FAQ

tmk edited this page 3 days ago · 73 revisions

READ FIRST

- **README** of top directory : https://github.com/tmk/tmk_keyboard/blob/master/README.md
- **README** of target project(keyboard/converter) directory.

Note that you'll need to read **both**.

Build

- FAQ/Build

Keymap

- FAQ/Keymap

Debug Console

hid_listen can't recognize device

When debug console of your device is not ready you will see like this:

```
Waiting for device:.........
```

once the device is pluged in then *hid_listen* finds it you will get this message:

```
Waiting for new device:.........................
Listening:
```

Check if you can't get this 'Listening:' message:

- build with `CONSOLE_ENABLE=yes` in **Makefile**.

You may need privilege to access the device on OS like Linux.

- try `sudo hid_listen`

Can't get message on console

Check:

- *hid_listen* finds your device. See above.
- Enable debug with pressing **Magic**+d. See Magic Commands.
- set `debug_enable=true` usually in `matrix_init()` in **matrix.c**.
- try using 'print' function instead of debug print. See **common/print.h**.
- disconnect other devices with console function. See Issue #97.

Linux or UNIX like system requires Super User privilege

Just use 'sudo' to execute *hid_listen* with privilege.

```
$ sudo hid_listen
```

Or add an *udev rule* for TMK devices with placing a file in rules directory. The directory may vary on each system.

File: /etc/udev/rules.d/52-tmk-keyboard.rules(in case of Ubuntu)

```
# tmk keyboard products     https://github.com/tmk/tmk_keyboard
SUBSYSTEMS=="usb", ATTRS{idVendor}=="feed", MODE:="0666"
```

Miscellaneous

NKRO Doesn't work

First you have to compile frimware with this build option `NKRO_ENABLE` in **Makefile**.

Try `Magic` **N** command(`LShift+RShift+N` by default) when **NKRO** still doesn't work. You can use this command to toggle between **NKRO** and **6KRO** mode temporarily. In some situations **NKRO** doesn't work you need to switch to **6KRO** mode, in particular when you are in BIOS.

If your firmeare built with `BOOTMAGIC_ENABLE` you need to turn its switch on by `BootMagic` **N** command(`Space+N` by default). This setting is stored in EEPROM and keeped over power cycles.

https://github.com/tmk/tmk_keyboard#boot-magic-configuration---virtual-dip-switch

TrackPoint needs reset circuit(PS/2 mouse support)

Without reset circuit you will have inconsistent reuslt due to improper initialize of the hardware. See circuit schematic of TPM754.

- http://geekhack.org/index.php?topic=50176.msg1127447#msg1127447
- http://www.mikrocontroller.net/attachment/52583/tpm754.pdf

Can't read comlumn of matrix beyond 16

Use `1UL<<16` intead of `1<<16` in `read_cols()` in **matrix.h** when your columns goes beyond 16.

In C `1` means one of **int** type which is **16bit** in case of AVR so you can't shift left more than 15. You will get unexpected zero when you say `1<<16` . You have to use **unsigned long** type with `1UL` .

http://deskthority.net/workshop-f7/rebuilding-and-redesigning-a-classic-thinkpad-keyboard-t6181-60.html#p146279

Pull-up Resistor

In some case converters needed to have pull-up resistors to work correctly. Place the resistor between VCC and signal line in parallel.

For example:

```
Keyboard       Conveter
               ,------.
5V------+------|VCC   |
        |      |      |
        R      |      |
        |      |      |
Signal--+------|PD0   |
               |      |
GND------------|GND   |
               `------'
R: 1K Ohm resistor
```

https://github.com/tmk/tmk_keyboard/issues/71

Arduino Micro's pin naming is confusing

Note that Arduino Micro PCB marking is different from real AVR port name. D0 of Arduino Micro is not PD0, PD0 is D3. Check schematic yourself.
http://arduino.cc/en/uploads/Main/arduino-micro-schematic.pdf

Bootloader jump doesn't work

Properly configure bootloader size in **Makefile**. With wrong section size bootloader won't probably start with **Magic command** and **Boot Magic**.

```
# Size of Bootloaders in bytes:
#   Atmel DFU loader(ATmega32U4)   4096
#   Atmel DFU loader(AT90USB128)   8192
#   LUFA bootloader(ATmega32U4)    4096
#   Arduino Caterina(ATmega32U4)   4096
#   USBaspLoader(ATmega***)        2048
#   Teensy   halfKay(ATmega32U4)   512
#   Teensy++ halfKay(AT90USB128)   2048
OPT_DEFS += -DBOOTLOADER_SIZE=4096
```

AVR Boot section size are defined by setting **BOOTSZ** fuse in fact. Consult with your MCU datasheet. Note that **Word**(2 bytes) size and address are used in datasheet while TMK uses **Byte**.

AVR Boot section is located at end of Flash memory like the followings.

```
byte     Atmel/LUFA(ATMega32u4)          byte     Atmel(AT90SUB1286)
0x0000   +----------------+               0x00000  +----------------+
         |                |                        |                |
         |                |                        |                |
         |  Application   |                        |  Application   |
         |                |                        |                |
         =                =                        =                =
         |                | 32KB-4KB               |                | 128KB-
0x6000   +----------------+               0x1E000  +----------------+
         |  Bootloader    | 4KB                    |  Bootloader    | 8KB
0x7FFF   +----------------+               0x1FFFF  +----------------+


byte     Teensy(ATMega32u4)              byte     Teensy++(AT90SUB1286)
0x0000   +----------------+               0x00000  +----------------+
         |                |                        |                |
         |                |                        |                |
         |  Application   |                        |  Application   |
         |                |                        |                |
         =                =                        =                =
         |                | 32KB-512B              |                | 128KB-
0x7E00   +----------------+               0x1FC00  +----------------+
         |  Bootloader    | 512B                   |  Bootloader    | 2KB
0x7FFF   +----------------+               0x1FFFF  +----------------+
```

And see this discussion for further reference.
https://github.com/tmk/tmk_keyboard/issues/179

Special Extra key doesn't work(System, Audio control keys)

You need to define `EXTRAKEY_ENABLE` in **makefile** to use them in TMK.

```
EXTRAKEY_ENABLE = yes          # Audio control and System control
```

http://deskthority.net/workshop-f7/tmk-keyboard-firmware-collection-t4478-60.html#p157919

Wakeup from sleep doesn't work

In Windows check `Allow this device to wake the computer` setting in Power **Management property** tab of **Device Manager**. Also check BIOS setting.

Pressing any key during sleep should wake host.

Using Arduino?

Note that Arduino pin naming is different from actual chip. For example, Arduino pin

`D0` is not `PD0`. Check circuit with its schematics yourself.

- http://arduino.cc/en/uploads/Main/arduino-leonardo-schematic_3b.pdf
- http://arduino.cc/en/uploads/Main/arduino-micro-schematic.pdf

Arduino leonardo and micro have **ATMega32U4** and can be used for TMK, though Arduino bootloader may be a problem.

Using PF4-7 pins of USB AVR?

You need to set JTD bit of MCUCR yourself to use PF4-7 as GPIO. Those pins are configured to serve JTAG function by default. MCUs like ATMega*U* or AT90USB* are affeteced with this.

If you are using Teensy this isn't needed. Teensy is shipped with JTAGEN fuse bit unprogrammed to disable the function.

See this code.

```
// JTAG disable for PORT F. write JTD bit twice within four cycles.
MCUCR |= (1<<JTD);
MCUCR |= (1<<JTD);
```

https://github.com/tmk/tmk_keyboard/blob/master/keyboard/hbkb/matrix.c#L67

And read **26.5.1 MCU Control Register – MCUCR** of ATMega32U4 datasheet.

Adding LED indicators of Lock keys

You need your own LED indicators for CapsLock, ScrollLock and NumLock? See this post.

http://deskthority.net/workshop-f7/tmk-keyboard-firmware-collection-t4478-120.html#p191560

Program Arduino Micro/Leonardo

Push reset button and then run command like this within 8 seconds.

```
avrdude -patmega32u4 -cavr109 -b57600 -Uflash:w:adb_usb.hex -P/dev/ttyACM0
```

Device name will vary depending on your system.

http://arduino.cc/en/Main/ArduinoBoardMicro https://geekhack.org/index.php?topic=14290.msg1563867#msg1563867

USB 3 compatibility

I heard some people have a problem with USB 3 port, try USB 2 port.

Mac compatibility

OS X 10.11 and Hub

https://geekhack.org/index.php?topic=14290.msg1884034#msg1884034