

**Andrew Schmelyun**

Posted on Aug 20, 2020



5



1

Converting my Windows PC to use MacOS keybindings

#showdev #tutorial #productivity

tl;dr:

- Download and install the latest .msi of [SharpKeys](#)
- Map Left Ctrl to Left Alt (and vice versa) in SharpKeys
- Download and install [AutoHotKey](#)
- Save [this script](#) somewhere on your machine
- Double-click the downloaded script to activate it

Want a more detailed walk-through? Keep reading!

A few months ago I put together a pretty decent Windows PC, mostly for gaming. Although of course I started tinkering here and there with programming on it, and eventually decided to go full in on making it a web dev machine. The timing couldn't be perfect either, as Windows released [WSL 2](#) recently and its performance with Docker Desktop has been incredible.

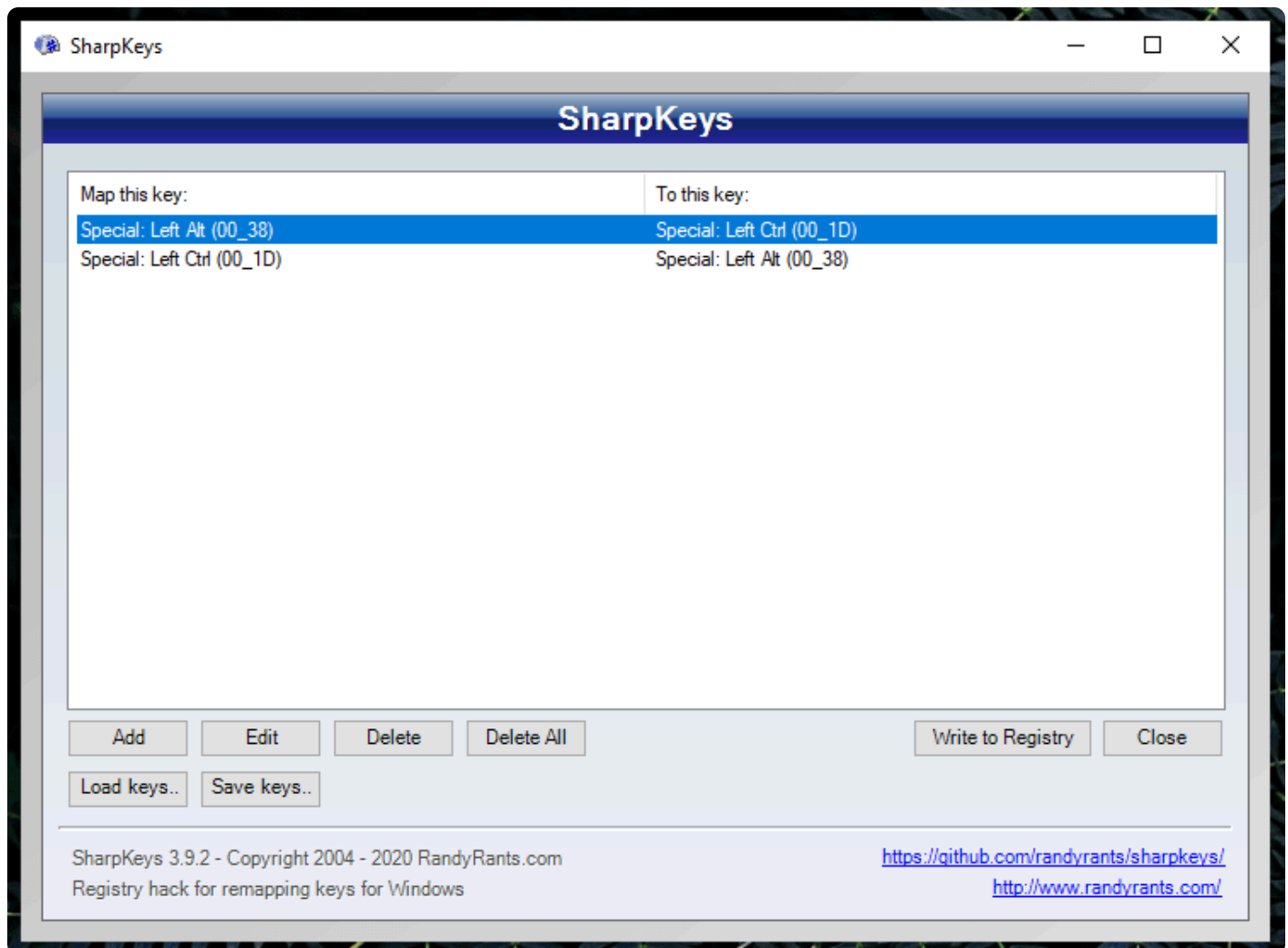
The downside though is that my day-to-day work is on a MacBook, and (although I feel alone in this) I absolutely *love* using Alt instead of ctrl for things like copying and pasting. I figured there had to be others like me out there who have solved this, so I went searching for a solution.

There's a few different methods out there, but I found a combination of two things that were easy to set up using free software, and have performed consistently well since I added them a few weeks ago. With that being said, **let's get started!**

Step #1 - Switching Alt and Ctrl

The first part of this will be dedicated to swapping out our `ctrl` key for the `Alt` key, which mimics the Apple keyboard `cmd` key placement. For this, you'll need to download and install a program called **SharpKeys**. The app is entirely open source, and you'll need to visit the [releases GitHub page](https://github.com/randyrants/sharpkeys/) downloading the latest `.msi` file available.

Once you have it installed, open it up and click the **Add** button toward the lower right. You're looking to map both the Left Ctrl key to Left Alt, and the Left Alt key to Left Ctrl. After you do that, your program's main screen should look like this:



If everything looks right, click the **Write to Registry** button towards the lower right to save your changes. Then, exit the application.

You can now try it out and see that your Alt and Ctrl keys have been switched! Instead of using **Ctrl+C** for copying, you'll now be using **Alt+C**. However, there's a small caveat. Because all we did was map the keys differently, keyboard shortcuts that made use of Alt now expect to use Ctrl.

If you're like me and use **Alt+Tab** to switch between windows a lot, that's where you'll notice this problem the most. Let's fix that!

Step 2 - Remapping Alt Tab

For this next step we'll need another free program, [AutoHotKey](#). Go ahead and download the program from that link and install it on your machine. Skip through any tutorial or help requests if you'd like, we're only going to be using it with a single (and fairly short) script.

After it's been installed successfully, download the following script and save it somewhere convenient on your machine, making sure that it contains a `.ahk` extension:

```
1  #NoEnv ; Recommended for performance and compatibility with future AutoHotkey releases.
2  ; #Warn ; Enable warnings to assist with detecting common errors.
3  SendMode Input ; Recommended for new scripts due to its superior speed and reliability.
4  SetWorkingDir %A_ScriptDir% ; Ensures a consistent starting directory.
5
6  LCtrl & Tab:: AltTab
7  !Tab:: Send ^{Tab}
8  !+Tab:: Send ^+{Tab}
9
10 ^Space:: Send ^{Esc}
11
12 ^Left::
13     Send {Home}
14     Return
15
16 ^Right::
17     Send {End}
18     Return
19
20 ^+Left::
21     Send +{Home}
22     Return
23
24 ^+Right::
25     Send +{End}
26     Return
27
28
29 ^Up::
30     Send ^{Home}
```

```
31  Return
32
33  ^Down::
34      Send ^{End}
35  Return
36
37  ^+Up::
38      Send ^+{Home}
39  Return
40
41  ^+Down::
42      Send ^+{End}
43  Return
```

windows-macos-keybindings.ahk hosted with ❤ by GitHub

[view raw](#)

From there, double-clicking on the file will silently start the AutoHotKey script, re-enabling your **Alt+Tab** window switching! ✨

Note: The above script also adds in some helpful MacOS-related keybindings for Alt+Arrow combinations.

Extra Step - Running AutoHotKey Script on Startup

Using the method above, you'll need to double-click on the script each time your computer turns on in order for your **Alt+Tab** fix to start working. There's a [simple workaround](#) for this though, that will get the script starting as soon as the computer turns on.

Just follow these three simple steps:

1. Press **Win+R** to open up the Run dialog.
2. Type in `shell:startup` and press Enter, this should open up a folder window.
3. Copy and paste your saved `.ahk` script file into this window.

As soon as your computer boots up, your AutoHotKey script and SharpKeys modifications will start running, essentially swapping out your Windows keybindings for those on a MacOS system. 🔥

That's it! If you have any questions about the above, any issues with what was outlined, or additions that might be helpful to the AutoHotKey script I provided, please feel free to let me know. You can leave a comment below, or message me on [Twitter](#).

Feel free to follow me on there as well, if you'd like to see fairly regular, shorter posts on web development and MacOS. You'll also see updates when I publish new video tutorials and finish writing articles.

👋 Before you go



Do your career a big favor. **Join DEV.** (The website you're on right now)

It takes **one minute**, it's free, and is worth it for your career.

[Get started](#)

Community matters

Top comments (1)



Adam Davis • Aug 20 '20



Cool idea! I did something like this a while ago where I wanted to use the mouse with my left hand (I'm left-handed) so I mirrored some common keybindings onto the opposite end of the keyboard to be able to type them comfortably with my right hand.

[Code of Conduct](#) • [Report abuse](#)



Andrew Schmelyun

Software engineer and content creator, enjoys starting side projects and helping others learn.

LOCATION

Orlando, FL

JOINED

Mar 9, 2019

More from [Andrew Schmelyun](#)

I built a free tool to help content creators collect ideas and feedback

[#showdev](#)

The crippling problem of being able to build whatever you want

[#programming](#) [#productivity](#) [#discuss](#)

Intercept and inspect http requests with reqon

[#showdev](#) [#tools](#) [#opensource](#)

Please check out our sponsors ❤️