6 Create a Touch Keyboard

Important differences about touch keyboards

In this tutorial, we distinguish between **touch** keyboards, which are pressed by fingers on a screen, and **typing** keyboards which involve tapping real keyboard keys.

Touch keyboard layouts must be treated differently than typing keyboards because they are primarily visual keyboards. Typing keyboards are usually learned by the feel of the keys and the use of muscle memory (where a repeated action becomes automatic), which is why sequences of keys are often used to produce a single character. In addition, for many typing keyboards, the user is expected to memorize a series of keystrokes to produce a desired character. For example, when laying out a typing keyboard you might use the following sequence to produce the letter ɓ:

; + b = ɓ

(that is, typing ";" followed by "b" produces "ɓ"). Users must have a typing guide of some sort to remind them, but this combination is quickly learned. The combination might use a deadkey, where the semicolon (the deadkey) produces no visual output on the screen, but typing "b" produces "ɓ". Or the keyboard might display the semicolon when it is typed, then replace it with "ɓ" when "b" is pressed. Typists will master this quickly and throw away the guide. Experienced typists won't need to look at the hardware keys when they type.

On a touch keyboard, however, there is an understanding that WYTIWYG (What you touch is what you get). That means that the letter ɓ ought to be visible somewhere on the keyboard. We'll look at the options that we have for touch keyboards later in this document.

Plan beyond the touchscreen

You may think that you only want to create a touch keyboard layout and don't want to bother with setting up a typing keyboard layout. However, the distinction between a computer and a mobile device is becoming less and less all the time. Often a tablet will be paired with a bluetooth keyboard to enable the user to type faster. The minute this happens, your keyboard will not work as expected. Long press features don't work on the attached keyboard. That is why when you create a keyboard package in Keyman Developer it is always assumed that you will be producing both and bundling them together.

Touch layout

At the top of the touch layout area, you can select "phone" or "tablet". These are represented by iPhone and iPad graphical representations, but they will work on Android as well. You can select vertical or horizontal mode as well.

You have the flexibility to have a different number of rows or columns of keys on the tablet and phone layouts, and so Keyman keeps them separate. If you want the exact same key structure on both, then delete one of the two (either "phone" or "tablet") and complete the other. It is a lot of extra work to maintain two identical layouts. Keyman will automatically use "tablet" if "phone" is not present or "phone" if "tablet" is not present. Only if you want to make the "tablet" and "phone" layouts different, would you want to have both.

On the touch layout tab, we can start handling the additional characters. Remembering that all special characters on a touch keyboard must be visible somewhere, there are at least four options to add them to an existing keyboard. In the case of adding a hooked b to a latin keyboard you could:

1. Place the ɓ on an unused key in the underlying keyboard, such as swapping v for ɓ, as v is not used in Fulfulde.
2. Place the ɓ on a long-press pop-up so that long-pressing b gives the option to select it.
3. Place the ɓ on a different layer that appears when the activating key is pressed. Keyman has some different layer names (based on keyboard modifiers) that are predefined, plus a "numeric" layer, but

custom layer names can be added.
4. Add a dedicated ɓ key to your keyboard in another row or another column.

Let’s look at each of these more closely to decide.

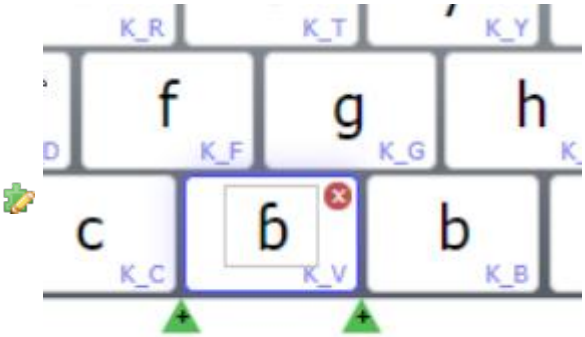
Swap with unused characters

In Fulfulde, the v, z, x letters are not used. So it may seem like a good idea to swap unused letter for the ones they use. However, in our global information age we must include access to a v even if it is not used frequently. The minute the user tries to pull up Vimeo to watch a video, they will be prevented from typing the URL! But since the v would be used quite infrequently, it would be possible to make the v available on the long-press popup for ɓ.

To swap v for ɓ, open the Keyman character map, and then drag the hooked b to the key.

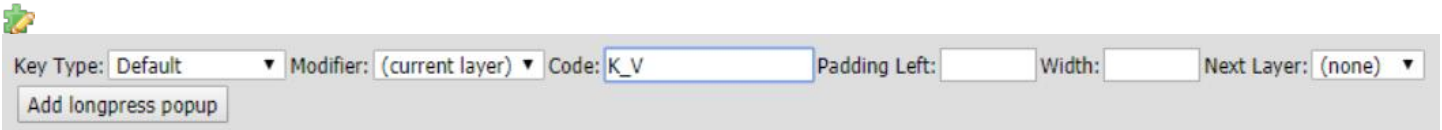
NB: When dragging from the character map, you can get the character, the code or the name. Right-click on the character, select "Insert Mode" from the menu, and pick one of the choices: "Code" (U+0253), "Character" (ɓ) or "Name" ($LATIN_SMALL_LETTER_B_WITH_HOOK).

After the letter appears on the key, hit enter to save it.



All this does is change the character displayed on the key. You’ll notice that the output is listed in the bottom right corner of the key: K_V which is the virtual key name for V. Note that keys on the touch layout with K_ names will use the behavior of the corresponding key in the .kmn file.

To change the output of the key, we need to look at the options below the keyboard when the key is selected:

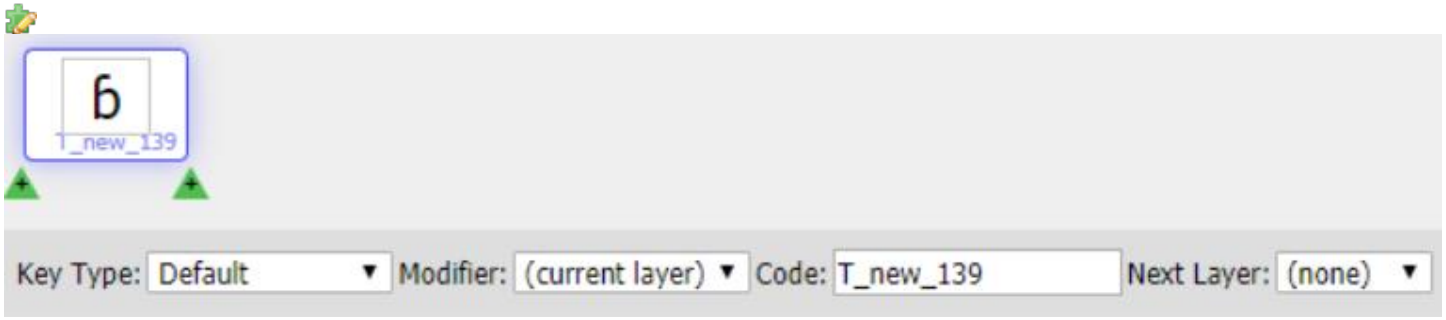


<strike< strike="">>- end tiki_plugin >></strike<>
In the code box, we can replace the K_V and reference the Unicode key value: U_0253. Note that it uses the underscore and not the plus symbol. Now, pressing that key would produce the hooked b. Remembering what we said about the danger of making the letter v inaccessible, we would need to add the v back as a long-press key. If you decide this is the best approach, long-press keys are covered next.

Longpress keys

The other approach in this scenario would be to place the hooked variants on longpress popup keys, such that a long press on b would offer the hooked b.

To do this, you would select the "b" key, and press the "Add longpress popup" button below. This opens another row of options below, starting with a new button. You can drag and drop the hooked b to that button and hit Enter.



You'll notice that the default code output by longpress keys is an automatically generated T variable. That is because Keyman doesn't try to guess what you are adding here—there are no defined longpress popup keys for your underlying keyboard. You can output the unicode value directly instead: U_0253.

If you wanted to add more longpress keys to this popup, you would click on the green triangle to the left or right to create a new programmable key on either side. To see what this looks like, click on the period key which has 8 more punctuation keys that pop up.

Touch key outputs

There are three types of codes to output from touch keys. Unicode characters, Keycodes, and Variables

Unicode characters

Any Unicode value may be represented by U_xxxx in the code line, but please note that only a single Unicode value may be output. For any output of multiple code points, you will have to use a variable and write a rule. See variable below.

Keycodes

Since you have also created a physical keyboard, you can invoke the same behavior as pressing a physical key by using the Keyman keycode for the code. For example, if your physical keyboard uses the number 1 to add a low tone on a vowel, you just need to put K_1 in the code box. This will perform the same function that the physical keyboard does when pressing the 1 key after a vowel.

Variables

The T variables are used in the coding section, for writing rules. These are always required when trying to output more than one character in the output. For example, the code can contain U_025B to output a single character ɛ, but can't contain U_025B U_0301 to output a composed character ɛ́. You would have to make a rule T_open_e_acute and then include this rule in the code:

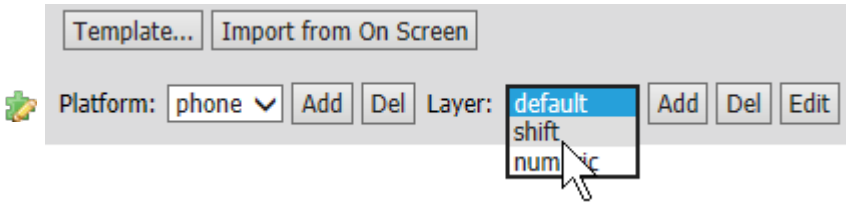
```
+[T_open_e_acute] &gt;&amp;nbsp;U+025B U+0301
```

You should **always rename** the proposed **T_new_xx** code to a named variable that you will recognize, like in the example above. Variable names are not case sensitive.

Additional layers

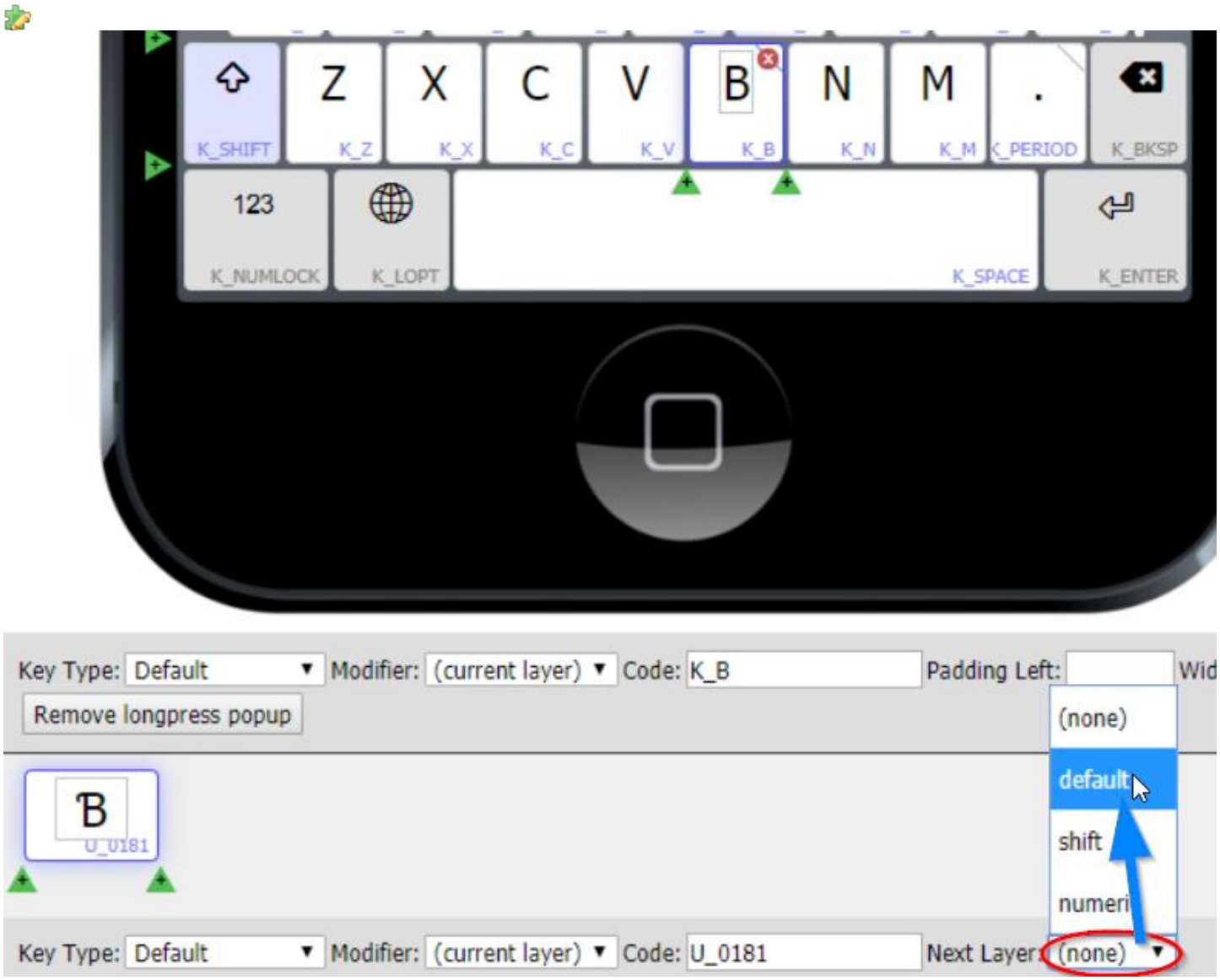
Touch keyboards have different layers. The most obvious is produced by the shift key, which when pressed, changes the characters produced by all the keys. But mobile keyboards also tend to have a number key layer. Some have a layer of symbols too. If you want, Keyman will allow you to create your own custom layer which can be produced by pressing a designated key of your choice. We won't take the time to do that now, but we will show you how to work in another layer.

The hooked b also has an uppercase version. This needs to be added after you switch the keyboard to the shift layer. At the top of the window is the Layer selector:



<strike< strike="">>- end tiki_plugin >></strike<>

Once the shift layer is selected, you can now edit the shifted keys, and may add a longpress popup key to the uppercase B like shown here:



<strike< strike="">>- end tiki_plugin <strike<
strike="">>></strike<></strike<>
The Next Layer select box tells the keyboard whether you want to change layers after pressing the key. By default, it is set to (none). If you leave it set to (none), when you type the capital hooked B the keyboard will remain on the shift layer and the next letter typed will also be a capital.

On most phones using Roman scripts, when you press shift and type a letter, the shift state then goes back to lowercase. If you want to have similar functionality on your keyboard, you need to set the Next Layer field to "default" (that is, the lowercase layer) for all the keys on the shift layer.

Squeeze in another key

If you are designing a tablet-sized keyboard, you can squeeze extra characters in. On a phone-sized keyboard, you could also do this, but then increase the chance of users missing the keys because they are so small. Using the green triangles next to any selected key, you can add extra characters. This is what

our keyboard would look like if we just added extra keys for 3 of our 5 special characters:



<strike< strike="">>- end tiki_plugin >></strike<>
It just so happens that adding the hooked y, d, and b to each row was consistent. To add the ŋ and the ɲ, we can place them on a longpress n.

How Longpress Keys are Shown

Even when no keys are selected in the keyboard layout, I can still see which ones have longpress popups, because Keyman Developer shows a faint gray line across the top right corner of the key. So you should have a faint diagonal line on your y, d, b, and n keys.

<strike< strike="">>- end tiki_plugin --></strike<>