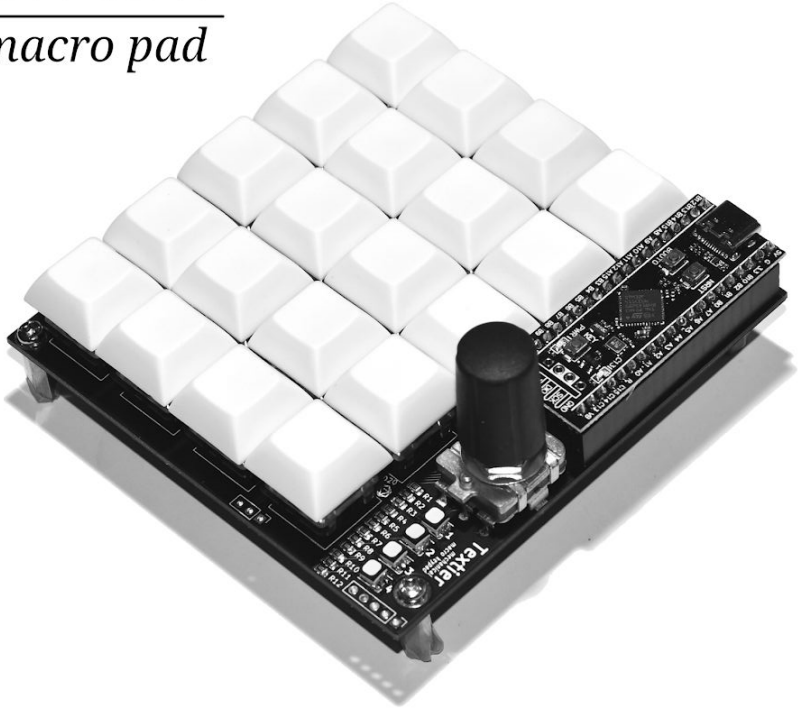
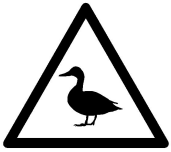
Textier

mechanical macro pad



Textier comes with ABSOLUTELY NO WARRANTY to the extent permitted by applicable law. For the purposes of US export law, Textier is a small aircraft. For the purposes of UK tax law, Textier is a gift. Names, characters, business, events and incidents are the products of the author's imagination. Any resemblance to actual persons, living or dead, or actual events is purely coincidental. Textier is known to the State of California to cause cancer or reproductive toxicity. Textier's hardware is open source; it may not fall under the definition of open source hardware. No animals were harmed in the making of Textier. This disclaimer was scanned for viruses. Please consider the environment before photocopying this disclaimer. No responsibility is accepted for inaccuracies in this disclaimer.

Safe cations



- Use the ducc or dance with these opera structs
- Do not let children play
- Keep aging away from children
- Never charge non batteries
- Never throw bats into the fire
- Before use, check for visible mage. Do not install or use a maged device.
- They are watching.
- Repairs can only be undertaken (never completed)
- No parts inside.

Congratulations $CUSTOMER on becoming a proud owner of this ***Textier™*** Mechanical Macro Pad! You are now in possession of the best 20-key CircuitPython keyboard to come out of the south-west of England! Every single ***Textier®*** is handmade. Thank you for making this project possible.

Date	**Unit**
/ /	# /5

Known Issues

"Blackpill" board

- STM32F411CE only has two USB endpoints - not enough for CircuitPython's MSC & CDC in addition to HID and/or AUDIO (MIDI).
    - Textier ships with MSC disabled in CircuitPython, requiring the use of ampy[1] to transfer files.
- Some LED IO is shared with the optional flash memory footprint.
    - Flash memory is not populated.
- Wide DSA profile keycaps conflict with the edge of the Blackpill if it is fully inserted in the socket.
    - Lifting the Blackpill slightly out of the socket avoids this while maintaining electrical connection to the PCB

These issues are all addressed by Greenpill[2].

"Blackpill" CircuitPython port

- `rotario` library is not implemented for STM devices
- Pin C15 is shown as "in use" when trying to initialise NeoPixels library.

Textier board

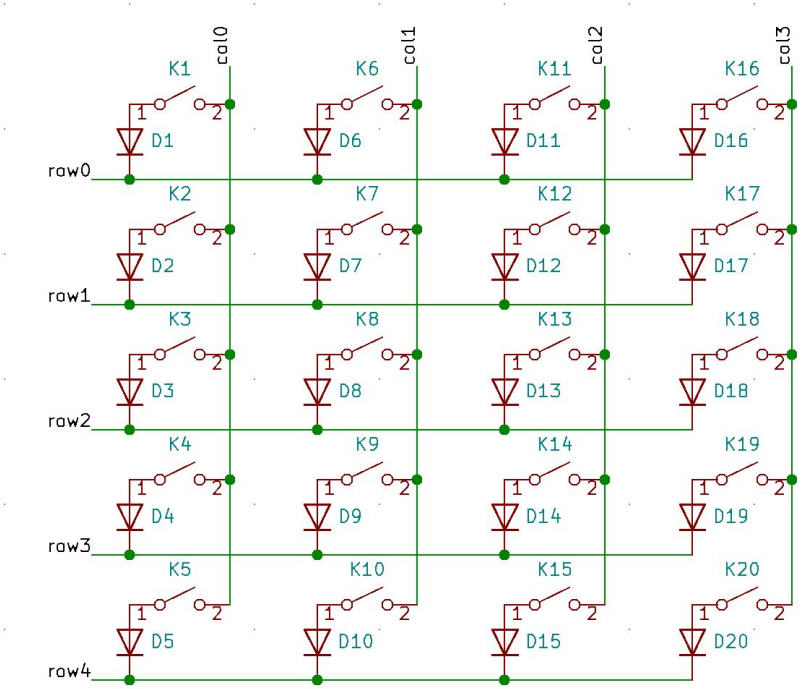
- LED pinout does not match silkscreen on reverse of board

[1] https://github.com/scientifichackers/ampy
[2] https://github.com/stary2001/greenpill

- A corrected pinout is available at the end of this manual
- Some SK2812b underglow LEDs may be fucked (ESD?)
  - An effort to repair/replace these will be made when the world is a bit less **global situation**-y

---

Key matrix



The key matrix is connected as shown. The use of diodes means that `col0-3` must be set as outputs, and `row0-4` must be set as inputs (with internal resistors set to pull-down). To scan the matrix, set one column high at a time, and read the rows. To set

up the matrix GPIO, use `textier.matrix_init()`, after which `textier.matrix_init()` can be used to get the set of keycodes[3] that are currently held down.

Example

```
import textier
textier.matrix_init()
keys_old = set()
while True:
     keys_new = textier.matrix_scan()
     print(keys_old - keys_new, “ released!”)
     print(keys_new - keys_old, “ pressed!”)
```

Indicator LEDs

There are four RGB leds on the top of Textier. At full current they are extremely bright, so it is highly advised to dim them with PWM. The pin mappings are given below (corrected from the table printed on the bottom of Textier).

	Red	**Green**	**Blue**
LED1	B1	B0	A8
LED2	A7	A6	B6
LED3	A3	A2	A1
LED4	B7	B8	B9

[3] Keycodes are numbered from zero by row, then by column.

Example

```
import texter, pulseio
LED1 = tuple(map(pulseio.PWMOut, textier.LEDs[0]))
LED1[0].duty_cycle = 64 * 2**8
LED1[1].duty_cycle = 32 * 2**8
LED1[2].duty_cycle = 16 * 2**8
```

Sudoku

			2			3		1
	2	3					8	
7	1		8		3			
			7					4
2				8	1			
8		7		6		1	5	
9		6						
	5		3			2		8
	8		4		7			9

made with <3 to the best of my ability
*ldm