SHARPIKEEBO 1.0

RASPBERRY PI ZERO + SHARP LS027B7DH01 Memory Display
Carrier Board + USB ATMEGA34U Keyboard

Description

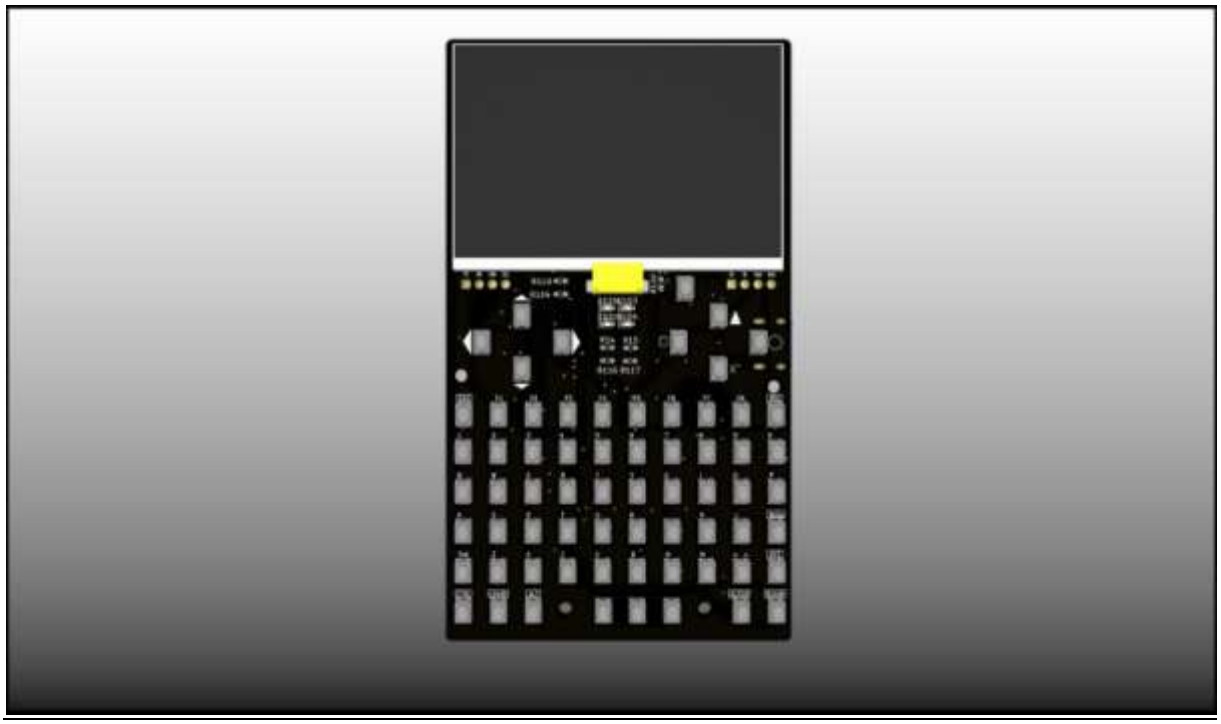
The SHARPIKEEBO carrier board has been designed to hold both a Raspberry(R) PI ZERO 2W or W + 2.7" (400x240) Sharp Memory Display. This display is capable of a fast refresh which allows the use of linux shells or display game images. A true USB Keyboard (driven by QMK firmware, so fully customized) has been added on the board.

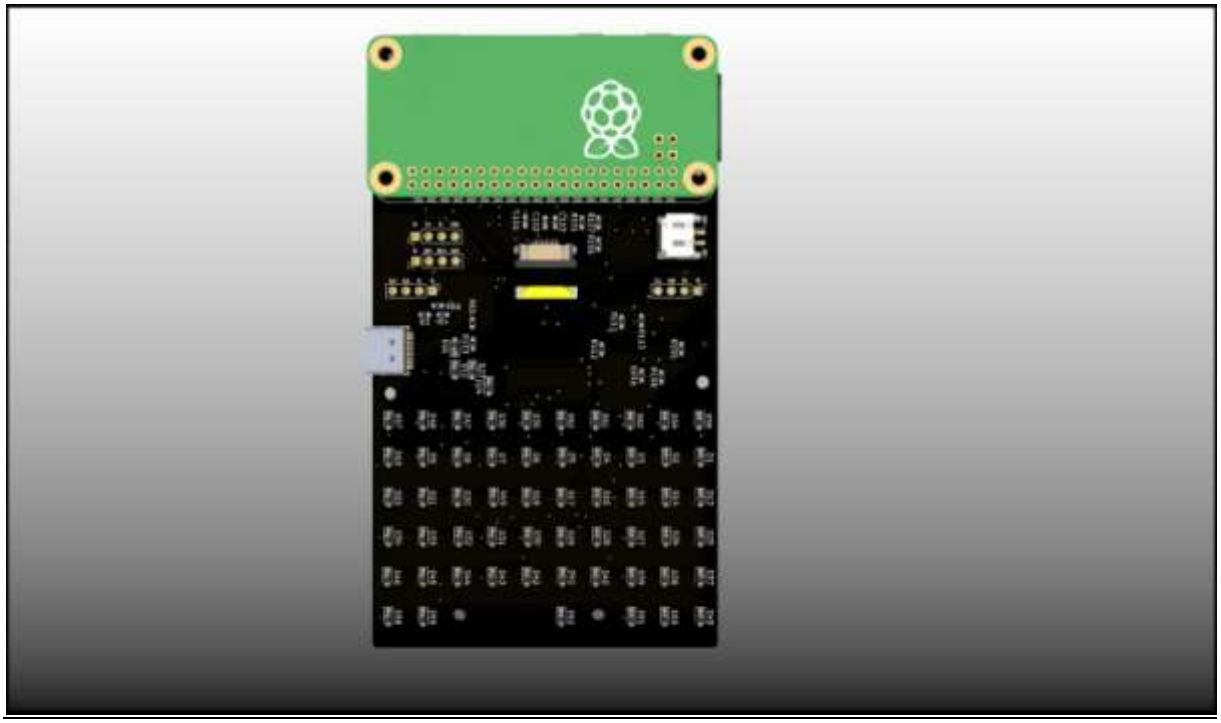
The board also has 4 independently controllable LEDs you can switch on/off with classic GPIO commands (C++ or Python)

The board also has 5 buttons you may also control via GPIO voltage detection (UP/Down voltage detection). You can use this board for gaming.

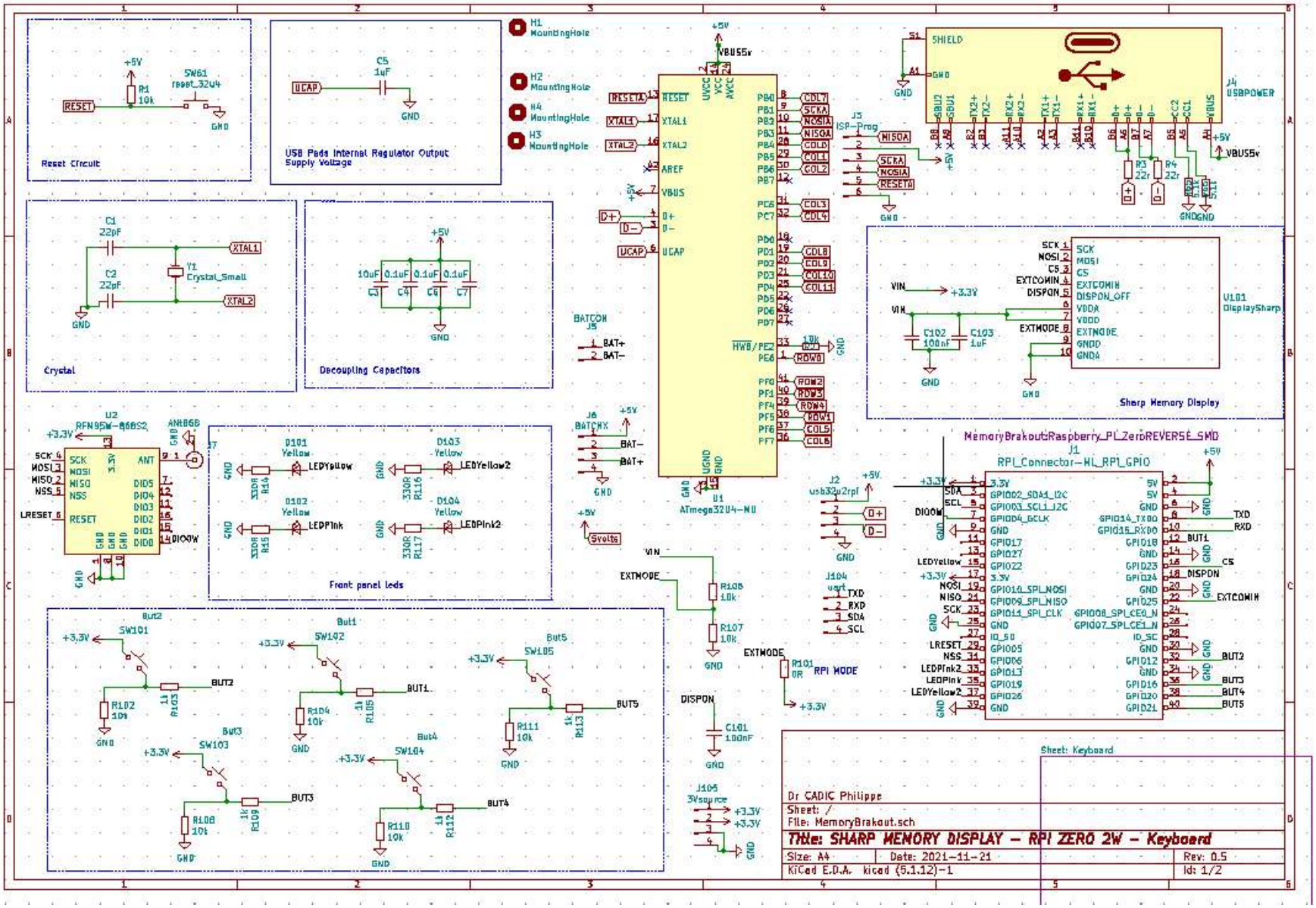
And more, you have UART GPIO and I2C GPIO easily available on board should you want to add a 3.3v or 5v module such a GPS , 4g or 5g modem etc ...

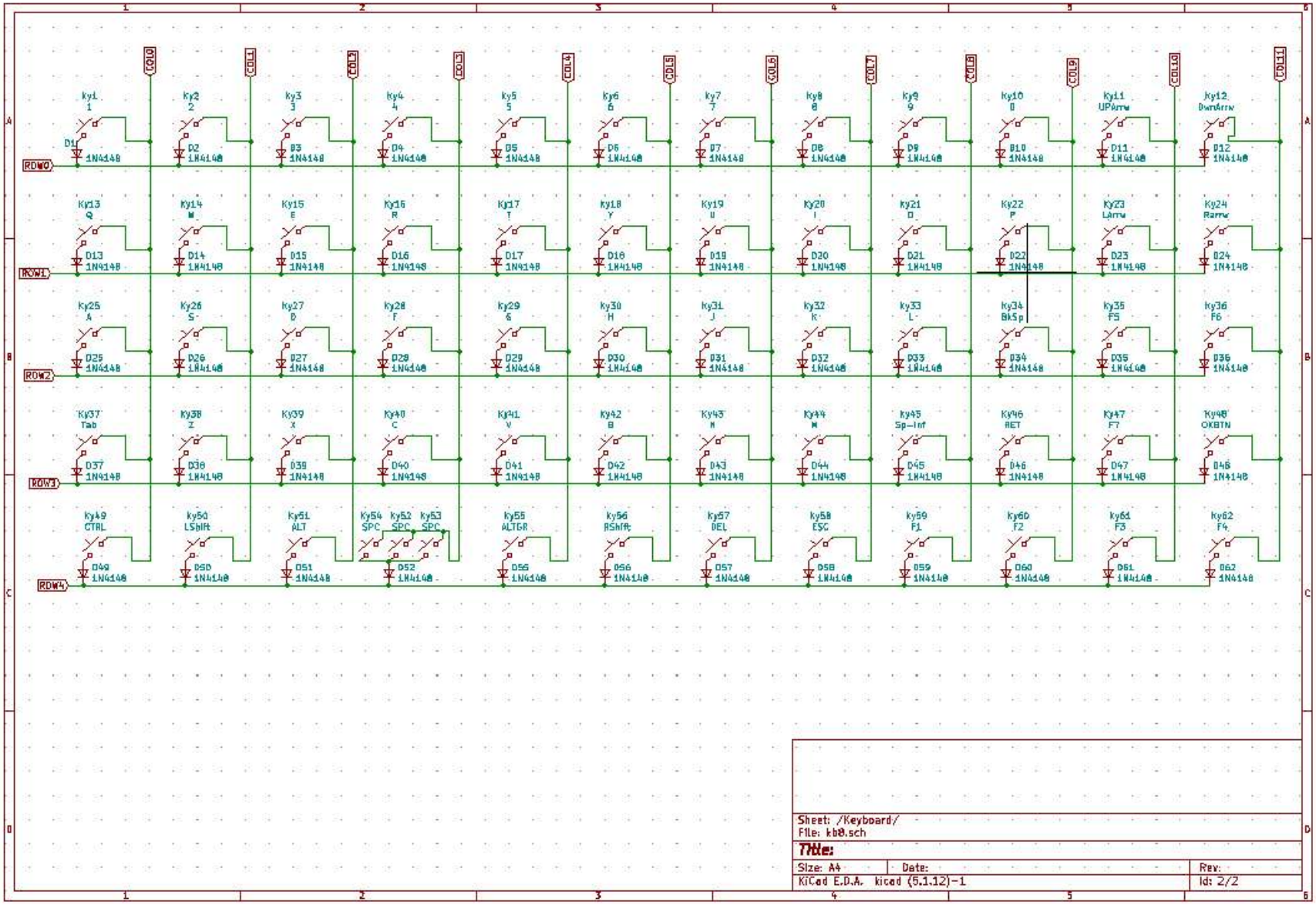
The board + RPI can be powered thought a USB C connector. The board has also been design to behave as a USB keyboard you can plug on a desktop or laptop or rack server.





The SHARPIKEEBO is probably one of the smallest linux computers with keyboard; 400x240 pixels Sharp Memory Display + Raspberry PI Zero 2W + RFM95 long range radio transceiver. **6 cm x 11 cm x 1.5 cm ...**

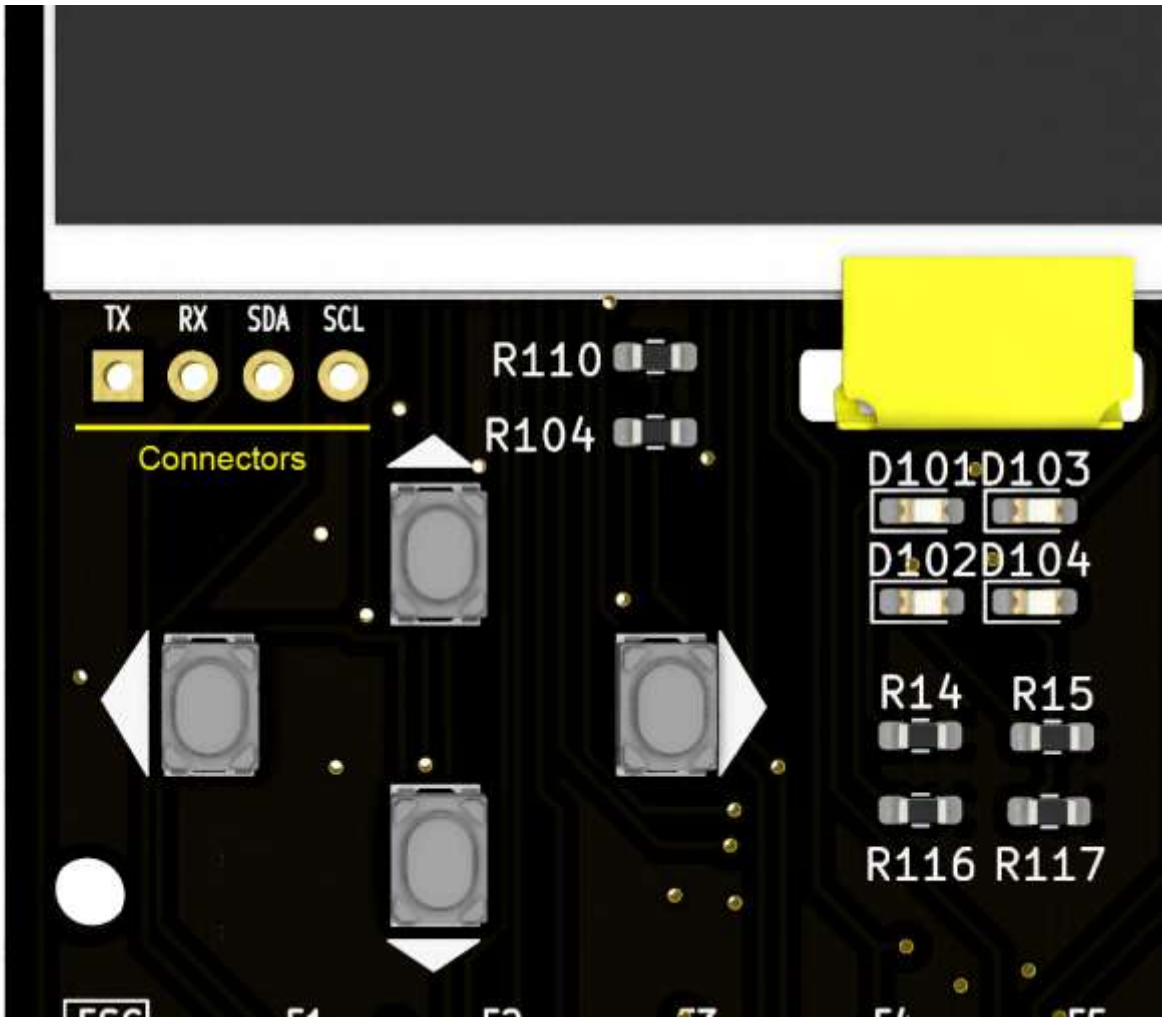

Reset Circuit
USB Pads Internal Regulator Output Supply Voltage
Crystal
Decoupling Capacitors
Front panel leds
Sharp Memory Display
ATmega32U4-MU
MemoryBrakout:Raspberry_PI_ZeroREVERSE_SMD
RPI_Connector-HL_RPI_GPIO
RPI MODE
Sheet: Keyboard
Dr CADIC Philippe
Sheet: /
File: MemoryBrakout.sch
Title: SHARP MEMORY DISPLAY - RPI ZERO 2W - Keyboard
Size: A4
Date: 2021-11-21
Rev: 0.5
KiCad E.D.A. kicad (5.1.12)-1
Id: 1/2


COL0
COL1
COL2
COL3
COL4
COL5
COL6
COL7
COL8
COL9
COL10
COL11
ROW0
ROW1
ROW2
ROW3
ROW4
Ky1 1
Ky2 2
Ky3 3
Ky4 4
Ky5 5
Ky6 6
Ky7 7
Ky8 8
Ky9 9
Ky10 0
Ky11 UPArrw
Ky12 DwnArrw
Ky13 Q
Ky14 W
Ky15 E
Ky16 R
Ky17 T
Ky18 Y
Ky19 U
Ky20 I
Ky21 O
Ky22 P
Ky23 LArrw
Ky24 RArrw
Ky25 A
Ky26 S
Ky27 D
Ky28 F
Ky29 G
Ky30 H
Ky31 J
Ky32 K
Ky33 L
Ky34 BkSp
Ky35 F5
Ky36 F6
Ky37 Tab
Ky38 Z
Ky39 X
Ky40 C
Ky41 V
Ky42 B
Ky43 N
Ky44 M
Ky45 Sp-Inf
Ky46 RET
Ky47 F7
Ky48 OKBTN
Ky49 CTRL
Ky50 LShift
Ky51 ALT
Ky54 SPC
Ky52 SPC
Ky53 SPC
Ky55 ALTGR
Ky56 RShift
Ky57 DEL
Ky58 ESC
Ky59 F1
Ky60 F2
Ky61 F3
Ky62 F4
D1 1N4148
Sheet: /Keyboard/
File: kb0.sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. kicad (5.1.12)-1
Id: 2/2

Extra GPIO available

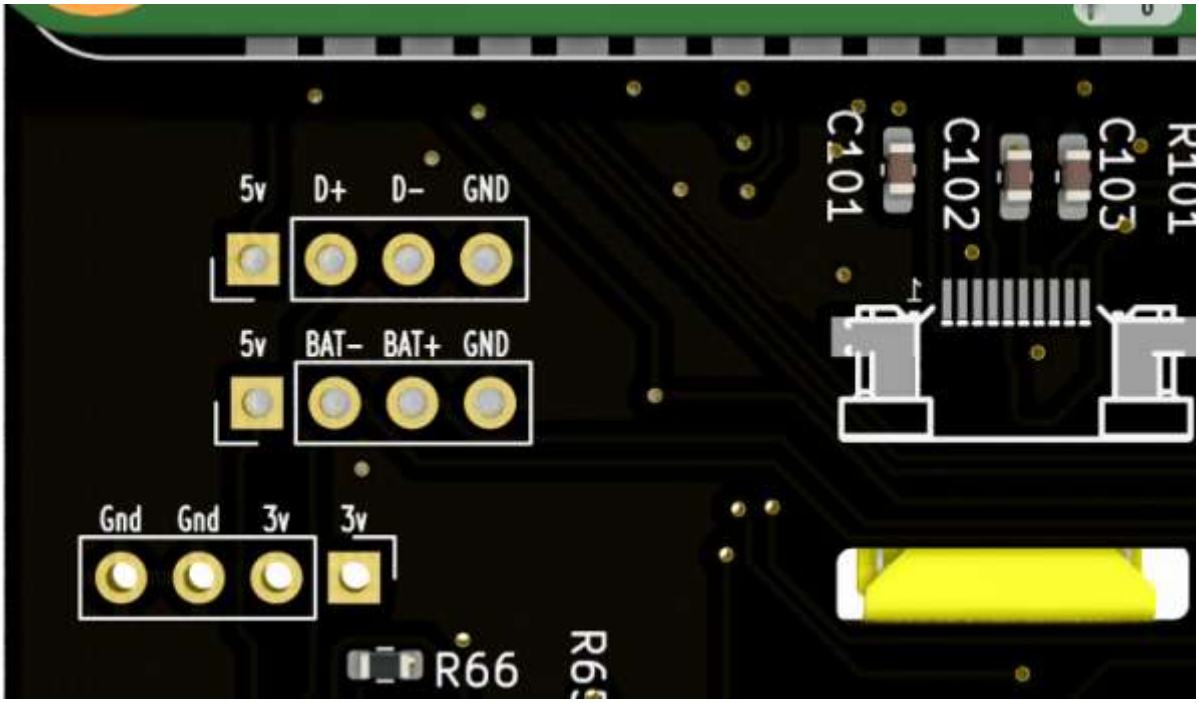
We added several extra GPIO, should you want to add an external UART extra board or a I2C board. For the maker, we also added access to the display pins with the right bottom line of 2.54 mm header pinout (SCK to GND). If you want, you can drive the LCD without the Raspberry PI ZERO , and replace it with a classic microcontroller. In the same idea, you have B1 to B5 buttons signals available on pins.

You get all the necessary connectors to add a battery charger , should your want to add a lipo and build a autonomous device or drive a 3G/4G/5G modem which require a lipo for energy reasons.

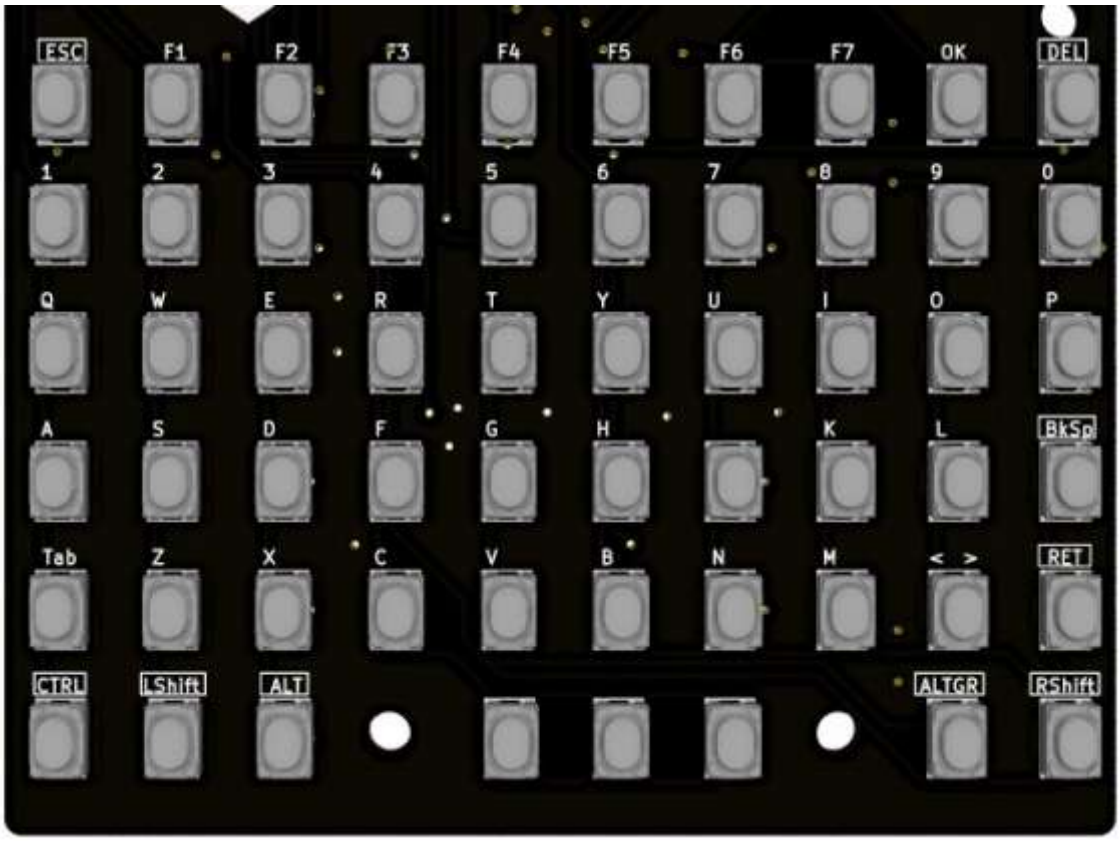


On the left you get 4 pins to connect a UART optional module or an I2C sister board. For example, you can add an optional display with the I2C interface, a GSM module such as SIM800L or newer which is often compatible with UART, an accelerometer … etc etc …

On the back side of the PC board, you get important extra connections to connect the RPI to the keyboard. You also have the necessary connectors to add the battery charging module.



The Keyboard

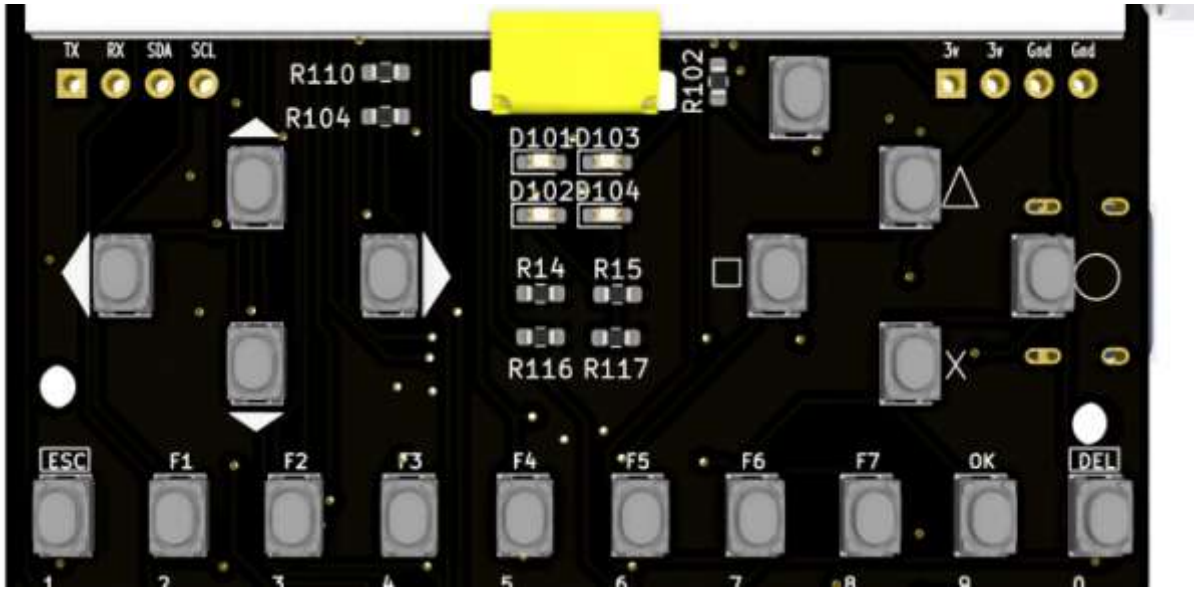


The keyboard has been designed to be a real PC keyboard. We are going to run linux in text mode so we need all kind of keys combination most keyboarded devices do not offer. To ease custom

configuration , we decided to let the keyboard behave as a classical USB device you just plug into the RPI. For this, the ATMEGA32U4 chip is used. We flashed the QMK opensource firware so it can work right out of the box. Now , with QMK, you can modify the keys to any one or any script you wish.

Two joypads

Should you want to work on a port of a game system, we kept 2 joypads with 9 buttons. It will then be possible for game developers to adapt their work onto SHARPIKEEBO.



The radio module for long range communications

Installation

If the board and display come separate, please connect the display's ribbon to the white connector, ribbon contacts facing the board surface.

Software setup

The Buttons and LEDs are controllable using classic python scripts.

For the display, you have to install 2 libraries/code and compile them

1° ONE BITE DISPLAY LIB: https://github.com/bitbank2/OneBitDisplay

Install with git clone https://github.com/bitbank2/OneBitDisplay

Go to /linux/Sharp_LCD directory and type make.

2° ARMBIANIO LIB: https://github.com/bitbank2/ArmbianIO

Install the lib with **git clone https://github.com/bitbank2/ArmbianIO**

and make

Go back to Go to /linux/Sharp_LCD directory and run the compiled executable ./sharp_lcd

Have fun !

Dr CADIC Philippe

Special credits to : Larry Bank for hi precious help and support in this project.
@fast_code_r_us

Link to new LCD if an accident occurs

Link to SHARP Display: https://fr.aliexpress.com/item/4000674316500.html

QMK Firmware : https://docs.qmk.fm/#/

Link to battery module: https://fr.aliexpress.com/item/32999118705.html

Link to RFM95 Radio module: https://www.digikey.fr/fr/products/detail/rf-solutions/RFM95W-868S2/5051755