

Instructions for using Marcel's Python program

Download Marcel's [matrixgenerator.py](#) program to a computer that can run Python 3. I used to use a Raspberry Pi since the Thonny IDE comes standard in the OS. This would involve transferring files back and forth to my PC with a memory stick. Lately I've been using Thonny on my Windows PC since I've loaded it for Pi Pico projects. You can download Thonny for Windows, Mac, or Linux [here](#).

Load the completed pin list file from step 11 in the same directory as the Python program so the startup menu can easily find it. Double click on the matrixgenerator.py program to launch Thonny.

On a Raspberry Pi, select "Build", "Compile" and then "Build", "Execute".

On a PC, select "Run", "Select interpreter", "The same interpreter which runs Thonny (default)". This last step is important because it's usually set to "MicroPython (Raspberry Pi Pico)". Next select "Run", "Run current script".

The start menu of the program is shown below. It lists all the .txt files in the folder and asks you to enter the index number for the file you want. Next it will ask you to enter the index number for the Teensy you are using so it can do the correct pin translation.

Marcel's Python program - Start Menu

```
index          file name
  1  Keyboard_without_number_pad.txt
-----
Enter the index number of the *.txt file you want.
OR: enter your own filepath:
1
-----

index  teensy device
  1      LC
  2     3.2
  3     4.0
  4     4.1
  5    2.0++
-----

The pin layout is different for each teensy version.
Please enter the index number of your teensy version:
1
```

The program then gives the results in the terminal window (example shown below) which you can copy and paste into a text file. The first results are the FPC connector pins that are inputs and outputs. Next it translates the pin numbers to Teensy I/O numbers according to the appropriate table shown in step 14. And finally it places all the key names in arrays for easy copy and paste into a USB keyboard routine.

Download one of the keyboard routines from my [repo](#) that is similar to your keyboard so you can modify it with the results from the Python program. If you are using a Teensy 2.0++ with your keyboard, search for "int" in the code and make them all "unsigned int" or it will give compilation errors.

An example output from Marcel's program (with added notes) is given below.

Results:

<pre>FPC 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]</pre>	<p>Keyboard FPC Input and Output pins</p>
<pre>TEENSY PINS (these have to be copied to the <u>arduino</u> file): cols_max=8 8 input 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]</pre>	<p>FPC pins translated to Teensy I/O pins</p>

In the USB Keyboard code, look for the line:

```
const byte cols_max = ;
```

For this example it should be set to 8;

Look for the line:

```
const byte rows_max = ;
```

For this example it should be set to 17;

Look for the line:

```
int Col_IO[cols_max] = {      }; // unsigned int is required for 2.0++
```

For this example it should list pins 8,16,9,15,10,14,11,26 inside the curly brackets

Look for the line:

```
int Row_IO[rows_max] = {      }; // unsigned int is required for 2.0++
```

For this example it should list pins 23,0,22,1,24,2,21,3,25,4,20,5,19,6,18,7,17 inside the curly brackets

The normal, modifier, media, and old_key matrixes are provided by Marcel's Python program and should be copied and pasted over the top of the existing array values as shown in the following screen captures.

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},
{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,0,0,0,0,0,0,0,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},
{1,1,1,1,1,1,1,1},
{1,1,1,1,1,1,1,1},
}
```