RETRO ENDEAVORS

Keyboard //e++ Adapter

User manual · firmware 0.9.8

For the Apple //e: unenhanced, enhanced and Platinum
November 2026
Copyright © 2026 Patrick Montelo. All rights reserved.
retroendeavors.com

Contents

1. Introduction
2. Installing the adapter (to be written)
3. Precautions (to be written)
4. Using the keyboard
5. The Keyboard //e++ Console program
6. Which pairing window do I use?
7. Connecting a computer over Bluetooth
8. Using the console
9. Command reference
10. Settings
11. Typing on the //e from a computer
12. Macros
13. Bluetooth keyboards
14. Firmware updates
15. Troubleshooting
16. Third-party software
17. Quick reference

 Copyright © 2026 Patrick Montelo. All rights reserved.

1 Introduction

What the adapter does, and what this manual covers.

The Keyboard //e++ Adapter plugs onto the keyboard connector inside an Apple //e and lets it use a modern keyboard: a **USB keyboard**, a **Bluetooth keyboard**, or both, alongside the //e's own keyboard, which plugs into the adapter and keeps working as before. It works on every //e: unenhanced, enhanced and Platinum. Keys behave as they do on the //e's keyboard, with extras: Open Apple and Closed Apple on the Alt keys, macros on the function keys, and copying and pasting whole BASIC listings from a Mac or PC instead of typing them in by hand.

Most people only need a keyboard. A Mac or Windows computer is useful for changing settings, pasting or typing text into the //e, defining macros and installing firmware updates. For that the adapter has a **console**: a text window on the computer where you type commands and the adapter answers. There are two ways to open it:

- **The Keyboard //e++ Console program** for Windows or Mac, downloaded from retroendeavors.com. It connects over a **USB cable** or over **Bluetooth**, and is needed for firmware updates and macro files. This is the recommended way.
- **A terminal program** you may already have - PuTTY on Windows, or the Terminal app on a Mac - connected over a **USB cable**. Every console command works this way too.

In this manual

- **Installing the adapter** and **Precautions** (chapters 2 and 3).
- **Using the keyboard**: the keys the adapter adds (chapter 4).
- **The Keyboard //e++ Console program**, **which Bluetooth pairing window** your computer needs, and **connecting a computer** over Bluetooth (chapters 5 to 7).
- **The console**: how commands work, every command, and the settings (chapters 8 to 10).
- **Typing on the //e** from a computer, **macros**, **Bluetooth keyboards** and **firmware updates** (chapters 11 to 14).
- **Troubleshooting** (chapter 15), and a one-page **quick reference** at the end.

This edition describes firmware 0.9.8. In this manual, **Command** is the key with the Windows logo on PC keyboards (the Command key on Mac keyboards), and **Alt** is labelled Option on Mac keyboards.

 Copyright © 2026 Patrick Montelo. All rights reserved.

2 Installing the adapter

Fitting the adapter in the Apple //e.

To be written. Fitting the adapter onto the keyboard connector (J17), plugging the //e's own keyboard cable into the adapter, and plugging in a keyboard.

 Copyright © 2026 Patrick Montelo. All rights reserved.

3 Precautions

Handling and safety.

To be written. Power off before plugging or unplugging anything inside the Apple //e; static; orientation of the adapter and cables; what not to connect.

Copyright © 2026 Patrick Montelo. All rights reserved.

4 Using the keyboard

Typing on the //e, and the keys the adapter adds. No Mac or PC is needed.

Type as you would on the //e's own keyboard. A key held down is held down on the //e too, so the //e repeats it as usual, and games that watch a held key work as they do with the //e's keyboard. Keys a //e keyboard does not have send what //e programs expect:

KEY	ON THE //E
Backspace	the left arrow (Delete is Delete)
Left `Alt` / right `Alt`	Open Apple / Closed Apple
Caps Lock	the //e's Caps Lock (or an extra Control key: setting `caps`, chapter 10)
The number pad	the //e's numeric keypad (built into a Platinum //e; an option on the others)

KEYS	WHAT THEY DO
`Ctrl`+`Delete`	Reset, like Control+RESET.
Left `Alt` + `Ctrl`+`Delete`	Reset with Open Apple: restarts the //e (boots the disk).
Right `Alt` + `Ctrl`+`Delete`	Reset with Closed Apple: the //e's built-in self-test.
`Command`+`Ctrl`+`L`	Toggles the //e's Caps Lock (upper case only, or upper and lower case).
`Command`+`Ctrl`+`R`	Records a macro (see chapter 12).
`Ctrl`+`]` or `End`	Stops a macro, or text being typed from a computer.
`Command`+`Ctrl`+`M`	Opens the **Low Energy** Bluetooth pairing window for 2 minutes, for Keyboard //e++ Console on Windows or a Mac, or an iPhone or iPad (chapter 6).
`Command`+`Ctrl`+`B`	Opens the **Classic** (serial port) Bluetooth pairing window for 2 minutes, for a computer without Bluetooth Low Energy (chapter 6).
`Command` + a letter or digit, `F1`-`F24`, `Home`, `Page Up` …	Play the macro stored on that key, if there is one.

The //e's Caps Lock is on when the adapter starts, as on the //e's own keyboard with its Caps Lock key down. If the //e's own Caps Lock key is locked down, Caps Lock stays on whatever the adapter does. On a keyboard without a right Alt key (some compact ones), Closed Apple is not available.

The light on the adapter

It blinks while no keyboard is connected, and stays on once one is.

Copyright © 2026 Patrick Montelo. All rights reserved.

5 The Keyboard //e++ Console program

A console for the adapter on Windows and on a Mac.

Keyboard //e++ Console opens the adapter's console in a window. It also does a few things a general-purpose terminal program (such as PuTTY or the Mac's Terminal, see the end of this chapter) cannot: installing firmware updates, typing a text file on the //e, and saving and loading macros. It is the best way to use the console over Bluetooth.



The program's icon

Getting it

Download Keyboard //e++ Console from the Keyboard //e++ page at **retroendeavors.com** (under *Products*), or from its *Downloads* page. There are three versions:

DOWNLOAD	FOR
Windows (`.exe`)	Windows 10 or 11, 64-bit
Mac, Apple silicon (`.dmg`)	Macs with an M1 or later chip; macOS 11 Big Sur or later
Mac, Intel (`.dmg`)	Macs with an Intel processor; macOS 11 Big Sur or later

To see which Mac you have, choose *Apple menu* > *About This Mac*: "Chip: Apple M..." is Apple silicon, and "Processor: ... Intel ..." is an Intel Mac. The same window shows the macOS version (Big Sur is version 11). On Windows, *Settings* > *System* > *About* shows the edition and "64-bit operating system".

> **The first time you open it**, Windows or macOS warns you about it. Keyboard //e++ Console is not yet signed by Microsoft or Apple, which is common for programs from small makers. Follow the steps below once; after that it opens normally.

Installing on Windows

1. Download the `.exe` file. If your web browser warns that the file is not commonly downloaded, keep it: in Microsoft Edge, choose **...** next to the download, then **Keep**, then **Show more**, then **Keep anyway**.
2. Move it from your Downloads folder to wherever you like to keep it, such as your Documents folder or the desktop.
3. Double-click it. The first time, Windows says *"Windows protected your PC"*. Click **More info**, then **Run anyway**. Windows remembers this, so it does not ask again.

Keyboard //e++ Console opens in a console window. If it stops with a message, for example because the adapter was not found, the window stays open until you press Enter.

Copyright © 2026 Patrick Montelo. All rights reserved.

Installing on a Mac

1. Double-click the downloaded `.dmg` file. A window opens with Keyboard //e++ Console, a shortcut to the Applications folder, and a *Read Me First* file holding these same steps.
2. Drag **Keyboard IIe++ Console** onto **Applications**. (That is Keyboard //e++ Console's name in Finder: a Mac file name can't hold a slash.)
3. Open **Terminal** (in *Applications* > *Utilities*), paste this line into its window and press Return:

   ```
   xattr -dr com.apple.quarantine "/Applications/Keyboard IIe++ Console.app"
   ```

   It prints nothing when it works. If it says *Permission denied*, paste it again with `sudo` and a space in front, press Return, and type your Mac's password.
4. Eject the disk with the eject button beside it in Finder's sidebar.
5. Open Applications and double-click **Keyboard IIe++ Console**. It opens in a Terminal window.

Why step 3: Keyboard //e++ Console is not yet signed by Apple. Without that step, macOS refuses to open it, or keeps its icon bouncing in the Dock. The command tells macOS you trust this program. You only do it once, and again after installing a newer version.

The first time Keyboard //e++ Console uses Bluetooth, macOS asks whether it may: choose **Allow**. (If you said no, turn it on in *System Settings* > *Privacy & Security* > *Bluetooth*.)

Connecting

The console program connects in one of these ways:

CONNECTION	WINDOWS	MAC
USB cable	Yes	Yes
Bluetooth	Yes, after pairing (chapter 7)	Yes, after pairing (chapter 7)

For the **USB cable**, connect the adapter's micro-USB socket to your Windows or Mac computer. Then start the program. The first time, if it finds more than one possible connection, it lists them and asks which to use; after that it remembers the choice.

To change connection - for example, from the USB cable to Bluetooth - press `F3` in the program: it lists the connections it can see, marks the one in use, and you type the number of the one you want. The new choice is remembered for next time. If the remembered connection is not there when the program starts, it lists the others instead.

Over **Bluetooth Low Energy**, the program connects only to a Keyboard //e++ adapter, even with other Retro Endeavors adapters (a Game I/O++, say) switched on nearby, and remembers which one.

Copyright © 2026 Patrick Montelo. All rights reserved.

In the program

Everything you type goes to the adapter, which answers in the window (chapter 8). The function keys belong to the program:

KEY	ACTION
`F1`	This help.
`F3`	Choose another connection: USB cable or Bluetooth.
`F5`	Update the adapter's firmware (chapter 14).
`F6`	Type a text file on the //e.
`F7`	Load macros from a file (chapter 12).
`F8`	Save the adapter's macros to a file.
`F9`	Third-party software notices (chapter 16).
`F10`	Quit.

> **On a Mac**, function keys usually need `fn` held down (`fn`+`F10` quits), and Terminal keeps Home and End for itself: start typing mode with `t` and Enter, and leave it with `Ctrl`+`Z` or `Ctrl`+`]`.

If the connection drops - the //e was switched off, or the computer moved out of Bluetooth range - the program says so and reconnects by itself when the adapter is back. `F10` still quits at once.

Using a terminal program instead

Over a USB cable the adapter appears to the computer as a serial port, so any terminal program can open the console. Commands, settings and typing mode (pasting text into the window) all work the same way. Firmware updates, typing a file with `F6`, and saving and loading macro files need Keyboard //e++ Console.

Windows: PuTTY

1. Connect the USB cable. In *Device Manager* > *Ports (COM & LPT)*, note the new **USB Serial Device (COM*n*)**.
2. In PuTTY choose **Serial**, type that port (for example `COM5`) under **Serial line**, and click **Open**. The speed setting does not matter.
3. Press Enter to get the adapter's prompt.

Mac: Terminal

1. Connect the USB cable and open **Terminal** (in *Applications* > *Utilities*).
2. Find the adapter's port: type `ls /dev/*.usbmodem*` and press Return. It is named like `/dev/cu.usbmodem1101`, often listed with a matching `/dev/tty.usbmodem1101`; the number differs from Mac to Mac and from one USB socket to another. If more than one appears (another adapter is also connected, say), unplug the adapter's cable and list them again to see which one goes away.

Copyright © 2026 Patrick Montelo. All rights reserved.

3. Type `screen`, a space and that name - the `cu.` one if there is one, for example `screen /dev/cu.usbmodem1101` - and press Return, then press Return again to get the adapter's prompt.
4. To leave, press `Ctrl`+`A`, then `K`, then `Y`.

Copyright © 2026 Patrick Montelo. All rights reserved.

6 Which pairing window do I use?

Low Energy for the Console program; Classic for a serial port.

The adapter has two Bluetooth **pairing windows**. For Keyboard //e++ Console, on Windows and on a Mac alike, use the **Low Energy** window: the program then finds the adapter and pairs by itself, with nothing to set up in the computer's Bluetooth settings. The **Classic** window is for a computer whose Bluetooth has no Low Energy, and for a terminal program that uses a Bluetooth serial (COM) port.

IF YOU USE	OPEN THIS PAIRING WINDOW	WITH THE KEYS	OR ON THE CONSOLE
Keyboard //e++ Console, on Windows or a Mac	**Low Energy**	`Command`+`Ctrl`+`M`	`L`
An iPhone or iPad app	**Low Energy**	`Command`+`Ctrl`+`M`	`L`
A computer without Bluetooth Low Energy, or a terminal program over Bluetooth	**Classic** (serial port)	`Command`+`Ctrl`+`B`	`P`

Press the keys on a USB or Bluetooth keyboard connected to the adapter. Each window stays open for 2 minutes, and opening one closes the other. Not sure, or Bluetooth will not cooperate? The **USB cable** always works, for everything.

A computer is paired **one way at a time**, Classic or Low Energy: pairing it the other way replaces its first pairing, on the adapter and in the computer's Bluetooth settings alike. When a window opens, the console names any computer already paired the other way.

Does my computer have Bluetooth Low Energy?

- **Mac:** yes. Every Mac that can run Keyboard //e++ Console (macOS 11 Big Sur or later, chapter 5) has Bluetooth Low Energy built in, desktops such as the Mac Pro and Mac mini included. To see your macOS version, choose *Apple menu > About This Mac*.
- **Windows PC:** almost certainly, if its Bluetooth is less than about ten years old. To check, open *Device Manager* and expand *Bluetooth*: a PC with Low Energy lists *Microsoft Bluetooth LE Enumerator*. If it is not there, use the Classic window (chapter 7) or the USB cable. A PC with no Bluetooth at all has no Bluetooth switch in *Settings > Bluetooth & devices*; a USB Bluetooth adapter adds it.

> A Mac running a newer macOS than Apple supports for it (installed with an unofficial patcher) may still have an older Bluetooth card without Low Energy. Keyboard //e++ Console then says it found no Bluetooth Low Energy devices at all. Use the Classic window and the adapter's Bluetooth serial port (chapter 7), or the USB cable.

Other computers, phones and tablets

- **iPhone and iPad:** the Low Energy window, with a Bluetooth terminal ("UART") app.
- **A terminal program** such as PuTTY or the Mac's Terminal: the USB cable (chapter 5), or over Bluetooth, the Classic window and the adapter's Bluetooth serial port (chapter 7).

Copyright © 2026 Patrick Montelo. All rights reserved.

If you opened the wrong window

WHAT YOU SEE	WHAT HAPPENED, AND WHAT TO DO
Keyboard //e++ Console cannot find the adapter over Bluetooth Low Energy, or says the computer has to pair first	The Low Energy window was not open (the Classic one, or none). Open the Low Energy window (`Command`+`Ctrl`+`M`) and start the program again within 2 minutes. If you added the adapter in the computer's Bluetooth settings during the Classic window, you can instead choose its Bluetooth serial port in the program (`F3`).
The adapter was added in the computer's Bluetooth settings, but no Bluetooth serial port appears for it (in the program's list, or a terminal program)	The computer paired during the Low Energy window, which gives no serial port. Remove the adapter in the Bluetooth settings, then add it again during the Classic window (`Command`+`Ctrl`+`B`).
Keyboard //e++ Console used to connect over Bluetooth Low Energy, but no longer does	The computer was paired during the Classic window since, which replaces its Low Energy pairing. On the console, `list` shows how it is paired: `forget N` that entry, then open the Low Energy window (`Command`+`Ctrl`+`M`) and start the program again.

With the USB console open, the adapter also says when a computer tried the wrong kind of pairing, and which window to open instead.

Copyright © 2026 Patrick Montelo. All rights reserved.

7 Connecting a computer over Bluetooth

Pairing once, then connecting without a cable.

A computer pairs with the adapter during a **pairing window** of two minutes. There are two kinds; chapter 6 explains which to use:

PAIRING WINDOW	OPEN IT WITH	ON THE CONSOLE	FOR
Low Energy	`Command`+`Ctrl`+`M`	`L`	Keyboard //e++ Console on Windows or a Mac; iPhone and iPad apps
Classic (serial port)	`Command`+`Ctrl`+`B`	`P`	Computers without Bluetooth Low Energy; terminal programs (serial port)

The keys work on any USB or Bluetooth keyboard connected to the adapter, with no computer connected; they also switch Bluetooth on if it was off. No code needs typing. Outside a window, computers that have not paired are turned away; a computer that has paired connects again at any time without one.

Pairing with Keyboard //e++ Console (Windows or Mac)

1. Press `Command`+`Ctrl`+`M` on a USB or Bluetooth keyboard connected to the adapter (or type `L` and Enter on the console). The Low Energy pairing window is open for 2 minutes.
2. Within those 2 minutes, start Keyboard //e++ Console. It finds the adapter over Bluetooth by itself. The first time, if it asks which connection to use, choose *ble* (Bluetooth Low Energy). It pairs the computer and connects; on Windows it says *Pairing this PC with the adapter*.
3. If it connects over the USB cable instead, press `F3` (`fn`+`F3` on a Mac) and choose *ble*.
4. On a Mac, if macOS asks whether the program may use Bluetooth, choose **Allow**; if it asks to confirm the pairing, accept. If the 2 minutes ran out meanwhile, open the window again and restart the program.

Nothing needs doing in the computer's Bluetooth settings: don't add the adapter there yourself. Afterwards the computer connects whenever the program starts. (Windows then lists the adapter in *Settings > Bluetooth & devices*; that is normal.)

Pairing for a serial port: Classic window

For a computer whose Bluetooth has no Low Energy (an older Mac or PC), or for a terminal program such as PuTTY over Bluetooth. The computer pairs in its own Bluetooth settings and gets a Bluetooth serial port for the adapter, which Keyboard //e++ Console (or the terminal program) then opens.

On Windows:

1. Open *Settings > Bluetooth & devices*, choose **Add device**, then **Bluetooth**. Leave the list open: Windows is now searching.

Copyright © 2026 Patrick Montelo. All rights reserved.

2. Press `Command`+`Ctrl`+`B` on a USB or Bluetooth keyboard connected to the adapter (or type `P` and Enter on the console). The Classic pairing window is open for 2 minutes, and the adapter appears in the list within a few seconds.
3. Choose **Keyboard //e++ Adapter** (shown as a PC).
4. Windows pairs and says *Your device is ready to go*, then shows the adapter as *Not connected*. **That is normal:** Windows only connects when a program opens the adapter's serial port.
5. Start Keyboard //e++ Console and choose the port marked *Bluetooth to Keyboard //e++ Adapter* (or `F3` if it connects another way), or open the adapter's Outgoing COM port in your terminal program.

Windows makes two Bluetooth serial ports for the adapter. Use the **Outgoing** one (Keyboard //e++ Console finds it by itself). In *More Bluetooth settings > COM Ports* the other is listed as **Incoming**: Windows waiting for the adapter to call it, which it never does. It can be removed.

> Windows chooses the COM port numbers. If a Bluetooth port gets the same number as the USB cable's port, change one in *Device Manager > Ports > (the port) > Properties > Port Settings > Advanced > COM Port Number*.

On a Mac:

1. Open *System Settings > Bluetooth* (*System Preferences > Bluetooth* on macOS 11 and 12).
2. Press `Command`+`Ctrl`+`B` on a USB or Bluetooth keyboard connected to the adapter (or type `P` and Enter on the console). The Classic pairing window is open for 2 minutes, and the adapter appears in the list within a few seconds.
3. Click **Connect** beside **Keyboard //e++ Adapter**. The Mac pairs, then shows the adapter as *Not Connected*. **That is normal:** the Mac only connects when a program opens the adapter's serial port.
4. Start Keyboard //e++ Console and choose */dev/cu.KeyboardeAdapter*, marked *Bluetooth serial port (Classic)* (`fn`+`F3` if it connects another way). In the Mac's Terminal, the same port works with `screen /dev/cu.KeyboardeAdapter`.

Good to know

- One computer can use the console over Bluetooth at a time, alongside up to two Bluetooth keyboards. The adapter remembers up to 8 pairings, keyboards and computers together.
- The console over Bluetooth can be switched off with `btcon off` (and on again with `btcon on`, over the USB cable). The keys above switch it on for the session; `save` keeps it on.
- `list` shows paired keyboards and computers; `forget` forgets all of them. After `forget`, remove the adapter on each computer that was paired before pairing it again (see *If pairing does not work* below).

If pairing does not work

The usual cause is that the computer still remembers an earlier pairing with the adapter, while the adapter has forgotten it (after `forget`, for instance). What you see:

Copyright © 2026 Patrick Montelo. All rights reserved.

- Keyboard //e++ Console cannot connect over Bluetooth, says the computer has to pair first, says Windows has an old pairing, or loses the connection after a few seconds.
- **Classic window:** the adapter does not appear in the computer's Bluetooth settings, or its serial port will not connect.

To pair afresh:

1. On the computer, remove the adapter. On Windows: *Settings > Bluetooth & devices*, remove every *Keyboard //e++ Adapter* entry (and for a COM port, check that its ports have gone from Device Manager). On a Mac: *System Settings > Bluetooth*, click ⓘ beside *Keyboard //e++ Adapter* (it may be listed twice), then **Forget This Device**. If it is not listed, go on.
2. Pair again as above: open the Low Energy window and start Keyboard //e++ Console (or, for a serial port, the Classic window, as above).
3. Only if that still fails, also clear the adapter's own pairings: type `forget` and Enter on its console, then `y` and Enter, and pair again. This forgets **every** paired keyboard and computer, so each of them has to be paired again too.

Copyright © 2026 Patrick Montelo. All rights reserved.

8 Using the console

How commands work.

On connecting, the adapter shows its firmware version and a `>` prompt:

```
[Keyboard //e++ Adapter 0.9.8 ready — type ? and Enter for help, s for status]
[Copyright (c) 2026 Patrick Montelo. All rights reserved.]
>
```

Commands end with Enter

Type a command and press **Enter**. Nothing happens until Enter, so text typed by mistake does no harm: the adapter just says it is not a command. `Backspace` corrects a mistake, and `Ctrl`+`U` clears the line.

- Commands are words, typed in capitals or not: `status`, `save`, `forget`.
- Most have a one-letter short form, which must be typed exactly as shown: `s` is status, but `S` sends keys to the //e.
- Settings take a value: `capslock off`, `speed warp`. A setting typed without a value shows what it is now.

Commands that ask first

A few commands are easy to regret: `forget`, `bootsel`, deleting all macros (`M -*`), and switching Bluetooth or the console over Bluetooth off while connected over Bluetooth. They ask first:

```
> forget
That forgets every paired Bluetooth keyboard and computer.
Go ahead? Type y and Enter (anything else cancels). 'forget now' skips this question.
```

Type `y` and Enter to go ahead; anything else cancels. Adding `now` to the command skips the question: `forget now`.

Keys that act at once

- `Home` starts typing mode (chapter 11), and `Ctrl`+`]` stops it or a macro.
- Everything typed in typing mode goes to the //e.
- While `pair` is showing a list of Bluetooth keyboards, type a keyboard's number to choose it, or press `Esc` to cancel.

9 Command reference

Every console command.

Type the command, then Enter. The short form, where there is one, does the same.

Help and status

COMMAND	SHORT	WHAT IT DOES
`help`	`?`	List the commands.
`status`	`s`	Version, keyboards connected, Bluetooth, the //e's key lines, settings.
`diag`	`D`	Diagnostics: firmware, board, clock, time since start, last restart, USB details.

Typing on the //e

COMMAND	SHORT	WHAT IT DOES
`type`	`t`	Typing mode: what you type or paste is typed on the //e (also: `Home`).
`S` *keys*		Send keys the terminal cannot: arrows, Esc, Ctrl keys (chapter 11).
`reset`	`r`	Reset the //e (Control+RESET).

Macros (chapter 12)

COMMAND	SHORT	WHAT IT DOES
`M`		List the macros.
`M` *key*		Define a key's macro: type it, then `Ctrl`+`Z`.
`M` *key* `=` *text*		Define it on one line.
`M –` *key*		Delete one. `M –*` deletes all (asks first).

Settings (chapter 10)

COMMAND	SHORT	WHAT IT DOES
`capslock on|off`	`u`	The //e's Caps Lock.
`caps real|control`	`c`	What the Caps Lock key does.
`speed` *slow|normal|basic|warp*	`g`	Typing speed.
`timings`	`G`	Change a typing speed's timings.

 Copyright © 2026 Patrick Montelo. All rights reserved.

`keytime`	`K`	How long each typed key is held, and the gap after it.
`save`	`w`	Keep the settings after power-off.
`factory`	`x`	Back to the default settings (not saved until `save`).

Bluetooth (chapters 7 and 13)

COMMAND	SHORT	WHAT IT DOES
`bt on|off`	`e`	Bluetooth.
`pair`	`p`	Pair a Bluetooth keyboard.
`list`	`k`	Paired keyboards and computers.
`forget`		Forget every pairing (asks first). Each computer then has to remove the adapter and pair again.
`btcon on|off`	`o`	The console over Bluetooth.
`L`		Low Energy pairing window, for Keyboard //e++ Console on Windows or a Mac, and phones (2 minutes).
`P`		Classic (serial port) pairing window, for computers without Low Energy (2 minutes).

Firmware (chapter 14)

COMMAND	SHORT	WHAT IT DOES
`update`	`U`	Receive a firmware update (Keyboard //e++ Console's F5 does this for you).
`bootsel`	`b`	Restart ready for a firmware file to be copied on (asks first).
`trace on|off`	`d`	Show every report a keyboard sends (for fault finding).

Copyright © 2026 Patrick Montelo. All rights reserved.

10 Settings

How the adapter behaves, and how to keep it.

Changes take effect immediately but are not stored permanently until you enter `save`. Unsaved changes are lost when the //e is switched off. `s` lists all the settings and says whether they are saved.

SETTING	WHAT IT DOES	DEFAULT
`capslock on|off`	**The //e's Caps Lock**, as its own Caps Lock key does when locked down: on, letters are capitals; off, they follow Shift. `Command`+`Ctrl`+`L` toggles it from any keyboard.	on
`caps real|control`	**Caps Lock key.** `real`: it switches the //e's Caps Lock on and off (its light shows it). `control`: an extra Control key, where CONTROL sits on the //e's own keyboard.	real
`speed slow|normal|basic|warp`	**Typing speed** for typing mode and macros (chapter 11).	normal
`keytime` *hold gap*	**Key timing.** How long each typed key is held down (5-250 ms), and let go before the next (2-250 ms); also the shortest time a quick tap on a keyboard is held. `keytime` alone asks for both; `keytime 40 15` sets them.	40 ms, 15 ms
`bt on|off`	**Bluetooth.** Off: the radio stays off and the adapter works with USB keyboards only.	on
`btcon on|off`	**Console over Bluetooth.** Lets a paired computer open the console without a cable.	off

`factory` puts every setting back as above, without saving; `save` then makes that the power-on state. Macros and Bluetooth pairings are not affected by either.

 Copyright © 2026 Patrick Montelo. All rights reserved.

11 Typing on the //e from a computer

Typing mode, typing a file, and sending single keys.

Typing mode

Type `t` and Enter (or press `Home`). From then on, whatever you type or paste into the console window is typed on the //e: handy for BASIC listings and monitor hex dumps.

- **Finishing:** `Ctrl`+`Z` leaves typing mode once everything before it has been typed. `Ctrl`+`]` or `End` stops at once.
- Line endings become Return, the terminal's arrow keys become //e arrows, and Backspace becomes the left arrow. Curly quotes and dashes from word processors become plain ones; other characters the //e cannot type are skipped.
- Keyboards attached to the adapter keep working meanwhile.

To type a whole file, press `F6` in Keyboard //e++ Console and choose the file.

Pacing

The //e has no keyboard buffer: a key that arrives before the program has read the last one replaces it. So keys are paced by the **typing speed**, with a longer wait after Return while BASIC takes in the line:

SPEED	BETWEEN KEYS	AFTER RETURN	FOR
`slow`	60 ms	1000 ms	programs that still drop keys at Normal
`normal`	30 ms	750 ms	general use (default)
`basic`	10 ms	500 ms	the Applesoft prompt
`warp`	5 ms	5 ms	programs that read the keyboard quickly

`timings` changes the times of any speed; `save` keeps them. Each key is also pressed for the key time and let go for the gap (`keytime` , 40 ms and 15 ms), so the //e's keyboard encoder sees every one; that limits the fastest speeds to about 18 characters a second. If a program still drops characters, choose a slower speed.

Letters are typed as the //e's Caps Lock says: with Caps Lock on (the default), as capitals, as the //e's own keyboard would; with it off, as they are.

Sending single keys: `S`

To send keys that a terminal program can't easily type (arrows, Esc, Ctrl keys), particularly from a phone or tablet app, type `S` and key letters, then Enter:

AFTER S	SENDS
`l r u d`	Left, Right, Up, Down arrows
`e` / `n` / `s`	Esc / Return / Space

Copyright © 2026 Patrick Montelo. All rights reserved.

`i j k m`	I J K M (to move the //e's cursor after Esc)
`^` and a letter	that Ctrl key: `^C` is Ctrl+C
`x` and two hex digits	any code from 00 to 7F: `x1B` is Esc
`@`	Ctrl+@

So `S dddd` sends four Down arrows, `S ^C` stops a BASIC program, and `S eiiiijs` moves the cursor up four and left one. The adapter lists what it sent.

Copyright © 2026 Patrick Montelo. All rights reserved.

12 Macros

Keys that type a stored string.

A macro is a string of keys stored in the adapter, typed on the //e with one key press.

Keys that can hold a macro

- `Command` with a letter or digit;
- `F1` to `F24`, `Home`, `End`, `Page Up`, `Page Down`, `Insert`, `Print Screen`, `Scroll Lock`, `Pause` and `Menu` - alone, or with `Shift` for a second macro. (Keyboards whose top row controls volume and brightness need `Fn` or their Fn lock.)

Recording on the keyboard

1. Press `Command`+`Ctrl`+`R`, then the key to record into.
2. Type the macro. It is typed on the //e as you go, and the Caps Lock light blinks.
3. Press `Command`+`Ctrl`+`R` again to save it (`Command`+`Ctrl`+`Esc` cancels).

Recording nothing deletes the key's macro. Press the key to play it; `Ctrl`+`]` stops it, and so does `End` unless End has a macro of its own.

From the console

COMMAND	DOES
`M`	List all macros, e.g. `F1 = CATALOG{Return}`.
`M F1=CATALOG{Return}`	Define F1's macro on one line.
`M F1`	Define F1's macro by typing it (as in typing mode), then `Ctrl`+`Z` to save, `Ctrl`+`]` to cancel.
`M -F1`	Delete F1's macro.
`M -*`	Delete all macros (asks first).

Key names: `A` means Command+A; `F1`, `Shift+F12`, `Home`, `PgUp`, `PgDn`, `Insert`, `PrtSc`, `ScrLk`, `Pause`, `Menu`. In a macro's text, names in braces stand for keys:

WRITE	FOR
`{Return} {Esc} {Tab}`	those keys
`{Left} {Right} {Up} {Down}`	the arrow keys
`{^C}`	a Ctrl key
`{x1B}`	any code 00-7F, in hex
`{{`	a `{`

Copyright © 2026 Patrick Montelo. All rights reserved.

The adapter has about 8 KB of storage shared by all macros (roughly 8,000 characters). They are saved as soon as they change, and survive firmware updates and `factory`. In Keyboard //e++ Console, F8 saves them all to a text file and F7 loads a file back: one macro per line, as `M` lists them.

 Copyright © 2026 Patrick Montelo. All rights reserved.

13 Bluetooth keyboards

Pairing a Bluetooth keyboard with the adapter.

1. Put the keyboard into pairing mode (usually: hold its Bluetooth or Connect button until a light flashes).
2. On the console, type `pair` and Enter. The adapter searches for ten seconds and lists the keyboards it found, numbered.
3. Type the keyboard's number. (`Esc` cancels.)
4. If the console shows a code, **type it on the Bluetooth keyboard and press Enter**.

After that the keyboard connects by itself whenever it wakes up. Two Bluetooth keyboards can be connected at once, alongside USB keyboards; the adapter remembers up to 8 pairings (keyboards and computers together).

- `list` shows the paired keyboards and computers, and which are connected.
- `forget` forgets them all (asks first). The keyboard may then need its own pairing cleared before it pairs again.
- `bt off` switches Bluetooth off (keyboards and console); `bt on` switches it on again.

> Keyboards with their own small USB receiver (Logitech Unifying and similar) are USB keyboards to the adapter: plug in the receiver, no pairing needed.

 Copyright © 2026 Patrick Montelo. All rights reserved.

14 Firmware updates

Installing new firmware.

With Keyboard //e++ Console

1. Connect Keyboard //e++ Console to the adapter, by USB cable or Bluetooth. The computer needs to be on the internet.
2. Press F5. The program asks retroendeavors.com for the newest firmware.
3. If there is newer firmware, the program shows its version and what is new, and asks *Install it?* Type `y` and Enter. If the adapter already has the newest firmware, the program says so; press Enter.

The program downloads the firmware and sends it to the adapter. The adapter checks that it is genuine Retro Endeavors firmware for this adapter, installs it, and restarts (about 10 seconds, during which the keyboard does not respond). Then the program reconnects and confirms the new version. Settings, macros and Bluetooth pairings are kept. If anything goes wrong, the adapter keeps its current firmware.

The program checks for new firmware only when F5 is pressed, never by itself. If it says the connection reaches another adapter (a Game I/O++, say), press F3 and choose the Keyboard //e++.

From a file

If Retro Endeavors support (**retroendeavors.com/contact**) sends you a firmware file (ending `.kfw`): press F5, then type `f` and Enter at the question, and choose the file. If the computer can't reach retroendeavors.com, the program says so and offers a file instead: type `y` and Enter. The adapter accepts only genuine firmware files from Retro Endeavors.

By copying a file (BOOTSEL)

For recovery only, with a `.uf2` file from Retro Endeavors support:

1. Switch the //e off and take the adapter off the keyboard connector. (On the //e, the adapter would look like Control+RESET held down while it waits for the file.)
2. Hold the adapter's **BOOTSEL** button while plugging its USB cable into the computer. A drive called **RP2350** appears.
3. Copy the `.uf2` file onto it. The adapter restarts with the new firmware; unplug it and fit it back on the //e.

The console's `bootsel` command (it asks first) does the same as holding the button, for an adapter already connected by USB cable and off the //e.

> If power fails while an update is being installed, the adapter starts with the RP2350 drive by itself. Ask Retro Endeavors support for the `.uf2` file and copy it on as above. An update cannot leave the adapter unusable.

 Copyright © 2026 Patrick Montelo. All rights reserved.

15 Troubleshooting

When something does not work.

PROBLEM	TRY
Keys do nothing	Check the light: blinking means no keyboard is connected. Type `s` on the console: it lists the keyboards attached.
Letters are all capitals	The //e's Caps Lock is on, the default: `capslock off`, Caps Lock, or `Command`+`Ctrl`+`L`. If the //e's own Caps Lock key is locked down, release it.
The //e drops characters in typing mode	Choose a slower `speed`, or a longer `keytime`.
Open Apple or Closed Apple does nothing	They are the left and right Alt (Option) keys. Some compact keyboards have no right Alt.
The adapter does not appear when pairing a computer	Open the right pairing window (chapter 6: `Command`+`Ctrl`+`M` for Keyboard //e++ Console, `Command`+`Ctrl`+`B` for a serial port) and pair within 2 minutes; then see *If pairing does not work* in chapter 7.
The computer shows the adapter as *Not connected*	Normal until a program connects to it.

Copyright © 2026 Patrick Montelo. All rights reserved.

Keyboard //e++ Console stopped connecting over Bluetooth	The computer and the adapter disagree about their pairing (often after `forget`). Remove the adapter in the computer's Bluetooth settings and pair again: *If pairing does not work* in chapter 7.
Windows says *Windows protected your PC* when Keyboard //e++ Console starts	Click **More info**, then **Run anyway** (chapter 5).
A Mac will not open Keyboard //e++ Console, or its icon keeps bouncing in the Dock	Force Quit it if it is bouncing, then do step 3 of *Installing on a Mac* (chapter 5): the `xattr` line in Terminal.
Keyboard //e++ Console cannot find the adapter	Check the cable, or that the computer is paired; over Bluetooth, the console over Bluetooth must be on (`btcon on`, over the USB cable).
A command does not work	Commands end with Enter, and short forms are case-sensitive: `?` lists them.

If the adapter restarts by itself

If the firmware ever stops responding, the adapter restarts by itself within a few seconds, and the console says why (*note: restarted after …*). `s` and `D` show it too. Please report that line to Retro Endeavors with what you were doing.

Getting help

If something here does not solve it, contact Retro Endeavors support through **retroendeavors.com/contact**. Saying which firmware the adapter has (`s` shows it), whether you use a Mac or a Windows PC, and what you were doing helps.

 Copyright © 2026 Patrick Montelo. All rights reserved.

16 Third-party software

The open-source software inside the adapter and the program, and where its licences are.

The Keyboard //e++ Adapter firmware and Keyboard //e++ Console are Copyright © 2026 Patrick Montelo, all rights reserved. They are built with open-source software from others, each under its own licence, listed below. The licences ask that their full texts come with the software: they are at **retroendeavors.com/legal/keyboard-iiepp**, and Keyboard //e++ Console shows its own with F9. On a Mac they are also in the disk image, as *Third-Party Notices.txt*.

In the adapter's firmware

SOFTWARE	LICENCE
Raspberry Pi Pico SDK	BSD 3-Clause
TinyUSB (Ha Thach)	MIT
Pico-PIO-USB (sekigon-gonnoc)	MIT
BTstack (BlueKitchen GmbH)	Raspberry Pi's licence for its chips
cyw43-driver (George Robotics)	Raspberry Pi's licence for its chips
micro-ecc (Kenneth MacKay)	BSD 2-Clause
Monocypher (Loup Vaillant)	BSD 2-Clause
MD5 (Solar Designer), Rijndael (Philip J. Erdelsky)	Public domain

In Keyboard //e++ Console

SOFTWARE	LICENCE
Python	Python Software Foundation License
OpenSSL (in Python)	Apache 2.0
pyserial (Chris Liechti)	BSD 3-Clause
bleak (Henrik Blidh)	MIT
PyObjC (Mac) or the Python WinRT packages (Windows)	MIT
certifi (web site certificates)	Mozilla Public License 2.0; source at github.com/certifi/python-certifi
Nuitka runtime (Kay Hayen)	AGPL 3.0 with the Nuitka Runtime Library Exception, which lets programs built with it be distributed under any terms

The Keyboard //e++ Adapter grew out of the Keyboard][++, which was inspired by Sebastian Kienzl's PS/2-to-//e converter (2008); none of its code or design is used.

Quick reference

The commands and keys used most.

Console: commands end with Enter

`?`	help
`s`	status
`t` / `Home`	typing mode (`Ctrl`+`Z` ends)
`S dddd`	send keys
`M`	list macros
`M F1=TEXT{Return}`	define a macro
`M -F1`	delete a macro
`reset`	reset the //e
`save`	keep settings for every power-on
`factory`	back to the defaults (until `save`)
`pair` / `list` / `forget`	Bluetooth keyboards
`L` / `P`	pairing window: Low Energy / Classic
`bt` / `btcon on|off`	Bluetooth / its console

Settings

`capslock on|off`	the //e's Caps Lock
`caps real|control`	Caps Lock key
`speed slow|normal|basic|warp`	typing speed
`keytime 40 15`	key held / gap (ms)

On the keyboard

`Ctrl`+`Delete`	reset
left / right `Alt`	Open / Closed Apple
`Command`+`Ctrl`+`L`	the //e's Caps Lock
`Command`+`Ctrl`+`R`	record a macro
`Ctrl`+`]` / `End`	stop
`Command`+`Ctrl`+`M`	pair: Low Energy (Console)
`Command`+`Ctrl`+`B`	pair: Classic (serial port)

Keyboard //e++ Console

`F1`	help
`F3`	choose another connection
`F5`	firmware update
`F6`	type a text file
`F7` / `F8`	load / save macros
`F9`	third-party notices
`F10`	quit (Mac: `fn` + key)

 Copyright © 2026 Patrick Montelo. All rights reserved.