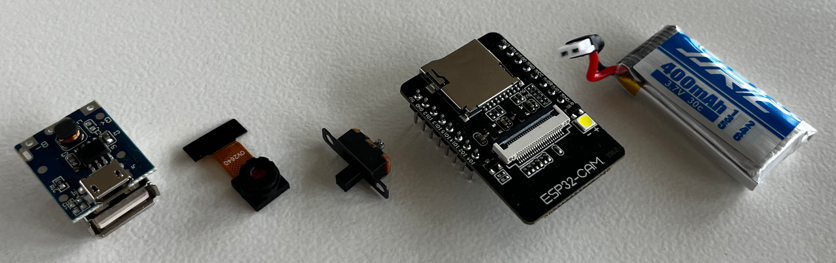
# **Manual Do it yourself Chameleon Scope**

## **Required material**

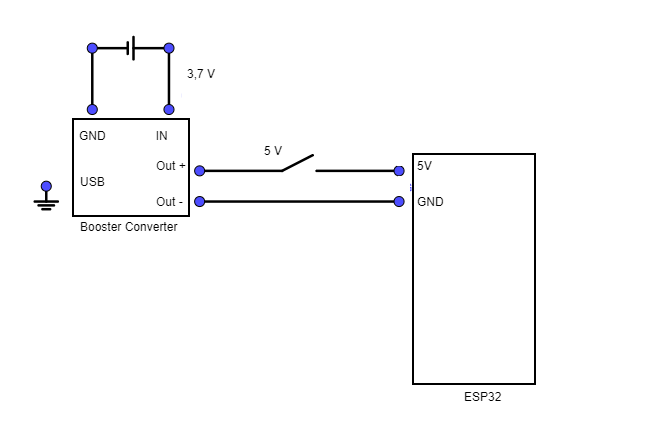
To create the R2D part of the Chameleon Scope you need the following parts:

* **Camera module ESP32-CAM**
* **400mAh LiPo battery 3.7V**
* **3D printer and filament (TPU and PLA) for the case**
* **Battery 5V booster module**
* **Camera extension**
* **Cable for soldering**
* **I/O switch**



## **Soldering**

The ESP32-CAM requires 5V, while the LiPo battery transfers 3.7V voltage. To transform the voltage, it needs a booster module in between. One advantage of the booster module is that it has an integrated USB-C interface, which allows the ESP32 to be charged via the booster module. With the help of this circuit diagram, the soldering is carried out and explained step by step.



**Step 1: Soldering the battery to the booster module**

|  |  |
| --- | --- |
| Ein Bild, das Kabel, Elektrische Leitungen, Elektronik, Elektrisches Bauelement enthält.  Automatisch generierte Beschreibung | 1. Start with the battery and solder its positive terminal to the B+ surface. 2. Solder the negative terminal of the battery to the B- surface. 3. Take the black cable and solder it to the outermost pin on the B side (blue marking). 4. Take the red cable and solder it to the outermost pin on the B+ side (red marking). |

**Step 2: Soldering the cables coming from the booster module to the switch**

|  |  |
| --- | --- |
| Ein Bild, das Kabel, Elektrische Leitungen, Schrumpfschlauch, Werkzeug enthält.  Automatisch generierte Beschreibung | 1. Begin at the booster module. 2. Solder the power supply (red cable) to the outer pin of the switch. 3. Solder another power supply (red cable) to the middle pin of the switch. 4. Connect the cable from the middle pin of the switch to the 5V pin of the microcontroller. 5. From the booster module, solder the earth (black cable) directly to the GND pin of the microcontroller, located below the 5V pin. 6. This creates a circuit that can be interrupted by the switch. |

## **Runtime**

**Runtime Microcontroller**

Battery: 400 mAh LiPo Battery– 3,7 V

Booster Modul: 5V

Battery power

400 mAh \* 5V

= 0,4 \*5

= 2 Wh (2 Watt per hour)

Active Camera

P = 140 mW

P = 0,14 W

Runtime = 2 Wh / 0,14 W

**Runtime = 14 Hours**

Active flashlight

i = 0,31 A

P = 5V \* 0,31 A

P = 1,55 W

Runtime = 2 Wh / 1,55 W

**Runtime = 1,29 h = 1 hour 18 minutes**

## **IT-Architecture**

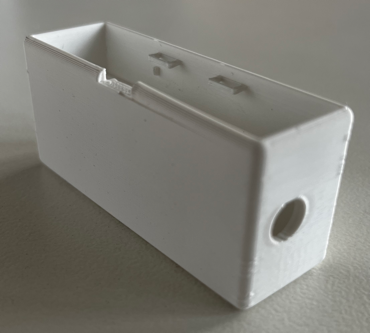
Ein Bild, das Text, Diagramm, Screenshot, Plan enthält.

Automatisch generierte Beschreibung

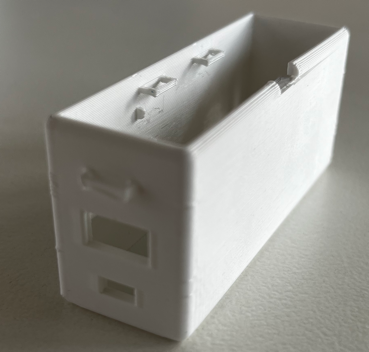
## **3D-Case explanation**

**Inner Case:**

* Variables: Dimensions and translations defined.
* Code structure: Uses difference() and cube() functions.
* Case: Outer case created with cube() function.
* Inside: Interior space created with cube() function.
* Camera: Cylindrical camera object created with cylinder() function.
* LED: Cylindrical LED object created with cylinder() function.
* Cap: Cap created with cube() function.
* Gap to open: Gap for opening case created with cube() function.
* Magnet: Magnet created with cube() function.
* Holding to open: Holding structures and gap holding objects created with cube() function.

 Ein Bild, das Im Haus, Fernbedienung enthält.

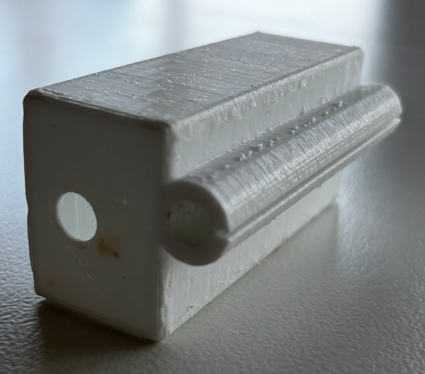
Automatisch generierte Beschreibung

 Ein Bild, das Rechteck, Materialeigenschaft, Wand, Im Haus enthält.

Automatisch generierte Beschreibung

**Outside-Case:**

* Variables: Case dimensions and translations defined.
* Code structure: Uses difference() and minkowski() functions.
* Camera: Camera object created with cylinder() function.
* Case: Outer case created with cube() function.
* Additional Components: Spheres and fixation objects created with minkowski(), cylinder(), and cube() functions.
* Fixation: Fixation objects created with cylinder() and cube() functions.

 Ein Bild, das Boden, Im Haus enthält.

Automatisch generierte Beschreibung

**Cap:**

* Translation and Cube: Cube created with cube() function.
* Plattform: Cylindrical and cubic objects created with cylinder() and cube() functions.
* Fixation: Cylindrical and cubic objects created with cylinder() and cube() functions