

epomaker keyboard detected as apple

Epomaker Galaxy80 keyboard

The Problem

Recently, my Keychron K8 keyboard has died (PCB [\[1\]](#)), so I started hunting for a new keyboard. Another Keychron was out of the question, so I decided to give Epomaker a try. I stumbled into a B-stock [Epomaker Galaxy80](#) and decided to give it a shot. I am very happy with the feel of the keyboard, typing feels much better than the K8, at least to me.

However, I ran into a problem in linux (Fedora Workstation, although this is not distro-specific): I was unable to use the F-row (F1 - F12), so that Alt + F4 would not close the window, F11 would not trigger full screen etc. When I used the online [keyboard testers](#), it would register keys/commands such as “Files”, “Calculator”, “Volume Up/Down” etc.:

“F-row keypresses in a keyboard tester”

I searched the keyboard manual, thinking it must be some sort of an Fn lock being activated, but there does not seem to be a feature like that on my keyboard. Even worse, using the Fn key + one of the F-row keys was not triggering the actual F-row key either. I had no (useful) way to use my F-row keys!

This issue has been reported with many other Epomaker keyboards, such as EP84, TH80, RT80, RT100 or GMK87, but also with other brands (including some Keychrons). The good news is, the fix should be the same for all of them!

The Solution

After some furious googling, I actually found the solution to the problem, which turns out to be on the system side rather than in the keyboard itself. Simply [reconfiguring](#) your [HID Apple module](#) fixes the issue. If you're technically savvy and/or do not care about any further details, go use the module configuration and enjoy using your keyboard.

```
echo "options hid_apple fnmode=2" | sudo tee /etc/modprobe.d/hid_apple.conf
sudo dracut --regenerate-all --force
# reboot and enjoy your fully functioning keyboard!
```

(courtesy of Reddit user [Outrageous_Opinion84](#) and [Joao-Peterson](#) on GitHub)

The Solution Explained

Below, I would like to go into more details for several reasons:

1. Reddit is going to paywall some of the content, and other content might get authwalled - therefore the solution may not be readily available.
2. It took me some time to find the solution, so by writing this article, I'm hoping the more elaborate explanation will be more SEO-friendly than some of the existing answers scatted over the internet.
3. I am sure I will forget this solution next time I install a new system and writing it up will hopefully mean I will not need to re-figure it out.
4. I wanted to learn (and share) in a bit more detail what happens with the configuration files and how the fix works.

First, let's take a look at how the Epomaker keyboard reports to the system when loaded up (the last but one row):

output of lsusb

Now the problem is that the keyboard is being detected as an Apple wireless keyboard clone - an abomination of a keyboard discontinued in 2015:

output of sudo dmesg

By default, the F-row keys on the Apple wireless keyboard emit the media commands, which is how those keypresses will get interpreted by the `hid_apple` kernel module. Luckily, the module offers configurable function modes (`fnmode`), as documented on [Arch Wiki](#) or [Ubuntu Wiki](#). We are interested in the `fnmode=2` option:

- 2 - normally function keys, switchable to media keys by holding Fn key (=auto on non-Apple keyboards)

You can temporarily test the parameter by sending the `fnmode` parameter to the kernel module via `sysfs`:

(You likely need to do a `sudo su` first)

```
echo 2 >> /sys/module/hid_apple/parameters/fnmode
```

Now your keyboard should work as GodEpomaker intended! The problem is that this change will not be persisted after reboot.

The way to do it is through the `modprobe.d` configuration files, which allow to pass parameters to the modules when they are loaded during system boot.

The following command will create the configuration file for the `hid_apple` module and pass the `fnmode=2` parameter to it:

```
echo "options hid_apple fnmode=2" | sudo tee /etc/modprobe.d/hid_apple.conf
```

Now to apply the changes to the initramfs (a mini-image with your kernel, including its modules), you need to run the `dracut` command to re-generate initramfs.

```
sudo dracut --regenerate-all --force
```

Note: Some distributions may require a different command to re-generate initramfs, such as `sudo update-initramfs -u` for Ubuntu or `sudo mkinitcpio -P` in Arch.

Finally, all you need to do is either reboot your machine, or reload the module:

```
sudo rmmod hdi_apple
modprobe hdi_apple
```

Revision #1

Created 13 August 2025 04:06:59 by Cesar Gzz

Updated 13 August 2025 04:07:26 by Cesar Gzz