Instructions for modifying the Teensyduino 3.2 code for a different keyboard matrix.

Modify the Teensyduino code from one of these Teensy 3.2 example keyboards: Sony VPCEB4, VPCEA, and Dell D630.

Each of the items that need to be modified are listed below.

If you are using Marcel's Python program, its output will automatically provide the information that needs to be added to the Teensyduino code. An example output from his program is given at the end of this document.

Const byte rows_max =

- Set this to the number of rows in your matrix (16 to 26)
- If you have 26 rows (and 8 columns), you are using all of the Teensies 34 I/O signals and must remove the LED on the Teensy 3.2. If you don't, the LED will be a heavy pull down that will fool the keyboard scan into thinking a key has been pressed and not released.

Const byte cols_max =

- Set this to the number of columns in your matrix (probably 8)

Int normal[rows_max][cols_max] = {

- This array should have 8 items on each line and 16 to 26 lines (based on rows_max).
- Transfer every normal key from your matrix table to this array. Yes it's monotonous.
- This array is only for the normal keys, not for Control, Alt, Shift, GUI, or Fn.
- If your matrix table has no normal key in a cell then put a 0 in the array.
- Put a 0 in the cell if your matrix has Control, Alt, Shift, GUI, or Fn keys listed at this location.
- The names given for each key must be as shown in the "All Key Codes" table at: www.pjrc.com/teensy/td_keyboard.html the exception is KEY_MENU, which is not listed on the PJRC table but it does work.
- If your keyboard has a key name that does not exist in the PJRC table, it can't be used.
- The PJRC table uses Tilde for the back tick `key (also known as grave accent key).

int modifier[rows_max][cols_max] = {

- This array should have 8 items on each line and 16 to 26 lines (based on rows_max).
- Transfer every modifier key from your matrix table to this array.
- If your matrix table has a normal key or no key listed in the cell, put a 0 in this position.

- The names for the modifier keys are as listed in the PJRC table except the "lefts" listed below:
- MODIFIER_LEFT_CTRL, MODIFIER_LEFT_SHIFT, & MODIFIER_LEFT_ALT are missing from the PJRC table but they work fine.
- MODIFIER_FN has been defined by me at the top of this code so I can watch for it in case anyone wants to add multimedia or other Fn things. The Fn key by itself is not sent over USB.

Int media[rows_max][cols_max] = {

- This array should have 8 items on each line and 16 to 26 lines (based on rows_max).
- This table is for the media keys and any other key that are accessed by holding down the Fn key
- You can only use items that are listed in the PJRC "All Key Codes" table for the Normal, Media Player, and System Control Keys.
- Put a 0 in the matrix if the key has no Fn function or if the function is not supported by PJRC.

You can see the key code definitions that Teensyduino loaded on your PC at:

C:\Program Files (x86)\Arduino\hardware\teensy\avr\cores\teensy3\keylayouts.h

boolean old_key[rows_max][cols_max] = {

- This array should have 8 ones on each line and 16 to 26 lines (based on rows_max).

int Row_IO[rows_max] = {   };

- Use the 3.2 translation table on the next page to convert each of the FPC pin numbers to Teensy 3.2 I/O numbers starting from the first row in your matrix table down to the last row.

int Col_IO[cols_max] = {   };

- Use the 3.2 translation table on the next page to convert each of the FPC pin numbers to Teensy 3.2 I/O numbers starting from the first column in your matrix table to the last column.

Comment the following CAPS LOCK LED lines of code if your keyboard uses FPC pin 34 (the LED I/O Pin):

#define CAPS_LED 13 // Teensy LED shows Caps-Lock

if (keyboard_leds & 1<<1) { // mask off all bits but D1 and test if set

 go_1(CAPS_LED); // turn on the LED

```
}
else {
  go_0(CAPS_LED); // turn off the LED
}
```

Teensy 3.2 Translation Table

FPC Pin #	Teensy 3.2 I/O #
1	23
2	0
3	22
4	1
5	21
6	2
7	20
8	3
9	19
10	4
11	18
12	5
13	17
14	6
15	24
16	7
17	25
18	8
19	33
20	9
21	26
22	10
23	27
24	11
25	28
26	12
27	32
28	31
29	30
30	29
31	16
32	15
33	14
34	13

An Example from Marcel's Python Program is given below:

```
------------------------------------------------------
Results:
------------------------------------------------------
```

```
FPC PINS:                          Keyboard FPC Input and
                                   Output pins
8 input pins:
[18, 19, 20, 21, 22, 23, 24, 25]

17 output pins:
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17]
------------------------------------------------------
```

```
TEENSY PINS (these have to be copied to the arduino file):
            cols_max=8
                                   FPC pins translated to Teensy
8 input pins:                      I/O pins
[8, 16, 9, 15, 10, 14, 11, 26]

            rows_max=17
17 output pins:
[23, 0, 22, 1, 24, 2, 21, 3, 25, 4, 20, 5, 19, 6, 18, 7, 17]
------------------------------------------------------
```

```
Normal Keys in a row column matrix

------------------------------------------------------
KEY   copy into int normal[rows_max][cols_max]=
------------------------------------------------------
{
{0,KEY_INSERT,0,KEY_F12,0,0,0,KEY_RIGHT},
{0,KEY_DELETE,0,KEY_F11,0,0,0,KEY_DOWN},
{KEY_UP,KEY_HOME,KEY_MENU,KEY_END,0,0,0,KEY_LEFT},
{0,KEY_F8,KEY_F7,KEY_9,KEY_O,KEY_L,KEY_PERIOD,0},
{KEY_QUOTE,KEY_MINUS,KEY_LEFT_BRACE,KEY_0,KEY_P,KEY_SEMICOLON,0,KEY_SLASH},
{KEY_F6,KEY_EQUAL,KEY_RIGHT_BRACE,KEY_8,KEY_I,KEY_K,KEY_COMMA,0},
{KEY_H,KEY_6,KEY_Y,KEY_7,KEY_U,KEY_J,KEY_M,KEY_N},
{KEY_F5,KEY_F9,KEY_BACKSPACE,KEY_F10,0,KEY_BACKSLASH,KEY_ENTER,KEY_SPACE},
{KEY_G,KEY_5,KEY_T,KEY_4,KEY_R,KEY_F,KEY_V,KEY_B},
{KEY_F4,KEY_F2,KEY_F3,KEY_3,KEY_E,KEY_D,KEY_C,0},
{0,KEY_F1,KEY_CAPS_LOCK,KEY_2,KEY_W,KEY_S,KEY_X,0},
{KEY_ESC,KEY_TILDE,KEY_TAB,KEY_1,KEY_Q,KEY_A,KEY_Z,0},
{0,0,0,KEY_PRINTSCREEN,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,KEY_PAGE_UP,KEY_PAGE_DOWN,0,0},
{0,0,0,0,0,0,0,0},
}
```

Modifier Keys in a row column matrix

```
----------------------------------------------------
MODIFIER Copy to int modifier[rows_max][cols_max]=
----------------------------------------------------
{
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{MODIFIERKEY_LEFT_ALT,0,0,0,0,0,0,MODIFIERKEY_RIGHT_ALT},
{0,0,MODIFIERKEY_LEFT_SHIFT,0,0,0,MODIFIERKEY_RIGHT_SHIFT,0},
{0,MODIFIERKEY_LEFT_CTRL,0,0,0,0,MODIFIERKEY_RIGHT_CTRL,0},
{0,0,0,MODIFIERKEY_GUI,0,0,0,0},
{0,0,0,0,0,MODIFIERKEY_FN,0,0},
}
```

Media Fn keys in a row column matrix

```
----------------------------------------------------
FN   Copy to int media[rows_max][cols_max]=
----------------------------------------------------
{
{0,0,0,KEY_MEDIA_NEXT_TRACK,0,0,0,0},
{0,0,0,KEY_MEDIA_PLAY_PAUSE,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,KEY_MEDIA_VOLUME_DEC,KEY_MEDIA_MUTE,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,KEY_MEDIA_VOLUME_INC,0,KEY_MEDIA_PREV_TRACK,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,KEY_MEDIA_EJECT,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0},
}
```

old_key matrix	copy to
ONE boolean old_key[rows_max][cols_max]=	

```
{
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
}
```