Manual CDCD

Content

Introduction

This manual describes the use and setup of the CDC-based C^4D (CDCD).
The CDCD provide 4 different types of use, which distinguish between the use of an amplifier on the detection head and the use of wireless data transmission. The configurations are summarized in the table below. Depending on the chosen configuration, the assembly of the hardware differs. All configurations can also be realized in a differential measurement mode setup.

Table 1: Configurations of CDCDs

Configuration	Wired data transmission (only one detector)	Wireless data transmission (up to 8 detectors, needs master device)
Without amplifier (needs link at the bottom of the detection head)	1	3
With amplifier (needs fully equipped bottom of the detection head)	2	4a-4d

Setup for the first time a CDCD-set

1. Chose the desired configuration of the CDCDs according to Table 1.
2. Build according to the chapters “Building instructions for detection head” and “Building instructions for the supply unit” the required units.
3. Configure according to the chapter “Initial configuration” the CDCDs

Use of preconfigured CDCDs

There are 4 different types of use, which distinguish between the use of an amplifier and the use of wireless data transmission, see Table 1.

Information: The Arduino Nanos must be preconfigured according to the section “Configuration” before use.

Wired data transmission

- Set switches 2,3,4 to ON; set the remaining switches to OFF
- Connect the Arduino’s USB port on the detection device to the PC
- Start the Software

Wireless data transmission, battery powered setups

- Connect Master device to PC
- On the detection unit: set switch 5 to “ON”, use switches 2-4 to set a unique id. Apply changes by pressing the momentary switch and wait until the Arduino restarts (see section “Coding”)
- Supply the detection unit by voltage (6 - 16V) at the ports mark ports in Figure 11.

Additional Information

Coding

The coding on the supply unit of the detection device must be unique, the coding can be changed at any time and must by applied by resetting the Arduino (disconnect the device from power or use one of the momentary switches).

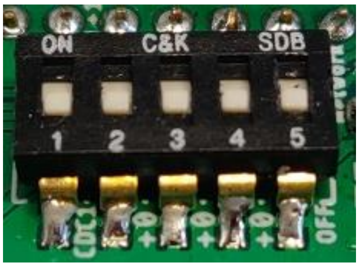


Figure 1:DIP-switches on the supply unit

Switches 2-4: Set the detection device ID according to: switch 2 adds 1 to CDC1 → CDC2; switch 3 adds 2, switch 4 adds 4.

All possibilities:

S 2	S 3	S 4	Name	S 2	S 3	S 4	Name
OFF	OFF	OFF	CDC1	OFF	OFF	ON	CDC5
ON	OFF	OFF	CDC2	ON	OFF	ON	CDC6
OFF	ON	OFF	CDC3	OFF	ON	ON	CDC7
ON	ON	OFF	CDC4	ON	ON	ON	CDC8

Switch 5: OFF: Configuration mode; except switches 2 to 4 are set to ON

→ Usage without amplifier and wireless communication → CDC1

ON: wireless communication is activated

Trigger

The trigger can be connected via a 4-pin terminal block. The trigger can be connected to any detection device or to the master. However, preferably on the master.

Two trigger mods are supported (change requires reprogramming)

- Prince Trigger-Mode (2 wire connection) (line 11: ce_ms = false)
    - The trigger T_G is connected to GND via a switch of the CE. Every change in potential is evaluated as a trigger, the maximum pulse length must not exceed 4 s. Start and stop triggers take place at the same cable.
- Agilent Trigger-Mode (4 wire connection) (line 11: ce_ms = true)
    - Adapter board with 2x D-SUB 9 ports (male and female) and small parts is necessary

Measurement mode

The detection head can also be equipped with a second pair of electrodes to allow a differential measurement setup. This requires also a change in the software for the Arduinos.

- Single-ended measurement mode (2 electrodes) (line 13: differential = false)
- Differential measurement mode (4 electrodes) (line 13: differential = true)

LEDs:

Wireless data communication:

- LED on Master and detection device flash fast (approx. 10 Hz): Connection established successfully
- LED on Master and detection device flash slow: establishing connection
- LED on detection device flash slow: detection device searches for master
- LED on detection device is not flashing: detection head cannot be addressed (see: Known hardware problems)

Software

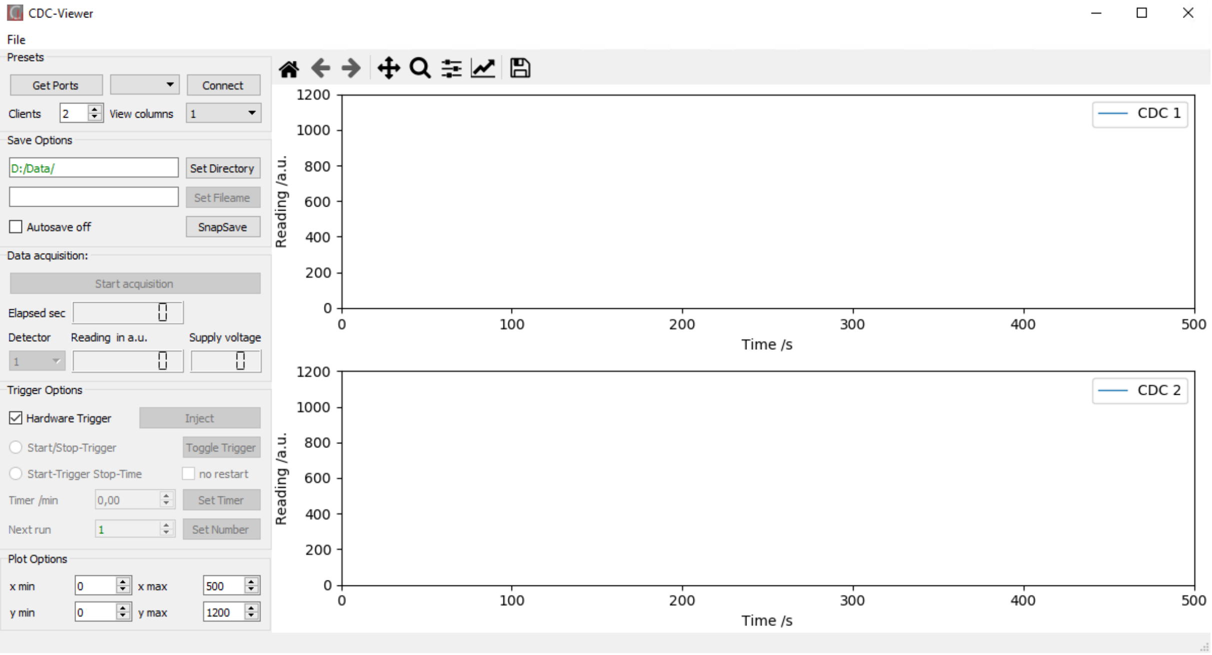
Required basic software

- Python 3.X along with PyQt5; matplotlib and pySerial

  → Anaconda Python 3.X : https://www.anaconda.com/distribution/
- System path "python" should be created in Windows (can be created automatically during installation, check the box next to "Add Anaconda to my PATH environment variable")
- After installation of "Anaconda" add installation of "pySerial" via: Enter in command line (search for "cmd" in Windows): `conda install -c anaconda pyserial` confirm after some time by entering "y". Installation is completed after the message "All requested Packages already installed", close the window.
- To check the installation of the driver for the Arduino connect it to the PC. For troubleshooting have a look at: https://sparks.gogo.co.nz/ch340.html or https://www.makershop.de/ch340-341-usb-installieren/ or search in the web for "CH340 driver"

Measurement software

0. Connect the USB port of the Arduino of the master to the PC and supply the detection devices with power. When operating without radio: Connect the detection device's Arduino to the PC.
1. Launch python software, pleas wait a moment until the UI appears.



Clients 2
View columns 1
D:/Data/
☐ Autosave off
Detector 1
☑ Hardware Trigger
○ Start/Stop-Trigger
○ Start-Trigger Stop-Time
☐ no restart
Timer /min 0,00
Next run 1
x min 0 x max 500
y min 0 y max 1200

Figure 2. Software UI

2. Choose serial port of the Arduino and apply by clicking "Connect", if port is not known, have a look at: Serial Monitor, COM-Port
3. Set number of detection devices (Clients), corresponds to the highest ID assigned via the DIP switches, and chose desired columns of plots, "0" disables live plot.
4. Save options: (Duplicates in the file name will result in the file being overwritten)
   1. Autosave mode:
      - Select Folder
      - Enter any desired filename without file extension (min. 3 letters) apply by clicking "Set Filename". Successful application is reported by green text color

Information: The chosen filename will be extended by _RX_DY whereas X increments the measurements (X is display in the box “Next run” / “Current run”) and Y denotes the number of the detection device (selected by the DIP switches)

2. Manual mode (Check “disable Autosave”)
   - Save current plots via “SnapSave”
     Information: The chosen filename will be extended by _DY, Y denotes the number of the detection device (selected by the DIP switches)
     Information: “SnapSave” is also available in autosave mode

5. Data acquisition:
   a) “Start acquisition” start the data acquisition
   b) The combobox “Detector” allows to select a specific detector by its id.
   c) “Reading in a.u.” displays the received value form the selected detector device in dependence of elapsed time since the last reset (“Elapsed sec”). “Supply voltage” displays the belonging voltage of the batteries.
6. Trigger Options:
   a) In autosave mode, the “Inject” button can be used to start a recording manually. Afterwards, the same button, now labeled “Stop run”, stops the recording by clicking. This button works in any case, despite of a connected hardware trigger or the checkbox “Hardware-Trigger” is checked or unchecked.
      See also section “Trigger”
   b) Unchecking the checkbox “Hardware-Trigger” deactivates the trigger from hardware, the “Inject” / “Stop run” button are working anyway.
   c) Start/Stop-Trigger: (preselected)
      Start and stop of a measurement are defined by a trigger event at the beginning and the end of the measurement.
   d) Start-Trigger Stop-Time:
      Start via trigger event and stop after a predefined period
      - Activation is only possible if the duration (“Timer /min”) is set to a value different form “0” and is applied by clicking “Set Time”.
      - If “no restart” is checked, only the first trigger event is recognized. Only a single measurement with the set duration is recorded. Following triggers are disregarded.
   e) Further options:
      - “Current run” / “Next run” informs about the number of the next/current measurement, green text color. Adjustment is possible by changing the number. After applying with “Set Number”, the text color is green again.
      - “Toggle Trigger” can be used in case of an error with the trigger. This button can be used to toggle whether the next trigger event is a start or a stop signal.
7. Plot:
   Joint adaptation of the plot area in x and y direction of all diagrams.

Known software issues

Information: All data is at any time recorded to a backup-directory which is in the temp folder in the root directory of the python script. The data is named: YYYY-mm-dd_HH-MM-SStemp_DX.csv

1.) Live-Plots stops working: no effect on measurement data if values in “Reading in a.u.” and / or “Elapsed sec” change.

2.) In case of errors regarding the matplotlib library: Enter in the Windows command line successively the following commands:
   - pip uninstall matplotlib
   - python -m pip install --upgrade pip
   - pip install matplotlib

Serial Monitor, COM-Port

Software: Arduino IDE (https://www.arduino.cc/en/Main/Software)

Accessing the serial monitor:

1. Connect the PC to the USB port on the Arduino Nano
2. Open Arduino IDE
3. Select the COM port at Tools >Ports> COM X
   - If the port is not known: unplug the Arduino, display the port list, remind and close it. Plug in the Arduino and select the new entry in the port list. At the bottom right you should see Arduino Nano, ATmega328 (Old Bootloader) on COM X.
   - If not: Select Tools > Board > Arduino Nano and select Tools > processor > ATmega328 (Old Bootloader)

Open the serial monitor (magnifying glass at the top right), select 19200 baud in the new window at the bottom right

Configuration

Initial configuration

In the initial configuration, the software is installed on the Arduinos and the detection sets are defined. The definition of the detection sets pairs all detection devices with a fixed master so that several sets can be used at the same time without crosstalk.

Preparatory steps

1. Make sure the Arduino IDE (https://www.arduino.cc/en/Main/Software) is installed
2. Copy the contents of the Arduino\libraries folder into the libraries folder of the working directory of the Arduino software (`%USERPROFILE%\Documents\Arduino`)
   The following folders are then contained in Arduino\libraries: RF24-master, SBNetwork-master
3. To check the installation of the driver for the Arduino connect the bare Arduino to the PC. For troubleshooting have a look at: https://sparks.gogo.co.nz/ch340.html or https://www.makershop.de/ch340-341-usb-installieren/ or search in the Web for „CH340 driver“
4. Define a set ID, this must be unique in the laboratory and must be entered later identically in all Arduinos of a detector set

Upload and configuration for the detection devices

Information: Start with the configuration of the detection device's Arduinos. Only connect one Arduino on the PC during configuration.

Connect the PC to the USB port on the Arduino Nano with installed custom-made PCB (set all switches to OFF) and radio module

1. Open the file „CDC_Funk_MC_Multi_v19.ino“ at Arduino\CDC_Funk_MC_Multi_v19
2. Select the COM port at Tools > Ports > COM X
   2.1. If the port is not known: unplug the Arduino, display the port list, remind and close it. Plug in the Arduino and select the new entry in the port list
3. At the bottom right you should see Arduino Nano, ATmega328 (Old Bootloader) on COM X. If not: Select Tools > Board > Arduino Nano and select Tools > processor > ATmega328 (Old Bootloader)
4. Check if line 10 contains the following: bool client = true; // Save bool for client/master
5. Set the desired trigger mode in line 11: for Prince: ce_ms = false; for Agilent: ce_ms = true
6. Set the desired measurement mode in line 13: for single-ended mode: differential = false, for differential measurement mode: differential = true
7. Upload CDC_Funk_MC_Multi_v19.ion with the arrow button „→„ or use Sketch > Upload; Success is confirmed by a message "Upload completed" below the sketch.
8. Set the detection device ID according to: switch 2 adds 1 to CDC1 → CDC2; switch 3 adds 2, switch 4 adds 4; switch 5 on ON activates the radio module.
   All options:

S 2	S 3	S 4	Name	S 2	S 3	S 4	Name
OFF	OFF	OFF	CDC1	OFF	OFF	ON	CDC5
ON	OFF	OFF	CDC2	ON	OFF	ON	CDC6
OFF	ON	OFF	CDC3	OFF	ON	ON	CDC7
ON	ON	OFF	CDC4	ON	ON	ON	CDC8

   Leave switch 5 on OFF for configuration
   It is only necessary for the configuration of the CDC8 that a detector is plugged in and supplied with voltage.

9. Open the serial monitor (magnifying glass at the top right), select 19200 baud in the new window at the bottom right

   The output should be:

   ```
   Firmware:CDC_Funk_MC_Multi_v19
   Hello at CDCX
   SetupRadio is active
   Enter 'N_X' to reset the wireless device and set it to client-master-set X
   Prince Trigger-Mode (2 wire connection)
   CDCX
   …
   ```

   If X does not correspond to the set value: set the switches again and press the reset button on the Arduino.

   For CDC8 a different output is displayed:

   ```
   Firmware:CDC_Funk_MC_Multi_v19

   Hello at CDC 8
   Please make sure that a detector with power supply is connected to configure
   CDC8 or use a CDC in USB-only mode, in the latter case CDC is always named
   CDC1
   „„„ Enter 'N_X' to reset the wireless device and set it to client-master-set
   X
   Prince Trigger-Mode (2 wire connection)
   Init...Get offset
   …
   done
   CDC1;373;0;8396142
   ```

10. Configuration of the set ID „S“: Enter N_“S“ in the input field above the output in the serial monitor and confirm with „Enter“

    The output should be: (here: „S“ = 9 and X = 2)

    ```
    Erasing device configuration data...Done
    SBNetwork Version 1.0.5
    ====================
    Try to read device config from internal flash...Failed
    Creating new device config and stroing it to internal flash...
    Done
    Device MAC = 0x5 0x4 0x4 0x9 0x2
    Master MAC = 0x0 0x0 0x0 0x0 0x0
    NetKey = 0
    Initializing NRF24L01 transmitter...Done
    Try to connect to master...Warning - Not paired to a master
    ```

11. Unplug the USB cable from the Arduino and set switch 5 to ON, radio is activated, connect the detection head

For additional detection devices it is in general possible to skip the steps 2 – 7. Doublecheck step 8 for each detection device to prevent identical IDs.

Upload and configuration for the master

Start analogously to steps 1-4 of section „Upload and configuration for the detection devices„:

5. Set line 10 for a master to: bool client = false; // Save bool for client/master
6. Set the desired trigger mode in line 11 : for Prince: ce_ms = false; for Agilent: ce_ms = true
7. Upload CDC_Funk_MC_Multi_v19.ion with the arrow button „→„ or use Sketch > Upload; Success is confirmed by a message „Upload completed“ below the sketch.
8. Open the serial monitor (magnifying glass at the top right), select 19200 baud in the new window at the bottom right

   Output:

```
Firmware:CDC_Funk_MC_Multi_v19
Enter 'N_X' to reset the wireless device and set it to client-master-set X
„„„ PRESS 'E' to handle the switch of adding new clients
Hello at CDC0
Prince Trigger-Mode (2 wire connection)
SBNetwork Version 1.0.5
====================

Try to read device config from internal flash...Done
Device MAC = 0x5 0x4 0x4 0x8 0x0
Master MAC = 0x0 0x0 0x0 0x0 0x0
NetKey = 0
Masterstorage Slot 0 0x5 0x4 0x4 0x8 0x1
Masterstorage Slot 1 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 2 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 3 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 4 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 5 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 6 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 7 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 8 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 9 0x0 0x0 0x0 0x0 0x0
Initializing NRF24L01 transmitter...Done
```

9. Configuration of the set ID „S“: Enter N_“S“ in the input field above the output in the serial monitor and confirm with „Enter“

   The output should be: (here: „S“ = 9)

```
Erasing device configuration data...Done
SBNetwork Version 1.0.5
====================

Try to read device config from internal flash...Failed
Creating new device config and stroing it to internal flash...
Done
Device MAC = 0x5 0x4 0x4 0x9 0x0
Master MAC = 0x0 0x0 0x0 0x0 0x0
NetKey = 0
Creating new Master Storage
Masterstorage Slot 0 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 1 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 2 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 3 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 4 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 5 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 6 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 7 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 8 0x0 0x0 0x0 0x0 0x0
Masterstorage Slot 9 0x0 0x0 0x0 0x0 0x0
Initializing NRF24L01 transmitter...Done
”””””
Configuration:Master client-master-set-number:9 Device-Number: 0
”””””
```

10. Activation of the automatic detection device adding (all detection device that should not be connected to the master must not be in use): Enter „E“ in the input field above the output in the serial monitor and confirm with Enter

    Output:

    ```
    ”””””
    Activating AutomaticClientAdding
    ”””””
    ```

11. Power one detection device after the other. The master’s output displays:

    ```
    Received 'SEARCH_MASTER'. Send MasterACK...Done
    Received 'PAIRING_REQUEST'. Send PairingACK... Done
    Storing new MAC to MasterStorage... Done
    CDC1;8388608;0;13195058
    …
    Received 'SEARCH_MASTER'. Send MasterACK...Done
    Received 'PAIRING_REQUEST'. Send PairingACK... Done
    Storing new MAC to MasterStorage... Done
    CDC2;8388608;0;13030720
    ```

    etc.

12. Enter „E“ in the input field above the output in the serial monitor and confirm with Enter to disable the automatic detection device adding, after adding all detector devices.

    Output:

    ```
    ”””””
    Deactivating AutomaticClientAdding
    ”””””
    ```

13. Disconnect the master from the USB cable
14. Configuration is finished

Follow-up configuration – detection device

Note: Only connect one Arduino on the PC during configuration.

1. Connect the PC to the USB port on the Arduino Nano with installed custom-made PCB (switch 4 on OFF) and radio module
2. Open Arduino IDE
3. Select the COM port at Tools > Ports > COM X
    - If the port is not known: unplug the Arduino, display the port list, remind and close it. Plug in the Arduino and select the new entry in the port list
4. Set the detection device ID according to: switch 2 adds 1 to CDC1 → CDC2; switch 3 adds 2, switch 4 adds 4; switch 5 on ON activates the radio module.

    All options:

S 2	S 3	S 4	Name	S 2	S 3	S 4	Name
OFF	OFF	OFF	CDC1	OFF	OFF	ON	CDC5
ON	OFF	OFF	CDC2	ON	OFF	ON	CDC6
OFF	ON	OFF	CDC3	OFF	ON	ON	CDC7
ON	ON	OFF	CDC4	ON	ON	ON	CDC8

    Leave switch 5 on OFF for configuration

    It is only necessary for the configuration of the CDC8 that a detector is plugged in and supplied with voltage.

5. Open the serial monitor (magnifying glass at the top right), select 19200 baud in the new window at the bottom right

   The output should be:

   ```
    Hello at Number: X
   „„„ PRESS 'N_4' to reset the wireless device and set it to client-master-set
   4,
   if no confirmation is displayed, retry it with disabled Network
   Init...done
   CDCX
   ```

   If X does not correspond to the set value: set the switches again and press the reset button.

6. Configuration of the set ID *S*: Enter N_*S* in the input field above the output in the serial monitor and confirm with “Enter”

   The output should be: (here: *S* = 9 and X = 2)

   ```
   Erasing device configuration data...Done
   SBNetwork Version 1.0.5
   ====================
   Try to read device config from internal flash...Failed
   Creating new device config and stroing it to internal flash...
   Done
   Device MAC = 0x5 0x4 0x4 0x9 0x2
   Master MAC = 0x0 0x0 0x0 0x0 0x0
   NetKey = 0
   Initializing NRF24L01 transmitter...Done
   Try to connect to master...Warning - Not paired to a master
   ```

7. Unplug the USB cable from the Arduino and set switch 5 to ON, radio is activated, connect the detection head

Reset of the master device

Note: all previously trained detector devices will be deleted

Procedure analogous to steps 1 - 3 of the “Follow-up configuration – detection device”

4. All coding switches are set to OFF or are not installed.
5. Open the serial monitor (magnifying glass at the top right), select 19200 baud in the new window at the bottom right

   Output:

   ```
   Firmware:CDC_Funk_MC_Multi_v19
   „„„ Enter 'N_X' to reset the wireless device and set it to client-master-set
   X
   „„„ PRESS 'E' to handle the switch of adding new clients
   Hello at CDC 0
   Prince Trigger-Mode (2 wire connection)
   SBNetwork Version 1.0.5
   ====================

   Try to read device config from internal flash...Done
   Device MAC = 0x5 0x4 0x4 0x9 0x0
   Master MAC = 0x0 0x0 0x0 0x0 0x0
   NetKey = 0
   Masterstorage Slot 0 0x5 0x4 0x4 0x8 0x1
   Masterstorage Slot 1 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 2 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 3 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 4 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 5 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 6 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 7 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 8 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 9 0x0 0x0 0x0 0x0 0x0
   Initializing NRF24L01 transmitter...Done
   CDC1;556;0;8396837
   ```

6. Configuration of the set number *S*: Enter N_*S* in the input field above the output in the serial monitor and confirm with “Enter”

   Output: (here: *S* = 9)

   Output:

   ```
   Erasing device configuration data...Done
   SBNetwork Version 1.0.5
   ====================

   Try to read device config from internal flash...Failed
   Creating new device config and stroing it to internal flash...
   Done
   Device MAC = 0x5 0x4 0x4 0x9 0x0
   Master MAC = 0x0 0x0 0x0 0x0 0x0
   NetKey = 0
   Creating new Master Storage
   Masterstorage Slot 0 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 1 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 2 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 3 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 4 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 5 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 6 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 7 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 8 0x0 0x0 0x0 0x0 0x0
   Masterstorage Slot 9 0x0 0x0 0x0 0x0 0x0
   Initializing NRF24L01 transmitter...Done

   „„„„„
   Configuration:Master client-master-set-number:9 Device-Number: 0

   „„„„„
   ```

7. Activation of the automatic detection device adding (all detection device that should not be connected to the master must not be in use): Enter “E” in the input field above the output in the serial monitor and confirm with Enter

   Output:

   ```
   „„„„„
   Activating AutomaticClientAdding
   „„„„„
   ```

8. Power one detection device after the other. The master’s output displays:

   ```
   Received 'SEARCH_MASTER'. Send MasterACK...Done
   Received 'PAIRING_REQUEST'. Send PairingACK... Done
   Storing new MAC to MasterStorage... Done
   CDC1;8388608;0;13195058
   …
   Received 'SEARCH_MASTER'. Send MasterACK...Done
   Received 'PAIRING_REQUEST'. Send PairingACK... Done
   Storing new MAC to MasterStorage... Done
   CDC2;8388608;0;13030720
   ```

   etc.

9. Enter “E” in the input field above the output in the serial monitor and confirm with Enter to disable the automatic detection device adding, after adding all detector devices.

   Output:

   ```
   „„„„„
   Deactivating AutomaticClientAdding
   „„„„„
   ```

10. Disconnect the master from the USB cable
11. Configuration is finished

Add additional detection devices to a master

Prepare the detection device with the same set ID as all other detection device according to the item “Upload and configuration for the detection devices” or “Follow-up configuration – detection device”.

Then continue according to “Reset of the master” WITHOUT performing step 6

Note: Another client can be added in while other clients are in operation

Hardware

A CDCD-set can consist out of two different devices which in turn consist of individual units:

1.) Detection device:
   - detection head
   - supply unit
2.) Master device (only needed with wireless data transmission)

A CDCD-set consist of a master device and up to 8 detection devices.

The different hardware configurations of the detection head and the supply unit are described in the chapters "Building instructions for detection head" and "Building instructions for the supply unit"

Known hardware problems

Error pattern	Solution
Data is received very rarely (frequency below 10 Hz)	Try another position of the master or detection device supply unit
The Arduino's pins get warm	Check whether there is no short circuit (start with: 3V3 against GND on the Arduino if there is a connection, remove the radio module and check again. If there is no more short circuit: replace the radio module)

Building instructions for detection head

1. Detach of the 4 mini-PCBs on the right of Figure 3 the result is shown in Figure 4.

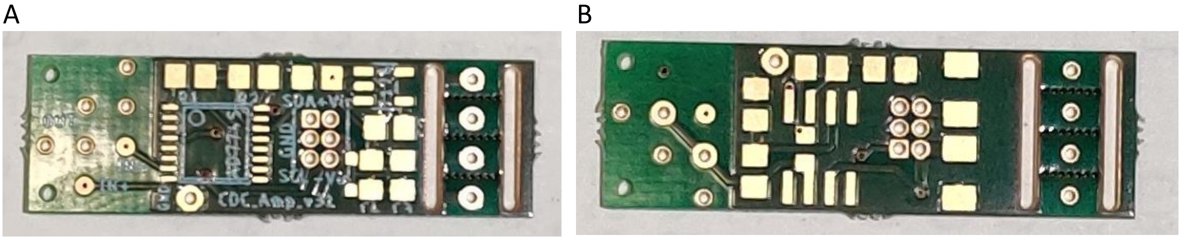


Figure 3: Images of the (A) top and (B) bottom of the entire detection head's PCB.

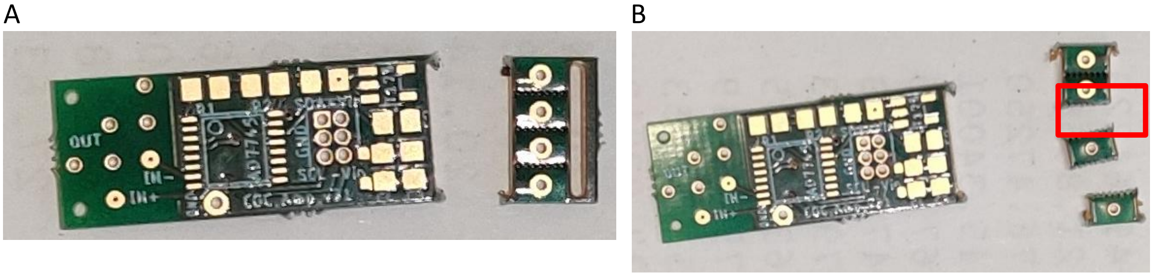


Figure 4: Images of the detection head circuit board with detached mini circuit boards. One of the further used mini PCBs is outlined in red.

2. Solder the required components according to Table 2 onto the larger PCB. The schematics is displayed in Figure 5. The equipped top and bottom of the detection head's PCB with and without installed amplifier can be found in Figure 6.

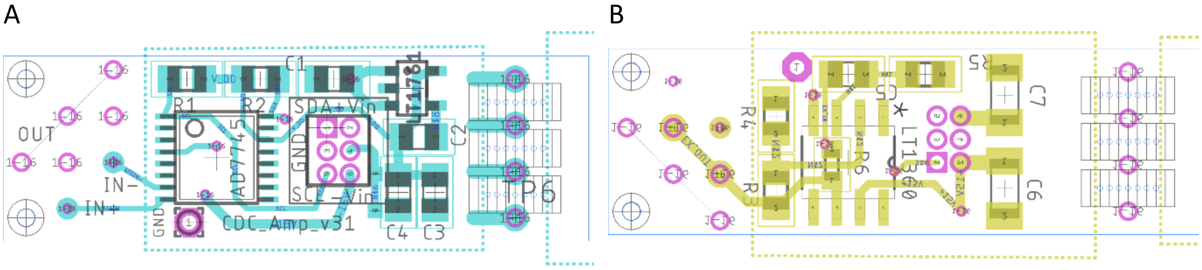


Figure 5: Schematics of the (A) top and (B) bottom of the detection head's PCB.

Table 2: Components of the detection head

Top:

Part	Value	Package
AD7745	AD7745	TSSOP16
LT1761-5	LT1761	SOT23-5
C1	1u	C0805
C2	10n	C0805
C3	10u	C0805
C4	100n	C0805
JP1	FTSH-103-01-F-D	

Bottom:

Configuration 1 and 3 (no Amplifier):

solder pads of R6 together

Configuration 2 and 4 (with amplifier):

Part	Value	Package
LT1360	LT1360CS8PBF	
C5	100n	C0805
R3	51k	R0805
R4	10k	R0805
R5	10k	R0805
C6	reserved	C1206K
C7	reserved	C1206K

Information: The head could also be supplied by the 5 V line of the Arduino, if the jumper Vopt|VA on the supply unit of the detection unit is connected and the test pad next to the upper pad of C4 is linked, see Figure 5

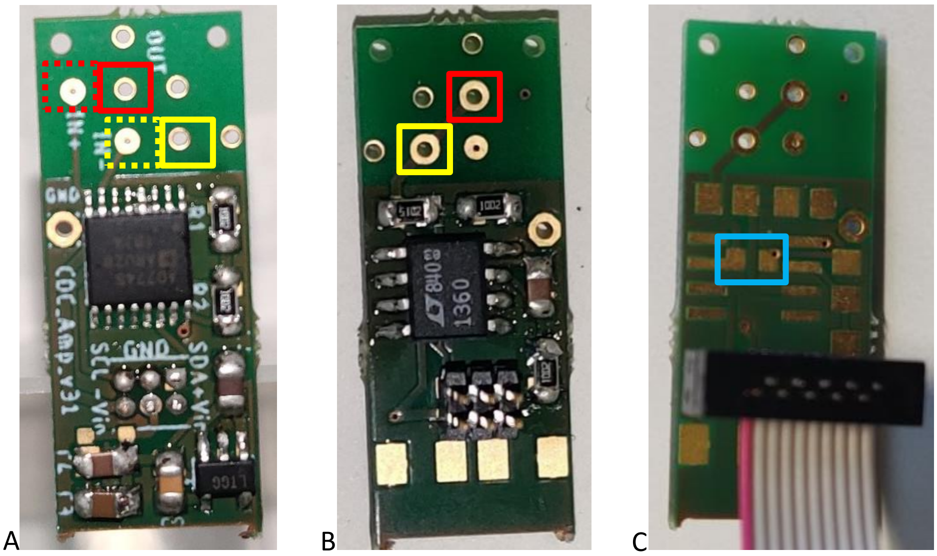


Figure 6: Images of the equipped (A) top and (B+C) bottom of the detection head's PCB with (B) and without (C) amplifier (link between the two pads in the blue box is missing)

3. Prepare the cannula segments by cutting a cannula to the desired length. Inset a capillary during cutting. Use cannulas with inner diameter of 0.4 mm (outer diameter 0.6, Gauge 23).

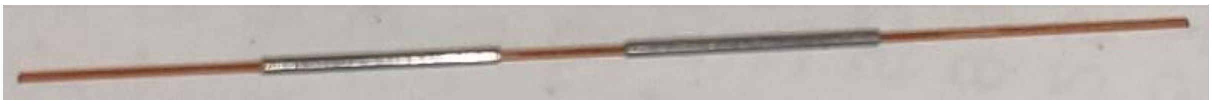


Figure 7: A pair of cannulas cut to a length of 10 mm each on a 360 µm outer diameter capillary.

4. Solder one cannula segment to the PCB of the detection head (outlined in red in Figure 6) and one to the mini-PCB. The soldering joint must be on the side with the larger soldering pad this is also the side where the electrode sticks out. The other side of the electrode should be at the same level as the PCB. The result is shown in Figure 8. For setups in differential measurement mode, a second pair of electrodes are mounted in the same way but to the yellow outlined boxes in Figure 6

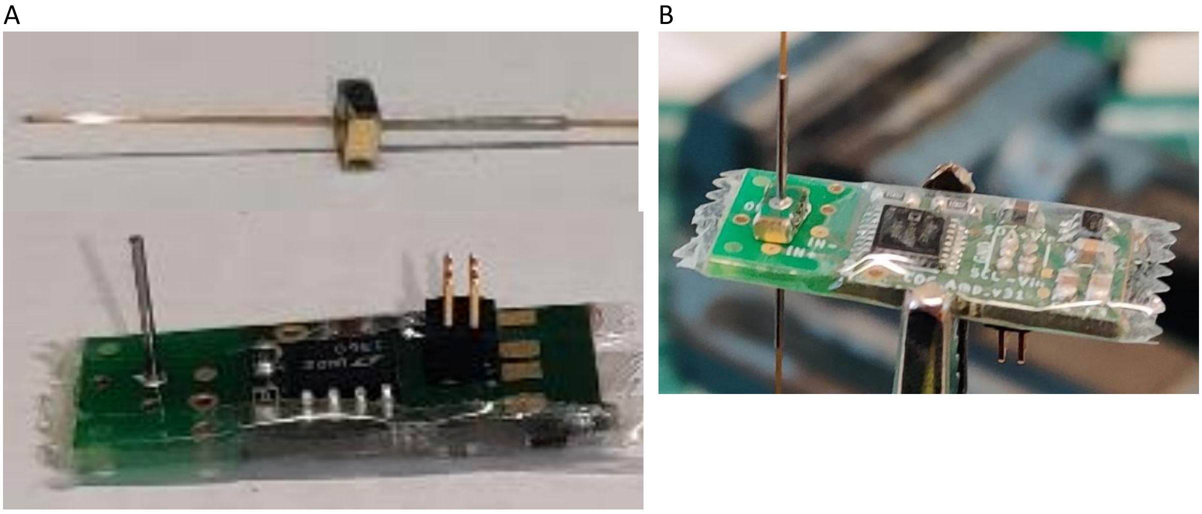


Figure 8: (A) mini-PCB and large PCB equipped with electrodes. The larger PCB is wrapped in transparent adhesive tape to prevent short circuits. The mini-PCB with an electrode is equipped with a capillary. (B) Both PCBs just before soldering them together (golden side of the mini-PCB and the pad next to the label IN+)

5. After wrapping the larger PCB in transparent adhesive tape, punch a hole with a needle in the tape, where the capillary must pass it.

6. Solder the mini-PCB to the larger PCB, it is important to solder the golden side connected electrically to the electrode of the mini-PCB to the pad connected to the input of the CDC-IC. The mounting pad IN+ is dotted outlined in red in Figure 6. The tape covering the mounting pad has not to be removed since it will melt during the soldering process anyway. Insert a capillary trough both electrodes while soldering.

7. Use heat shrink tubing to cover the electrodes and carefully wrap the PCBs with a layer of transparent adhesive tape. Keep the tape away from the golden side of the large PCB.

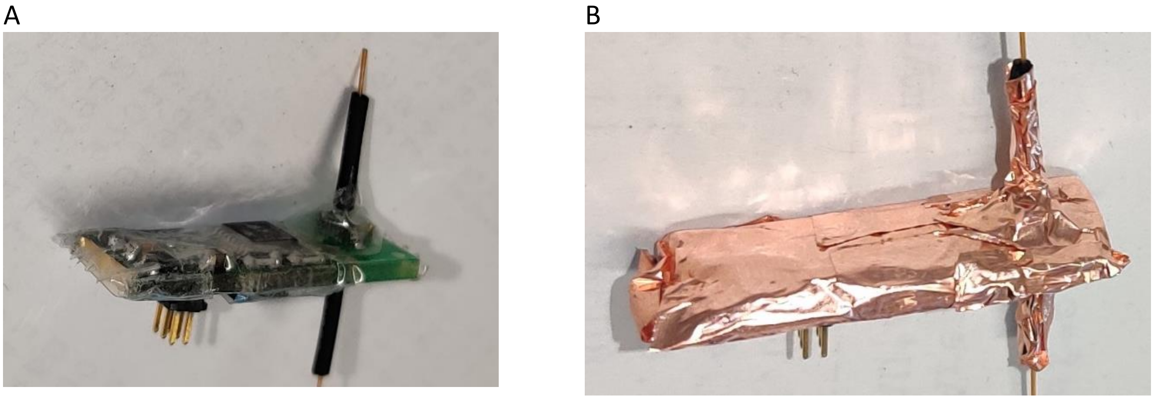


Figure 9: detection head (A) before and (B) after wrapping with adhesive copper foil

8. Wrap the detector carefully with adhesive copper foil, solder it to the golden side of the large PCB to connect it to ground. If necessary, solder the individual layers of the outer copper layer together to ensure proper electrical connection among themselves.

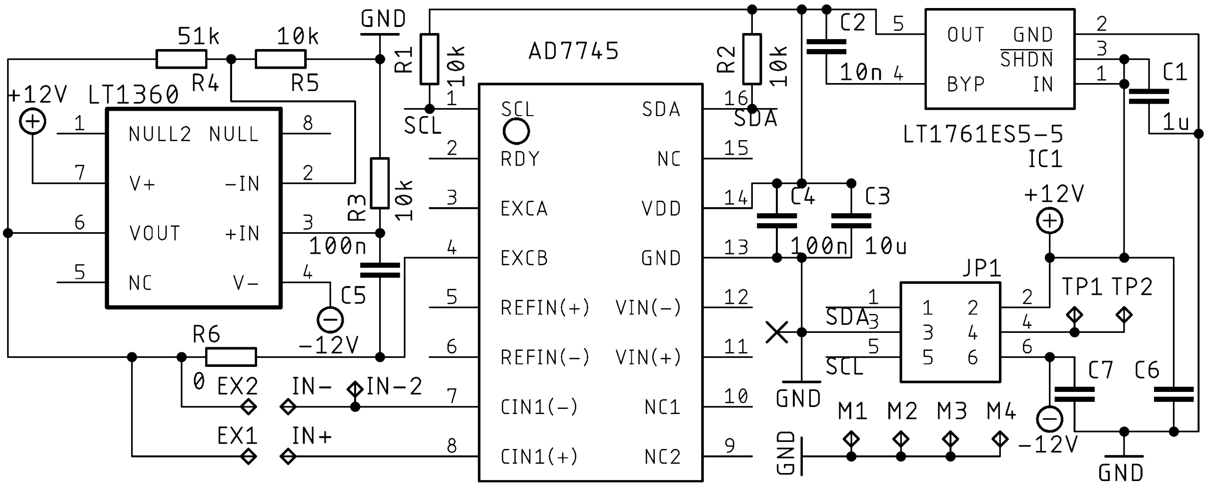


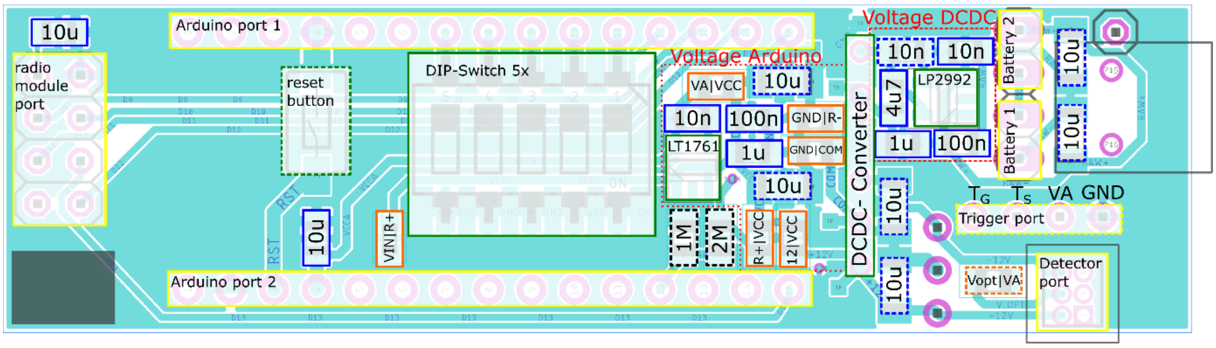
Figure 10. Schematic diagram of the detection head's circuit. Capacitors without values are reserved. Resistor R6 is a jumper.

Building instructions for the supply unit

The supply unit consists except for a PCB described below out of an Arduino Nano and a NRF24L01+ radio module. It provides pins for the connection to the detection head as well as an optional PCB terminal block for a trigger signal with an additional 5 V output. The DIP-switches on the supply unit are used for configuration.

The necessary components on the supply unit depending on the desired configuration can be taken from the figures in this section.

Fully annotated PCB of a supply unit:



Color code: yellow: connectors; green: IC, buttons, and switches; black: resistors; blue: capacitors; orange: jumpers

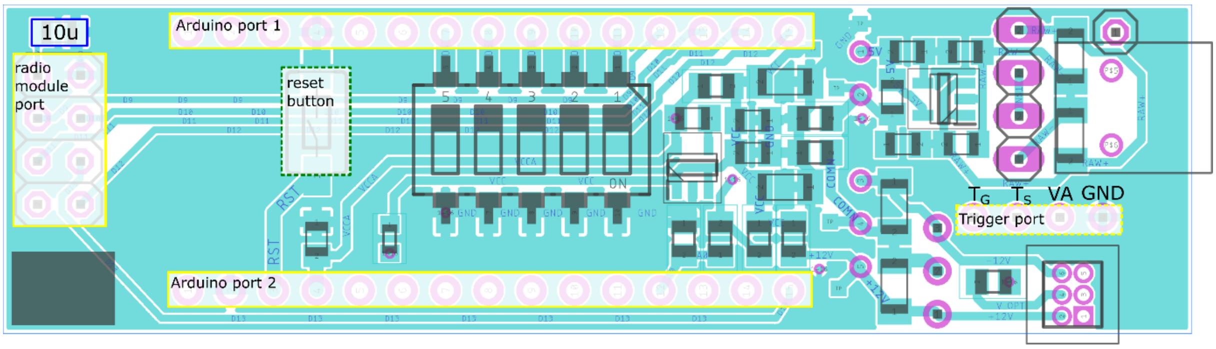
Information: the dashed outlined 1 MΩ and 2 MΩ resistors are optional and allow tracking of the battery voltage.

Information: the dashed outlined reset button and the trigger port PCB block terminal are optional as well as all dashed outlined 10 µF capacitors.

Information: the dashed outlined 10 nF capacitor in Voltage DCDC is necessary if LP2992-5 is replaced by an LT1761-5. Further, the solid outlined 10 nF capacitor must not be installed and the 4.7 µF capacitor must be replaced by a 10 µF capacitor.

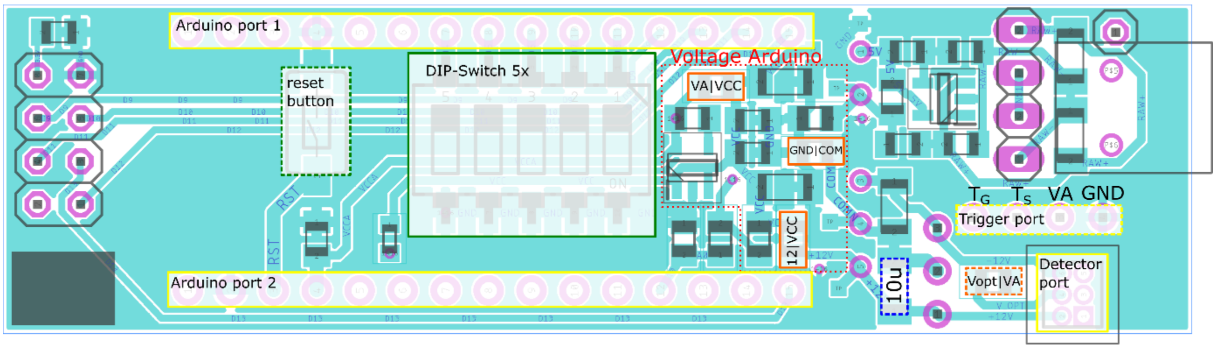
Information: The detection head can also be supplied directly from the 5 V supplying the Arduino by linking the dashed outlined jumper Vopt|VA

Supply unit of a master device

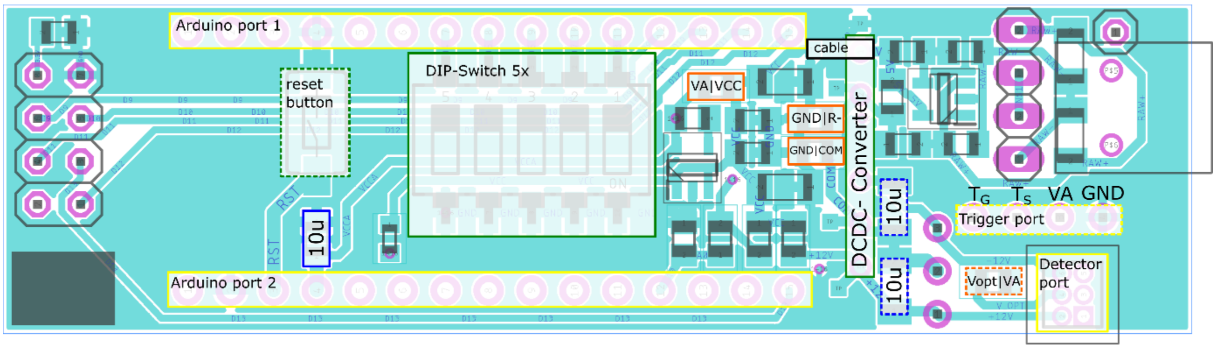


Supply units for wired data transmission

Configuration 1: Use of only one client without radio and without amplifier [performance not tested]

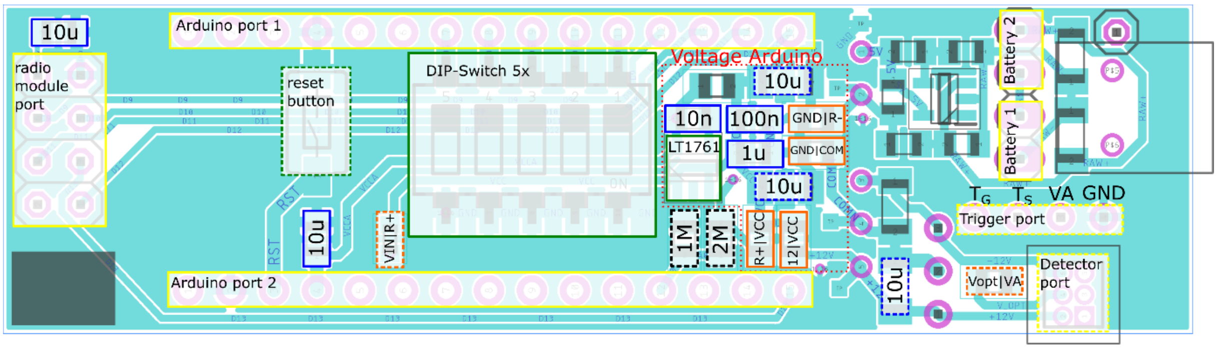


Configuration 2: Use of only one client without radio and with amplifier [performance not tested]



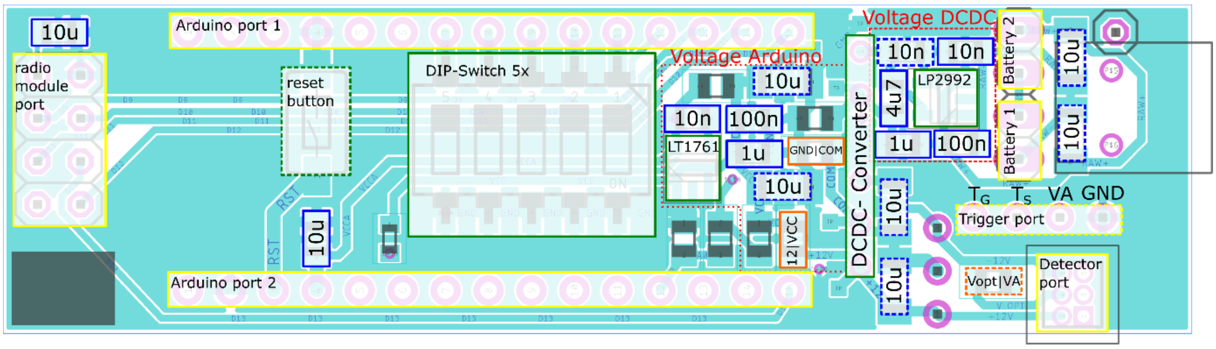
Supply units for wireless data transmission and battery supply:

Configuration 3: Usage without amplifier

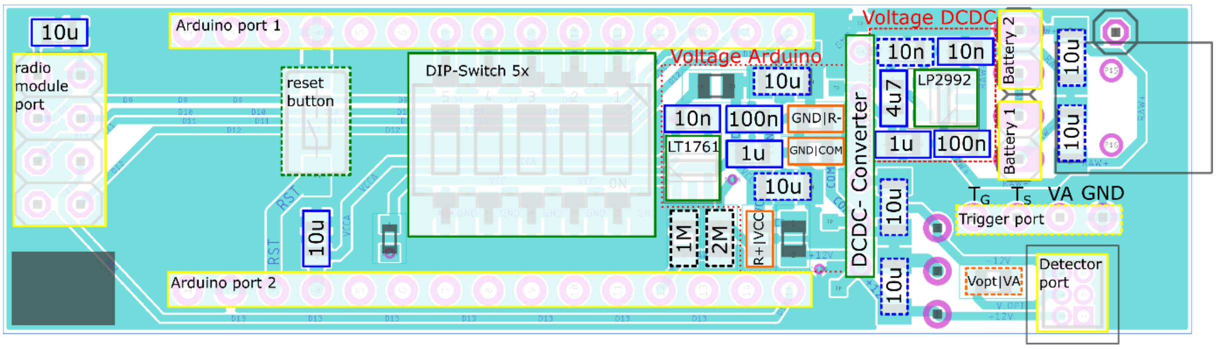


Configuration 4: Usage with amplifier

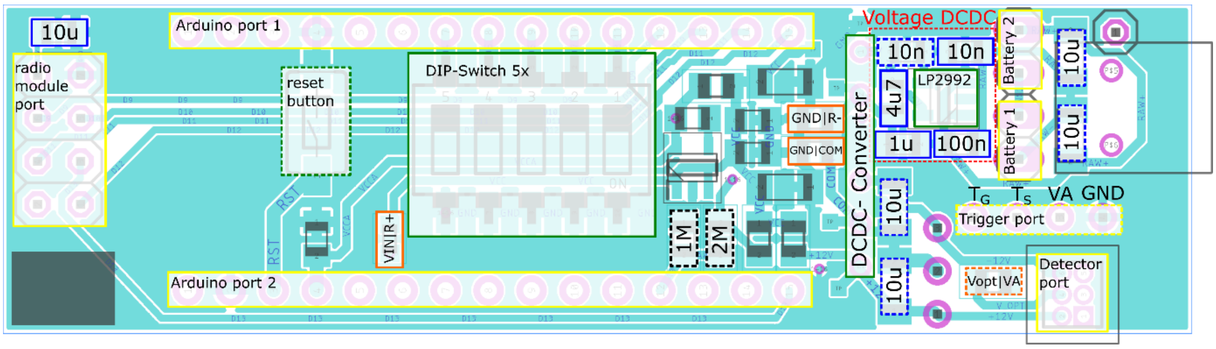
a) galvanic separation from the input voltage (the setup is powered entirely through the DCDC-Converter):



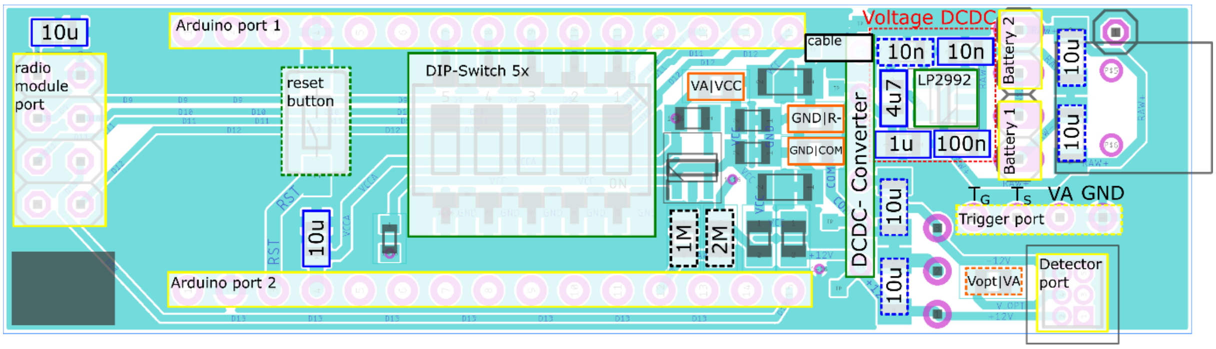
b) Supply Arduino via LT1761 from batteries



c) Supply Arduino via its own voltage regulator [performance not tested]



d) Supply Arduino via LP2992: (this setup also allows a wired or wireless data transmission which is powered via the 5 V line of the Arduino (Configuration 2)) [performance not tested]



Images of a supply unit in configuration 4d, Arduino is supplied via LP2992

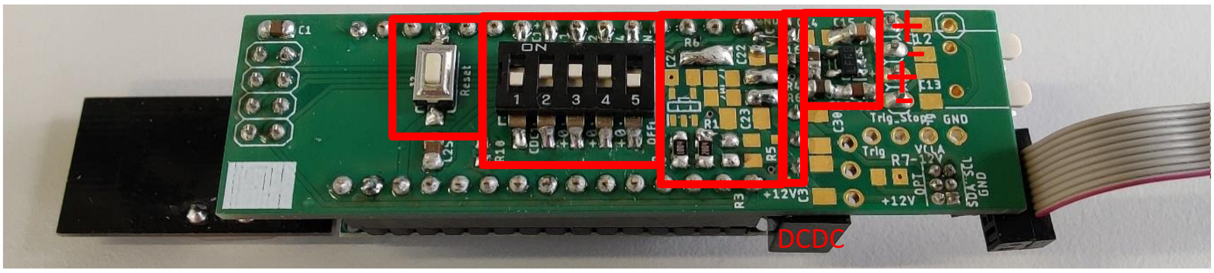


Figure 11. Supply unit of detection device – top
Left: Reset-Taster
Center left: DIP-switch. switch 2-4 to configure id; switch 5: enable radio module (switch on OFF)
Center: Area for voltage regulator for Arduino and resistors (with values 1M, 2M), here Arduino powered directly from LP2992 (VA|VCC are linked and a wire is installed between the test pad next to the print VCC and Pin 1 of the DCDC; see top right corner)
Center right: DCDC solder pads
Right: Area for voltage regulator for DCDC-converter
"+" and "-" denote the polarities of two in series connected ports for the batteries (3.7 V each)

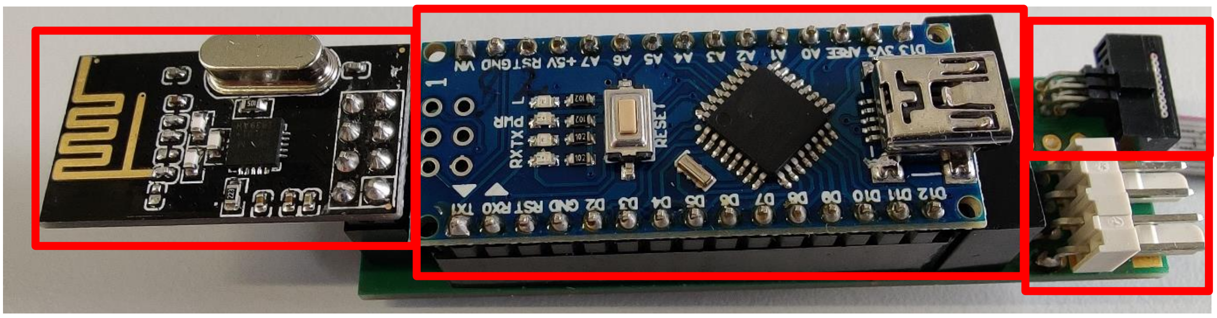


Figure 12. Supply unit of detection device – bottom, equipped Left: radio module
Center: Arduino Nano
Top right: connector for 10-core ribbon cable to detection head
Bottom right: two connectors to attach the necessary pair of batteries (2x 3.7 V)

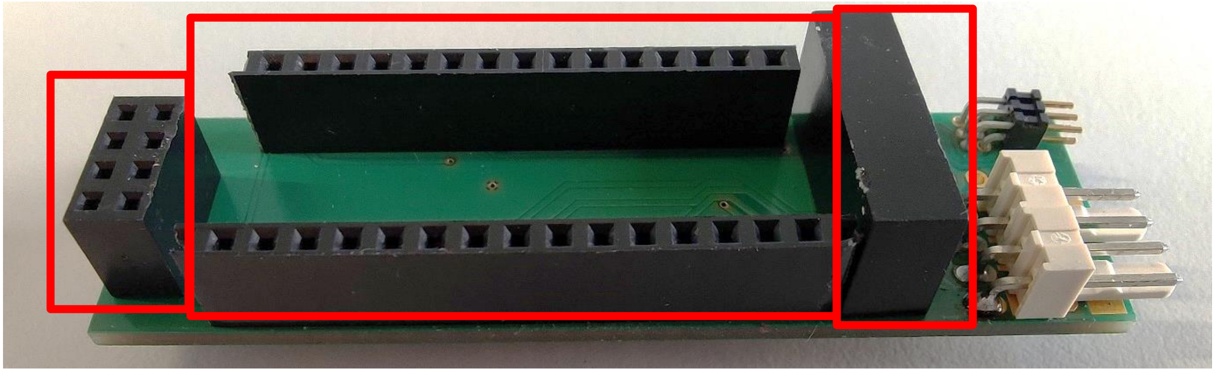


Figure 13. Supply unit of detection device – bottom
Left: 2x4 socket for radio module
Center: 2x 1x15 socket for Arduino Nano
Right: DCDC-Converter

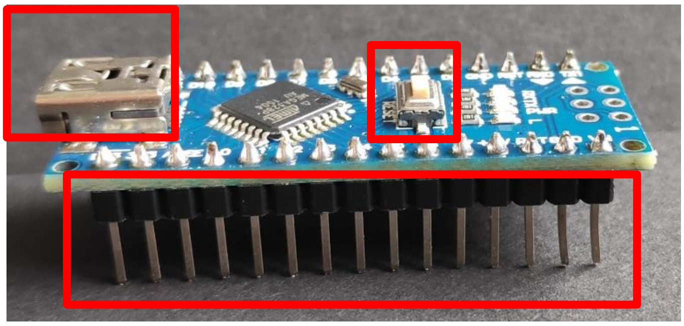


Figure 14. Arduino Nano Top
Left: Mini-USB port for configuration or communication and supply in case of wired data communication
Top right: Reset-Button, restarts Arduino
Bottom: Pins

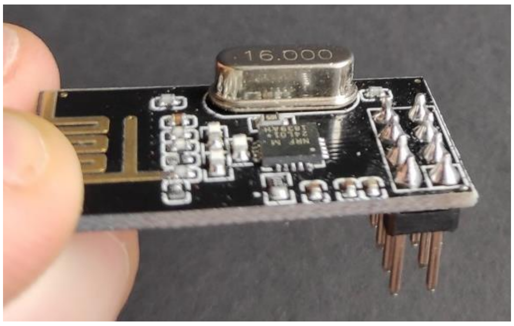
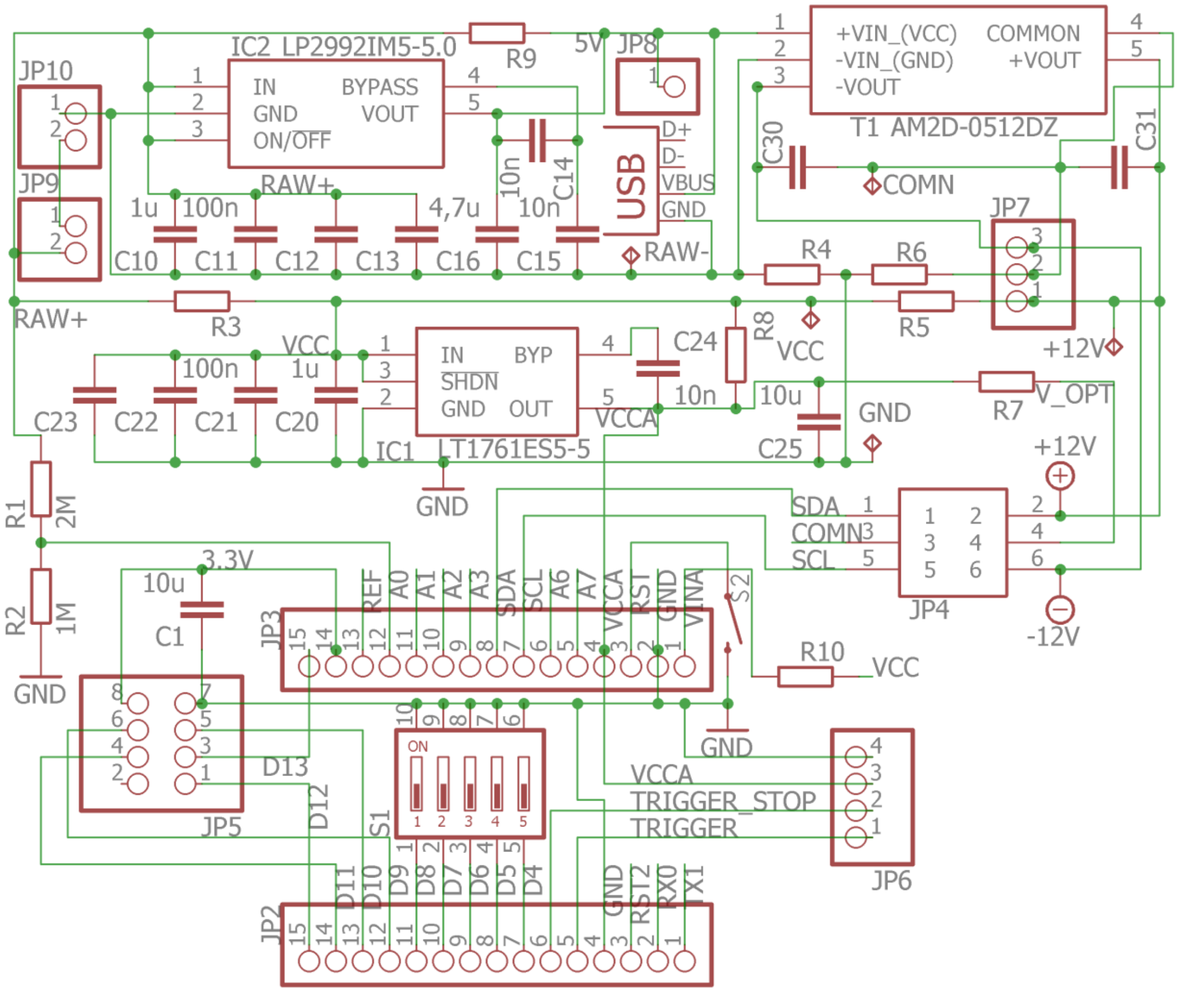


Figure 15.Radio module (NRF24L01)
Bottom right: Pins 2x4

Scheme of the supply unit's circuit



IC2 LP2992IM5-5.0
IN
GND
ON/OFF
BYPASS
VOUT
R9
5V
JP8
JP10
JP9
RAW+
1u
100n
4,7u
10n
10n
C14
C10
C11
C12
C13
C16
C15
USB
D+
D-
VBUS
GND
RAW-
+VIN_(VCC)
-VIN_(GND)
-VOUT
COMMON
+VOUT
T1 AM2D-0512DZ
C30
C31
COMN
JP7
R4
R6
R5
R8
VCC
+12V
RAW+
R3
VCC
100n
1u
IC1 LT1761ES5-5
IN
SHDN
GND
BYP
OUT
C24
10n
VCCA
10u
C25
GND
R7
V_OPT
C23
C22
C21
C20
GND
+12V
SDA
COMN
SCL
JP4
-12V
R1
2M
R2
1M
GND
3.3V
10u
C1
JP3
REF
A0
A1
A2
A3
SDA
SCL
A6
A7
VCCA
RST
GND
VINA
S2
R10
VCC
JP5
D13
D12
S1
ON
GND
VCCA
TRIGGER_STOP
TRIGGER
JP6
D11
D10
D9
D8
D7
D6
D5
D4
GND
RST2
RX0
TX1
JP2