# PX-HER0 Board User Manual

TM

From Zer0 to TM
32-bit HER0

~ STM32 Edition ~

# Contents

# 1 Introduction

The PX-HER0 is a low-power ARM Cortex development board that's been carefully crafted with embedded education in mind - hardware, software, and documentation. It maps the journey to becoming a true embedded hero. Above all, it should be a fun-filled adventure!!

*"What I cannot create, I do not understand."*

*Richard Feynman*

# 2 Disclaimer

This development / evaluation tool is suitable for research and development in a controlled laboratory environment and not designed or tested as a finished appliance for end users.

⚠ *Precautions must be taken against improper use such as Electrostatic Discharge (ESD) damage, over-voltage / over-current electrical stress, incorrect power sequencing, etc.*

THE HARDWARE IS PROVIDED BY PICONOMIC DESIGN CC "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PICONOMIC DESIGN BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS HARDWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# 3 Features & Specifications

- Microcontroller: STM32L072RB ultra low-power ARM Cortex-M0+
    - 128k FLASH
    - 20k SRAM
    - 6k EEPROM
- Memory: Adesto AT25SF041 4 Mbit serial flash memory
- Storage: Spring-loaded push-push microSD card slot
- Serial communication: Rock-solid FTDI FT230XS USB-serial bridge
- Rich user interface:
    - Low-power 128 x 64 monochrome graphic LCD with LED backlight
    - User LED and battery charger LED
    - Piezo buzzer
    - 6 LARGE finger-friendly user buttons
    - Reset button
- Peripheral connectors:
    - PWR (+3.6 V to +5 V power bus)
    - ADC x 4
    - DAC x 1
    - GPIO x 8 / PWM x 4
    - UART x 2
    - I$^2$C x 3 (single I$^2$C bus)
        - 1 x Sparkfun Qwiic / STEMMA QT I$^2$C connector
        - 1 x Seeed Grove I$^2$C connector
        - 1 via 0.1" header
    - SPI x 1
- Power:
    - Microchip MCP73831 Li-Po battery charger
    - Zero-burden voltage monitoring circuit
    - Efficient power-path management
    - True power on/off circuit
    - 95% efficient TI LM3670 step-down DC-DC regulator
- Dimensions:
    - Board: 100 x 80 mm (3.94 x 3.15")
    - Display active pixel area: 48.6 x 24.9 mm (1.91 x 0.98") / 54.6 mm (2.15") diagonal

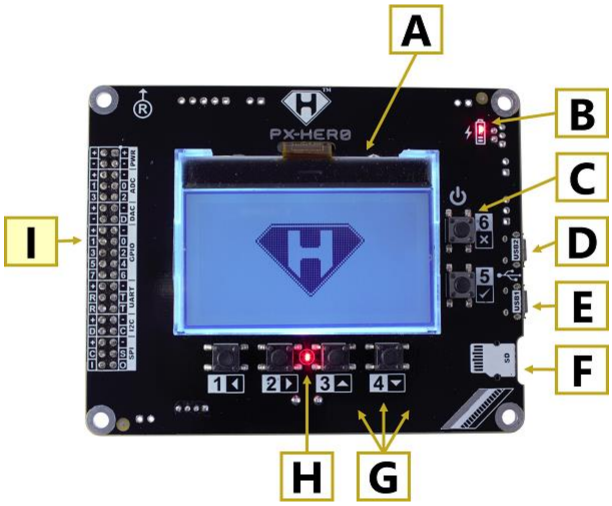
# 4 Potential applications

- Practical embedded engineering course for students and professionals
- Low-power foundation for IoT sensors
- Long-term data logger
- Environmental monitoring for citizen scientists
- Control panel
- Test fixture
- Sexy conference badge flair 😊

# 5 Kit content

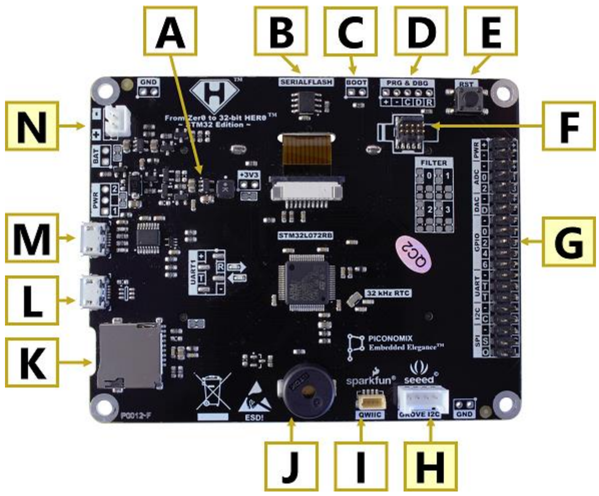
- 1 x PX-HER0 board
- 1 x JST Li-Po battery connector (1 x 2)
- 1 x Seeed GROVE connector (1 x 3)
- 1 x Vertical header (2 x 20)

## 5.1 Top view




| | |
|---|---|
| **A**. 128 x 64 Graphic LCD with backlight | **F**. microSD Card Slot |
| **B**. Charger LED | **G**. User Buttons |
| **C**. User / Power Button | **H**. User LED |
| **D**. USB2: USB-Serial Bridge | **I**. Peripheral Connector* |
| **E**. USB1: STM32 | |

## 5.2 Bottom view




| | |
|---|---|
| **A**. +3V3 Step-down DC-DC Regulator | **H**. Seed GROVE I$^2$C |
| **B**. Serial Flash | **I**. Sparkfun Qwiic / STEMMA QT I$^2$C |
| **C**. BOOT Selection | **J**. Piezo Buzzer |
| **D**. Prog & Debug connector | **K**. microSD Card Slot |
| **E**. Reset Button | **L**. USB1: STM32 |
| **F**. ARM Cortex Debug Connector | **M**. USB2: USB-Serial Bridge |
| **G**. Peripheral Connector* | **N**. Li-Po Battery Connector* |

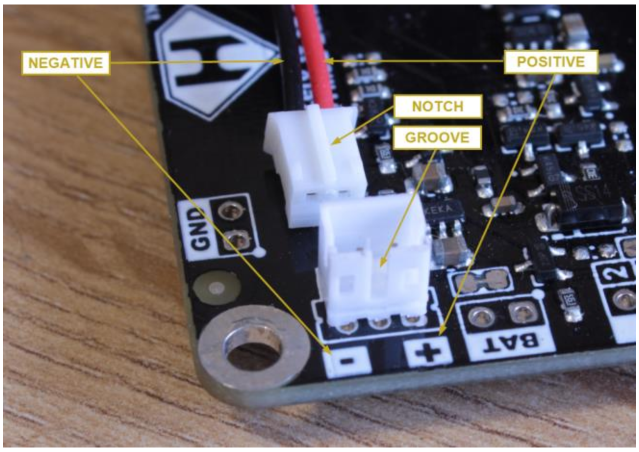
## 5.3 Assembly

*Easy hand assembly is required*. The three through-hole parts (Seeed GROVE, Li-Po Battery and Peripheral Connector) are supplied separately with the board to provide flexible choices.

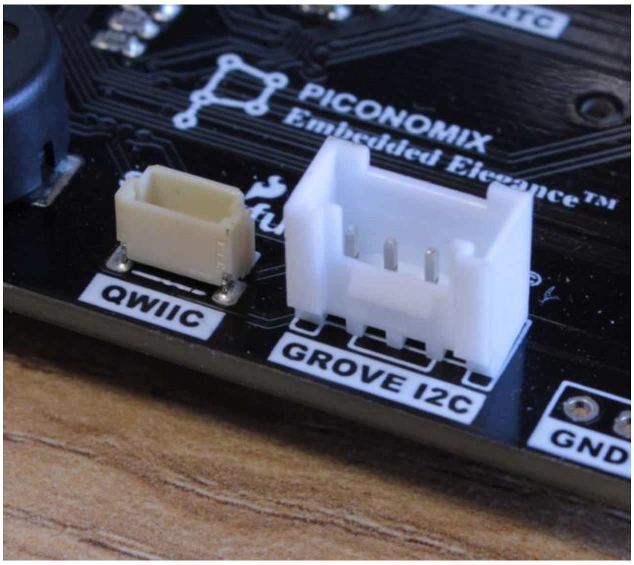
There isn't a uniform standard for the positive and negative terminal of a Li-Po battery and the vertical through-hole [JST connector](#) can be rotated 180° to cater for both permutations. The dual PCB footprint also allow a 3-pin [Molex connector](#) to be mounted.

Similarly, the 40-pin vertical peripheral connector can be mounted on the bottom or the top side. It can also be swapped for a longer, right-angle or female header.

To fit the JST connector, make 100% sure that the negative terminal of the battery (black cable) is closest to the minus (-) side of the white silkscreen. Note that the battery connector is polarised: it has a notch that fits into a groove:




Here is a picture that shows how to fit the Seeed GROVE I2C connector:

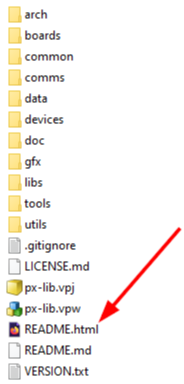


# 6 Getting started

A complete getting started guide is [HERE](), but a brief version is documented here for the super excited 😊

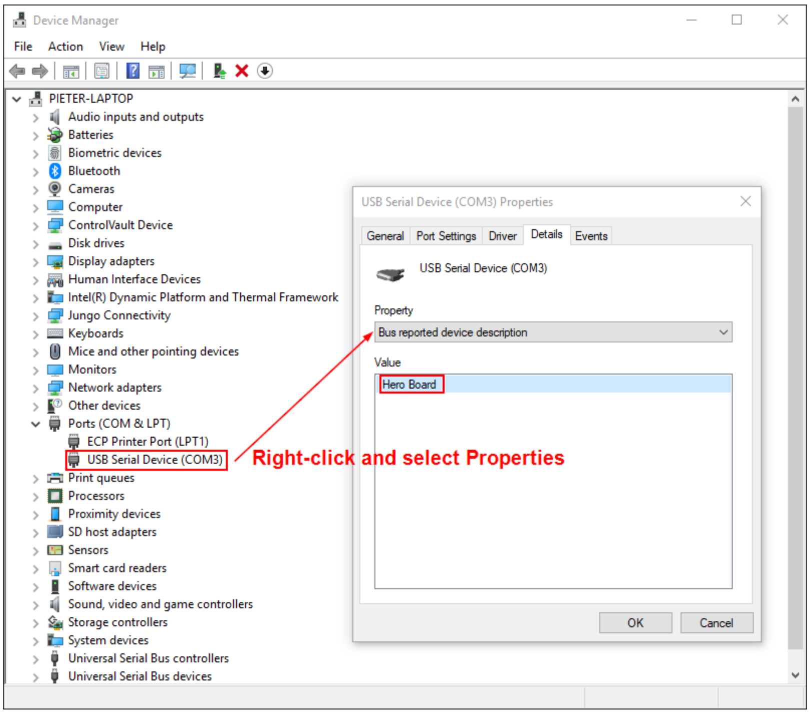
💡 *For your convenience all the downloads are listed in Section 11 [Download links]()*

The first step is to download the latest release of px-lib [HERE]() (for example [px-lib_0.9.0_2020-05-12.zip]()) and decompress it in a convenient location. The documentation is hosted online [HERE](), but the release also contains a copy of the documentation for easy offline access:

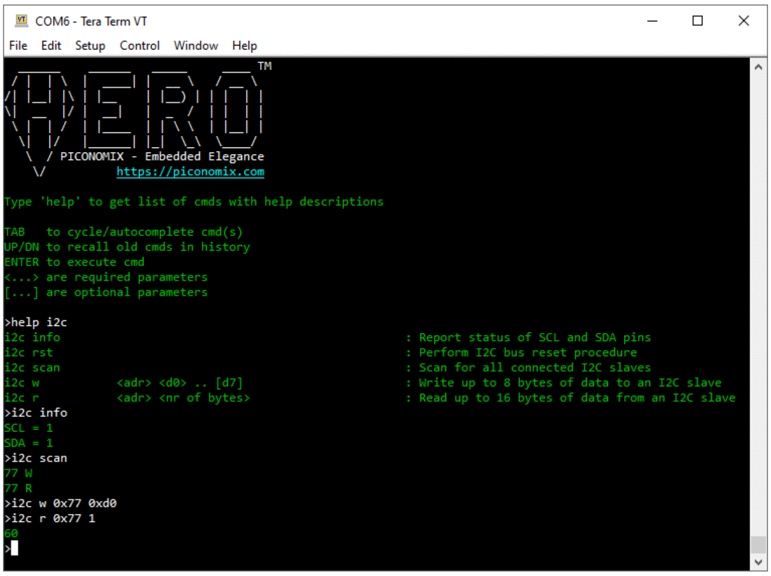



The PX-HER0 board is shipped with a [USB UF2]() bootloader and the [CLI Explorer]() app. Connect a USB cable to the port labelled USB1 and the board will power up and display the Hero logo.

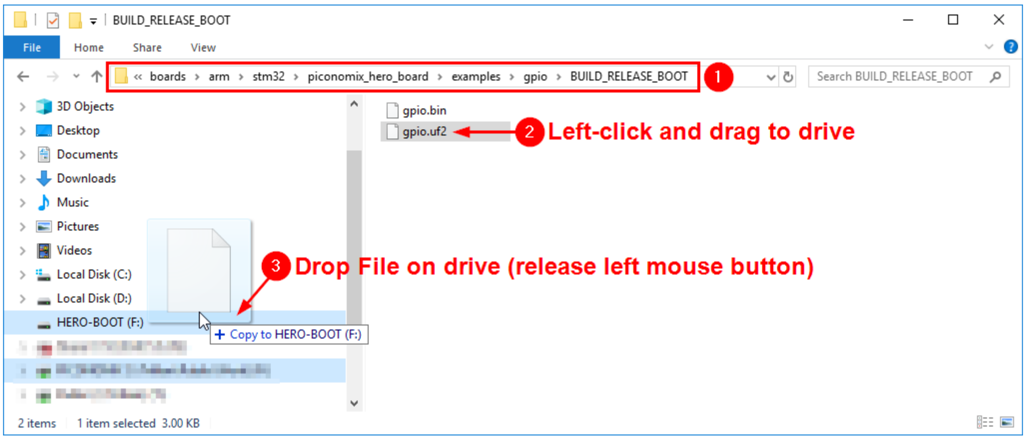
The app implements a Virtual COM port (USB CDC class) driver for communication. When USB1 is connected to the PC, the Windows 10 operating system will detect that a new USB device has been connected and automatically install a driver for it. Take note which COM port has been assigned. The Device Manager utility can also be used to find out. Right-click "Start (Windows icon bottom left of task bar) > Device Manager":



The next step is to install Tera Term (or use an equivalent VT100 terminal emulation software that support ANSI escape codes). The download page is [HERE](). Full installation and setup instructions are [HERE](). Open a new connection to the correct Serial Port (File > New Connection... Alt+N) and you are ready to interact with the CLI Explorer app (documentation [HERE]()):

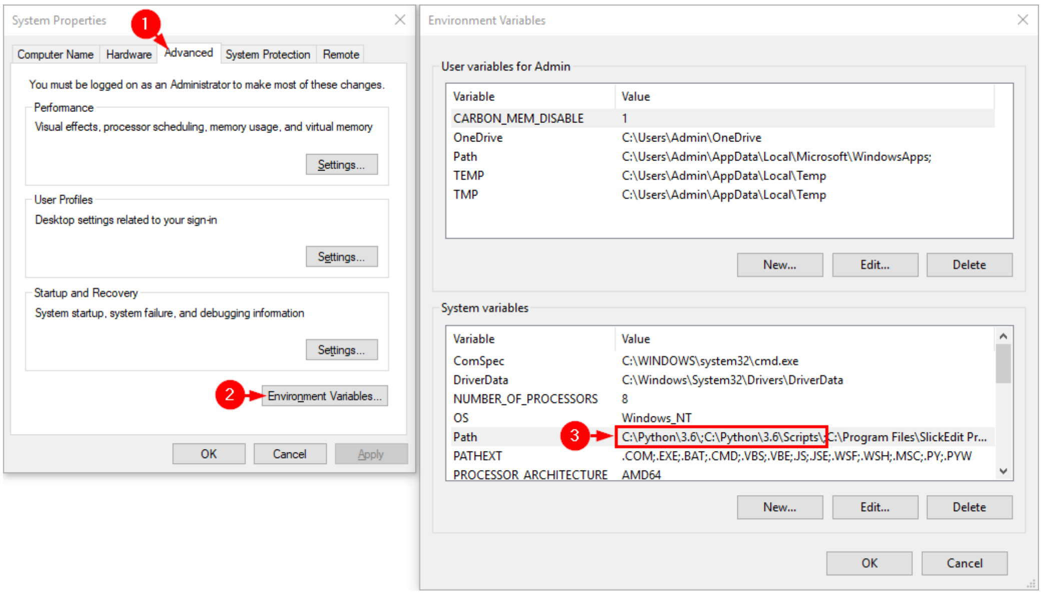



It's super easy to upload new FW to the board. Give the RESET button a quick double-tap and it will activate the USB UF2 bootloader (instructions [HERE]()). The board will appear as a removable drive with "HERO-BOOT" as the volume label. In Windows, drag and drop the UF2 file on the removable drive and the upgrade will begin:



You are now ready to install the development environment.

First you need to install Python 3 (downloads [HERE]). Make sure that the path has been set up to find Python and its scripts:



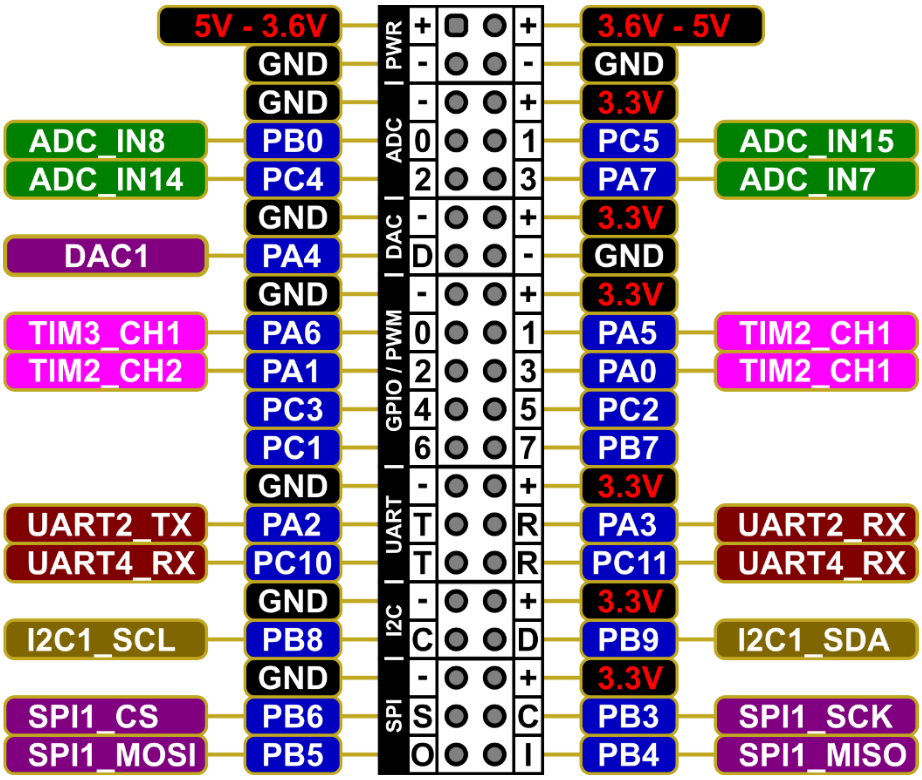

Next install STM32CubeIDE (full instructions [HERE])

# 7 Connectors

## 7.1 Peripheral Connector

Each set of peripheral pins are logically grouped and labelled with their own ground and power connections making it effortless to interface with other devices or components using [Dupont](#) cables.

Here is an annotated drawing showing to which STM32 GPIO pin it is connected to, as well as the peripheral function assigned to it:




For example, ADC0 is connected to PB0 and assigned to peripheral function ADC_IN8. GPIO2 is connected to PA1 but can also be used as a PWM output using Timer 2, Channel 2.

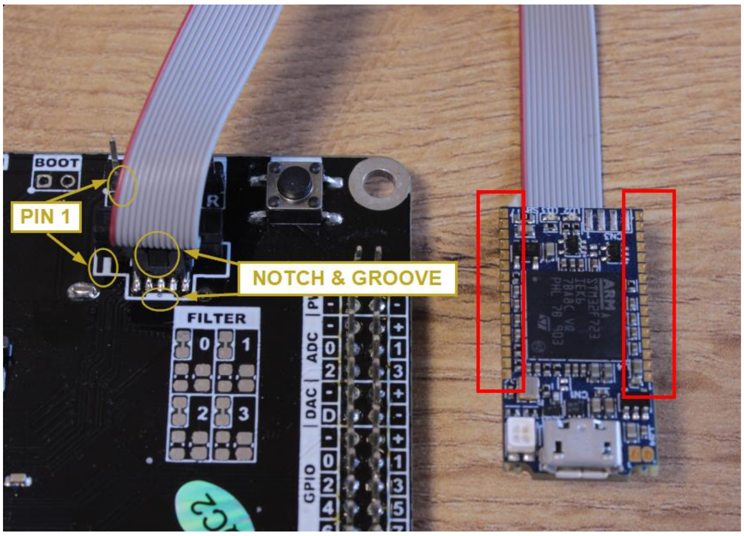
By grouping the pins together it is also possible to plug in a 40-pin [IDC ribbon cable](#) and the multiple GND pins preserve [signal integrity](#).

# 8 Programming & Debugging

## 8.1 ST-LINK-V3MINI

The 10-pin polarized connector (1.27 mm / 0.50" pitch) makes it straight forward to connect the ST-LINK-V3MINI. The open sides allow you to grip the sides of the cable connector with your fingers to pull it out gently and not yank it out with the ribbon cable.

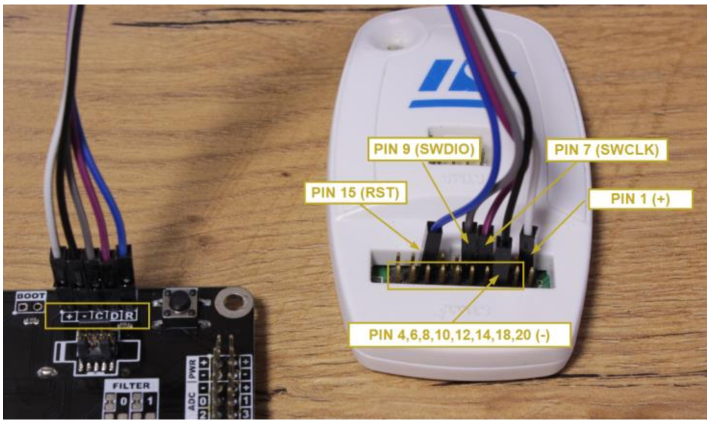
The white silkscreen provides two clues showing how the connector should be inserted. Conductor 1 of the cable is red and there is a corresponding white band on the left side. The connector has a notch preventing it from being inserted the wrong way around and there is a corresponding white groove indication on the bottom.




⚠ *It is advisable to add clear heat shrink around the exposed pads of the ST-LINK-V3MINI, because the castellated edge pads are exposed to ESD damage when handled. Note the clear heat shrink on the SEGGER J-Link EDU Mini.*

## 8.2 ST-LINK/V2

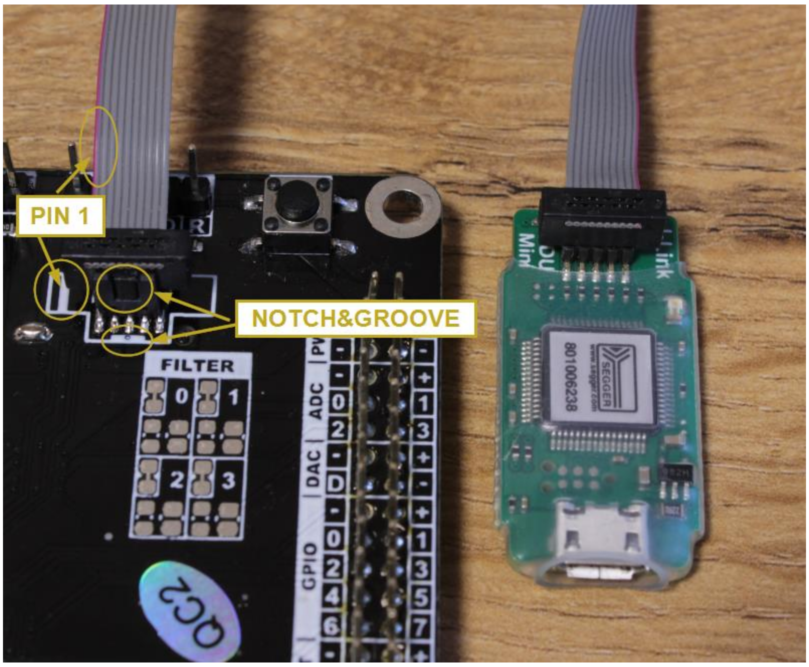
The older 20 x 2 header standard (2.54 mm / 0.1" pitch) used on the ST-LINK/V2 can also be connected easily using Dupont cables. Care must be taken to ensure that the correct pins are connected as there is no polarity protection. The white silkscreen on the board marks each pin:



## 8.3 SEGGER J-Link EDU Mini

The J-Link EDU Mini is an affordable debug probe from SEGGER to give students and hobbyists access to professional debug technology (for non-commercial educational purposes only).

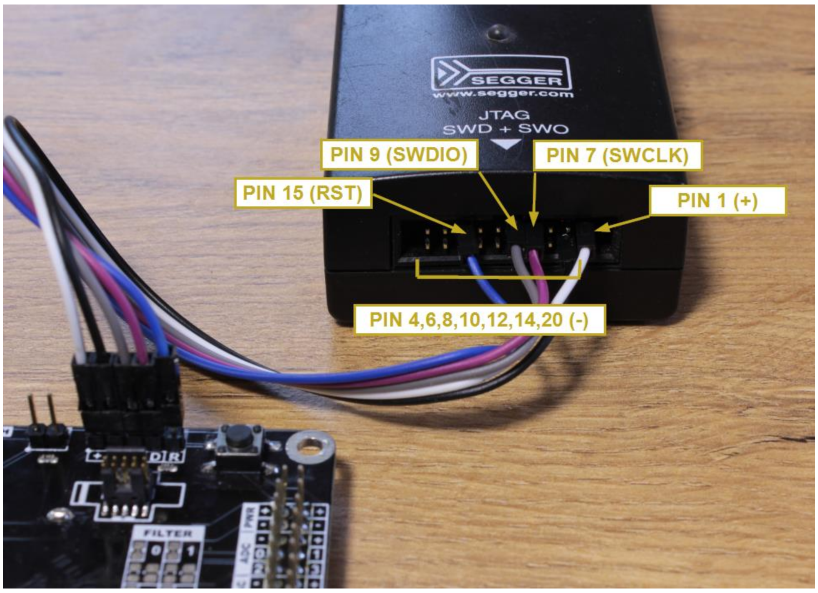
The white silkscreen provides two clues showing how the connector should be inserted. Conductor 1 of the cable is red and there is a corresponding white band on the left side. The connector has a notch preventing it from being inserted the wrong way around and there is a corresponding white groove indication on the bottom.




## 8.4 SEGGER J-Link

The J-Link debug probe supports a vast array of CPUs and "just works". It's the tool of choice for pros but comes with a corresponding price tag.

The pin connections are the same as the ST-LINK/V2 and can be connected easily using Dupont cables. Care must be taken to ensure that the correct pins are connected as there is no polarity protection. The white silkscreen on the board marks each pin.



# 9 SCH Hardware Blocks

The PX-HERO board Schematic is contained on three pages (Power, Microcontroller and Peripherals) and three versions are concatenated:

- The first set shows all the components.
- The second set indicates which parts are not populated for the deluxe edition.
- The third set indicate which parts are not populated for the lite edition.

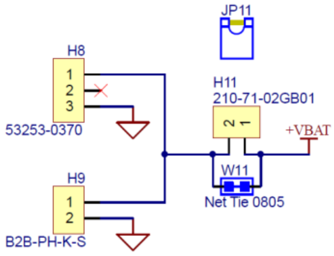
The following sections give an overview of each hardware block and should be read in conjunction with the schematic.

## 9.1 Power

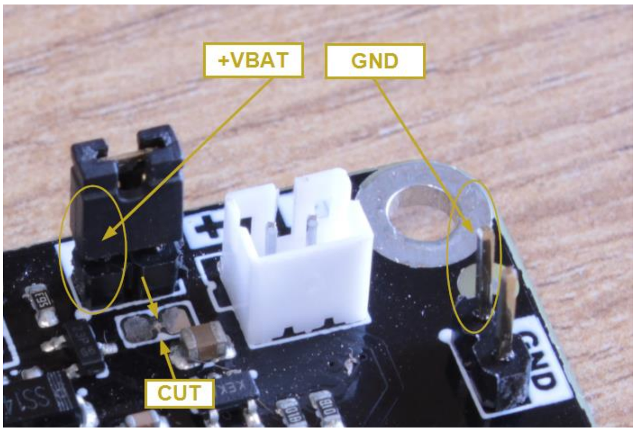
### 9.1.1 Li-Po Battery Connector

The board can be powered from a Li-Po battery. The dual footprint allows a 2-pin JST connector or 3-pin Molex connector to be mounted.

**Li-Po Battery Connector (Molex or JST)**



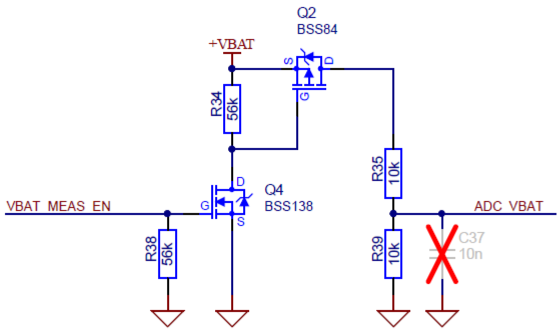

Optionally, track link W11 can be cut with an X-Acto knife and a vertical header H11 can be populated. This allows the option of disconnecting the battery by removing the jumper and powering the board from an external battery simulator using Dupont cables:



### 9.1.2 Battery Voltage Measurement

The battery voltage ranges from +4.2 V down to +3V, but the STM32's ADC input range is from +3.3 V down to 0 V. A resistive divider is required to scale the battery voltage down, but even high value resistors will drain the battery eventually. The STM32's ADC sampling stage also needs a relatively low source impedance to charge the sampling capacitance and make an accurate measurement. The STM32 enables the circuit by taking GPIO pin PC7 (net VBAT_MEAS_EN) high. The STM32 disables the circuit after making the measurement quickly to prevent draining the battery.

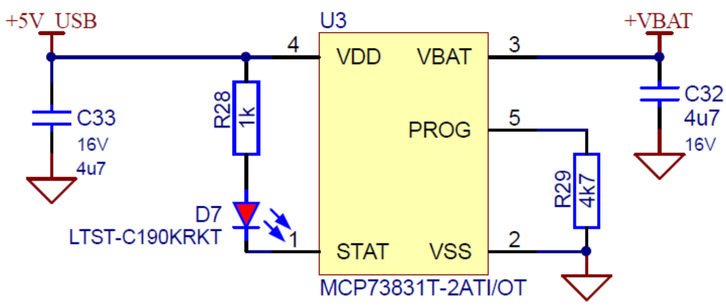
**Battery Voltage Measurement**




### 9.1.3 Li-Po Battery Charger

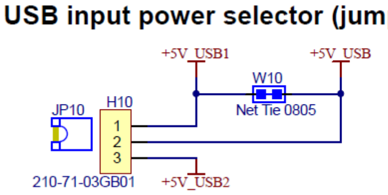
The board has a Microchip linear charger to charge the battery automatically when +5V from USB is present. Charging is indicated by D7. The fast-charging current is set with R29 to 213 mA. If a higher fast charging current is required, this resistor can be replaced with a lower value (down to 2 kΩ for 500 mA).

**Li-Po Battery Charger**

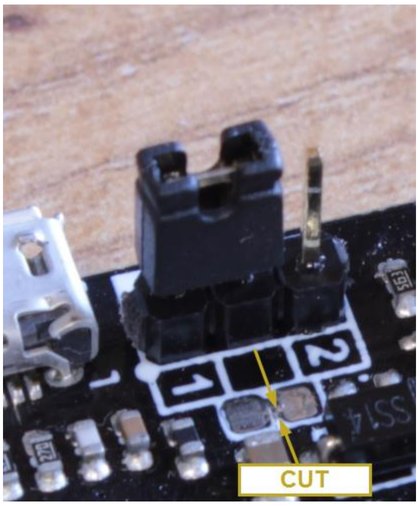


### 9.1.4 USB input power selector

The board is normally powered with +5 V via USB1 (actually +4.4 V to +5.25 V). Optionally, track link W10 can be cut with an X-Acto knife and a vertical header H10 can be populated. This allows the choice of powering the board via USB1 (STM32) or USB2 (FTDI USB-Serial bridge) with a jumper.



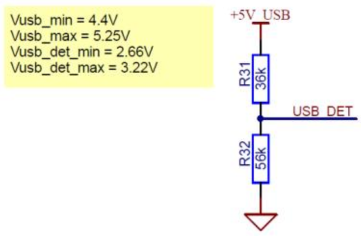

The picture shows the jumper in position 1 (power from USB1):



### 9.1.5 USB power detect

A resistive divider scales +5V from USB (actually +4.4 V up to +5.25 V) down to a valid GPIO input logic level (+2.66 V to +3.22 V) so that the STM32 can detect when it is connected to USB using GPIO pin PA8 (net USB_DET):

**USB power detect**

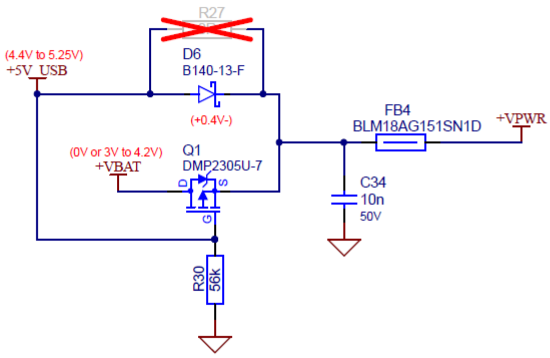



### 9.1.6 Automatic power path selection

The board can be powered from USB or the battery. Schottky diode D6 sources current (with a 0.4 V drop) when USB is present. When USB is absent and the Li-Po battery is present, pull-down resistor R30 switches P-channel MOSFET Q1 on and supplies current to the system with almost no volt drop (less than 90 mV @ 1 A).

A low pass filter consisting of C34 and Ferrite Bead FB4 blocks the high frequency noise of the board from radiating out on the battery or USB cable.

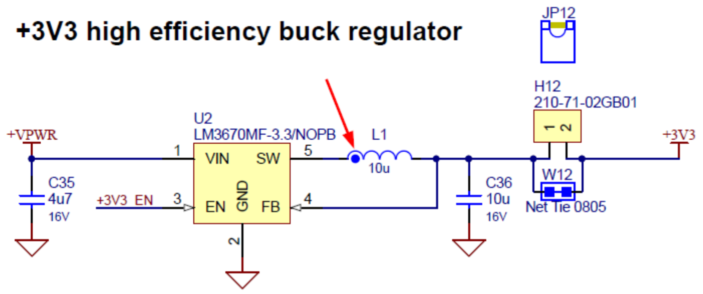
**Automatic power path selection**



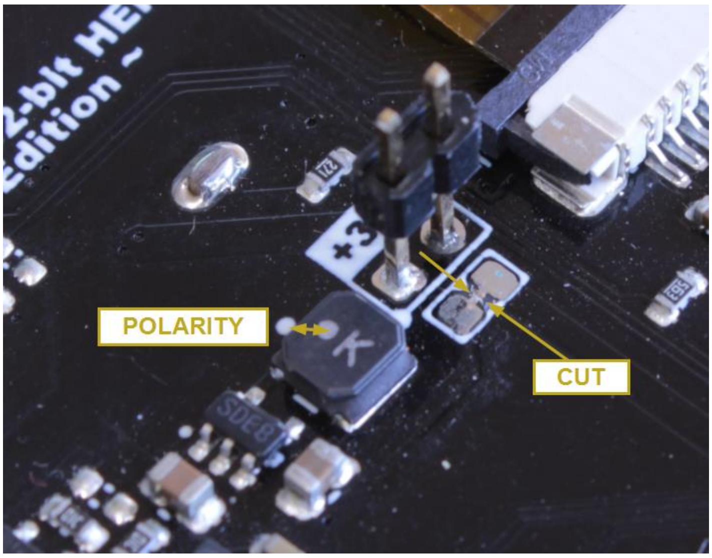
### 9.1.7 +3V3 high efficiency buck regulator

The TI [LM3670](LM3670) step-down DC-DC regulator was selected as the best compromise between maximum output current (350 mA @ 95% efficiency) and no-load quiescent current (15 µA), while still having a developer friendly SOT23-6 SMD footprint. It can be replaced with a TI [LM3671](LM3671) if up to 600 mA is required.

⚠ *The LM3670 has a maximum recommended operating voltage of +5.5 V. The absolute maximum rating is +6 V, but exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.*

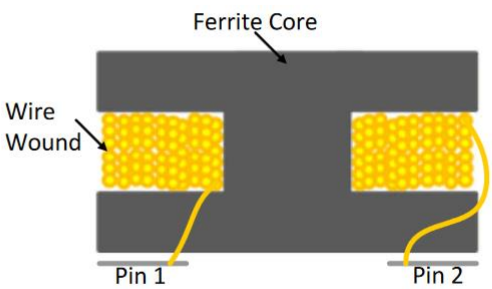



Optionally, track link W12 can be cut with an [X-Acto](X-Acto) knife and a vertical header H12 can be populated. It is more difficult because the LCD blocks access to solder the pins. The solution is to rotate and insert the header and the solder the pins carefully from the front as indicated in the picture. The black plastic holder can then be removed, rotated 180° and placed back and pushed down all the way until it is almost flush with the PCB.




This modification allows the option of measuring the board current at 3.3 V or supplying the board directly with 3.3 V. *Under normal circumstances this modification is not required.*

A white dot on the inductor marks the winding that starts on the inside of the bobbin. It is connected to the switching node of the buck regulator and the outer windings form a capacitive shield to reduce electrical field coupling (EMI).



### 9.1.8 Power on button and microcontroller hold

Here is the block that enables or disables the +3V3 buck regulator.

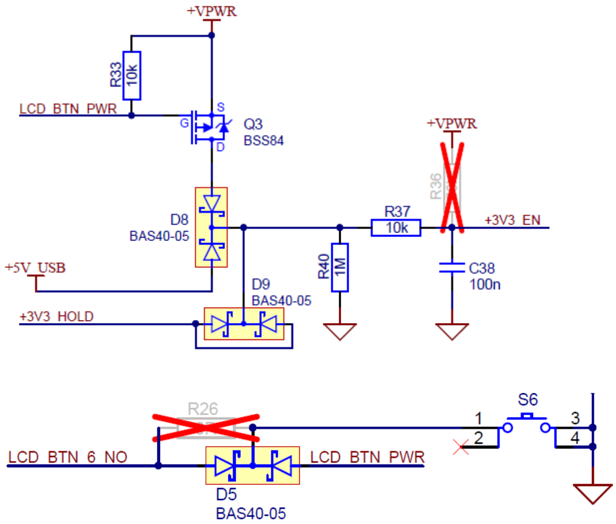
When the 6/NO button (S6) is pressed, it pulls the gate of P-channel MOSFET Q3 to GND via Schottky diode D5 (net LCD_BTN_PWR). Q3 turns on and source current via D8. +3V3_EN is pulled up to +VPWR and the buck regulator switches on.

The STM32 powers up and the firmware configures GPIO pin PC8 (net +3V3_HOLD) to output high. It sources current via D9 and keeps +3V3_EN high. The buck regulator will stay on as long as the STM32 keeps the pin high. Observe that about 3.2 µA will be wasted in R40. To switch off, the STM32 configures the pin to output low. This allows the STM32 to monitor the Li-Po battery voltage and switch off when it falls below +3.6 V

The buck regulator will switch on and stay on if +5V from USB is present.

R37 and C38 forms an RC time delay before the buck regulator will switch on. R40 forms a much longer delay to switch off the buck regulator.

**Power on button and microcontroller hold (always on when USB connected)**



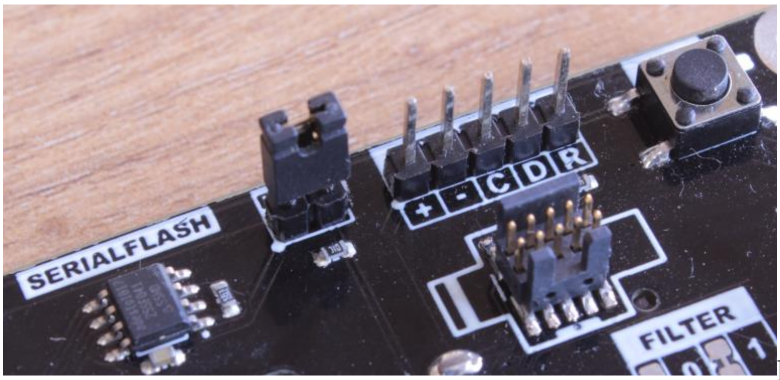
## 9.2 Microcontroller

### 9.2.1 STM32L072RB

The STM32 SCH symbol lists each possible pin function and the firmware assigned function is underlined in RED, for example pin 23, net ADC3_IN (PA7) is assigned to ADC_IN7. It's also possible to see that pin 21, net GPIO1 can be used as a GPIO pin (PA5), but it can also be used for PWM output (TIM2_CH1).

**STM32 ARM Cortex M0+ Microcontroller**


The BOOT0 pin can be pulled high if header H7 is mounted and a jumper is connected. The built-in STM32 ROM bootloader will be executed upon reset / power up and the FW can be programmed using connector USB1 and STM32CubeProg or the Arduino IDE with STM32duino. Remove the jumper and press the RST button for normal operation.

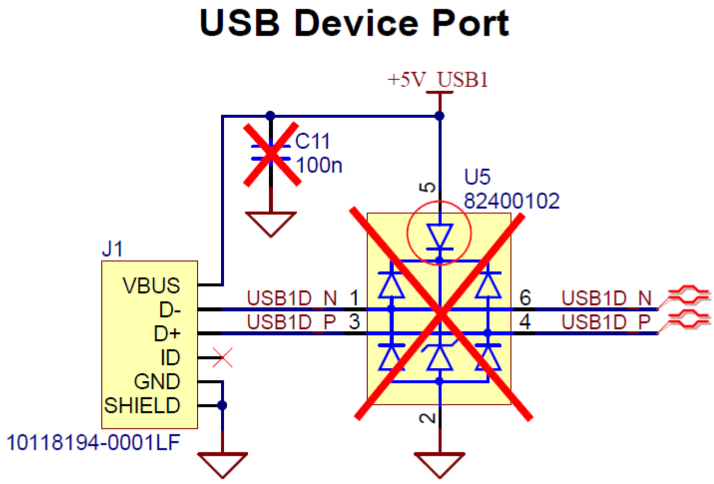



An external 32.768 kHz crystal (±20 ppm tolerance; ±1.73 second error per day) is connected to OSC32 to provide an accurate clock source for the RTC (Real Time Clock) and LPTIM (Low Power Timer) peripheral.

Lastly, there are two external 4.7 kΩ pull-up resistors on PB8 (I2C1_SCL) and PB9 (I2C1_SDA) required for proper I$^2$C bus operation.

### 9.2.2 USB Device Port (USB1 connected to STM32)

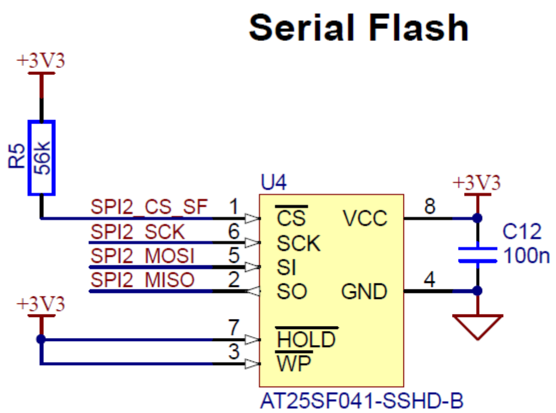
USB1 is connected to the STM32. Optionally a WE 82400102 ESD protection and 100 nF capacitor can be mounted if robust protection is required.




The STM32 enables an internal 1.5 kΩ pull-up resistor on the USB D+ line to signal to the host (your PC) that a full-speed device is connected (read more about it HERE). The protection diode could form an unintentional current path and source current to +5V_USB1, but fortunately this protection device has a blocking diode (circled in red) that prevents the unintentional path.

### 9.2.3 Serial Flash

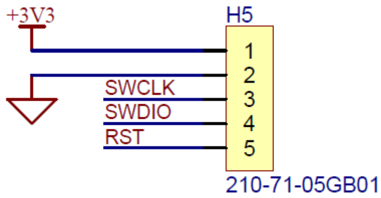
A 4-Mbit Adesto AT25SF041 Serial Flash is included on the board for energy efficient IoT data logging, etc. In deep power-down it consumes only 2 µA. px_log_fs has been created as a basic but resilient record-based file system to demonstrate how to take advantage of this (for more info see HERE).



### 9.2.4 Program & debug header

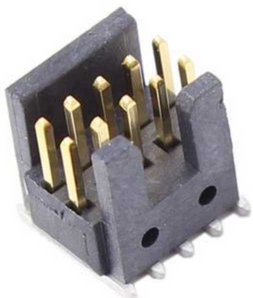
A 2.54 mm (0.1") pitch vertical header footprint is included on the board to allow easy connection of an [ST-LINK/V2]() or [Segger J-Link]() programmer & debugger to the SWD (Serial Wire Debug) port of the STM32.


**Program & Debug**

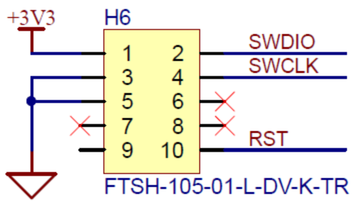



### 9.2.5 Standard ARM Cortex Debug Connector

The 10-pin polarized connector with open sides is very expensive, but we did not want to compromise a great out-of-the-box debugging experience. It prevents wrong-way-around or off-by-one connections. The open sides also allow one to grip the sides of the connector with your fingers to pull it out gently and not yank it out with the ribbon cable



**Standard ARM Cortex Debug Connector**



## 9.3 Peripherals

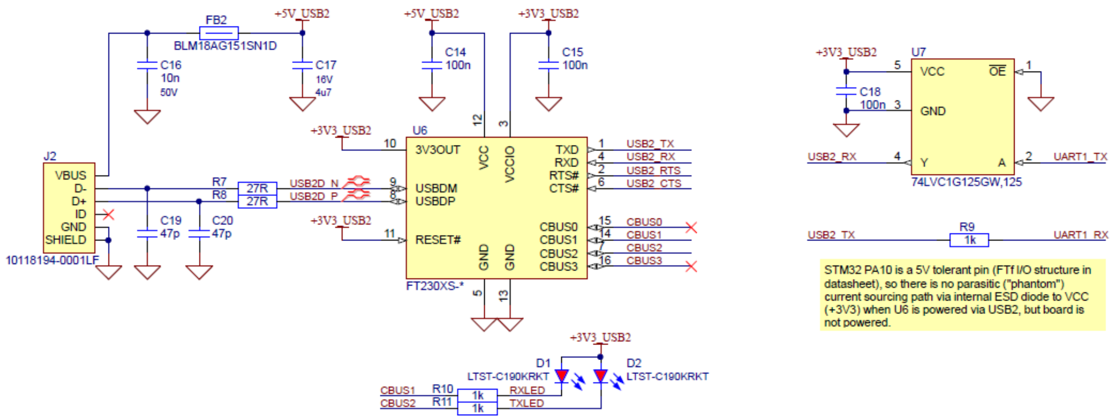
### 9.3.1 USB-Serial Bridge (USB2)

A rock-solid FTDI FT230XS USB-serial bridge that *just works* on Windows, Mac, and Linux is connected to UART1 of the STM32. This provides an easy serial communication facility without having to struggle with the complexity of a USB firmware stack. It can be used for "fire & forget" debug output or a tiny XMODEM-CRC bootloader.

The FTDI FT230XS USB-Serial bridge consumes 125 μA when USB is disconnected so it is powered separately from USB +5 V while an NXP 74LVC1G125GW buffer is used to isolate the serial interface and block parasitic current in the TX direction (consuming only 1 μA in the process).

PA10 (net UART1_RX) on the STM32 is a 5V tolerant pin (FTf I/O structure in datasheet), so there is no parasitic ("phantom") current sourcing path via internal ESD diode to VCC (+3V3) when U6 is powered via USB2, but the board is not powered. This eliminates the need for a NXP 74LVC1G125GW buffer in the RX direction.

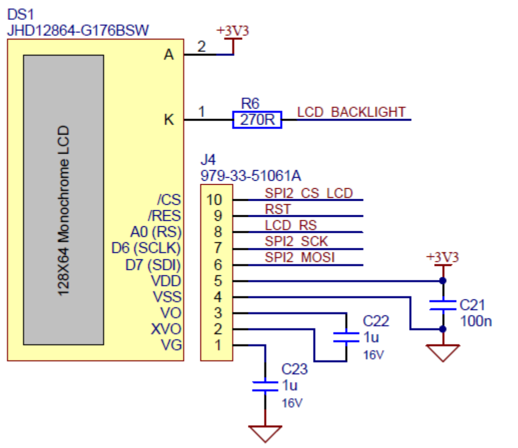
**USB-Serial bridge** **USB-Serial partial power-down isolation (USB2 disconnected)**




### 9.3.2 LCD

The JHDLCM JHD12864-G176BSW 128 x 64 pixel monochrome LCD contains a Sitronix ST7567 driver. It consumes only 150 μA when displaying a SNOW pattern. In power down mode, it consumes only 8 μA.

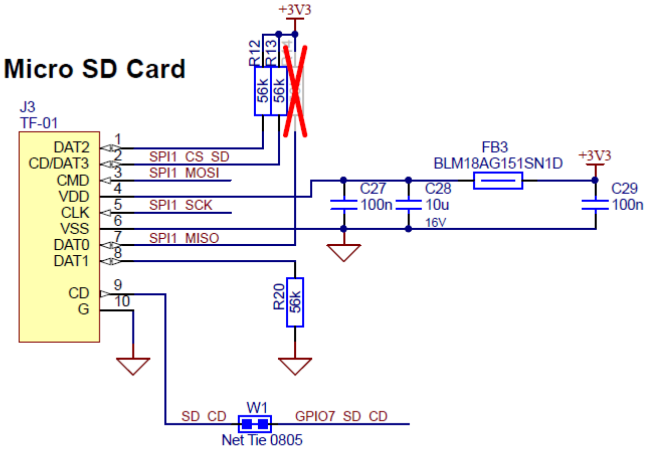
**LCD**




R6 sets the backlight current to 2.5 mA but can be replaced with a lower value if a brighter backlight is desired (up to 45 mA).

### 9.3.3 microSD Card Slot

A spring-loaded push-push microSD card slot expands data logging capability and can also be used for other interesting applications such as an SD Card Bootloader with LCD choice menu.




**(option to disconnect SD CD from GPIO7)**

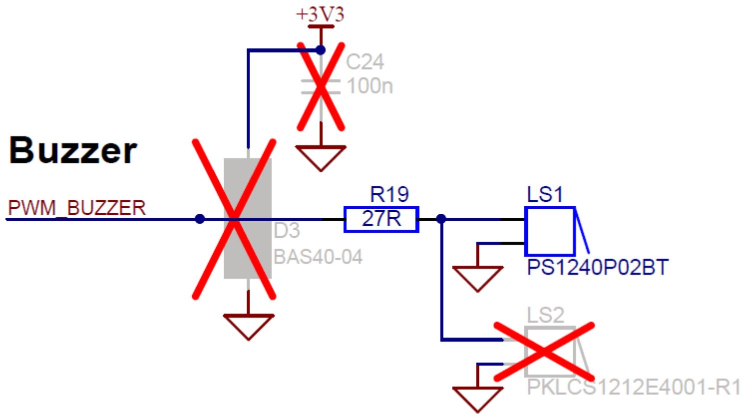
The ferrite bead FB3, C28 and C27 forms an "energy reservoir" to filter voltage dips on +3V3 when the microSD card is inserted.

Pin 7 (DAT0) requires a pull-up resistor to function properly. R14 is not populated, because it is pulled up internally by the STM32 firmware.

Pin 9 (CD) is pulled low to signal to the STM32 when a microSD card is inserted (net GPIO7_SD_CD), but this net is also shared with GPIO7 of the peripheral connector. If this functionality is not required, then the track link W1 can be cut with an X-Acto knife. A 0 Ω resistor can be populated if it is required in the future.

### 9.3.4 Piezo Buzzer

A low-cost piezo buzzer TDK PS1240 (or equivalent) is provided for audio feedback. It is resonant (loudest) at 4 kHz.

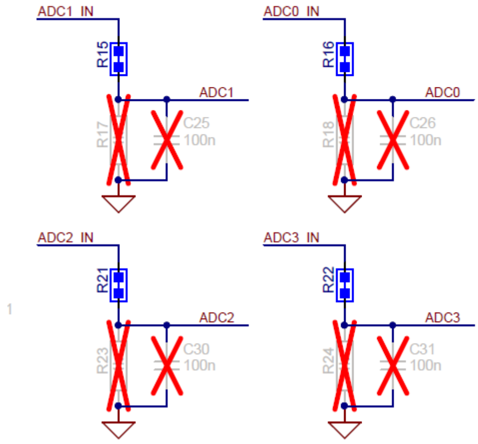



An extremely hard knock /bump can cause the piezo buzzer to generate a high voltage and can possibly damage the STM32's GPIO pin. This is known as the piezoelectric effect. D3 and C24 can be populated to protect the STM32 under these harsh conditions.

### 9.3.5 ADC scale and filter circuitry

Provision has been made on the board to add discrete resistor divider and capacitive filters on the ADC inputs. The wire links R15, R16, R21 and R22 can be cut, and 0805 sized discrete parts can be soldered.

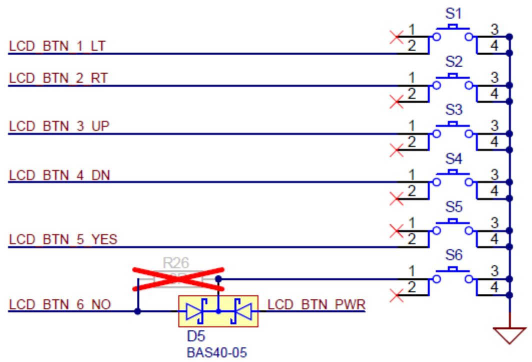
**ADC scale and filter circuitry**




### 9.3.6 LCD Buttons

6 labelled buttons (1/LEFT, 2/RIGHT, 3/UP, 4/DOWN, 5/YES and 6/NO) are arranged around the LCD to form a user interface. The STM32's internal pull-up resistors must be enabled to detect when a button is pressed.

**LCD Buttons**




**(pull-up resistors on micro must be enabled)**

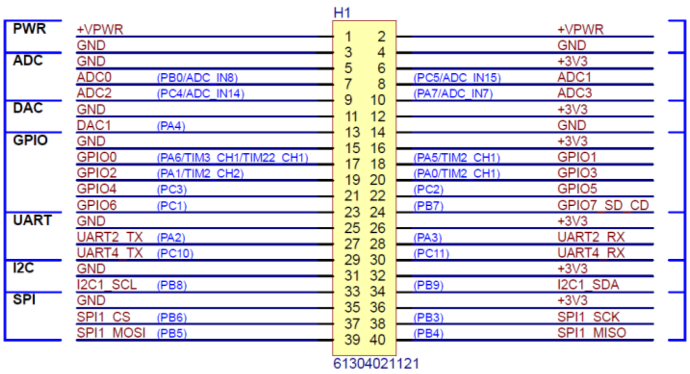
The 6/NO button also functions as the Power button when the board is disconnected from USB and a charged Li-Po battery is present. While it is being pressed, the +3V3 buck regulator is enabled. It is then the STM32 firmware's task to keep the +3V3 buck regulator enabled by configuring PC8 (net +3V3_HOLD) to output a logic high.

To switch the unit off the firmware must detect a long press for example, switch off the LCD to indicate to the user that the device is powering down and when the button is released, PC8 (net +3V3_HOLD) must be set to output a logic low.

### 9.3.7 Peripheral connector

Each set of peripheral pins are logically grouped and labelled with their own ground and power connections making it effortless to interface with other devices or components.

**Peripheral connector**




For convenience the actual pin is indicated with a BLUE label, for example net GPIO2 is routed to PA1, but can also be used as a PWM output using Timer 2, Channel 2.

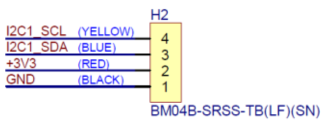
The function of each pin is assigned in firmware, and it is possible to assign a different function if required.

By grouping the pins together it is also possible to plug in a 40-pin IDC ribbon cable and the multiple GND pins preserve signal integrity.

### 9.3.8 Sparkfun Qwiic / STEMMA QT Connector

The Sparkfun Qwiic / STEMMA QT I²C connector allows a whole ecosystem of +3V3 boards to be connected to the PX-HER0 board.

**Sparkfun QWIIC / STEMMA QT Connector**

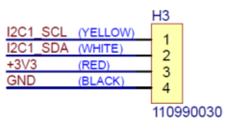



For convenience the wire colors are indicated with a BLUE label, for example the I2C SCL wire is YELLOW.

### 9.3.9 Seeed Grove I²C Connector

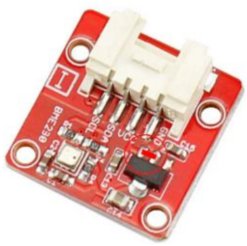
The Seeed Grove I²C connector opens the door to another wide range of inexpensive +3V3 boards that can be connected.

**Seeed Grove I2C Connector**




For convenience the wire colors are indicated with a BLUE label, for example the I2C SDA wire is WHITE.

WARNING! check first if the board requires a +5V supply and if it can be modified to work with a +3V3 supply. For the Crowtail-BME280 board, I had to short two LDO pins to make it work:



# 10 Power consumption

A Nordic Power Profiler Kit and the Low Power STOP Mode example was used for measurements:

| Mode | Current @ 3.6V |
|---|---|
| Active power consumption | 12.2 mA |
| STOP mode with LCD on; Backlight off; SD card inserted | 219 µA |
| STOP mode with LCD on; Backlight off; SD card removed | 171 µA |
| STOP mode with LCD off; Backlight off; SD card inserted | 84 µA |
| STOP mode with LCD off; Backlight off; SD card removed | 26 µA |

Here is a table that lists the individual contributions:

| Part | Description | Current |
|---|---|---|
| U1 | STM32L072 in STOP mode with 32 kHz RTC running @ 3.3V | 1.6 µA |
| U4 | AT25SF041 in Deep Power-Down @ 3.3V | 2 µA |
| DS1 | JHD12864 in Power Down @ 3.3V | 8 µA |
| R40 | 1 MΩ pull-down resistor pulled high by STM32 (net +3V3_HOLD) @ 3.3V | 3.2 µA |
| J3 | SD card @ 3.3V | 57 µA |
| U2 | LM3670 quiescent current @ 3.6V | 15 µA |

# 11 Download links

## 11.1 Software

Required:

- px-fwlib - a collection of open source C firmware and documentation for microcontrollers.
- FTDI Virtual Com Port (VCP) driver – driver for FT230XS USB-Serial bridge on USB2
- Tera Term – Free terminal emulation software to interact with the CLI Explorer app
- Python 3 – used to run a script that creates a UF2 file suitable for uploading via the USB UF2 bootloader
- STM32CubeIDE – a free cross-platform IDE to build and debug STM32 microcontroller applications

Optional:

- STM32CubeMX – graphical tool to configure and generate code for STM32 microcontrollers
- STM32CubeProg – software tool for programming STM32 microcontrollers
- GNU Arm Embedded Toolchain - A GNU GCC build toolchain for C and C++
- UnixShellUtils - a minimal collection of Unix shell utilities for Windows to build projects using Makefile scripts.

## 11.2 Design documents

- PX-HERO board Schematic
- PX-HER0 board BOM (Deluxe Edition)
- PX-HER0 board Component Assembly Drawings
- PX-HER0 board Mechanical Dimension Drawings
- PX-HER0 board PCB Layout
- PX-HER0 board 3D STEP file

## 11.3 Reference documents

- STM32L072 Datasheet
- STM32L072 Errata
- STM32L0x2 series Reference Manual
- STM32L0 Series ARM Cortex M0+ Programming Manual
- ARMv6-M Architecture Reference Manual
- JHDLCM JHD12864-G176BSW LCD Datasheet
- Sitronix ST7567 LCD driver Datasheet

# 12 PX-HER0 origin story

It all started way back in 1983 when I got a second-hand 8-bit Sinclair ZX Spectrum 16K when I was 9. The detailed story is HERE, but the short version is that I taught myself to modify obfuscated machine code to turn my puny 16K into a 128K machine and it turned out to be an April fool's joke. Fast forward 32 years... [insert obligatory 80's Cassette tape fast forward sound effect here]

During my career I got frustrated with hard to comprehend, poorly commented, unstructured code that I encountered in my industry and created px-fwlib to start a dialogue and demonstrate how it could be improved. It has been a passion project for over 10 years now.

At one stage I was the hardware engineer that wanted to find a wiring fault with a test fixture and wanted to put a GPIO pin in a specific state so that I could use a multimeter to trace the path to the DUT (Device Under Test) without bothering the overloaded firmware engineer on the project. That was the inspiration behind the CLI Explorer app and I try to build a small version of the CLI for every embedded device that I work on to test the hardware blocks and confirm that it works.

I used to be an Atmel AVR and Atmel SAMD21 man almost exclusively except for a short crossover to the dark side (Microchip PIC24 series) but forced myself (pun intended) to learn the STM32 series about two years ago and this is where the journey with the PX-HER0 board started, because I wanted to map the route for people to follow. I started to consult independently a year ago and lady luck smiled at me: much of my work has been for the STM32L0 series, so the investment has already born fruit. Hint: there is an STM32L082 in the Murata Type ABZ LoRa module.

I hope that the concise STM32 intro and carefully crafted tutorials will pave the road to a deeper insight into the embedded C universe and create that *Eureka!* moment of enlightenment for you as the 8-bit Sinclair ZX Spectrum 16K did for me.

# Change Details

| Revision | Date | Description |
|---|---|---|
| 1.0.0 | 2020-05-20 | First release |
| 1.0.1 | 2020-05-25 | Added origin story |
| 1.0.2 | 2020-05-28 | Added SEGGER J-Link and J-Link EDU Mini |
| 1.0.3 | 2020-06-01 | Added program & debug connector hardware blocks |
| 1.0.4 | 2020-06-01 | Explain that hardware blocks should be read in conjunction with schematic |
| 1.0.5 | 2021-09-01 | Updated HTTP links |