®

Macintosh Portable

Developer Note

For backlit
Macintosh Portable

®

Developer Note

Contents

Figures and tables

About this note

This developer note describes the backlit Macintosh® Portable computer and points out features that are new and different from those of the original Macintosh Portable computer. This note assumes that you are already familiar with both the functionality and programming requirements of Apple® Macintosh computers, and in particular the original Macintosh Portable computer. If you are unfamiliar with Macintosh computers or would simply like more technical information on the hardware, you may want to obtain copies of related technical manuals as explained in the following section, "Supplemental Reference Documents."

The developer note may contain information or specifications that are still under consideration by Apple Computer, Inc. The primary reason for releasing product information early is to provide the development community with essential product specifications, theory, and application information for the purpose of stimulating work on compatible third-party products.

♦ **Important** This developer note does not constitute a manual and is not complete in its present form. While every attempt has been made to verify the accuracy of the information presented, it is subject to change without notice. ♦

The information in this developer note will be included in future editions of *"Guide to macintosh Family Hardware"* and *"Desiging Cards and Drivers for the Macintosh Family."*

Supplemental reference documents

To supplement the information in this document, you might wish to obtain related documentation such as *Guide to the Macintosh Family Hardware,* Second Edition; *Designing Cards and Drivers for the Macintosh Family,* Second Edition; and *Inside Macintosh,* Volumes I through VI. These documents are available through APDA® (Apple Programmers and Developers Association). APDA is an excellent source of technical information for anyone interested in developing Apple-compatible products.

Through APDA you can purchase Apple technical documentation, programming tools, and utilities. For information on available products and prices, please contact

APDA
Apple Computer, Inc.
20525 Mariani Avenue, Mailstop 33-G
Cupertino, CA 95014

800-282-APDA (800-282-2732)
FAX: 408-562-3971
Telex: 171-576
AppleLink®: APDA

Introduction

The backlit Macintosh Portable is a cost reduced version of the original Macintosh Portable with some improvements, the most noticeable of which is a backlit active matrix display. The feature set and functional capabilities of the backlit Macintosh Portable are identical to those of the original Macintosh Portable. For detailed technical information on the original Macintosh Portable, refer to *Guide to Macintosh Family Hardware,* Second Edition. A companion book, *Designing Cards and Drivers for the Macintosh Family*, provides essential information on designing expansion cards, ROM cards, RAM cards, and modem cards for the original Macintosh Portable. Most of the information about the original Macintosh Portable in these books also applies to the backlit Macintosh Portable

The improvements to the backlit Macintosh Portable include the following:

- PSRAM (pseudostatic RAM) replaces the SRAM (static RAM) used in the original Macintosh Portable.

- A shredded, more flexible cable provides easier access to the processor-direct expansion slot.

- The RAM expansion connector has been keyed, and the pinout has been slightly changed (two pins are different) from that of the original Macintosh Portable.

- Backlighting, which can be controlled by either a user (via the control panel's Portable CDev) or third-party application software, has been added to improve the backlit Macintosh Portable computer's display quality.

Of these improvements, those most interesting to hardware/software developers are the changes to the RAM expansion connector and the addition of software-controllable backlighting.

The sections that follow briefly describe each of these topics.

RAM expansion

The backlit Macintosh Portable is designed to accommodate a RAM expansion card that can provide up to 4 megabytes of additional RAM for the system. The RAM expansion card connects to the backlit Macintosh Portable through a single 50-pin connector (slot) on the main logic board.

◇ **Important**   Because an Apple RAM expansion card does not fully decode RAM expansion space, the card must be removed before additional third party RAM can be added to the backlit Macintosh Portable. It is possible to expand RAM up to 8 MB, however the zero wait state /DTACK signal and the /REFRESH signal are generated only for the first 5 MB of RAM address space. Apple does not recommend that you use the processor-direct slot for RAM expansion, but if you are planning to design a processor-direct slot (PDS) expansion card for additional RAM, keep in mind that timing, particularly the /DTACK signal, is critical. ◇

The connector

The RAM expansion connector (slot) is physically identical to the connector used on the original Macintosh Portable with the exception that the backlit Macintosh Portable connector is keyed. Electrically, however, there is a slight difference because two of the pins have different assignments from those of the original Macintosh Portable. Pin assignments that have changed are

Pin number	Original Macintosh Portable	Backlit Macintosh Portable
28	/AS	/RAM.CS
32	/DELAY.CS	/REFRESH

◆ *Note:* The /RAM.CS signal is directly related to the /REFRESH signal. /RAM.CS goes high to signal that reading and writing are no longer valid just before /REFRESH goes low to refresh the RAM memory. /REFRESH is a normally high signal that goes low (active state) approximately every 16 microseconds and remains low for 180 nanoseconds during which time the RAM memory is refreshed.

Table 1 provides the pin number, name, and description of each signal in the backlit Macintosh Portable's RAM connector. The descriptions of these signals, except for /RAM.CS and /REFRESH, are identical to the descriptions provided for the original Macintosh Portable.

• **Table 1**   Backlit Macintosh Portable RAM expansion connector signals

Pin number	Signal name	Signal description
1	+5V	+5-volt power supply.
2–24 A23.	A1–A23	Unbuffered 68HC000 address signals A1–
25–26	GND	Logic ground.
27 Portable is	/SYS.PWR	Controls whether the backlit Macintosh in the operating state or sleep state.
28 goes low to longer valid.	/RAM.CS	This signal goes high before /REFRESH signal that reading and writing are no
29	R/W	68000 Read/Write signal.
30	/UDS	Upper data strobe.
31	/LDS	Lower data strobe.
32 normally high signal ns and refreshes RAM	/REFRESH	Approximately every 16 μs, this goes low (active state) for 180 memory.
33–48 D15.	D0-D15	Unbuffered 68HC000 data signals D0–
49–50	+5V	+5-volt power supply.

RAM expansion slot timing

Unlike the original Macintosh Portable which required one wait
state, the backlit Macintosh Portable requires zero wait states when its
processor is accessing memory locations in the expansion RAM.
Figure 1 shows the RAM expansion slot timing for the backlit
Macintosh Portable computer.

◇ **Important** By inserting an additional 64 wait states between CLK
cycles 4 and 5 (see Figure 1) when the backlit
Macintosh Portable is in idle mode, the processor
timing will effectively slow down to 1 MHz. ◇

• **Figure 1** RAM expansion slot timing

CLK 0 1 2 3 4 Wait State 5 6 7 0 1 2 3 4 Wait State 5 6 7 0 1 2 3 4

/AS

R/W

/DS

/RAM.CS

RAM R/W

/OE

/EN 245

DIR245

ADDR Valid Valid
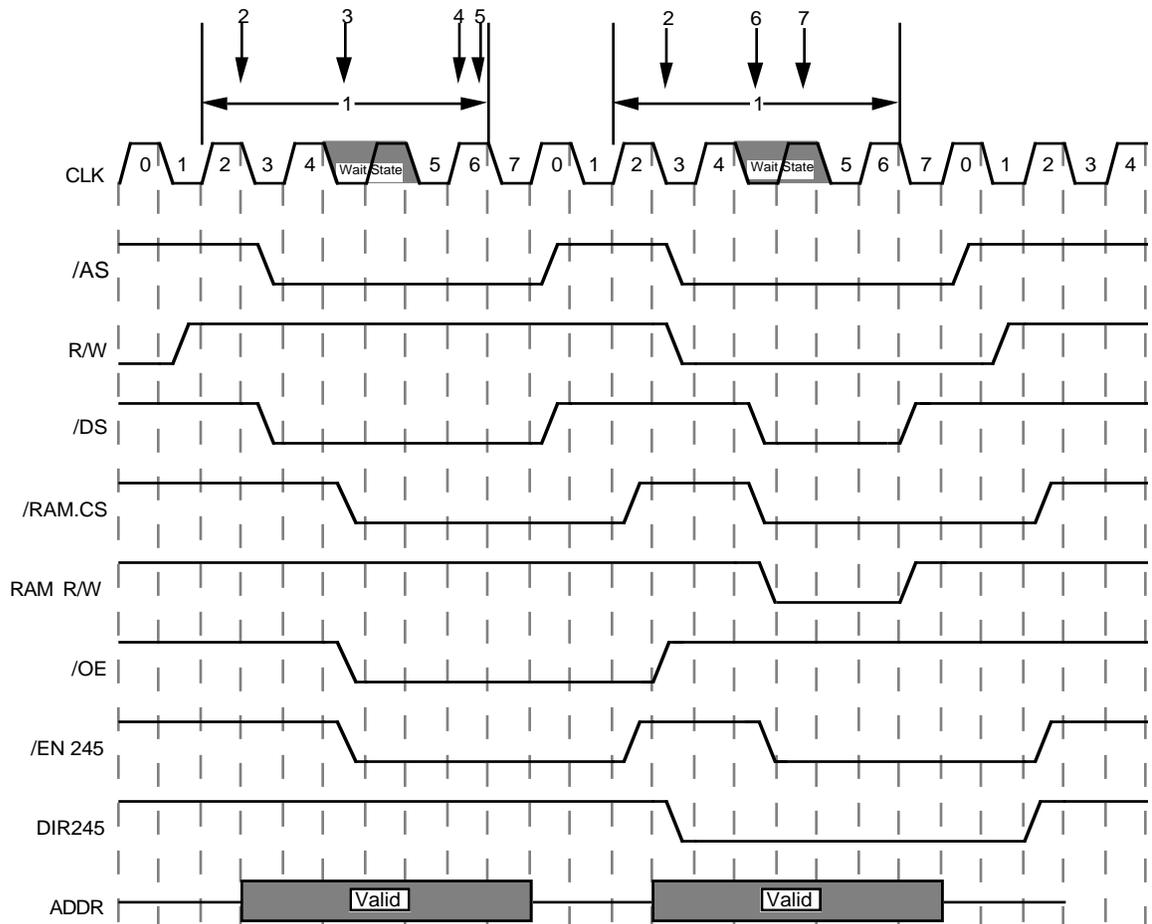
NOTES:
1. Total length of cycle read or write 5 x 32ns = 160 ns.
2. Delay clock to /AS = 30ns.
3. Delay AS/ to /RAM.CS = 10ns.
4. Delay through 74ac245 buffer = 7ns.
5. Setup before s7 = 5ns.
6. Delay s4 to /DS falling = 30ns.
7. Delay /DS to RAM R/W falling = 20ns.

When the portable is in idle mode, an additional 64 wait states are added between clk 4 & 5 effectively slowing the CPU timing down to 1 mhz. During normal operation these wait states are not present.

Read timing:		Write timing:	
Cycle		Cycle	160
/AS		Delay to s4	64
/RAM.CS		s4 to /DS	30
buffer		/DS to R/W	20
setup		Write cycle	100
RAM		Required	60
Margin		Margin	30 ns

The RAM expansion card

Figure 2 is a design guide showing the physical specifications you need to design a RAM card for the backlit Macintosh Portable. Remember that the connector on your RAM card must be keyed to match the RAM expansion connector on the main logic board. One possible source for the 50-pin keyed connector is

AMP Incorporated
Harrisburg PA, 17105

AMP's part number for this connector is 535954-1.

■ **Figure 2** RAM expansion card design guide

57.15 (2.25)

4.0 (.157)
/.45˚ chamfer

This area to remain clear of components
to allow room for board stabalization from lid.
5 (0.1969) deep by 8 (0.157) wide

68.66
(2.703)

81.20
(3.197)

Pin 1

50 – pin connector (keyed)

27.28 (1.074)

1.45 (.057)

10.1
(.398)

123.33 (4.855)

Dimensions in millimeters
with inches in parentheses.
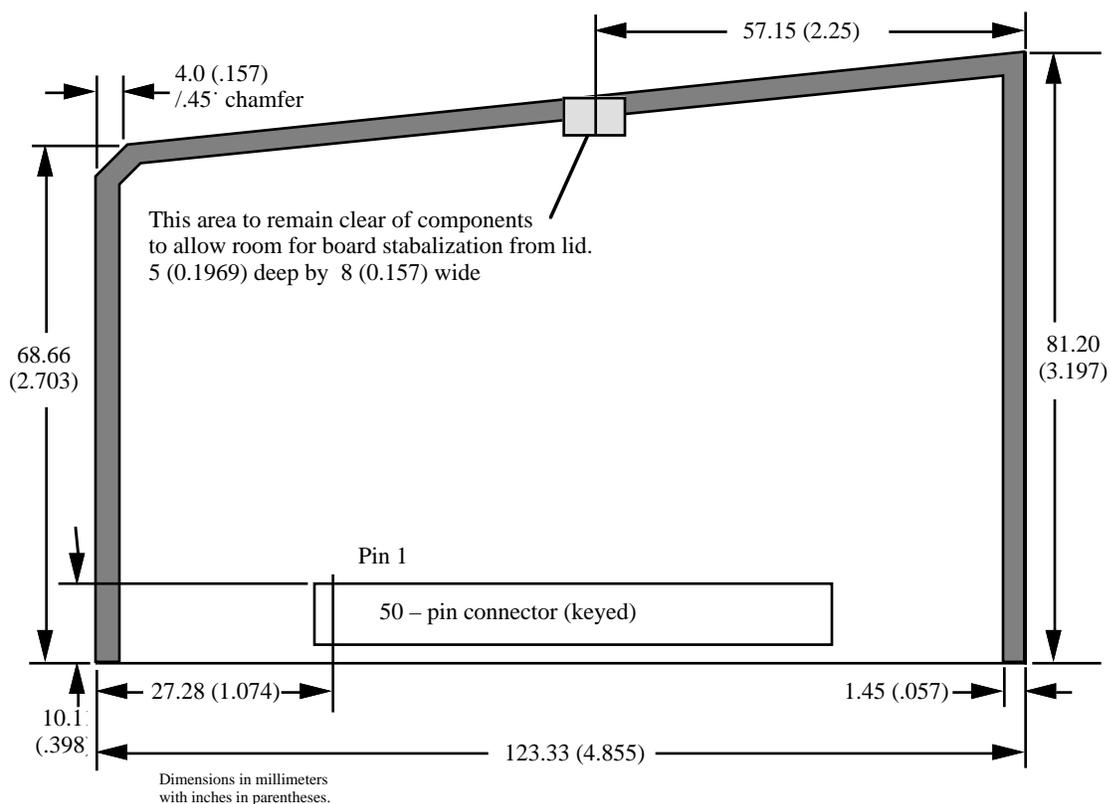
Design considerations

You must design your RAM expansion card to operate at zero wait states if you expect it to work in the backlit Macintosh Portable.

Because of different pin functions and the requirement of zero wait states, a RAM card designed for the backlit Macintosh Portable must have a keyed connector to prevent it from being installed in the original Macintosh Portable. However, a RAM expansion card designed for the original Macintosh Portable can be installed in the backlit Macintosh Portable but it will not function unless it

• is designed to run at zero wait states, and

• does not include a connection to the /DELAY.CS signal (/REFRESH for backlit Macintosh Portable)

◇ **Important**   Developers of expansion cards for the processor-direct slot must remember that /EXT.DTACK will no longer delay /DTACK for accesses to the first 5 megabytes of address space ($00 0000 thru $4F FFFF.) RAM or other devices responding to addresses in the first 5 megabytes are required to run at full speed with no wait states. ◊

Backlighting

This section describes the new backlighting feature of the backlit Macintosh Portable computer and describes the calls that can be issued by your application program to control the backlighting capability.

Controlling the backlighting

The backlit Macintosh Portable backlighting feature allows either an application or a user (via the control panel's Portable CDev) to use a driver to control the backlight brightness of the display. The driver that provides the software control for the CDev or application program is referred to as .Backlight. This driver also responds to certain key sequences from the keyboard and adjusts the brightness level accordingly. Figure 3 shows the possible inputs to the .Backlight driver for controlling the brightness of the display.

- **Figure 3** Inputs to the .Backlight driver.

```
┌──────────────┐        ┌─────────────┐
│ Portable CDev│        │ Application │
└──────┬───────┘        └──────┬──────┘
       │                       │
       │    ┌─────────────┐         ┌──────────┐
       └────┤.Backlight Driver├ ─ ─ ─┤ KeyMap  │
            └──────┬──────┘         └──────────┘
                   │
            ┌──────┴───────┐
            │Backlight Panel│
            └──────────────┘
```
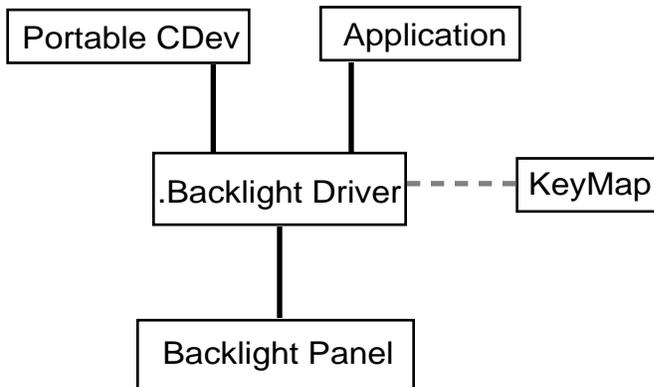
The backlight driver

The backlight driver (.Backlight) is a standard Macintosh driver. You use standard Macintosh driver calls to open, control, and gain access to it. Your application can issue special calls to gain status information and to control the brightness of the backlighting on the screen.

Call Descriptions

The following are descriptions of each of the calls that can be made to the backlight driver. Two of the calls, Open and Close, are issued by the Macintosh Operating System and should not have to be made by your application. They are included in the call descriptions, however, so that you can understand the significance of their functions.

Open

The Open call initializes the backlight driver. First, the driver creates its global storage. If storage is not available, an error is returned. Once global storage has been obtained, the driver proceeds to set the brightness potentiometer to the value stored in PRAM, the keyboard sequence is read from PRAM, and a VBL task is installed. The VBL task monitors the keyboard and adjusts the brightness level.

Errors returned:

Error	Code	Description
noErr	0	No errors, all went well
openErr	−23	No memory available for global storage

Close

The Close call does just the opposite of the Open call. It removes the VBL task from the queue and returns global storage to the Memory Manager.

♦ *Note:* Your application should not have to make the Close call.

Errors returned:

No errors are returned from the Close call.

Prime

The Prime call is not supported by the backlight driver.

Errors returned:

Error	Code	Description
readErr	−19	Always returned on Prime read call as reading is not supported from this driver
writErr	−20	Always returned on Prime write call as writing is not supported to this driver is

Status

The Status calls provide information about the backlight settings.
The selectors supported are Get Current Brightness, Get Current Key
Equivalents, Get Range, Get Saved Screen Brightness, Get Saved Key
Equivalents, and Get Maximum Brightness.

Get Current Brightness   csCode = $5301
This call returns the current setting for the backlight brightness in
the PB.csParam[0].

Get Current Key Equivalents   csCode = $5302
This call returns a bitmap of the current key equivalents in the
PB.csParam[0].

♦ *Note:* Key equivalents are modifier keys that you press while
   pressing one of the arrow keys to adjust screen brightness.

If a bit corresponding to a key is set (on), then that bit is included in
the key combination. The key combination feature is turned off if all
bits are 0 (off). The bit definition is the same as shown in the
previous example.

Get Range   csCode = $5303
This call returns the range of brightness values in the
PB.csParam[0,1]. csParam[0] will contain the maximum value, and
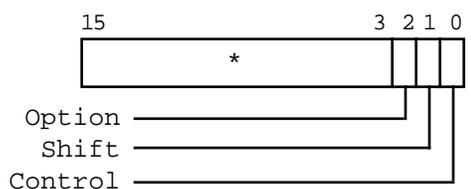csParam[1] will contain the minimum value.

Get Saved Screen Brightness   csCode = $5304
This call returns the saved value of the backlight brightness in the
PB.csParam[0]. This is the brightness value that the backlighting will
be set to when the system is rebooted as stored in Parameter RAM.

Get Saved Key Equivalents   csCode = $5305
This call returns a bitmap of the saved key equivalents in the
PB.csParam[0]. These are the key equivalents that will be used when
the system is rebooted as stored in the Parameter RAM. If a bit
corresponding to a key is set (on), then that bit is included in the key
combination. The key combination feature is turned off if all bits are
0 (off). Following is a definition of the bits.

```
CONST          Control    =    1
               Shift      =    2
               Option     =    4
```

```
               15                        3 2 1 0
              ┌──────────────────────────┬─┬─┬─┐
              │             *            │ │ │ │
              └──────────────────────────┴─┴─┴─┘
        Option ────────────────────────────┘ │ │
         Shift ──────────────────────────────┘ │
       Control ────────────────────────────────┘

              * reserved for future use
```

Get Maximum Brightness   csCode = $5306
This call returns the current maximum value allowed for the
backlight brightness setting. This value may be less than the
maximum value returned by Get Range, depending upon the
current battery charge.

Errors returned:

Error	Code	Description
statusErr	−18	Status selector not supported

Control

The Control calls adjust the brightness settings. Selectors supported
are Set Brightness, Save Brightness, CDev Active, Set Key
Equivalents, and Save Key Equivalents.

Set Brightness   csCode = $4301
This call adjusts the brightness level of the backlight to the value
contained in PB.csParam[0]. The value will be pegged to a maximum
or a minimum if it is out of range.

Save Brightness   csCode = $4302

This call saves to Parameter RAM the current backlight settings and makes the current backlight setting the default when you reboot.

CDev Active   csCode = $4303
This call tells the driver whether the CDev or an application (third-party software) is using the driver. A value of nonzero in PB.csParam[0] indicates that the CDev is active; a value of zero indicates that the CDev is inactive. When the CDev is active, this call also turns off the key equivalents feature because the CDev has its own key equivalents.

♦ *Note:* To avoid possible interaction when attempting to control the backlight brightness, your application should make sure that the CDev is inactive.

Set Key Equivalents   csCode = $4304
This call sets the key equivalents combination. The corresponding key bits that get set are defined and become part of the key combination. This feature is turned off if all the bits are 0 (off). Following is a definition of the bits.

```
CONST          Control    =    1
               Shift      =    2
               Option     =    4
```



```
                    15                      3  2 1 0
                   ┌──────────────────────┬─┬─┬─┬─┐
                   │          *           │ │ │ │ │
                   └──────────────────────┴─┴─┴─┴─┘
        Option ───────────────────────────┘ │ │
         Shift ─────────────────────────────┘ │
       Control ───────────────────────────────┘

              * reserved for future use
```

Save Key Equivalents   csCode = $4305
This call saves to Parameter RAM the key equivalents and makes the current key equivalents setting the default when you reboot.

Errors returned:

Error	Code	Description
controlErr	–17	Control selector not supported