EMOJIBOARD

Standalone feedback collecting keyboard with interchangeable emojis.

Version 2.0

Timo Luukkonen

University of Lapland

User Experience Research Group LUX

23.1.2025

1 TABLE OF CONTENTS

Device and Program Documentation

2 OVERVIEW

The hardware setup includes a Raspberry Pi (version 3 B v1.2) connected to:

- A 3x7 + 2x2 button matrix.
- NeoPixel LEDs (26 units).
- A buzzer (connected to GPIO pin 16).
- The program logs keypresses into a CSV file.

The program is implemented in Python and uses libraries such as `RPi.GPIO`, `rpi_ws281x`, and `datetime`. It starts automatically when the device boots using a cron job and saves keypresses to the file `keypress_log.csv` located in the `/home/emojikeyboard` folder.

3 PROGRAM FUNCTIONALITY

3.1 NEOPIXEL LED CONTROL

Each button in the matrix is linked to a specific NeoPixel LED. When a button is pressed, its corresponding NeoPixel lights up green and returns to its default state once the button is released. NeoPixel 10 is programmatically set to remain off at all times and is not used in the program.

3.2 BUTTON MATRIX

The matrix is divided into two sections:

1. A 3x7 matrix.
2. A 2x2 matrix.

The buttons are assigned the following functions:

- BACKSPACE: Clears the keypresses in the current session.
- RETURN: Saves the current session data and starts a new session.
- Other buttons: Add the pressed key to the current session.
- Key to Emojis conversion table attached to this document.

3.3 KEYPRESS LOGGING

The program logs each session in the following format:

- A username (randomly generated combination of an adjective, animal, and unique code).
- Keypresses.
- A timestamp of when the session is saved.

The data is stored in the file `keypress_log.csv`.

4 AUTOMATIC PROGRAM STARTUP

The program is configured to run automatically on device boot using a cron job. Add the following line to the user's `crontab` file by running `crontab -e`:

```
@reboot python3 /home/emojikeyboard/keyboard-140.py
```

5 INSTRUCTIONS FOR COPYING *KEYPRESS_LOG.CSV* TO A USB STICK

1. Connect HDMI display and a regular keyboard to the Emojiboard.
2. Connect power cable.
3. Insert the USB stick into the Raspberry Pi.
4. Identify the USB stick's mount point using the command:

   ```
   lsblk
   ```

   The stick is typically listed as `/dev/sda1`.

5. Mount the USB stick using the command (assuming it is `/dev/sda1`):

   ```
   sudo mount /dev/sda1 /mnt
   ```

6. Copy the log file to the USB stick:

   ```
   cp /home/emojikeyboard/keypress_log.csv /mnt
   ```

7. Safely unmount the USB stick:

   ```
   sudo umount /mnt
   ```

8. Remove the USB stick from the device.

6 TROUBLESHOOTING

If the program does not start or the NeoPixel LEDs do not function correctly:

1. Verify that all hardware connections are correct.
2. Ensure the required libraries are installed:
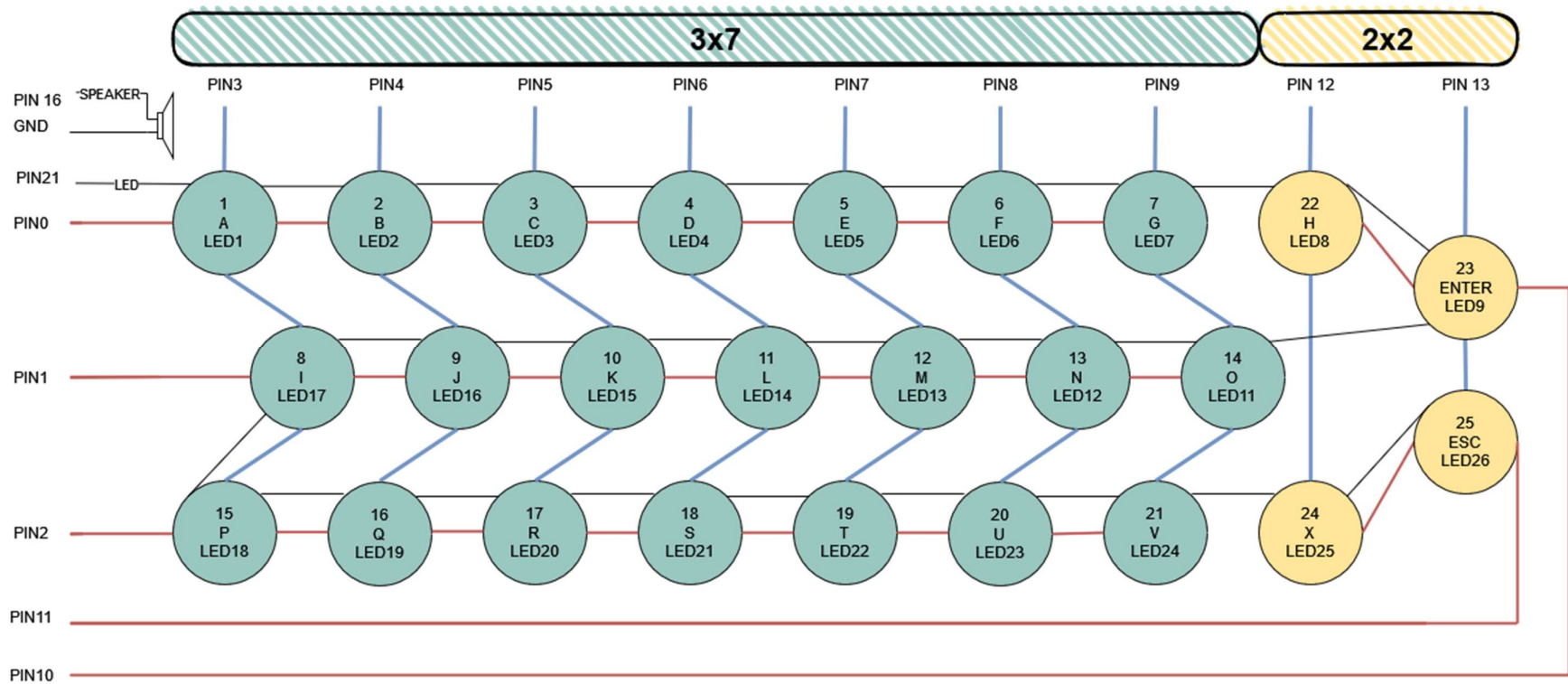
   ```
   pip3 install rpi_ws281x RPi.GPIO
   ```

3. Check `/home/emojiboard/keyboard_debug.log`
4. Check `/var/log/syslog` for error messages.

7 ATTACHEMENTS

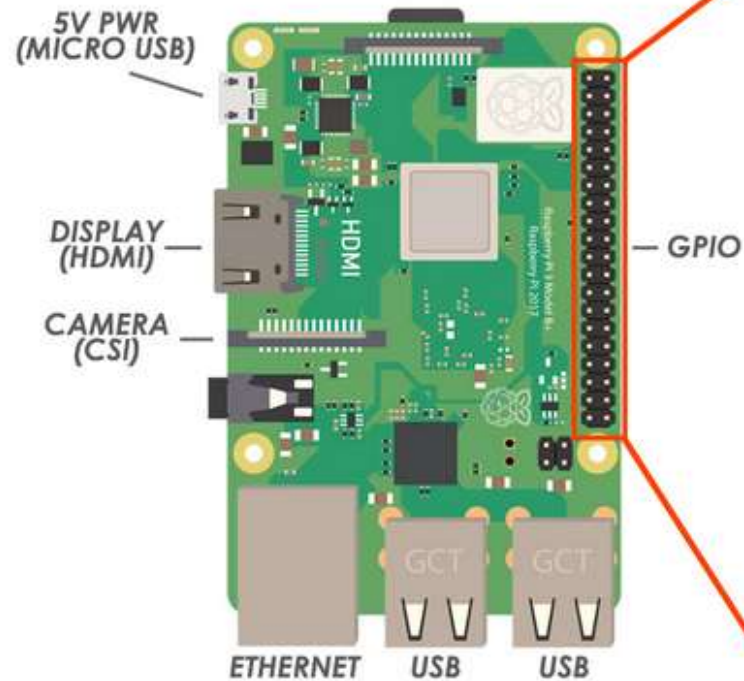## 7.1 LOGICAL TABLE & LETTERS TO EMOJI CONVERSION

Button Number	Button Label	Key Mapped	NeoPixel Number	Matrix	Row	Column	Emoji
1	A	'A'	1	3x7	1	1	😊
2	B	'B'	2	3x7	1	2	😍
3	C	'C'	3	3x7	1	3	🤩
4	D	'D'	4	3x7	1	4	🤑
5	E	'E'	5	3x7	1	5	🤔
6	F	'F'	6	3x7	1	6	😬
7	G	'G'	7	3x7	1	7	😥
8	H	'H'	17	3x7	2	1	😭
9	I	'I'	16	3x7	2	2	😱
10	J	'J'	15	3x7	2	3	🤮
11	K	'K'	14	3x7	2	4	🤬
12	L	'L'	13	3x7	2	5	💩
13	M	'M'	12	3x7	2	6	👍
14	N	'N'	11	3x7	2	7	👎
15	O	'O'	18	3x7	3	1	🦌
16	P	'P'	19	3x7	3	2	🔥
17	Q	'Q'	20	3x7	3	3	🌍
18	R	'R'	21	3x7	3	4	💿
19	S	'S'	22	3x7	3	5	🪓
20	T	'T'	23	3x7	3	6	🪓
21	U	'U'	24	3x7	3	7	❤️
22	V	'V'	8	2x2	1	1	🚫
23	Enter	KEY_RETURN	9	2x2	1	2	
24	X	'X'	25	2x2	2	1	♻️
25	Backspace	KEY_BACKSPACE	26	2x2	2	2	

7.2 Wiring schematic

Timo Luukkonen

University of Lapland

User Experience Research Group LUX

23.1.2025

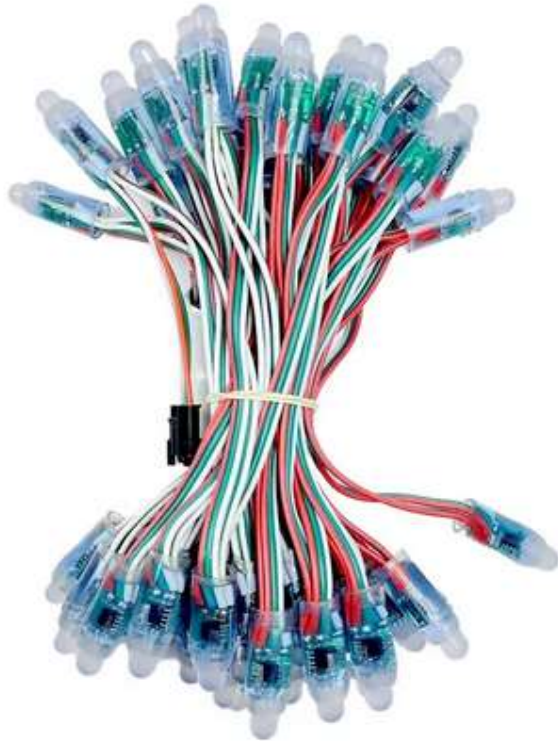# 7.3 RASPERRY PI 3B V. 1.2 PINOUT

YoungWonks

7.4 Button specifications

60mm LED Illuminated Push Button Switch Arcade DC5V/12V
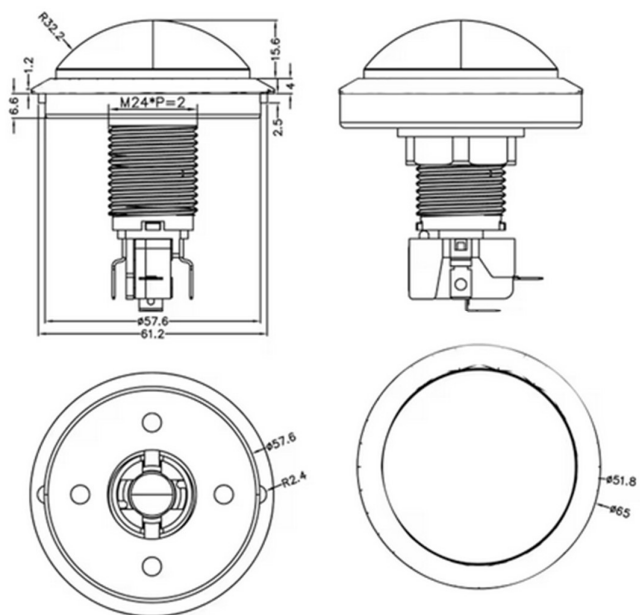
7.5 LED String specification

Dc5v Ws2811 Full Color Led Pixel Light Module 12mm

7.6 Build Photos