registers to override</td>
</tr>
</tbody>
</table>
Table is an array of BYTES.

0Ch  WORD  Number (0-256) of video DAC Color registers in table
0Eh  WORD  Index (0-255) of first video DAC Color register in table
10h  DWORD  ptr to table of video DAC Color registers to override
         Table is ??? triplets ??? of BYTES???
14h  BYTES  array of mode values to which this profile is to pertain
         BYTE  FFh end of array

--------*-M004000AC--------------------------
MEM 0040h:00ACh - RESERVED
Size: 4 BYTES
--------b-M004000B0--------------------------
MEM 0040h:00B0h - Phoenix 386 BIOS 1.10 10a - LOOP COUNT FOR HARD DISK TIMEOUT
Size: BYTE
Desc: number of times a tight software delay loop should be executed to
generate the sub-55ms delays used internally by the BIOS
Note: also used for delaying when beeping due to full keyboard buffer
SeeAlso: MEM 0040h:00ECh"Dell",INT 15/AH=BCh
--------d-M004000B0--------------------------
MEM 0040h:00B0h - PTR TO 3363 OPTICAL DISK DRIVER OR BIOS ENTRY POINT
Size: DWORD
Notes: When 3363 BIOS present, the ASCIZ signature "OPTIC "occurs 3 bytes
         beyond this entry point
         When 3363 BIOS and 3363 File System Driver present, the ASCIZ signature
         "FILE SYSTEM DRIVER" occurs 3 bytes beyond this entry point
--------b-M004000B5--------------------------
MEM 0040h:00B5h - 1988 Phoenix 386 BIOS 1.10 03 - PARAMS FOR TYPE 48 HARD DISK
Size: 16 BYTES
SeeAlso: INT 41,INT 46,MEM 0040h:00C0h"HARD DISK"
--------*-M004000B4--------------------------
MEM 0040h:00B4h - RESERVED
Size: WORD
--------b-M004000B5--------------------------
MEM 0040h:00B5h - Dell 4xxDE
Size: BYTE

Bitfields for Dell 4xxDE flags:
Bit(s)  Description  (Table M0050)
  2  ??? (related to disk drives)
  5  page tables set to allow Weitek addressing in real mode
  6  Weitek math coprocessor present
--------b-M004000B5--------------------------
MEM 0040h:00B5h - Tandy BIOS DATA FLAGS
Size: BYTE
SeeAlso: MEM F000h:C000h

Bitfields for Tandy BIOS data flags:
Bit(s)  Description  (Table M0131)
  0  set if drive A: is 720 Kb
1. set if drive B: is 720 Kb
2-7 unknown
Note: Before checking these bits, the Tandy ROM BIOS ID byte at F000h:C000h should be verified to be equal to 21h.

MEM 0040h:00E5h - Gigabyte AWARD v4.51PG - ASSOC DRIVE NUMS TO PHYS INTERFACES
Size: BYTE
SeeAlso: MEM 0040h:00E5h"AWARD"

Bitfields for drive number/interface mapping:
Bit(s) Description (Table M0129)
7-6 interface for drive 83h (F:)
  00 primary master
  01 primary slave
  10 secondary master
  11 secondary slave
5-4 interface for drive 82h (as for bits 7-6)
3-2 interface for drive 81h (as for bits 7-6)
1-0 interface for drive 80h (C:) (as for bits 7-6)
SeeAlso: #M0128
----------M004000B6--------------------------
MEM 0040h:00B6h - RESERVED FOR POST???
Size: 3 BYTEs
----------M004000B9--------------------------
MEM 0040h:00B9h - ???
Size: 7 BYTEs
--------b-M004000BC--------------------------
MEM 0040h:00BCh - 1993 Phoenix 486 BIOS 1.03 PCI - CPU TYPE/MASK REVISION
Size: WORD
Desc: the high byte contains the CPU type, the low byte the mask revision (stepping level), as reported to the BIOS in DX by the CPU at startup
SeeAlso: INT 15/AH=C9h
--------b-M004000C0--------------------------
MEM 0040h:00C0h - 1988 Phoenix 386 BIOS 1.10 03 - PARAMS FOR TYPE 49 HARD DISK
Size: 16 BYTEs
SeeAlso: INT 41,INT 46,MEM 0040h:00B0h"HARD DISK"
--------*M004000C0--------------------------
MEM 0040h:00C0h - RESERVED
Size: 14 BYTEs
--------K-M004000C2--------------------------
MEM 0040h:00C2h - AMI BIOS 1.00.12.AX1T - KEYBOARD TYPE
Size: WORD
Desc: this word contains an indication of the type of keyboard (controller???) attached to the system
Note: AMI’s APM code checks for 4147h vs. other value (5047h seen on Intel "Plato" motherboard)
SeeAlso: #00586,INT 16/AH=F2h
--------b-M004000CE--------------------------