get.vial.today /docs/custom_keycode.html

Custom Keycodes

1. Porting guide
2. Custom Keycode

Custom keycodes in QMK

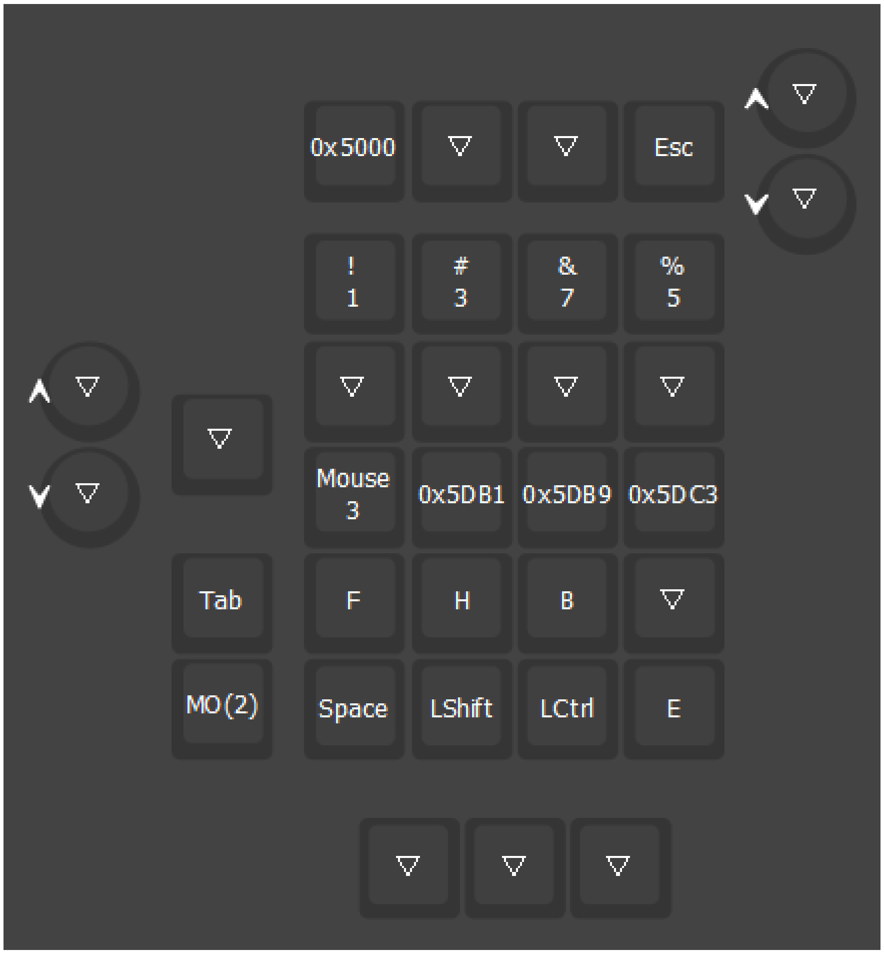
QMK provides a way for user to redefine behavior for existing key or create new keycode.

For example, here 3 custom keycodes are defined within `keymap.c`. An `enum` block is used to assign each keycode a unique integer code, and then the behavior of each keycode is defined inside `process_record_user()`. The keycodes are then assigned inside the keymap as usual.

```
enum blender_keycode {
    B_VPAN = SAFE_RANGE,
    B_DOLLY,
    B_UNDO,
}

bool process_record_user(uint16_t keycode, keyrecord_t *record) {
        switch (keycode) {
                case B_VPAN:
            if (record->event.pressed) { ... }
            return 0;
        case B_DOLLY:
            if (record->event.pressed) { ... }
            return 0;
        case B_UNDO:
            if (record->event.pressed) { ... }
            return 0;
        }
}
```

When you load a board with custom keycodes, they will appear as unique hexcode. You can use the “any key” button and type in values starting with `0x7E40` to assign them but this isn’t very user friendly.



Custom keycodes in Vial

Vial provides a way to alias custom keycodes so that they appear as a named keycode you can assign in the GUI.

In `vial.json` you can create a new list named `customKeycodes` with each of the keycodes having three properties:

- `shortName`: the name that will appear on the key in the GUI; it needs to be short otherwise the text will overflow.
- `title`: a short description that will appear when you hover over the keycode
- `name`: appears before title

For those three keycodes you'd have something like this in your `vial.json`. Watch out for any syntax errors. The last item in the list should not have a comma.

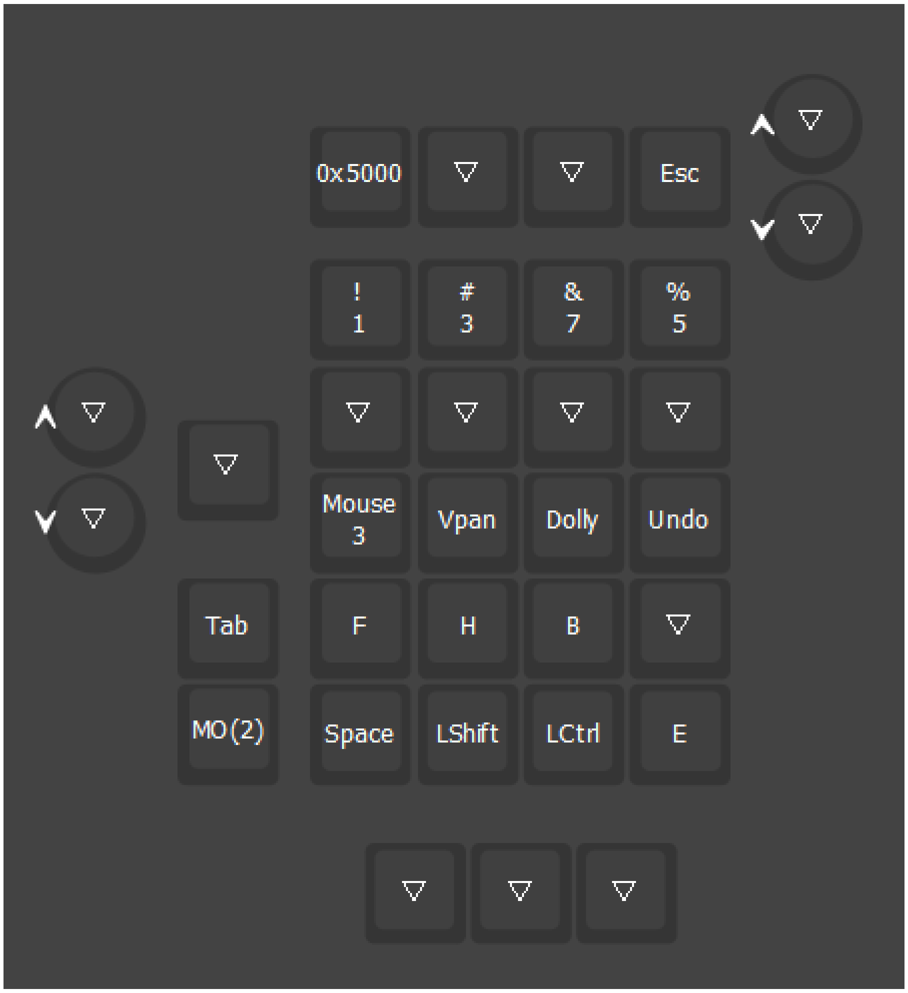
```
....
"customKeycodes": [
        {"name": "Vpan",
         "title": "Viewport pan",
         "shortName": "B_VPAN"
        },
        {"name": "Dolly",
         "title": "Dolly zoom",
         "shortName": "B_DOLLY"
        },
        {"name": "Undo",
         "title": "",
         "shortName": "B_UNDO"
        }
],
"matrix": { "rows": 6, "cols": 5 },
....
```

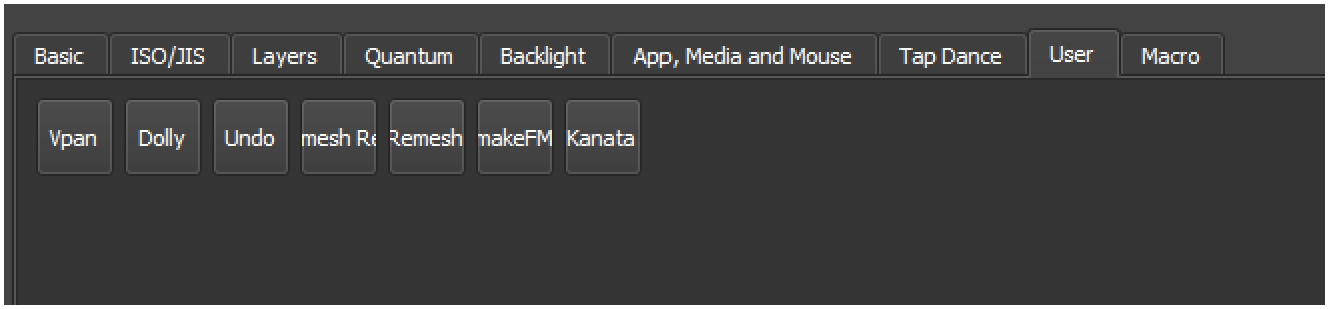
In the keymap.c file, replace `SAFE_RANGE` with `QK_KB_0`:

```
enum blender_keycode {
    B_VPAN = QK_KB_0,
    B_DOLLY,
    B_UNDO,
}
```

Custom keycodes in the json file **must** match what’s inside the enum block, both in order and number of keycodes.

If everything goes smoothly the hexcode will be replaced with name of the keycode. You can access all your custom keycode inside the User tab. Do note that the firmware size will increase slightly (this example is ~100bytes).


0x5000
▽
▽
Esc
▽
▽
!
1
#
3
&
7
%
5
▽
▽
▽
▽
▽
▽
▽
Mouse
3
Vpan
Dolly
Undo
Tab
F
H
B
▽
MO(2)
Space
LShift
LCtrl
E
▽
▽
▽


Basic
ISO/JIS
Layers
Quantum
Backlight
App, Media and Mouse
Tap Dance
User
Macro
Vpan
Dolly
Undo
mesh Re
Remesh
makeFM
Kanata