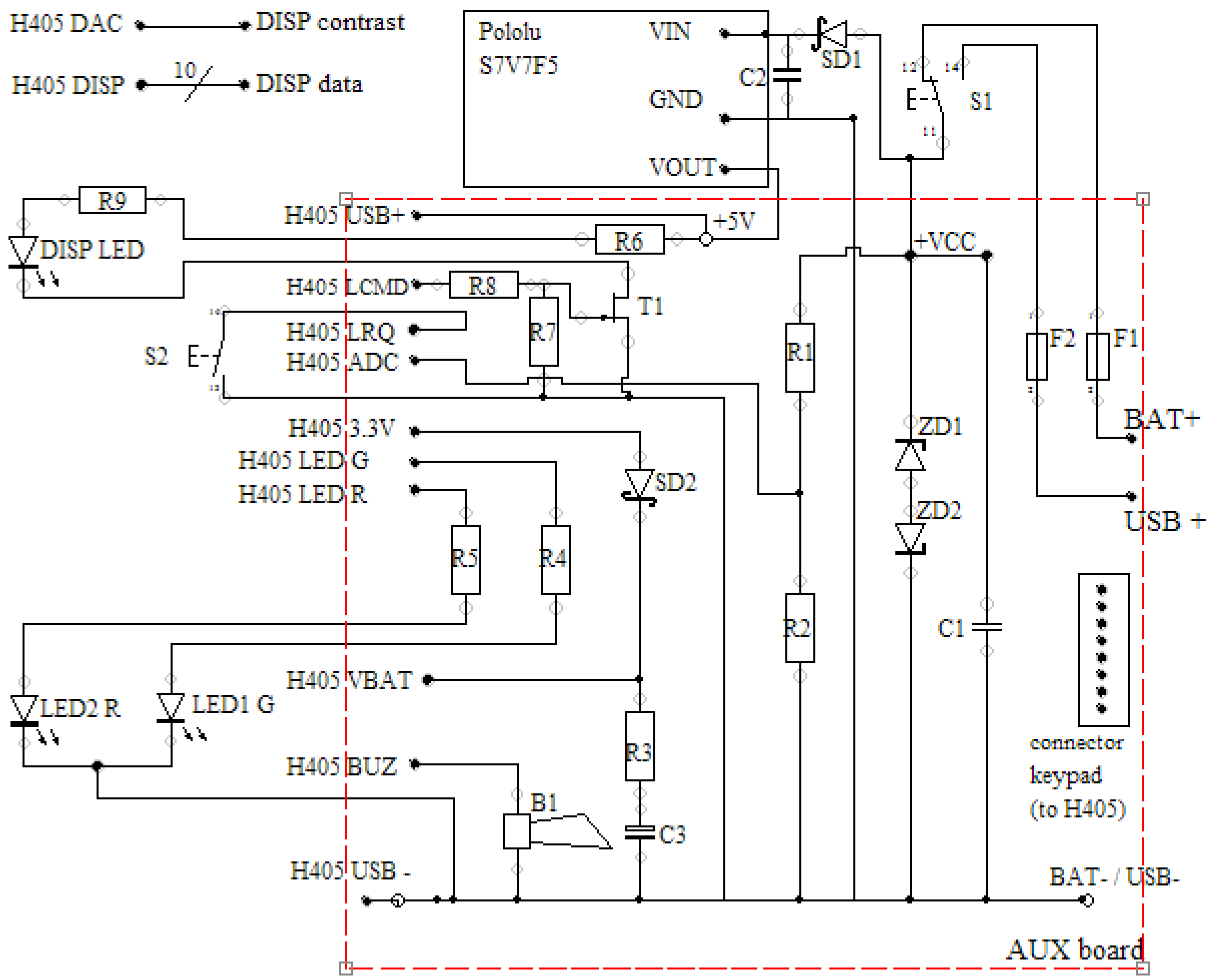
Auxiliary Board

This document specifies the auxiliary board with a schematic diagram, explanation of the individual components and the part list. **Make sure to follow the rework instruction at the end of this document.**

The auxiliary board interfaces power sources, power switch, keypad, extra LEDs and the H405 main board.

Here the schematic diagram:



Explanations:

- **F1** and **F2** are the main fuses, one for each power source. It is a good idea to include a third fuse holder so that a replacement fuse is readily available.

- The Zener diodes **ZD1** and **ZD2** will protect the device from overvoltage with both polarities (more than about 8 V). An unsuited power supply (e.g. 12 V instead of 5 V) will make the fuse for the active power source melt down to protect the diodes from damage.

- The Schottky diode **SD1** is the polarity protection for the overall system. Wrong polarity (under 8 V) will have the device not running, but without melting a fuse. That happens if the AA type batteries are put into the battery holder in the wrong way.

- The capacitor **C1** is for dealing with high frequency noise on the VCC line.

- The capacitor **C2** protects the Pololu *S7V7F5* step-up/down converter against possible LC spikes from the power switch; it must not be a low ESR type, according to Pololu. Should be located as close to the VIN/GND pins of the Pololu board as possible.

- The **C3** super-cap powers the VBAT domain of the main board. The Schottky diode **SD2** prevents C3 from reverse powering the whole board. The resistor **R3** protects SD2 from overcurrent if C3 is in discharged state during power-up, limiting the charging current to 80 mA.
At the same time, R3 limits the current C3 can deliver. Putting R3 and SD2 together this way ensures that upon power-on, the VBAT input of the CPU has a good voltage, and also that the fuse will not melt down if the note at the end of this document is not being followed.
With the RC time constant of R3 and C3, powering the device for two minutes will charge C3 to 95 %. That should be fine for at least one week without powering the device.

- **R1** and **R2** are a voltage divisor: The ADC of the H405 is for up to 3.3 V, but the measured supply voltage is usually 5 V and overvoltage protected from 8 V on. So the supply voltage is divided down for the ADC measurement. In case of wrong polarity of the power supply, R2 will also limit the wrong polarity inrush current to less than 1 mA. 5 mA reverse injection is the specified maximum for an STM32F05 pin as per the data sheet, so that's safe.

- **R4**, **R5**, **R6** limit the current through the LEDs.

- **B1** is a low current, DC operated piezo buzzer.

- **S1** is the main switch to select between wall power and battery power.
- **S2** is a button that can put the LREQ input of the controller to ground, which means "light request".

- **T1** is a MOSFET. The controller answers the light request from S2 by setting the LCMD (light command) pin to high. That will open T1, and current will flow so that the display backlight is active.
An IO pin of the controller can deliver up to 25 mA, but the display backlight requires 30 mA – and that is already a low-current one. Others need something between 100 mA and 200 mA. So the solution is to drive the display backlight indirectly via this MOSFET, which basically does not draw any current.
MOSFETs are quite sensitive to ESD damage – always properly ground yourself before touching a MOSFET.
- **R7** pulls the gate of T1 to ground level if the controller is not ready. This resistor should be quite high because it will draw parasitic current when the microcontroller sets the output pin LCMD to 3.3 V level (logical true).

- **R8** is for robustness. Actually, the T1 gate does not draw current during steady states (both high and low) because a MOSFET is voltage-controlled, not current-driven. Unlike with a bipolar transistor, there is no need for a base resistor.
However, T1 has a gate capacity. In the very first moment when switching from low to high level, this capacity acts like a short-circuit to ground. This is only a transient, but it might affect the system stability. R8 limits the maximum current in this situation to stay within the specified limits.

- **R9** is not part of the auxboard because it is closely tied to the forward voltage of the display backlight, and that can vary somewhat between different specimen even of the same display. Together with R6, is limits the backlight current.

- The connection from the H405 DAC to the contrast voltage input of the display is electrically direct. Same for the command and data lines. For the prototype, I used a plug/socket connector on the board. This way, I can remove the front plate if need be. Keep the total cable length short, i.e. less than 20 cm.
The enable line (**display pin 6, EN**) is the most critical one; you might want to route it separately. In case you experience problems with the display and have ruled out wrong wiring and bad soldering spots, you can try a series (not parallel!) resistor of 33 Ω...330 Ω as close to the Cortex-M4 board as possible (i.e. not at the display side). The optimum value depends on your cabling, but a 100 Ω should be a good starting value. This will reduce eletrical reflections.
If that also fails, the next idea would be to insert a Schmitt trigger right at the display pin 6 (i.e. not at the Cortex-M4 side).
The other lines don't matter much because the software delays are long enough so that there will be a steady state even with long cables.

- The Pololu **S7V7F5** board generates +5 V on its output which is rated 500 mA when stepping up and 1 A when stepping down. The input voltage can be anything from 2.7 V to 11.8 V (remember that SD1 will have a drop of about 0.4 V). The intended input supply voltages are:

- +5 V in case of USB plug powering (USB tolerance specifies 4.75 V to 5.25 V)
- +6 V in case of 4xAA alkaline batteries (may vary from 6.4 V initially down to 4.0 V when empty)
- +4.8 V in case of 4xAA rechargeable batteries (may vary from 5.2 V initial full charge voltage down to 4.0 V when empty)
- maximum device current draw with overclocking and display backlight is around 150 mA.

<u>Note on external power over USB</u>:

The intended USB power source is an adaptor from wall power to USB, typically used for charging mobile phones.

For computer USB ports, the CT800 would have to pull one of the data lines to 3.3 V. But as per the USB spec, only 100 mA would be allowed. Since up to 150 mA is possible, the CT800 would have to ask the USB host (the PC / laptop) whether more than 100 mA are fine. The USB host can allow or deny this. This is part of the so-called USB enumeration process.

The issue is that this would require to implement a USB software stack in the CT800 just for powering the device. Things like game or configuration data transfer are out of question anyway since there is not enough RAM free for any kind of filesystem.

In practice however, most computer USB ports will work fine even without signalling the device's presence and without enumeration. For the mainboard manufacturers, it is simpler and cheaper to just deliver 5 V / 500 mA and be done with it.

Part list:

F1, F2: melting fuses, fast type, 1 A (plus fuse holders). Actually, 500 mA would be enough, but 500 mA fuses have 2 Ω (or even 3 Ω) resistance while 1 A fuses only have 0.1 Ω. So during normal operation at 200 mA current draw, we would lose about 0.5 V at the fuse alone using a 500 mA type – that would be 10 % of the supply voltage. With a 1 A type, the drop will only be 20 mV.

C1: ceramic capacitor, 100 nF, 50 V

C2: electrolytic capacitor, 33 µF, 35 V, no low ESR type! 33 µF is a bit too much for usual USB ports which specify 10 µF maximum device capacity, but the basic design is for USB wall power converters anyway, not for PC USB ports. However, that is not much above the specification so that it still works in practice.

C3: super-cap, 1.0 F, 5.5 V
R1: 20 kΩ, 0.6 W, 0.1 % (voltage divisor for the ADC)
    Alternatively, R1 can be 47 kΩ if up to 12V supply voltage are intended. In this case, the PCB pin programming pins PA5 – PA7 must be set accordingly, see chapter Connections, and ZD1 / ZD2 must be chosen for 12 V.
R2: 10 kΩ, 0.6 W, 0.1 % (voltage divisor for the ADC)
R3: 39 Ω, 0.6 W, 1 % (protection for SD2)
R4: 649 Ω, 0.6 W, 1 % (protection for the green LED1; slightly more than 2mA to reduce the brightness difference to the red LED)
R5: 806 Ω, 0.6 W, 1 % (protection for the red LED2)
R6: 56 Ω, 0.25 W, 1 % (protection for the display LED)
R7: 56 kΩ, 0.25 W, 1 % (pull-down resistor for T1)
R8: 220 Ω, 0.25 W, 1 % (gate resistor for T1)
R9: 56 Ω (calibrated protection for the display LED)
LED1 (green): 5 mm low-current LED, 2 mA, Uf=1.9 V
LED2 (red): 5 mm low-current LED, 2 mA, Uf=1.7 V
SD1: Schottky diode, 1N5817, 1 A, Uf=0.4 V, Ir=1 mA
SD2: Schottky diode, BAT46, 150 mA, Uf=0.25 V, Ir=0.5 µA
ZD1, ZD2: Zener diode, 6.8 V, 5 W
B1: Piezo buzzer for DC, capable of running at 3 V with a few mA (e.g. Conrad article 717702)
connector for keypad: pin row 2.54 mm, 8 pins
S1: changer switch
S2: on/off button
T1: N-channel MOSFET, IRF3708 (this one is clearly over-dimensioned, but it is in TO220 housing, and my prototype board is not for SMD. Anyway, if you choose another MOSFET, make sure it is capable of 3.3 V logic drive. A low RDS_on of maybe some 30 mΩ must be specified for V_GS=2.5..2.8 V.)
board: euro epoxy board, 2.54 mm hole distance

Pololu S7V7F5: Step-up/down converter with 5 V output, 1 A down, 500 mA up (e.g. via Ebay)

Display: Lumex LCR – U02004DSF-WH. R6 must match the specification for the backlight, which in this case is 30 mA / 3.3 V. Since the supply is 5 V, R6 = (5 V – 3.3 V)/30 mA = 56 Ω. T1 is chosen to have a negligible resistance when switched ON.
However, depending on what exact voltage level the Pololu regulator generated, and depening on the display LED tolerance, R6 ensures only that the display LED will not be destroyed.
With the prototype, the Pololu board gave 5.08 V, and the voltage drop at R6 was 1.9 V. The LED forward voltage thus was 3.2 V and not 3.3 V, and the current was 34 mA. That is quite close to the absolute maximum of 40 mA, and clearly more than the typical 30 mA.
Something around 20 mA is already sufficient, and the white LED will last much longer. With R9=56 Ω right in the display line 15 (LED+), I measured 19 mA, which is fine. **Alternatively, you can also just drop R9 and replace R6 by 120 Ω.**

When choosing the display, refer to the datasheet to make sure that the minimum HIGH level for the logical pins is compatible with the 3.3 V outputs of the mainboard. This display has 2.4 V as minimum input voltage for HIGH level.
There are other displays that need 70 % of the supply voltage, which in case of 5 V supply would be 3.5 V minimum high level. Reliable operation with 3.3 V IO pins is not possible with these displays, so you would need a level shifter.

The display internal controller itself would probably work with 3.3 V, but then the contrast voltage will be too low so that the display visibility would break down. That's why it needs 5 V supply.

Notes on the display with regard to the software:

- this display is interfaced with an 8 bit parallel data bus. Changing this to 4 bit would imply minor changes in the software. Going with SPI instead would require to write an SPI driver.

- the display must offer 4 lines with 20 characters each. Otherwise, the whole user interface would have to be rewritten because it is designed for 20x4 characters. That would be a major change.

Regarding the overvoltage protection:

- The Z diodes are rated at 700 mA continuous current, but according to the data sheet, they can sustain single current bursts of more than 10 A for 500 ms.
- The ESKA datasheet for the 1.0 A fast type fuses shows that 4 A for 300 ms are enough to melt the fuse, or 10 A for 20 ms.
- Both combined means that applying overvoltage will melt down the fuse before the diodes get damaged.
- (Maybe transient voltage suppressor diodes at 6.8 V would be even better.)

However, there is one corner case where the protection will not work: if a current limited overvoltage is applied. That can be e.g. a lab power supply at 12 V with the current limited to 1 A. It will be enough voltage to force destructive current through the Z diodes, but not enough current to trigger the fuse. Catching that with just two diodes is not possible; the resulting protection circuit would involve some MOSFETs.

Then again, that would be overdone since the overvoltage protection already starts at device design level:

- the batteries are 4xAA type, which will not yield more than about 6 V.
- the power socket is not a barrel power jack because these are commonly used for anything between 5 V DC and 24 V DC so that it would be possible to accidentally plug in a wrong wall power adaptor. Instead, the power socket is USB type B, which comes at 5 V only.

Regarding the resistor values:

For a more cost-efficient production, you can use resistors with somewhat different values. The LEDs especially are not that sensitive to the exact resistor value, just choose something close by. E.g. R6 and R9 can also be put together in one 120 Ω resistor (when still using the Lumex display).

The ADC resistors have to be precise, at most 1 % tolerance (less is better), but they don't have to be exactly 10 kΩ and 20 kΩ. The bigger one just has to be twice the smaller one so that they give an 1:2 voltage divisor. The resistors should be in that order of magnitude, like 10 kΩ – 20 kΩ.

Connections:

Supply Voltage:

BAT+	to	positive terminal of the 4xAA batteries (put in series, 4.8 V DC)
USB+	to	external power supply at 5 V / 200 mA DC, e.g. a USB plug
BAT- and USB-	to	H405 USB-
H405 USB+	to	EXT2-26 and DISPLAY pin 2
H405 USB-	to	EXT2-25 and DISPLAY pin 1, 5

VBAT powering:

H405 3.3V	to	EXT1-5
H405 VBAT	to	EXT1-23

Buzzer:

H405 BUZ	to	EXT2-19	(PB15)

ADC:

H405 ADC	to	EXT2-10	(PA3)

The ADC is for monitoring the battery input voltage.

H405 DAC (EXT-2-11 / PA4):
directly to DISPLAY pin 3 (via 16 pin socket, see below).
The DAC is for controlling the display contrast.

Display (electrically direct connection, physically via 16 pin 1:1 socket):

Display pin 1	VSS	to	BAT- / USB-	
Display pin 2	VDD	to	H405 USB+	
Display pin 3	VEE	to	EXT 2-11	(PA4 / DAC)
Display pin 4	RS	to	EXT1-12	(PB5)
Display pin 5	RW	to	BAT- / USB-	
Display pin 6	EN	to	EXT1-13	(PB6)
Display pin 7	D0	to	EXT1-15	(PB7)
Display pin 8	D1	to	EXT1-16	(PB8)
Display pin 9	D2	to	EXT1-17	(PB9)
Display pin 10	D3	to	EXT2-14	(PB10)
Display pin 11	D4	to	EXT2-15	(PB11)
Display pin 12	D5	to	EXT2-17	(PB12)
Display pin 13	D6	to	EXT2-16	(PB13)
Display pin 14	D7	to	EXT2-18	(PB14)
Display pin 15	LED+	to	+5 V (via R6)	
Display pin 16	LED-	to	T1	

Keypad (electrical direct connection, physically via 14 pin socket):

Keypad pin 1	to	EXT1-19	(PC0)
Keypad pin 2	to	EXT1-20	(PC1)
Keypad pin 3	to	EXT2-2	(PC2)
Keypad pin 4	to	EXT2-9	(PC3)
Keypad pin 5	to	EXT2-13	(PC5)
Keypad pin 6	to	EXT2-20	(PC6)
Keypad pin 7	to	EXT2-21	(PC7)
Keypad pin 8	to	EXT2-22	(PC8)

LED:

H405 LED G	to	EXT1-21	(PB0)
H405 LED R	to	EXT1-26	(PB1)

Light button:

H405 LREQ	to	EXT2-8	(PA1)
H405 LCMD	to	EXT2-7	(PA2)

PCB pin programming for 5 V supply voltage (R1 = 20kΩ)

not connected	to	EXT1-18	(PA5)
not connected	to	EXT1-14	(PA6)
not connected	to	EXT1-22	(PA7)

PCB pin programming for 12 V supply voltage (R1 = 47kΩ)

GND	to	EXT1-18	(PA5)
not connected	to	EXT1-14	(PA6)
not connected	to	EXT1-22	(PA7)

14 pin socket layout for the keypad, LEDs and LREQ:

Keypad pin 1 (row 1):	2
Keypad pin 2 (row 2):	4
Keypad pin 3 (row 3):	6
Keypad pin 4 (row 4):	8
Keypad pin 5 (col 1):	10
Keypad pin 6 (col 2):	12
Keypad pin 7 (col 3):	14
Keypad pin 8 (col 4):	13

LED G+:	11
LED G GND:	9
LED R+:	7
LED R GND:	5
LREQ +:	3
LREQ GND:	1

Rework instruction:

On the H405 mainboard, the VBAT bridge has to be removed (cf. mainboard user manual / schematics). Since that bridge is wired, not soldered, it is not possible to remove this using a soldering iron. Instead, a sharp knife is suitable.

Before doing so, there is a direct connection between EXT1-5 (3.3 V line) and EXT1-23 (VBAT). After removing the VBAT bridge, verify (multimeter) that this connection is not there anymore.

Warning: If you leave out that step, the supercap will not only power the backup RAM, but the whole CPU. This will draw much more current, and the backup saving may not work for more than a few seconds.

Rasmus Althoff, December 2018


```
/*
 * License notice:
 *
 *  Copyright (C) 2016-2018, Rasmus Althoff
 *
 *  This file is part of the CT800 (hardware interface / ARM).
 *
 *  CT800 is free software: you can redistribute it and/or modify
 *  it under the terms of the GNU General Public License as published by
 *  the Free Software Foundation, either version 3 of the License, or
 *  any later version.
 *
 *  CT800 is distributed in the hope that it will be useful,
 *  but WITHOUT ANY WARRANTY; without even the implied warranty of
 *  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 *  GNU General Public License for more details.
 *
 *  You should have received a copy of the GNU General Public License
 *  along with CT800. If not, see <http://www.gnu.org/licenses/>.
 *
 */
```