Technical Note TN2450

Remapping Keys in macOS 10.12 Sierra

Important: This document is no longer being updated. For the latest information about Apple SDKs, visit the documentation website.

In macOS Sierra 10.12, we introduced a new way to keyboard key remapping. This document will discuss the various ways to do so.

Introduction

Under macOS Sierra 10.12, the mechanism for key remapping was changed. This Technical Note is for developers of key remapping software so that they can update their software to support macOS Sierra 10.12. We present 2 solutions for implementing key remapping functionality for macOS 10.12 in this Technical Note. The command line `hidutil` tool is useful for executable scripts. macOS applications can use the IOHIDEventSystemClient API to achieve this functionality. The scope of the key remapping function applies to all users and will remain in effect so long as there is an active keyboard service. Key remappings are lost when the system is restarted or if the keyboard service is removed (for example when the last keyboard is disconnected.) No special privileges are required to use key remapping.

Back to Top

Scripting Key Remapping

Keys can be remapped via the command-line tool `hidutil`. For example, use the `hidutil` command-line tool to remap the 'A' key to the 'B' key as shown in Listing 1. The map array consists of two key/value pairs that contain the source (`HIDKeyboardModifierMappingSrc`) and destination (`HIDKeyboardModifierMappingDstKey`) of the key remapping. The keys take a hexadecimal value that consists of 0x700000000 or'd with the desired keyboard usage value (see Table 1 for usage values).

Listing 1: Remapping keys in a script

```
$ hidutil property --set '{"UserKeyMapping":
```

```
[{"HIDKeyboardModifierMappingSrc":0x700000004,"HIDKeyboardModifierMappingDst":0x700000005},
{"HIDKeyboardModifierMappingSrc":0x700000005,"HIDKeyboardModifierMappingDst":0x700000004}]}'

)
```

A script can check the key remapping state by using the `hidutil` command-line tool as shown in Listing 2. A `null` result indicates that there are no key remappings active.

Listing 2: Checking Key Remapping state

```
$ hidutil property --get "UserKeyMapping"

(null)
```

Back to Top

Programmatic Key Remapping

The IOKit HID APIs can be used for key remapping. The user will provide a dictionary of key remapping that the HID event system will apply to the keyboard.

Listing 3 : Key Remapping using IOKit HID APIs.

```
// compiled with Xcode 8.2.1


#import <Foundation/Foundation.h>


#import <IOKit/hidsystem/IOHIDEventSystemClient.h>


#import <IOKit/hidsystem/IOHIDServiceClient.h>


#import <IOKit/hid/IOHIDUsageTables.h>


int main(int argc, char *argv[])

{

    IOHIDEventSystemClientRef system;

    CFArrayRef services;


    uint64_t aKey = 0x700000004;

    uint64_t bKey = 0x700000005;
```

```
    NSArray *map = @[
                    @{@kIOHIDKeyboardModifierMappingSrcKey:@(aKey),
                       @kIOHIDKeyboardModifierMappingDstKey:@(bKey)},
                    @{@kIOHIDKeyboardModifierMappingSrcKey:@(bKey),
                       @kIOHIDKeyboardModifierMappingDstKey:@(aKey)},
                    ];


    system = IOHIDEventSystemClientCreateSimpleClient(kCFAllocatorDefault);
    services = IOHIDEventSystemClientCopyServices(system);
    for(CFIndex i = 0; i < CFArrayGetCount(services); i++) {
        IOHIDServiceClientRef service =
(IOHIDServiceClientRef)CFArrayGetValueAtIndex(services, i);
        if(IOHIDServiceClientConformsTo(service, kHIDPage_GenericDesktop,
kHIDUsage_GD_Keyboard)) {
            IOHIDServiceClientSetProperty(service, CFSTR(kIOHIDUserKeyUsageMapKey),
(CFArrayRef)map);
        }
    }


    CFRelease(services);
    CFRelease(system);


    return 0;
}
```

Back to Top

Key Table Usages

Table 1 presents a list of keyboard usages and their usage IDs for use in key remapping. This list is from the USB HID Usage Tables Specification, Section 10 Keyboard /Keypad Page.

Table 1: List of keyboard usages and their usage IDs.

Usage	Usage ID (hex)	Usage	Usage ID (hex)	Usage	Usage ID (hex)	Usage	Usage ID (hex)
Keyboard a and A	0x04	Keyboard 5 and %	0x22	Keyboard F7	0x40	Keypad 6 and Right	0x5E

						Arrow	
Keyboard b and B	0x05	Keyboard 6 and ^	0x23	Keyboard F8	0x41	Keypad 7 and Home	0x5F
Keyboard c and C	0x06	Keyboard 7 and &	0x24	Keyboard F9	0x42	Keypad 8 and Up Arrow	0x60
Keyboard d and D	0x07	Keyboard 8 and *	0x25	Keyboard F10	0x43	Keypad 9 and Page Up	0x61
Keyboard e and E	0x08	Keyboard 9 and (	0x26	Keyboard F11	0x44	Keypad 0 and Insert	0x62
Keyboard f and F	0x09	Keyboard 0 and)	0x27	Keyboard F12	0x45	Keypad . and Delete	0x63
Keyboard g and G	0x0A	Keyboard Return (Enter)	0x28	Keyboard Print Screen	0x46	Keyboard Non-US \ and |	0x64
Keyboard h and H	0x0B	Keyboard Escape	0x29	Keyboard Scroll Lock	0x47	Keyboard Application	0x65
Keyboard i and I	0x0C	Keyboard Delete (Backspace)	0x2A	Keyboard Pause	0x48	Keyboard Power	0x66
Keyboard j and J	0x0D	Keyboard Tab	0x2B	Keyboard Insert	0x49	Keypad =	0x67
Keyboard k and K	0x0E	Keyboard Spacebar	0x2C	Keyboard Home	0x4A	Keyboard F13	0x68
Keyboard l and L	0x0F	Keyboard - and _	0x2D	Keyboard Page Up	0x4B	Keyboard F14	0x69
Keyboard m and M	0x10	Keyboard = and +	0x2E	Keyboard Delete Forward	0x4C	Keyboard F15	0x6A
Keyboard n and N	0x11	Keyboard [and {	0x2F	Keyboard End	0x4D	Keyboard F16	0x6B
Keyboard o and O	0x12	Keyboard] and }	0x30	Keyboard Page Down	0x4E	Keyboard F17	0x6C
Keyboard p and P	0x13	Keyboard \ and |	0x31	Keyboard Right Arrow	0x4F	Keyboard F18	0x6D

Keyboard q and Q	0x14	Keyboard Non-US # and ~	0x32	Keyboard Left Arrow	0x50	Keyboard F19	0x6E
Keyboard r and R	0x15	Keyboard ; and :	0x33	Keyboard Down Arrow	0x51	Keyboard F20	0x6F
Keyboard s and S	0x16	Keyboard ' and "	0x34	Keyboard Up Arrow	0x52	Keyboard F21	0x70
Keyboard t and T	0x17	Keyboard Grave Accent and Tilde	0x35	Keypad Num Lock and Clear	0x53	Keyboard F22	0x71
Keyboard u and U	0x18	Keyboard , and "<"	0x36	Keypad /	0x54	Keyboard F23	0x72
Keyboard v and V	0x19	Keyboard . and ">"	0x37	Keypad *	0x55	Keyboard F24	0x73
Keyboard w and W	0x1A	Keyboard / and ?	0x38	Keypad -	0x56	Keyboard Left Control	0xE0
Keyboard x and X	0x1B	Keyboard Caps Lock	0x39	Keypad +	0x57	Keyboard Left Shift	0xE1
Keyboard y and Y	0x1C	Keyboard F1	0x3A	Keypad Enter	0x58	Keyboard Left Alt	0xE2
Keyboard z and Z	0x1D	Keyboard F2	0x3B	Keypad 1 and End	0x59	Keyboard Left GUI	0xE3
Keyboard 1 and !	0x1E	Keyboard F3	0x3C	Keypad 2 and Down Arrow	0x5A	Keyboard Right Control	0xE4
Keyboard 2 and @	0x1F	Keyboard F4	0x3D	Keypad 3 and Page Down	0x5B	Keyboard Right Shift	0xE5
Keyboard 3 and #	0x20	Keyboard F5	0x3E	Keypad 4 and Left Arrow	0x5C	Keyboard Right Alt	0xE6
Keyboard 4 and $	0x21	Keyboard F6	0x3F	Keypad 5	0x5D	Keyboard Right GUI	0xE7

Back to Top

Document Revision History

Date	Notes
2017-08-21	fixed typo - missing lower case "z" in Table 1
2017-03-21	New document that new document which describes how to remap keys in macOS 10.12 Sierra.