Fundamentals of C Programming in ISOBUS (1)

Franz Höpfinger

Contents

myst: enable_extensions: [“colon_fence”] html_meta: “description lang=en”: “metadata description” “description lang=de”: “metadata description” “description lang=fr”: “description des métadonnées” “keywords”: “Sphinx, MyST” “property=og:locale”: “de” 6

1 Wiki 1: C Programming in ISOBUS 7

1.1 Master Craftsmen’s Schools at Munich East Station 7

1.1.1 📋 This is in the curriculum (Framework Curriculum 2025) 7

2 Atom Extensions 9

3 Eagle 13

3.1 🎧 Podcast 13

3.2 📺 Video 13

3.3 End of life 13

3.4 Installer 13

3.5 Problems 13

4 📦 Installing and using Gitkraken 15

5 🔌 Hardware 31

5.1 🎧 Podcast 31

6 Homepage 33

6.1 🎧 Podcast 33

7 DIN Rail Moped Inputs 35

7.1 Differentiation of Limit Switches for Pneumatic Cylinders 35

7.1.1 Reed Switch 35

7.1.2 Semiconductor Magnetic Sensors 35

7.2 Connecting the Limit Switch to the Input 38

7.3 Grove System 39

7.4 Series Connection of Limit Switches 40

7.5 Button Boards 40

7.5.1 🌐 Related Topic Subpages on ms-muc-docs.de 40

8 DIN Rail Moped Extensions 41

9 DIN Rail Moped Connectors 51

9.1 1. Basic Wiring and Connections 51

9.2 **2. Insulation and Touch Protection** . . . 54
9.3 **3. Detail View: Special Signal Connections** . . . 55
9.4 **4. Connector Housing and Overall View** . . . 56
9.5 **5. Operating State** . . . 60

10 DINRailMoped . . . 63
10.1 🎧 **Podcast** . . . 63
10.2 📂 **Repository & Resources** . . . 63
10.3 ⚡ **Connection & Pinout** . . . 63
10.3.1 Pin Assignment . . . 63
10.4 ⚙️ **Configuration (4-pin vs. 6-pin)** . . . 63
10.4.1 Operation with a 4-pin cable . . . 63
10.5 📸 **Gallery & Mounting** . . . 64
10.5.1 🔌 Hardware Views . . . 64
10.5.2 Mounting Options . . . 66
10.5.3 🌐 Related topic subpages on ms-muc-docs.de . . . 67

11 DIN Rail Moped XL . . . 69
11.1 🌟 **Highlights** . . . 69
11.2 🎖️ **Certification** . . . 69
11.3 🛠️ **Design & CAD** . . . 70
11.3.1 Repository . . . 70
11.4 📸 **Gallery** . . . 70
11.5 **i Technical Details** . . . 71
11.5.1 🌐 Related topics on ms-muc-docs.de . . . 71

12 ISOBUS cable, small . . . 73

13 📦 Installation . . . 79

14 IsoAgLib-ISOMAN . . . 81

15 JTAG-S3-USB . . . 85
15.1 🌐 **Relevant topic subpages on ms-muc-docs.de** . . . 92

16 JTAG . . . 93
16.1 **JTAG Step 01** . . . 93
16.1.1 Switching the USB interface with ZADIG . . . 93
16.2 **JTAG Step 2** . . . 101
16.3 **JTAG Step 3** . . . 102
16.4 **JTAG Step_04** . . . 102
16.5 **JTAG Step_05** . . . 103
16.6 **JTAG Step_06** . . . 103
16.7 **JTAG Step_07** . . . 103

17 Joystick . . . 105
18 Hardware Cost Calculation . . . 107
18.1 🌐 Related topic subpages on ms-muc-docs.de . . . 107
19 Lego . . . 109
19.1 🎧 Podcast . . . 133
19.1.1 🌐 Related topic subpages on ms-muc-docs.de . . . 133
20 Optional . . . 135
21 Package . . . 137
21.1 🎧 Podcast . . . 137
21.2 📺 Video . . . 138
22 Self-Test . . . 139
23 Setting up (LINUX) . . . 141
24 Setting-up-II . . . 143
25 Installation Setup . . . 153
25.1 Step 1 . . . 153
25.2 Step 2 . . . 153
25.3 Step 3 . . . 154
25.4 Step 4 . . . 155
25.5 Step 5 . . . 156
25.6 Step 6 . . . 158
25.7 Step 7 . . . 159
25.8 Step 8 . . . 160
25.9 Step 9 . . . 161
25.10 Step 10 . . . 162
25.11 Step 11: (Information only) . . . 163
25.12 Step 12 . . . 164
26 blink_on_atom . . . 165
26.1 Step 14 . . . 165
26.2 Step 15 . . . 165
26.3 Step 16 . . . 166
26.4 Step 17 . . . 167
26.5 Step 18 . . . 170
26.6 Step 20 . . . 170
26.7 Step 21 . . . 171

26.8 Step 22 174
26.9 Modifying the Program 176

27 cci_EasyExample 177

28 cci_EasyExample_CAN2IP 179

29 📋 This is on the curriculum: C programming in ISOBUS 183

29.1 🎯 Relevance of C Programming & ISO 11783 C Libraries in the 2025 Curriculum 183

29.2 📚 Curriculum Modules & Focus Areas 183

29.2.1 Part 1 – Learning Unit 1.4: Automated Functions (80 hours) 183
29.2.2 Part 2 – Learning Unit 1.1: Systems Engineering (120 hours) 183
29.2.3 🌐 Relevant subpages on ms-muc-docs.de 183

29.3 🔗 Overview, Main Portal & Direct Links to all 6 Wikis 183

29.3.1 📄 PDF Downloads for All Wikis 184

0.1 myst: enable_extensions: [“colon_fence”] html_meta: “description lang=en”: “metadata description” “description lang=de”: “metadata description” “description lang=fr”: “description des métadonnées” “keywords”: “Sphinx, MyST” “property=og:locale”: “de”

1. Wiki 1: C Programming in ISOBUS



Meisterschulen
am Ostbahnhof

1.1 Master Craftsmen's Schools at Munich East Station

1.1.1 📋 This is in the curriculum (Framework Curriculum 2025)

C/C++ programming for embedded systems is the technical basis for implementing standards-compliant ISOBUS communication stacks (ISO 11783) on agricultural and construction machinery (Part I LE 1.4 and Part II LE 1.1).

- 📋 **This is in the curriculum – C Programming in ISOBUS**

Welcome to the documentation for C programming in ISOBUS.
This documentation is part of the knowledge base of the Master Craftsmen's Schools at Munich East Station.

Useful Links & Main Portal:

- 🌐 **Main Portal Master Schools Munich**
- 📋 **What's in the Curriculum (Curriculum 2025)**
- 📋 Interactive Framework Curriculum 2025
- 📄 Central PDF Download Hub
- 🔍 Super Search (all Wikis)
- 📄 Download PDF Handbook

!!! Note
This project is under active development.

- **Contents**

---

2. Atom Extensions

Basically, all extensions from M5 Stack work on the GROVE port: https://shop.m5stack.com/collections/m5-sensor and all extensions that offer the GROVE port: https://wiki.seeedstudio.com/Grove_System/ https://www.youtube.com/watch?v=1Rc_OiebDPo

We will be working with the following:

https://shop.m5stack.com/collections/m5-sensor/products/mini-dual-button-unit

https://shop.m5stack.com/collections/m5-sensor/products/angle-unit

https://shop.m5stack.com/collections/m5-sensor/products/mini-button-unit

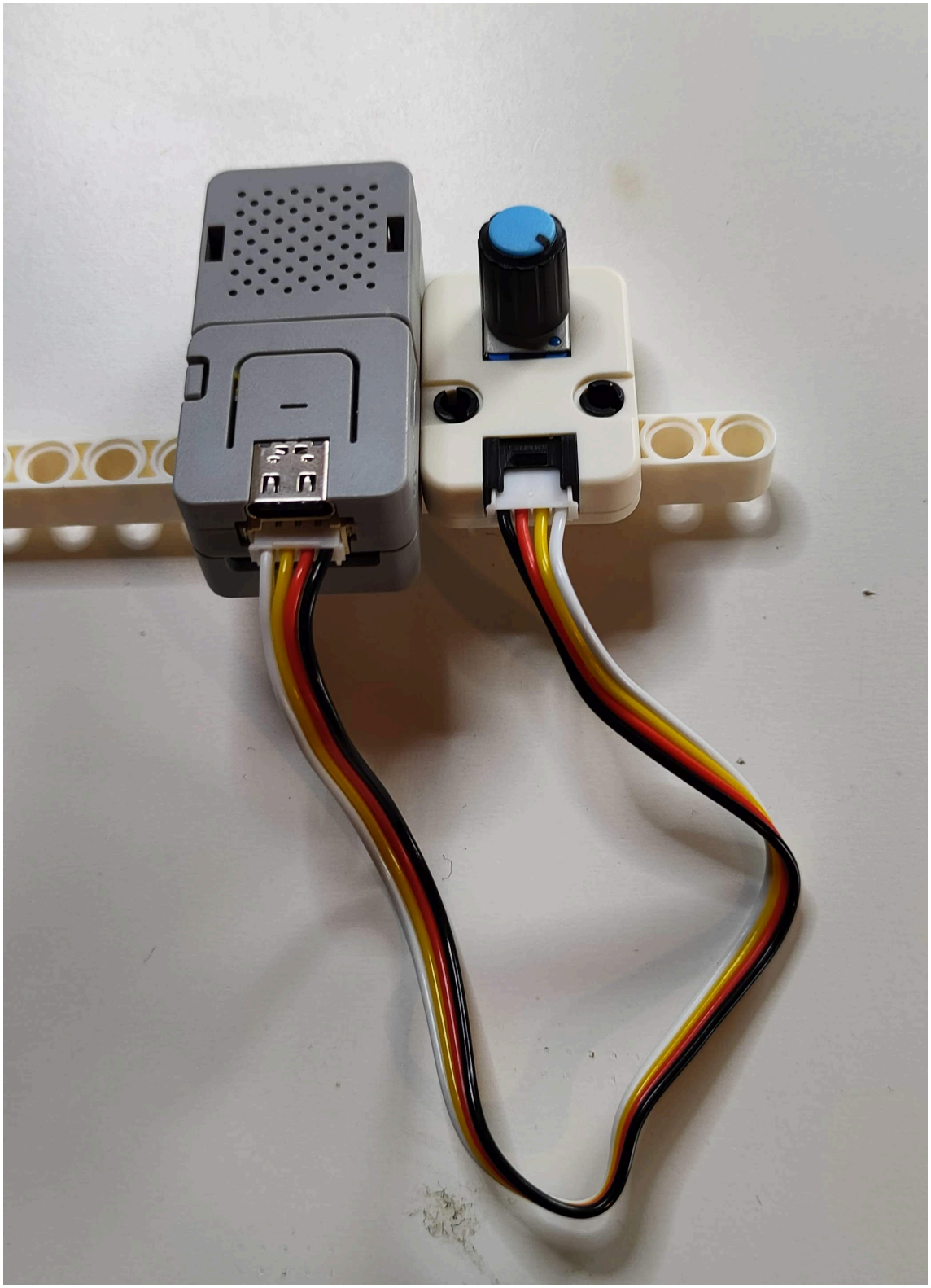
Other interesting ones:

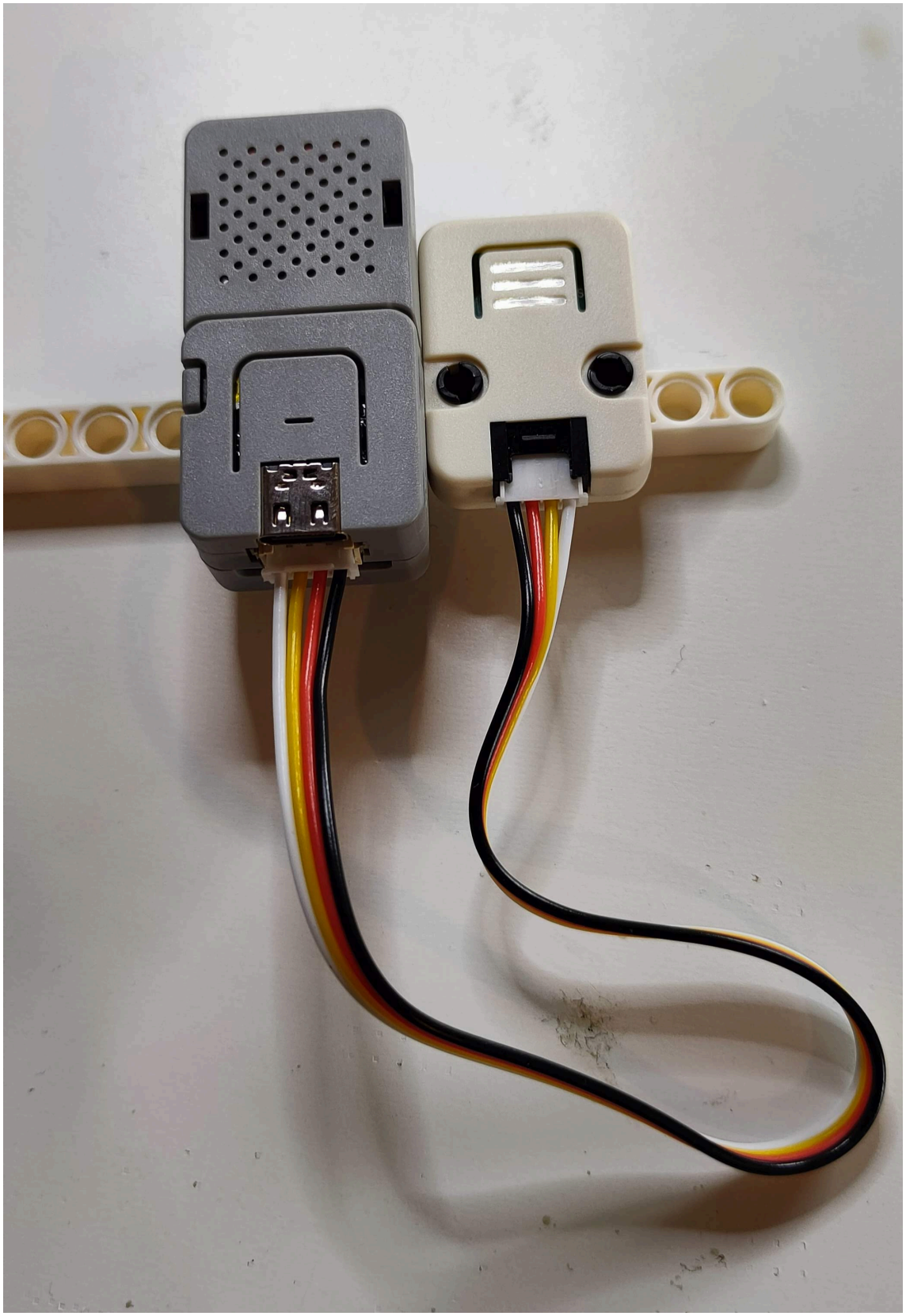
Soil Moisture: https://shop.m5stack.com/collections/m5-sensor/products/earth-sensor-unit

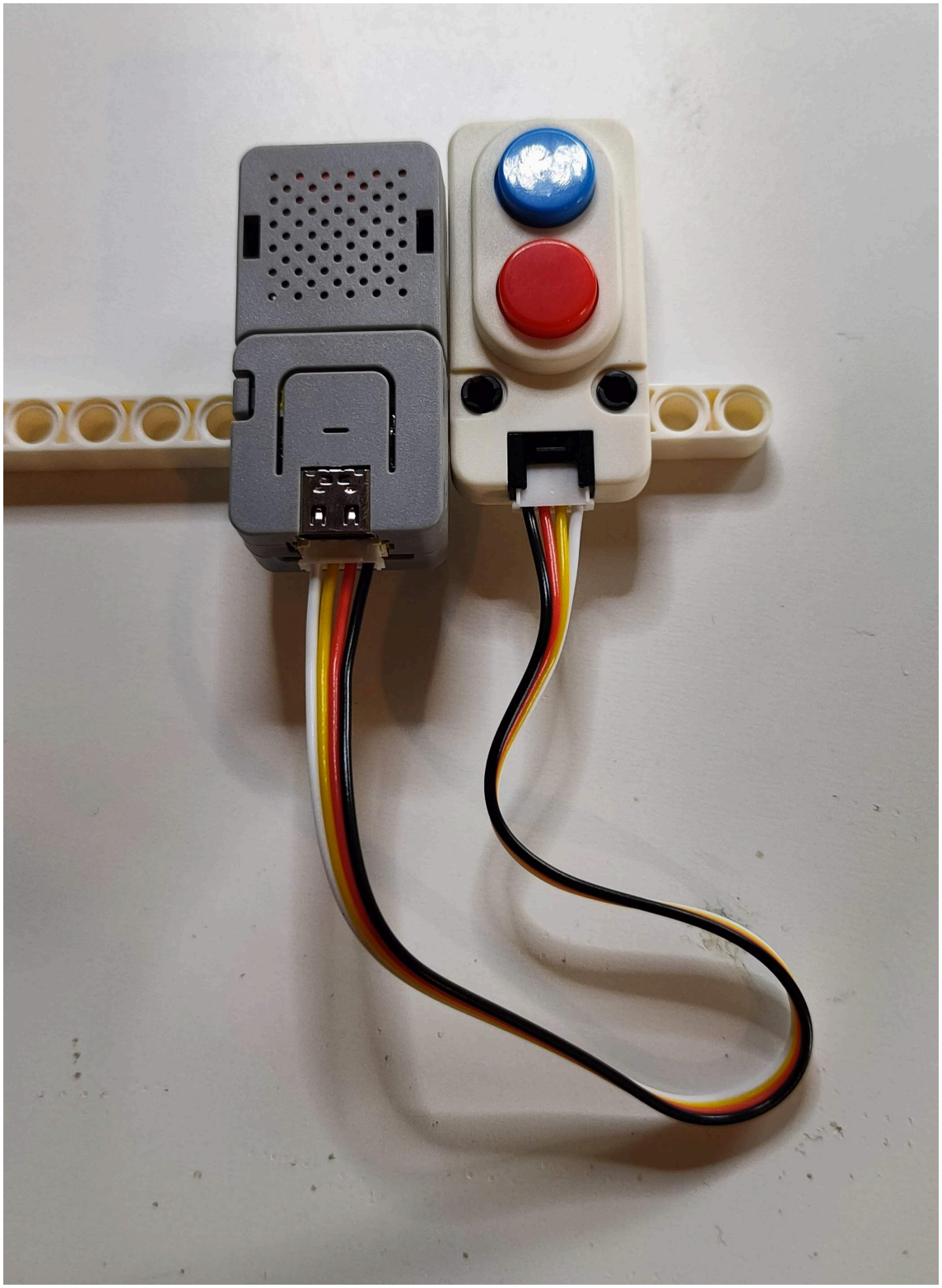
also Soil Moisture: https://wiki.seeedstudio.com/Grove-Capacitive_Moisture_Sensor-Corrosion-Resistant/

https://wiki.seeedstudio.com/Grove-LED_Button/ <https://wiki.seeedstudio.com/Grove-LED_Button/

https://shop.m5stack.com/collections/m5stack-new-arrival/products/fader-unit-with-b10k-potentiometer-sk6812







3. Eagle

3.1 🎧 Podcast

- From a Bavarian village to the Autodesk Cloud: The fascinating metamorphosis of the software legend EAGLE CAD

3.2 📺 Video

- EAGLE CAD: A Legend

3.3 End of life

On June 7, 2026, EAGLE will no longer be available.
https://www.autodesk.com/products/fusion-360/blog/future-of-autodesk-eagle-fusion-360-electronics/

3.4 Installer

Autodesk_EAGLE_9.6.2_English_Win_64bit.exe

3.5 Problems

Eagle has a problem under Windows 11 on certain HP and Lenovo laptops:
Eagle opens briefly and then immediately closes.
https://www.autodesk.com/support/technical/article/caas/sfdcarticles/sfdcarticles/Eagle-crashes-seconds-after-launching-splash-screen.html
https://www.youtube.com/watch?v=pyK8LCPOxkk
The solution is to download and replace two files, “libeay32.dll” and “ssleay32.dll”, from the package https://www.totalcommander.ch/win/tools/openssl-1.0.2u-x64_86-win64.zip)
Eagle will then run normally.
After that, Eagle will run as usual.

4. 📦 Installing and using Gitkraken

Before you begin, please read the section on GitHub.

... It is important that you have registered with your school email address on GitHub,

and you should also see the private repositories, e.g., this one: https://github.com/Meisterschulen-am-Ostbahnhof-Munchen/ blink_on_atom. Remove the space before “blink”.

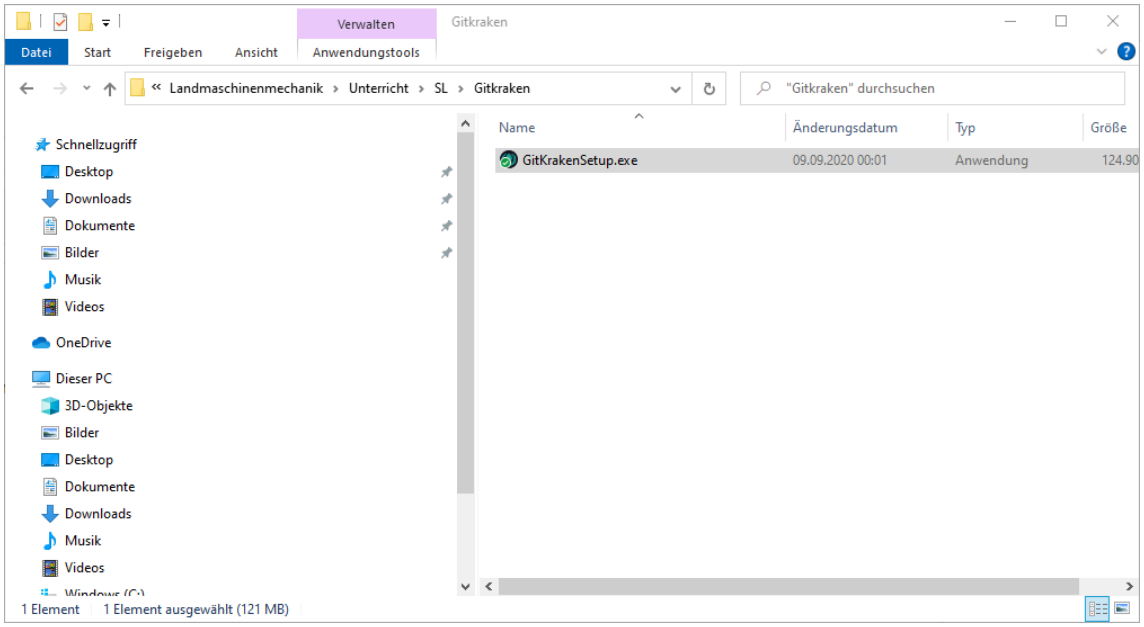
You can find GitKraken at M:\Landmaschinenmechanik\Unterricht\SL\GitKraken

Alternatively, you can also find it at <https://www.gitkraken.com/download/windows64, where you will also find numerous explanatory videos:

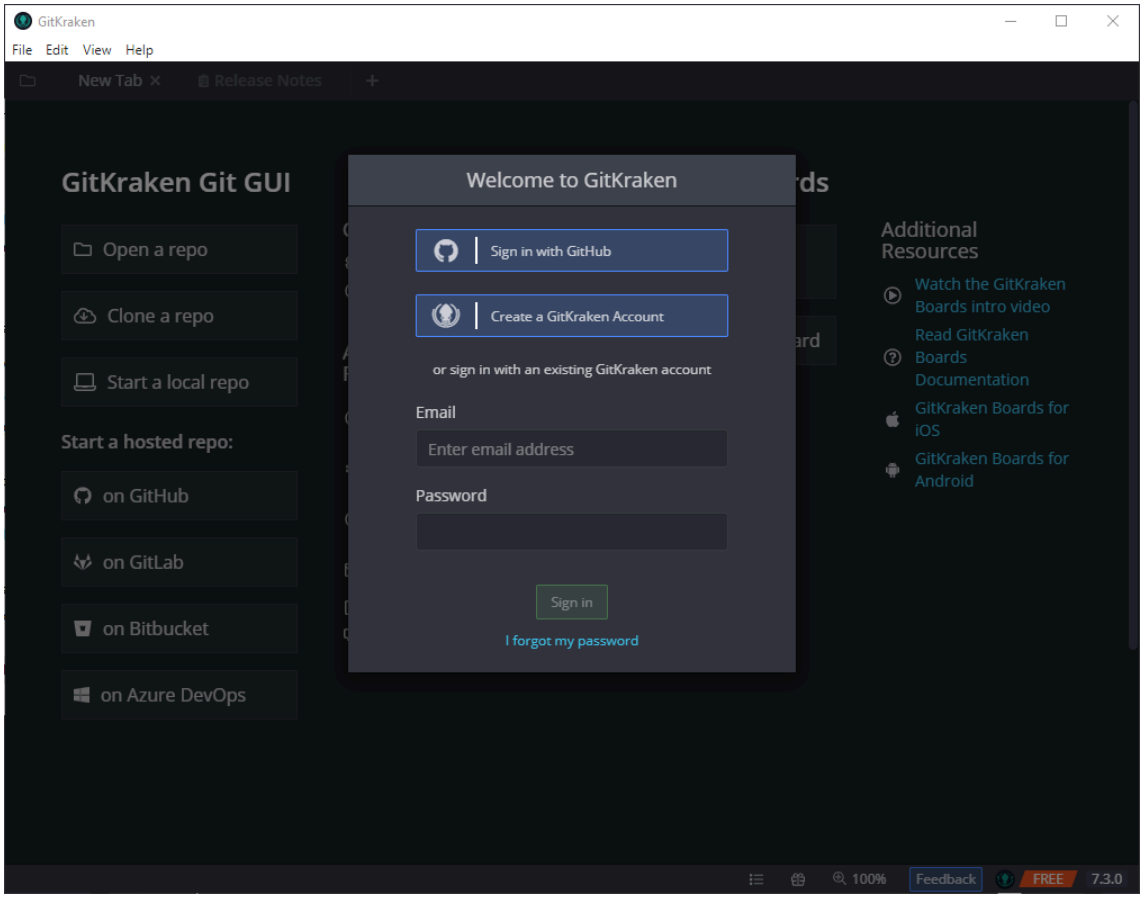
https://www.youtube.com/watch?v=ub9GfRziCtU&feature=emb_title

https://www.youtube.com/channel/UCp06FAzrFalo3txskS1gCfA

There, double-click on GitKrakenSetup.exe



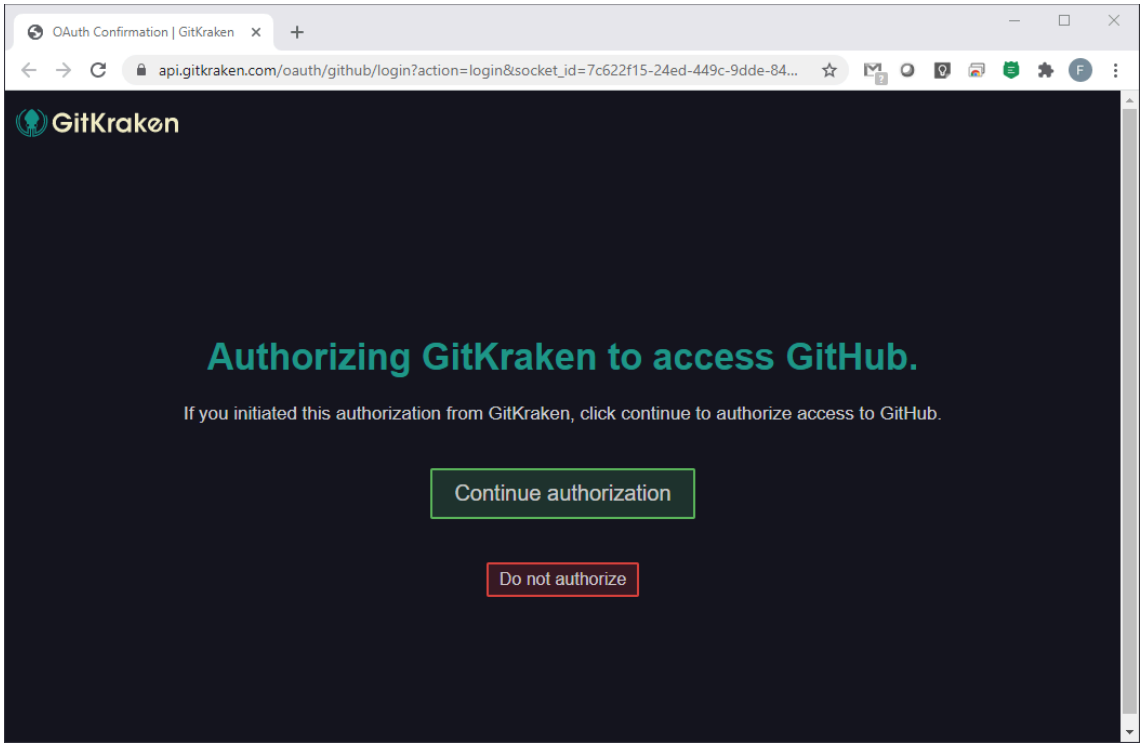
My start screen looks like this:



Now you have two options:

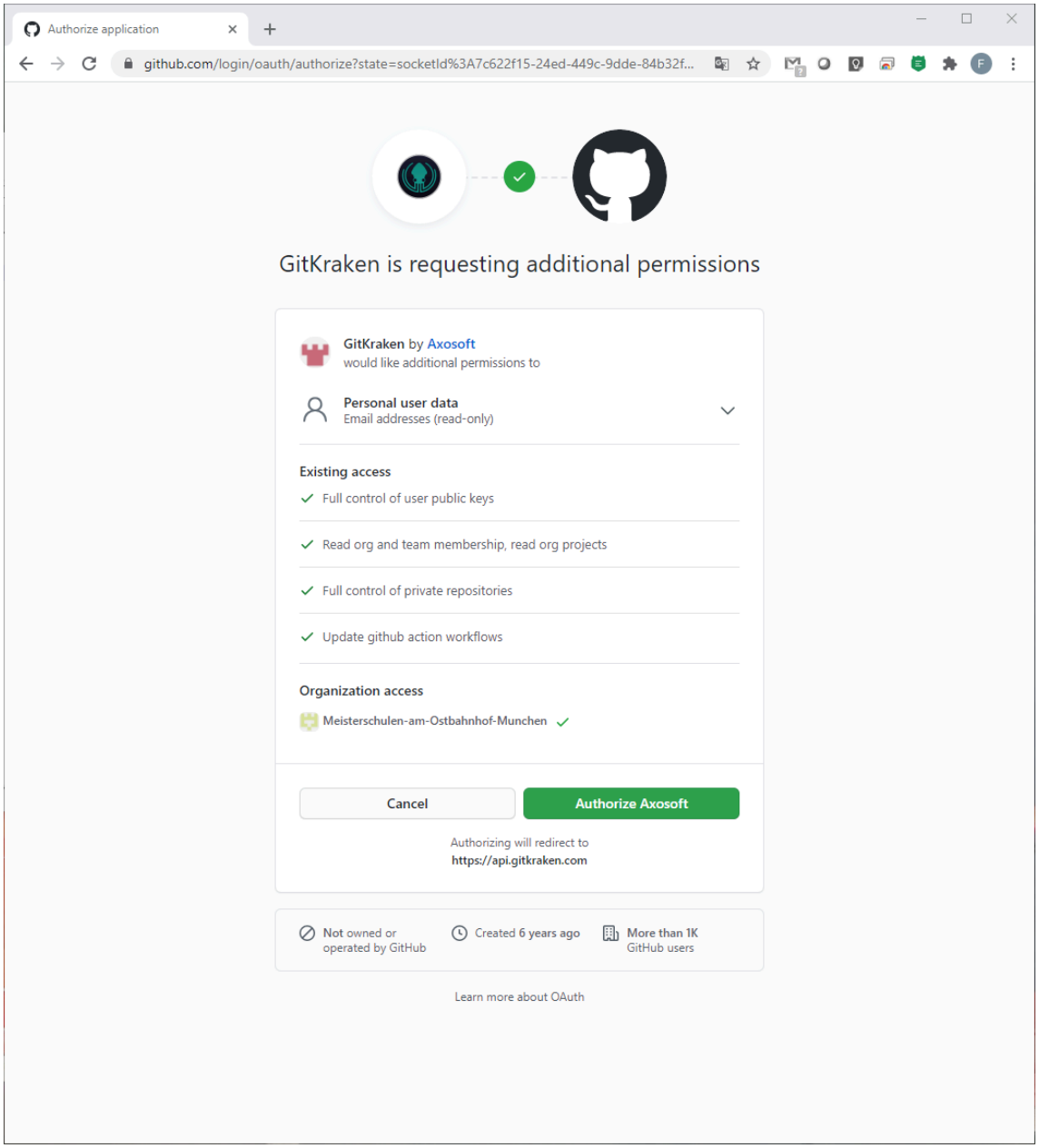
1. for For those who are impatient: “Sign in with GitHub,” which we’ll explain below. … 2. https://www.gitkraken.com/student-resources Here you can get a free GitKraken Pro, which normally costs $49 per year per person: https://www.gitkraken.com/pricing

So, we’ll only describe the first option here:
If you click “Sign in With Github” (you must leave the fields below blank >`_), then the browser will open:



Click “Continue authorization”

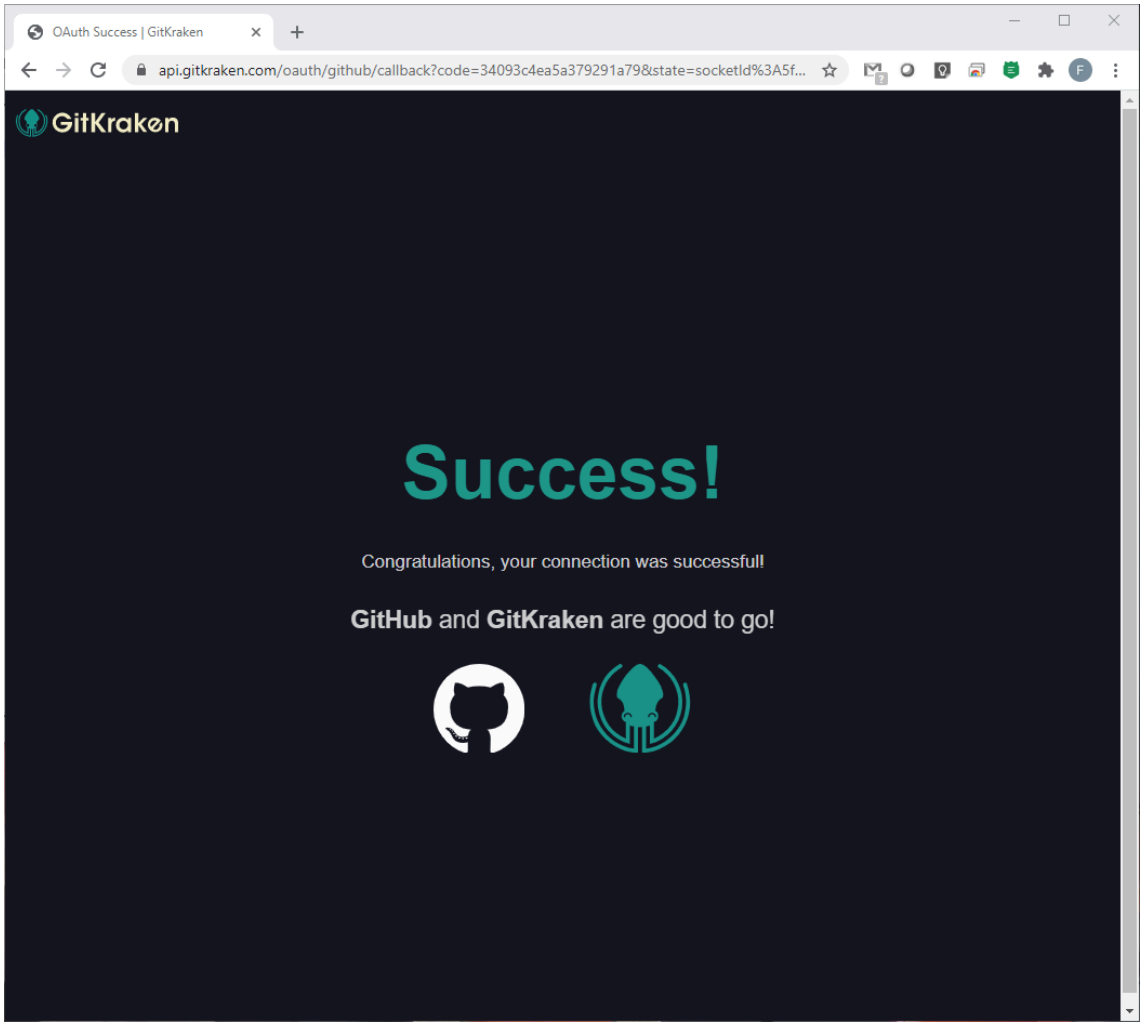
Now you should see something like this:



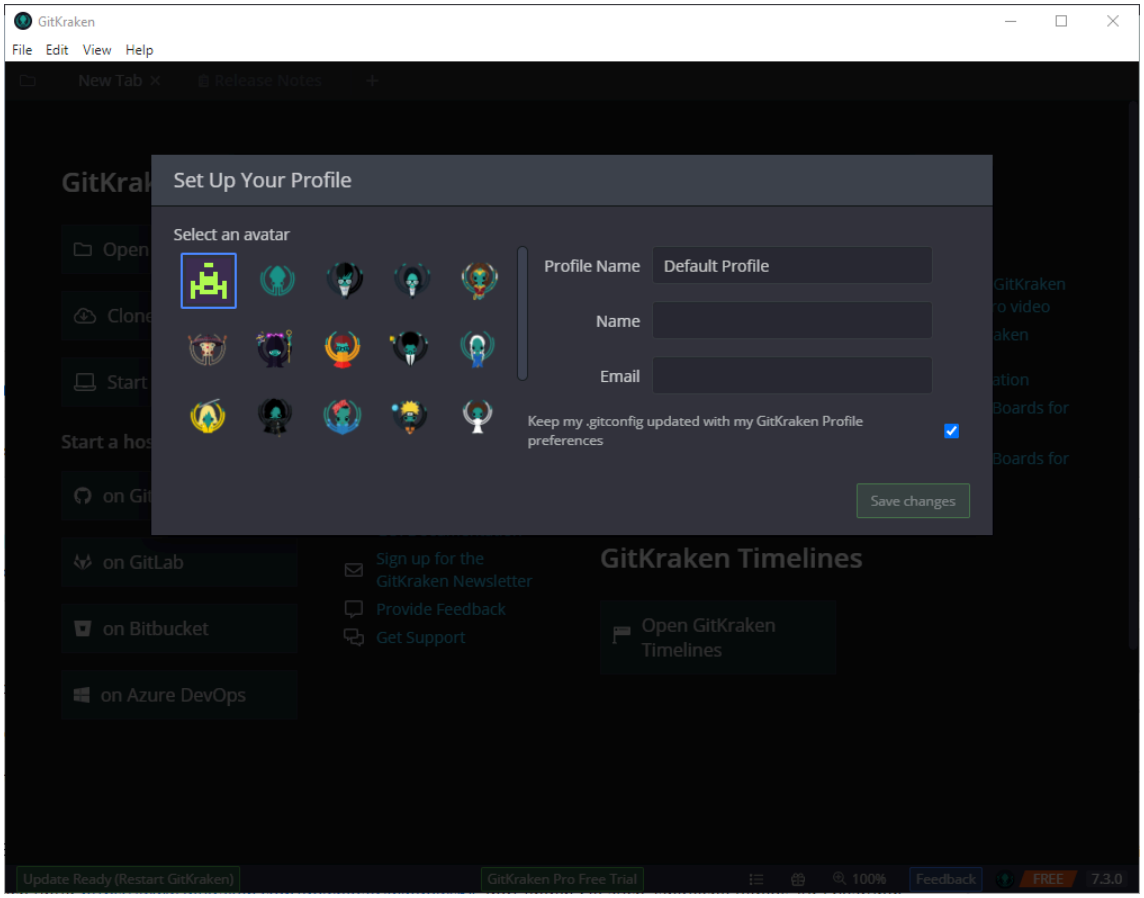
By the way: if you don't like your photo, you can freely customize it on Github.

Click on "Authorize Axosoft" and check the box next to Meisterschulen-am-Ostbahnhof-Munich if it isn't already there.

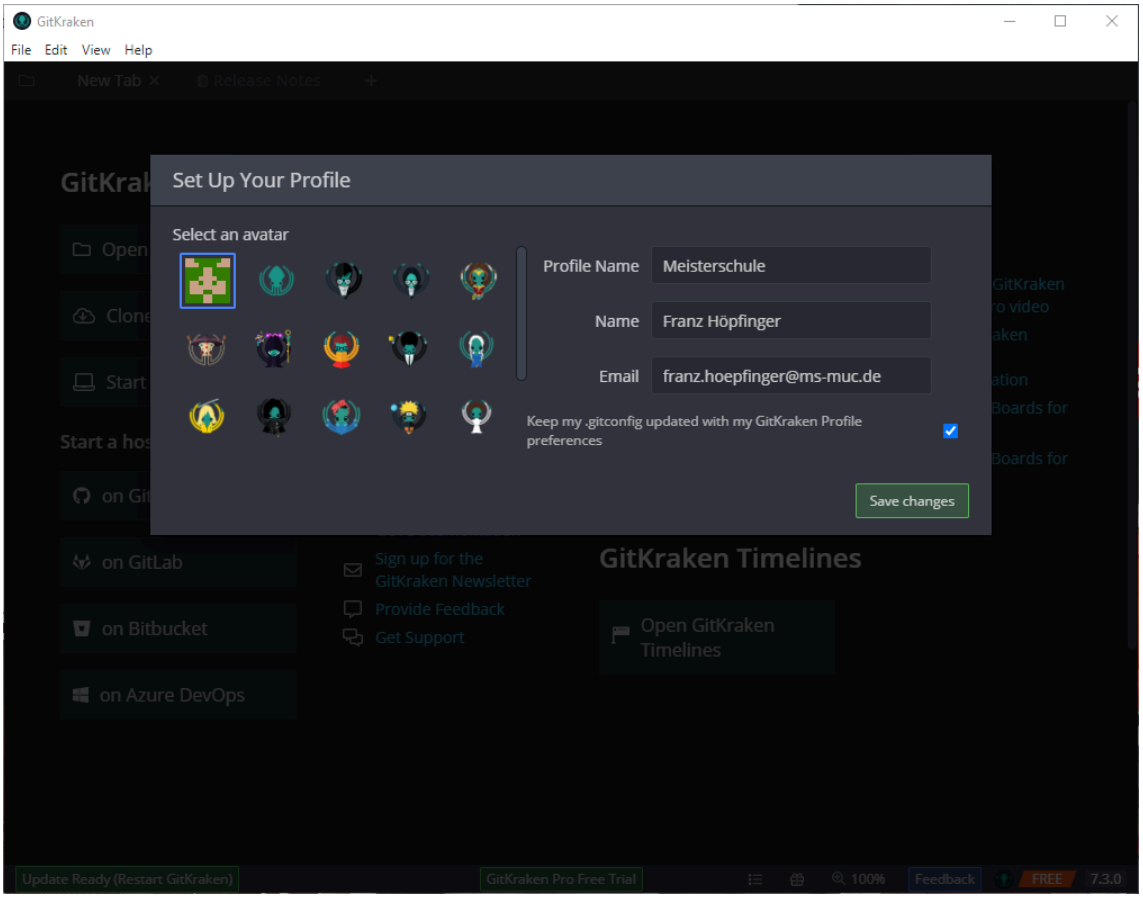
If this window appears, it It works:



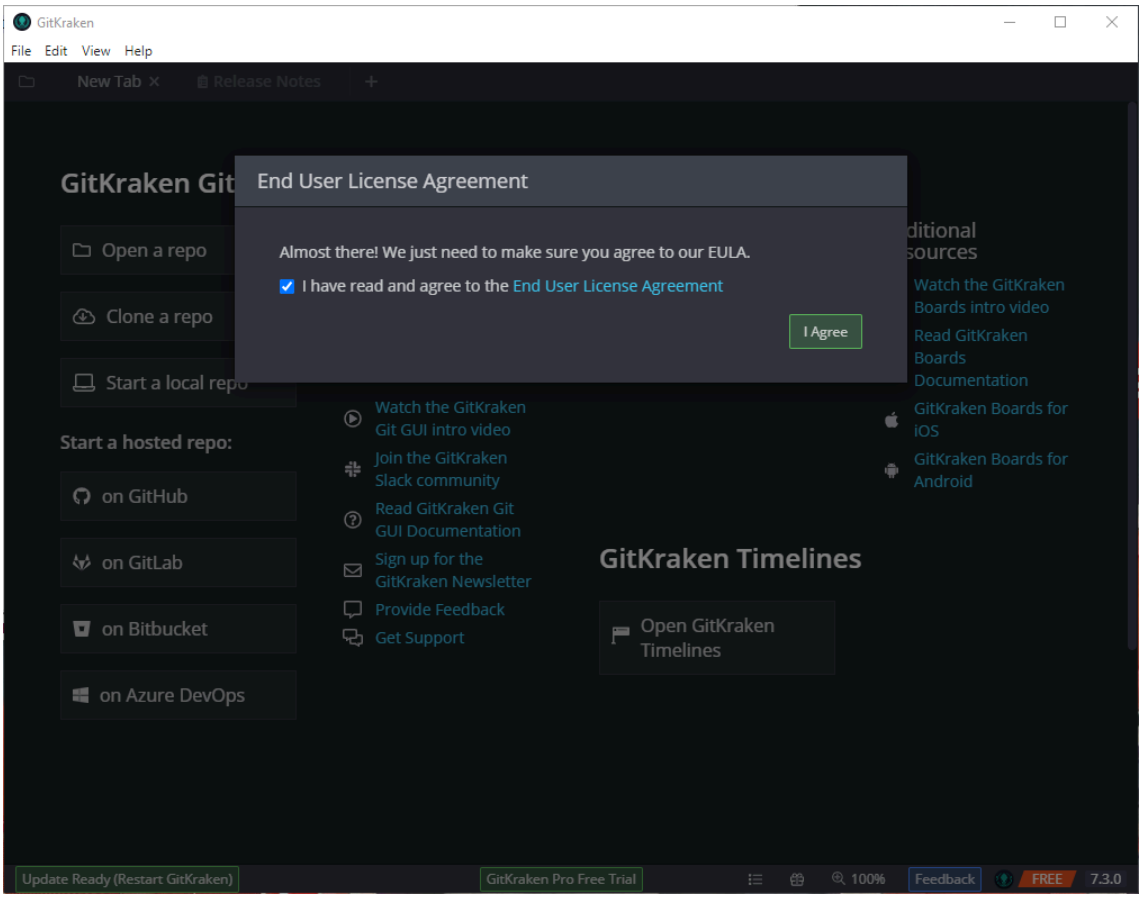
Back in Gitkraken, you should see this:



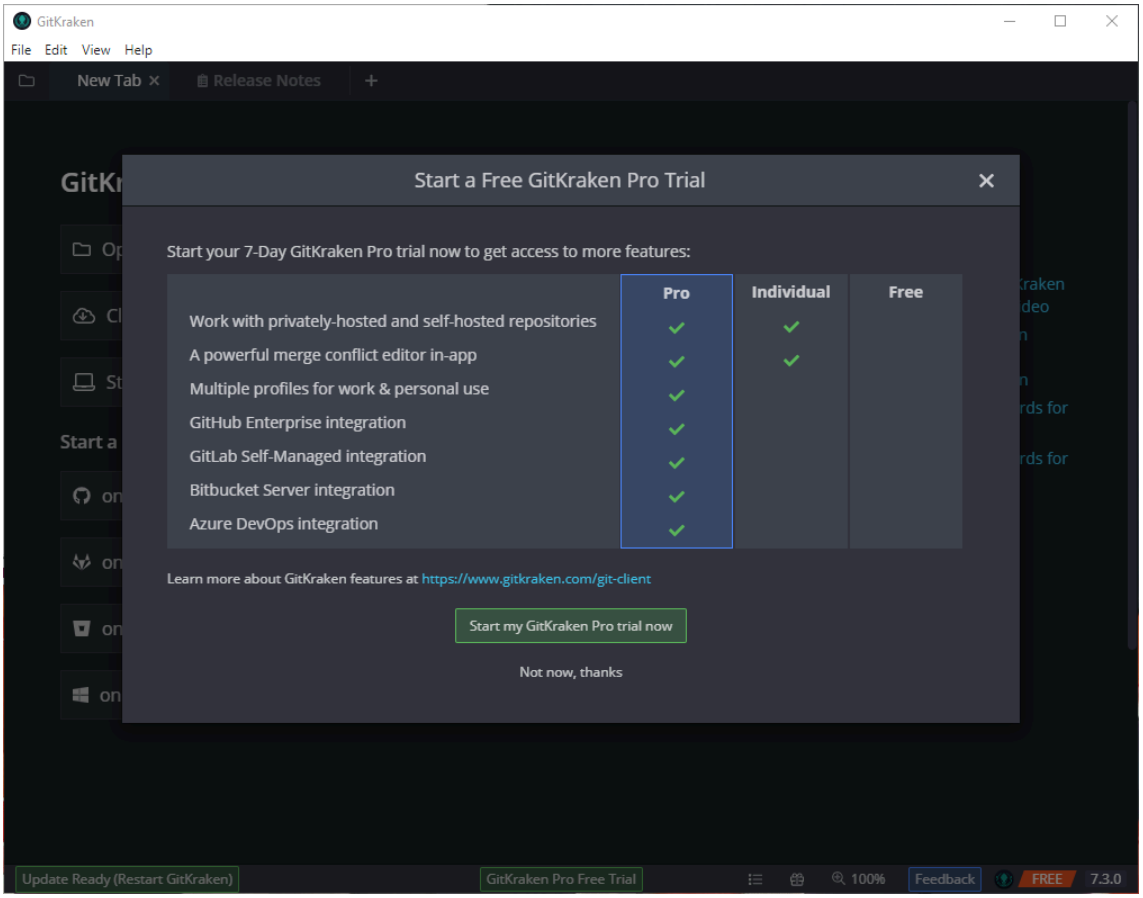
Enter your name, etc.,
like this, for example:



and then click on Save changes.

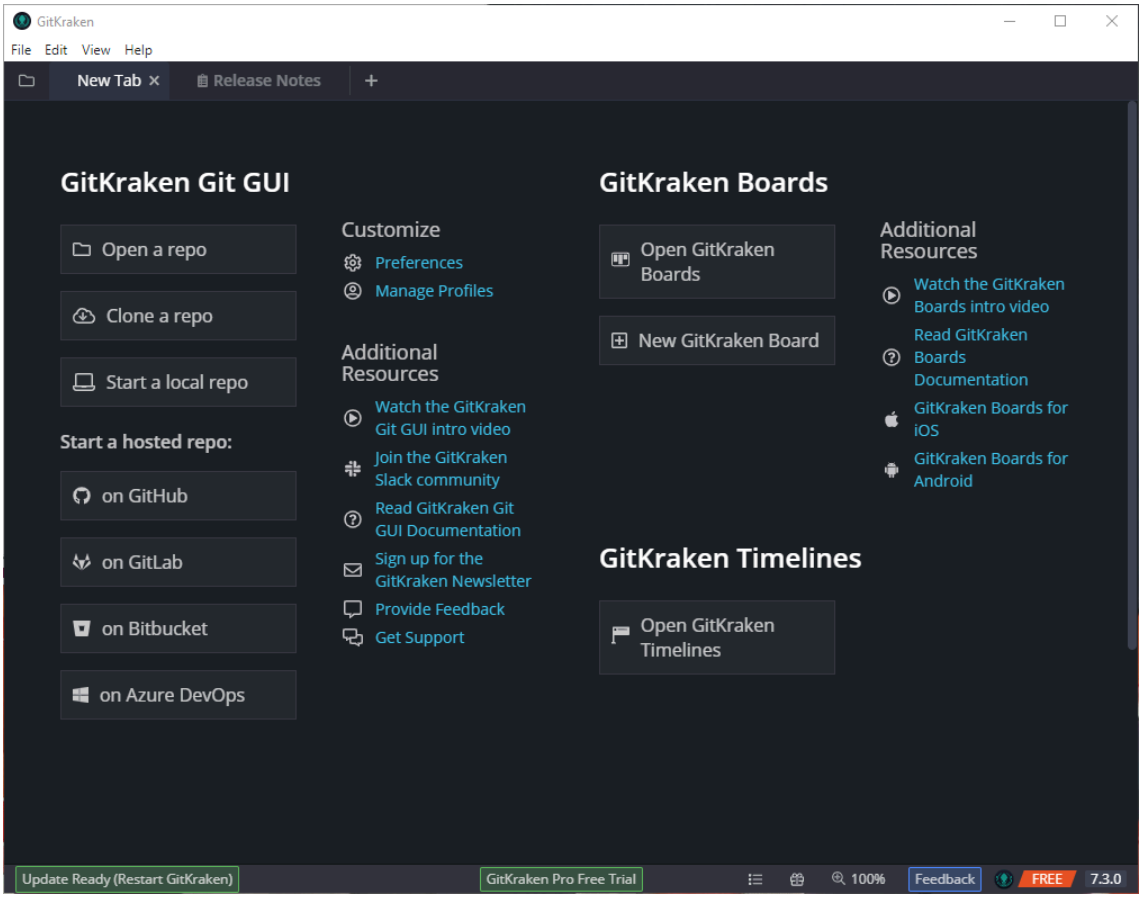


Check the box here and click “I Agree”.

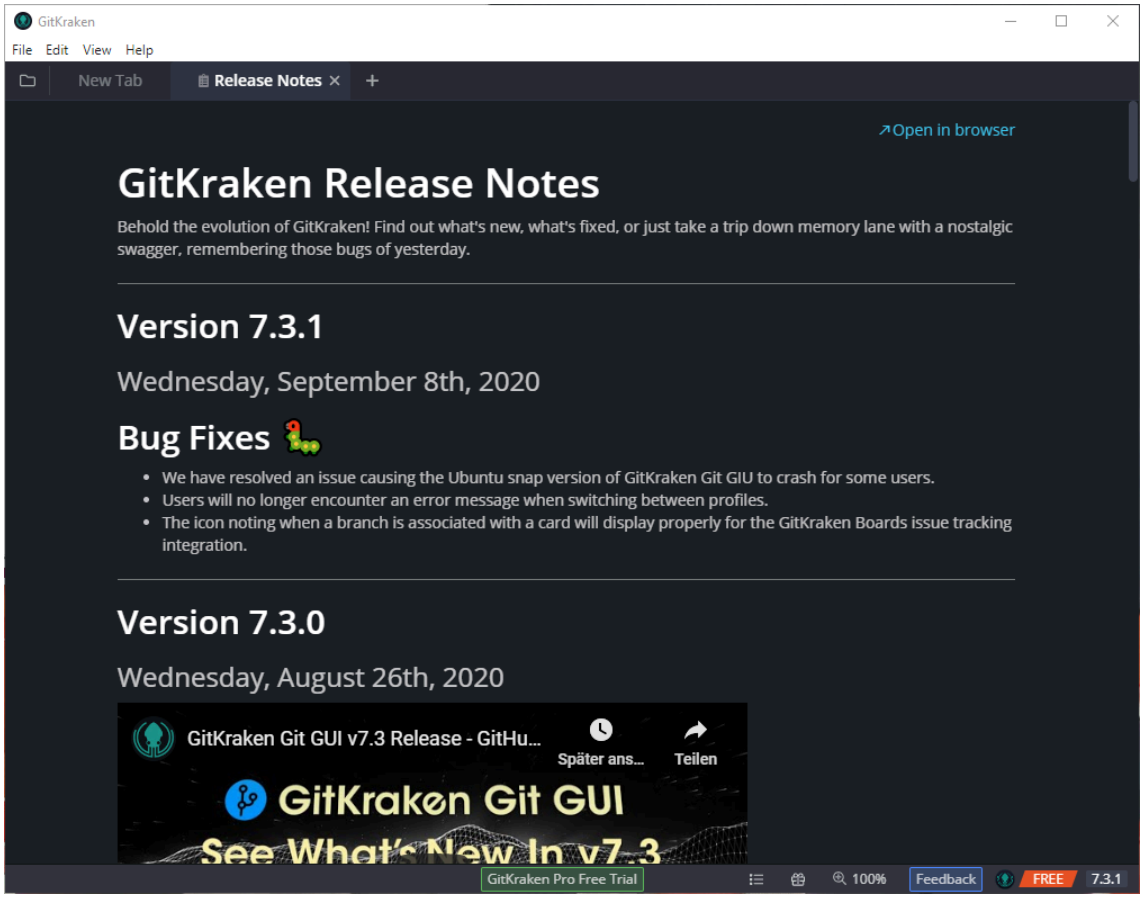


It doesn’t matter what you choose here; Gitkraken will automatically give you a trial of the Pro version. Therefore, we select “Start my Pro trial”.

(See above for a way for students to get a free Pro trial for more than 7 days >`_



Now click on Update Ready,
and then it should look like this. Appearance:



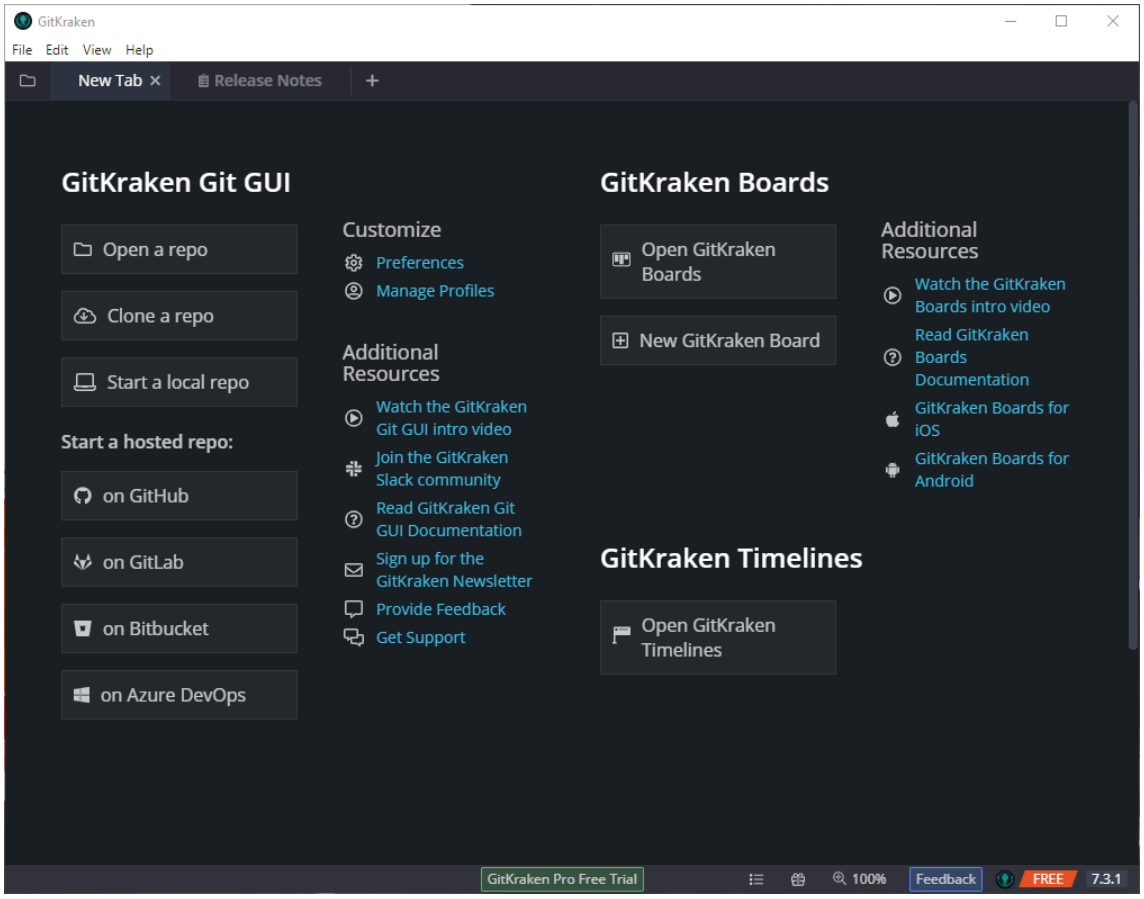
Next, we'll get two repositories.

First, we need a location.

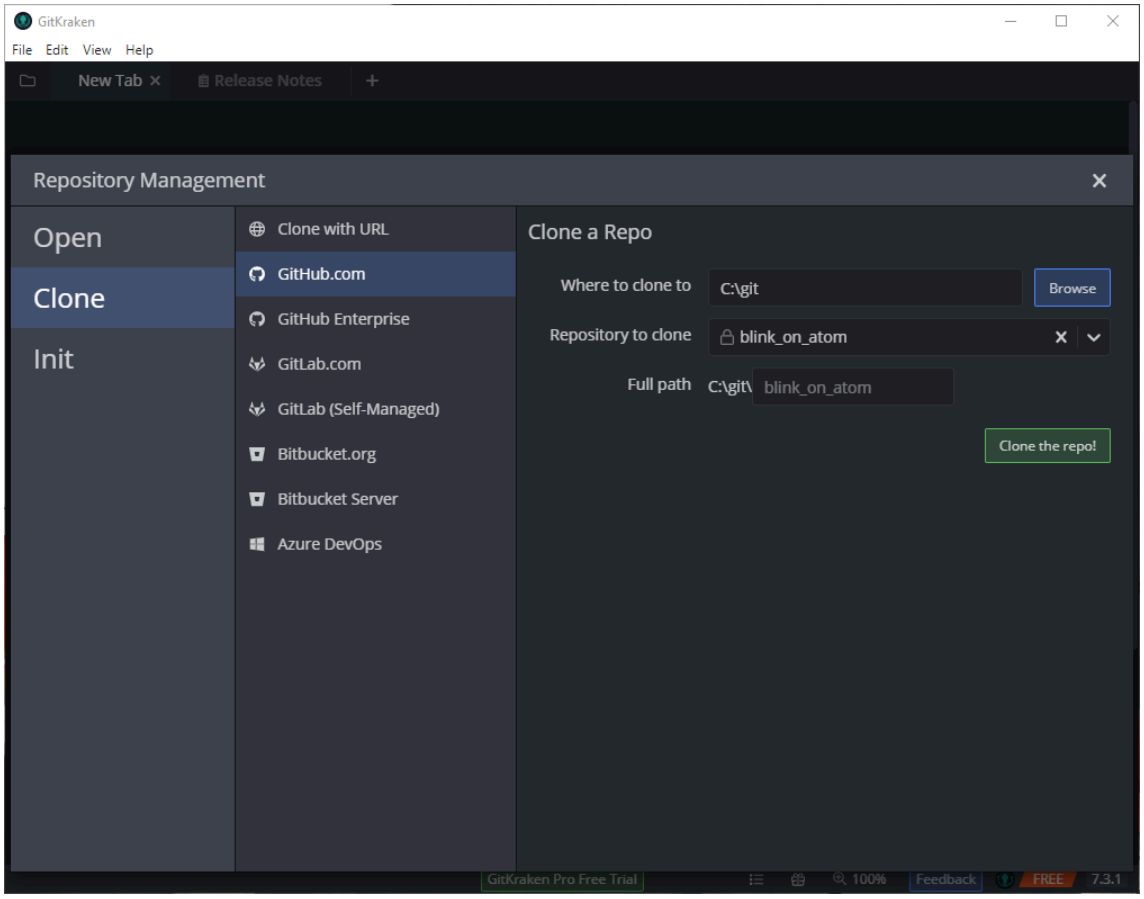
A good location is C:\git or D:\git if your laptop has two hard drives.

A network drive like E:\ is a bad idea.

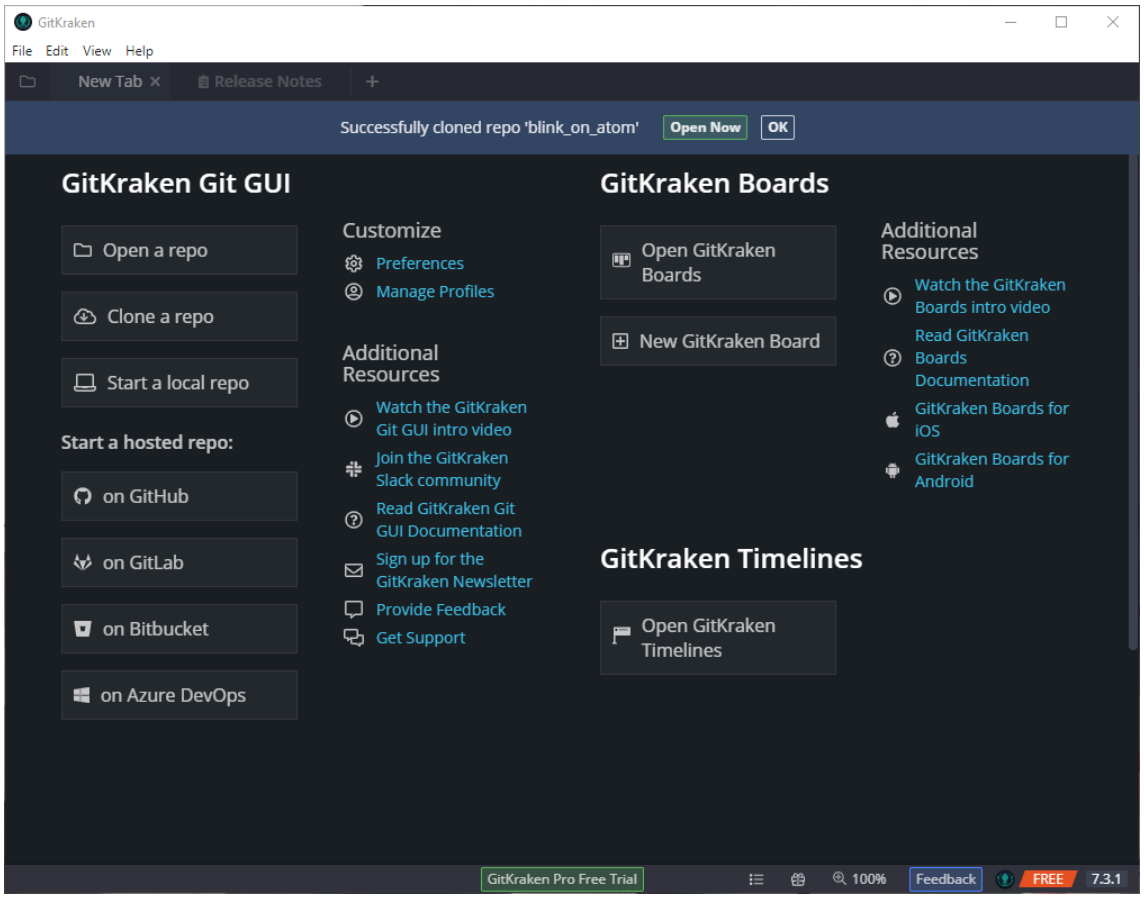
Click "new tab" and "clone a repo"



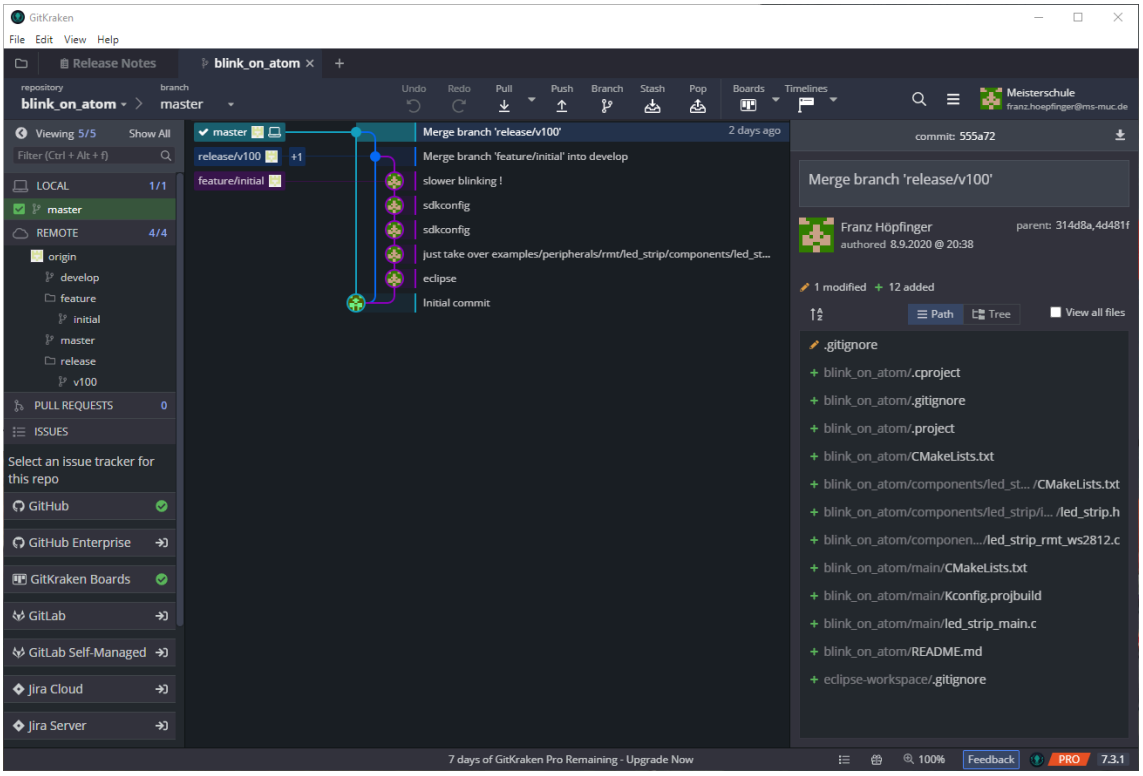
In the next window, select “GitHub.com” and your previously chosen location, and select the “blink_on_atom” repository.

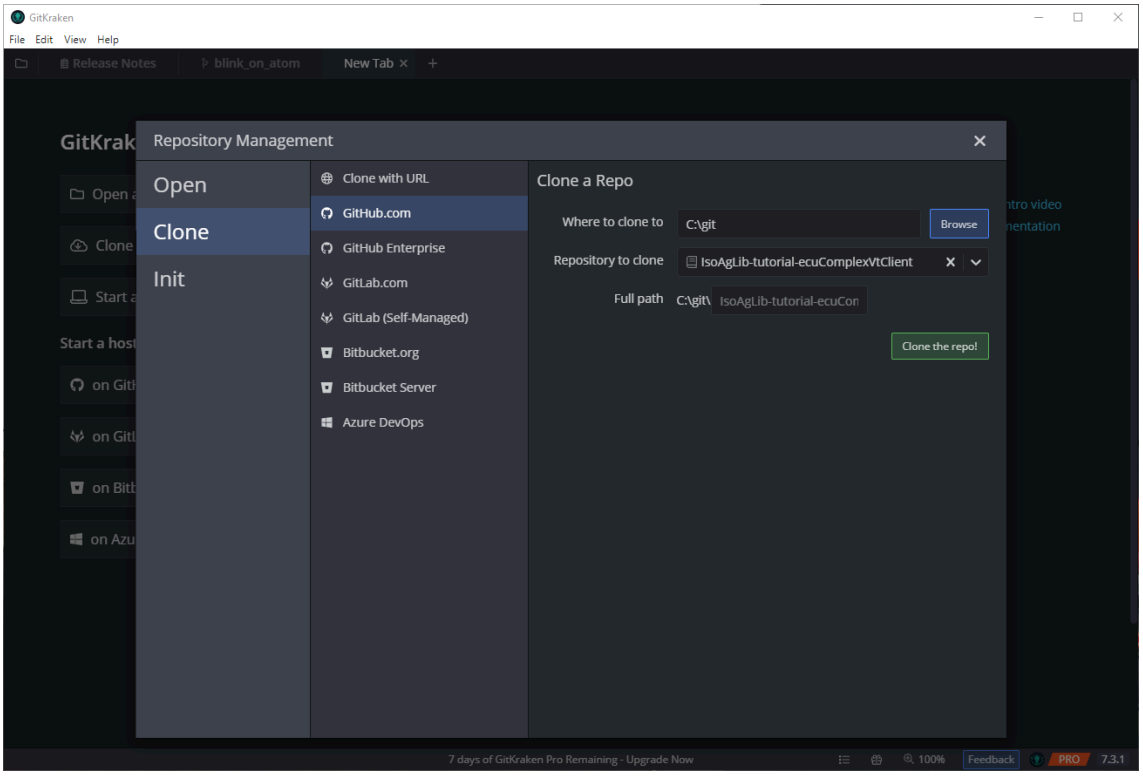


Click “Clone the repo!” and then “Open Now”

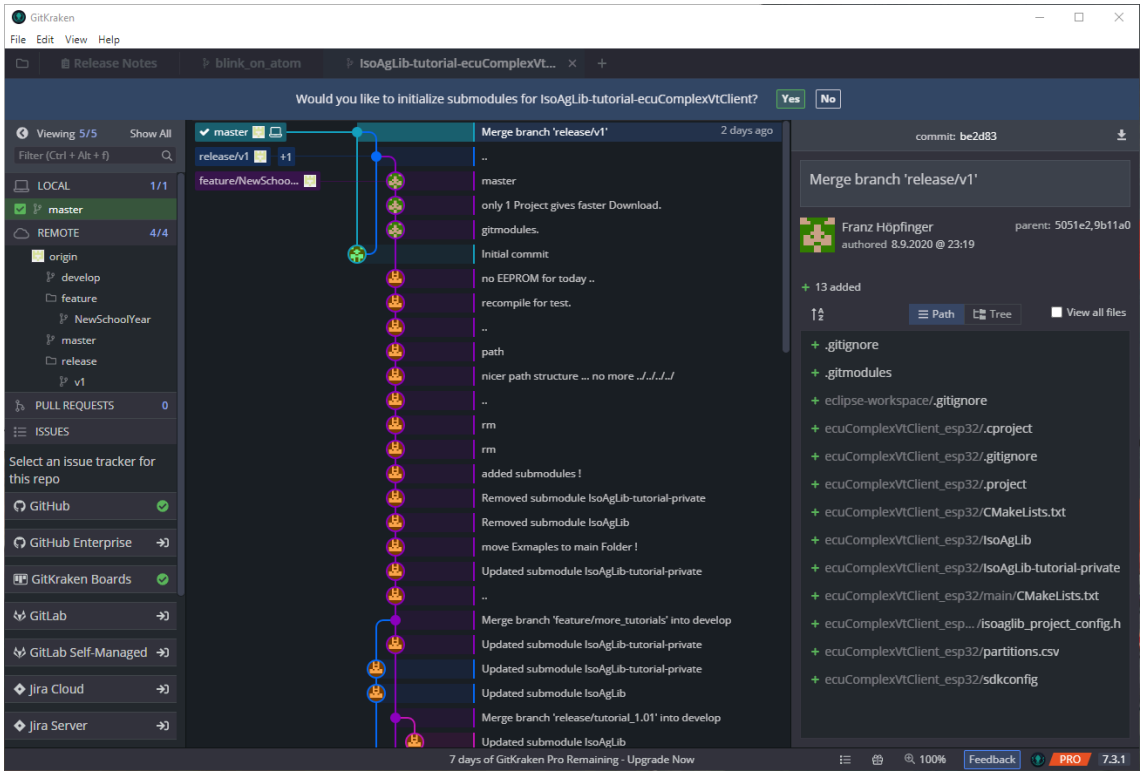


Now click the “+” at the top and repeat the process. Once:

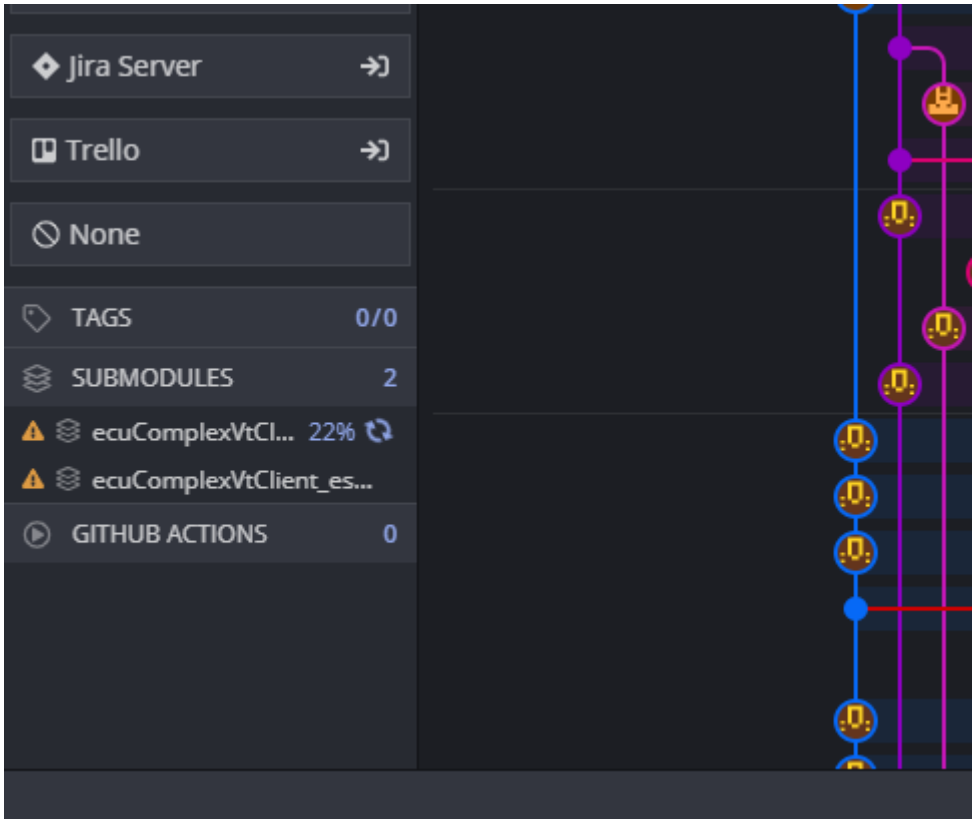




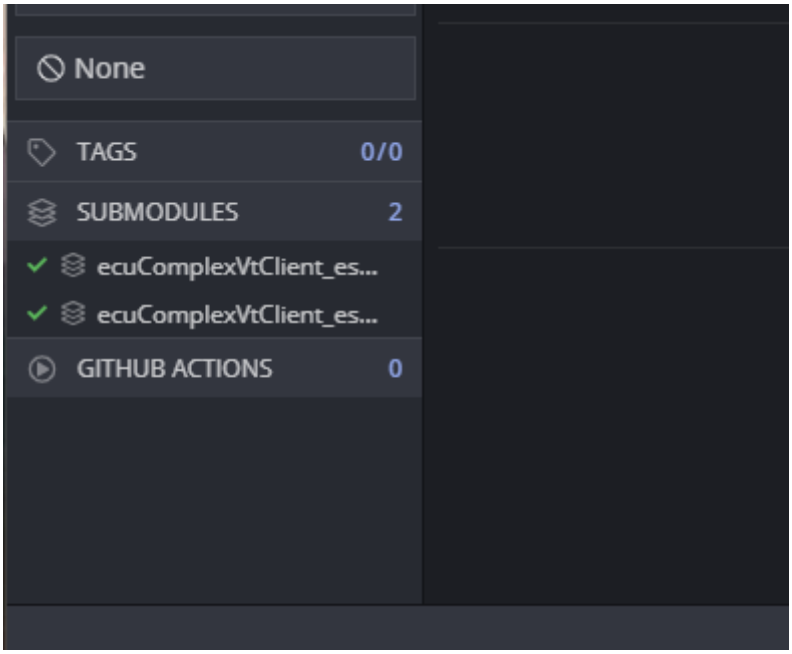
This time, select “ISOMAN”.
Click “Open Now” again
and confirm with YES.



The bottom left should then look like this:



After a while, it should look like this:



So, that's it for Gitkraken today.

We'll spend many more hours with it, but for now, we don't need Gitkraken anymore.

You can close it.

5. 🔌 Hardware

5.1 🎧 Podcast

- Automation 4.0: Why Software Is Overtaking Hardware and What That Means for Your Skills
- PLC: The Invisible Brain of Industry – From Robust Hardware to IT/OT Convergence
- Universal Automation: How to Decouple Software and Hardware for the Future of Industry

6. Homepage

6.1 🎧 Podcast

- [logiBUS®: Revolutionizing Agricultural IT – How ISOBUS Becomes a Smart Home for Farmers

7. DIN Rail Moped Inputs

7.1 Differentiation of Limit Switches for Pneumatic Cylinders

https://www.ifm.com/at/de/shared/produkte/zylindersensoren/technologie

7.1.1 Reed Switch

A reed switch is an electrical switching element consisting of two ferromagnetic contacts inside a hermetically sealed glass tube. These contacts are held apart by a thin, flexible glass rod coated with a special metal alloy. The main characteristics of a reed switch are:

1. Magnetic Activation: A reed switch is activated by a magnetic field. When the field is strong enough, the contacts move and close an electrical circuit.
2. Small Size: Reed switches are very small and can be installed in confined spaces.
3. High Switching Speed: Due to their small size, reed switches can switch very quickly, making them ideal for applications such as circuits and sensors.
4. High Reliability: Because reed switches are hermetically sealed, they are protected against moisture, dust, and other environmental influences, which increases their lifespan and reliability.
5. Low Power Consumption: Reed switches require very little power to switch, making them ideal for battery-powered applications.
6. High Sensitivity: Reed switches can respond to even very weak magnetic fields, making them ideal for applications such as magnetic field sensors and ammeters.

https://de.wikipedia.org/wiki/Reedschalter
Reed Switch Characteristics:

- Normally Open: always 2-wire technology
- Normally Closed: always 2-wire technology
- Changeover (rare): 3-wire technology
- Only the maximum voltage is specified in the datasheet
- Current Direction: irrelevant
- Ground Terminal: not present
- Usually no LED (except for special designs)
- Not short-circuit protected!!!

https://www.ifm.com/de/de/product/MR0100

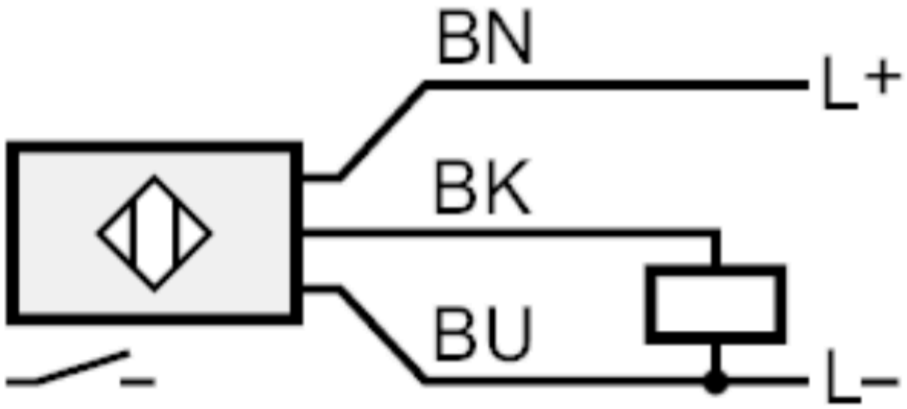
7.1.2 Semiconductor Magnetic Sensors

https://de.wikipedia.org/wiki/Magnetoresistiver_Effekt
Always Specified: e.g., 5…36 V DC

- Minimum and maximum voltage specified!!!
- Normally open (NO) contact: usually 3-wire technology
- Normally closed (NC) contact: usually 3-wire technology
- Changeover contact: 4-wire
- Current direction: alternating, see PNP/NPN, L+ L-
- Ground terminal: not present
- Usually no LED (except for special designs)
- Modern designs are short-circuit protected, cannot burn out

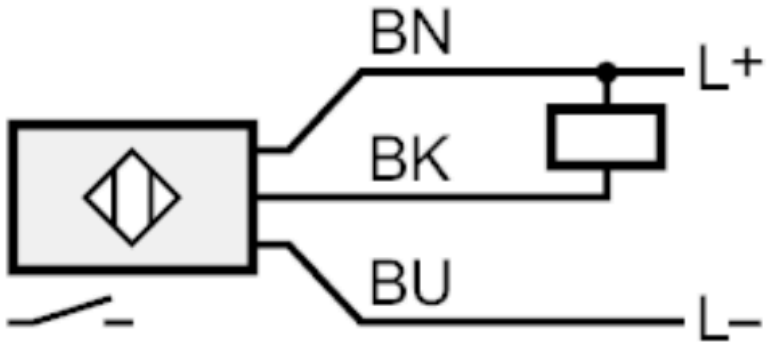
Example: https://www.ifm.com/de/de/product/MK5214 (short-circuit proof; overload proof)

7.1.2.1 PNP Sensor



Source: https://www.ifm.com/de/de/product/MK5140

7.1.2.2 NPN Sensor



Source: https://www.ifm.com/de/de/product/MK5309

7.1.2.3 AMR Cell

An AMR cell (Anisotropic Magnetoresistive Cell) is a type of An AMR cell is a magnetoresistive cell used in the semiconductor and magnet industries. It consists of a thin layer system of ferromagnetic and non-magnetic layers deposited on a silicon substrate.

The key characteristics of an AMR cell are:

1. Magnetic Sensitivity: The AMR cell is highly sensitive to magnetic fields and can detect even minute magnetic changes.

2. Low Power Consumption: The AMR cell requires very little power to operate.
3. High Accuracy: The AMR cell can perform highly precise measurements and is therefore ideal for applications requiring high accuracy, such as magnetic field sensing.
4. High Speed: The AMR cell can respond very quickly to changes in the magnetic field and is therefore ideal for applications requiring rapid measurements, such as data processing.
5. Low Cost: The AMR cell is relatively inexpensive and easy to manufacture, making it suitable for a wide range of applications.

Overall, the AMR cell is a very versatile component that can be used in many different applications, from magnetic field sensing to data processing and many other areas.

7.1.2.4 GMR Cell

A GMR cell (Giant Magnetoresistive Cell) is a type of magnetoresistive cell used in the semiconductor and magnet industries. The GMR cell consists of several thin layers of ferromagnetic and non-magnetic materials deposited in a layered structure on a substrate.

The most important characteristics of a GMR cell are:

1. High sensitivity: The GMR cell is very sensitive to magnetic fields and can detect very small changes.
2. High accuracy: The GMR cell can perform very precise measurements and is therefore ideal for applications where high accuracy is required, such as magnetic field sensing.
3. Low power consumption: The GMR cell requires very little power to operate.
4. High Speed: The GMR cell can react very quickly to changes in the magnetic field and is therefore ideal for applications requiring fast measurements, such as in data processing.
5. High Temperature Stability: The GMR cell is very temperature stable and can be used even at high temperatures.
6. Low Cost: The GMR cell is relatively inexpensive and easy to manufacture, making it suitable for a wide variety of applications.

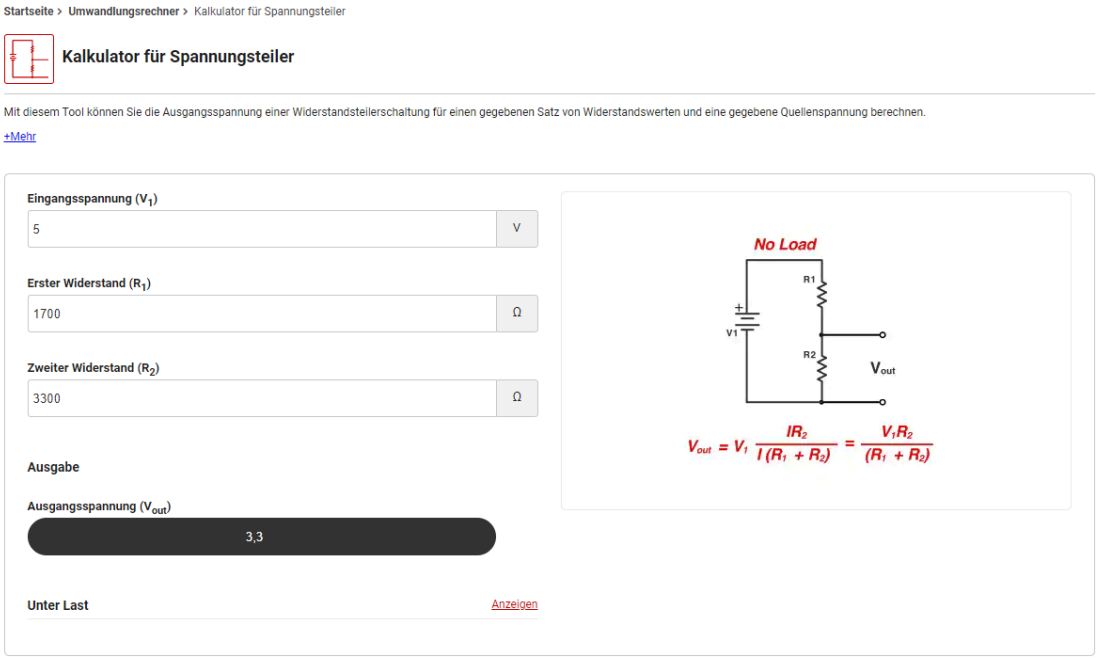
Overall, the GMR cell is a very versatile component that can be used in many different applications, from magnetic field sensing to data processing and many other areas.

https://www.ifm.com/de/de/product/MK5117

7.2 Connecting the Limit Switch to the Input



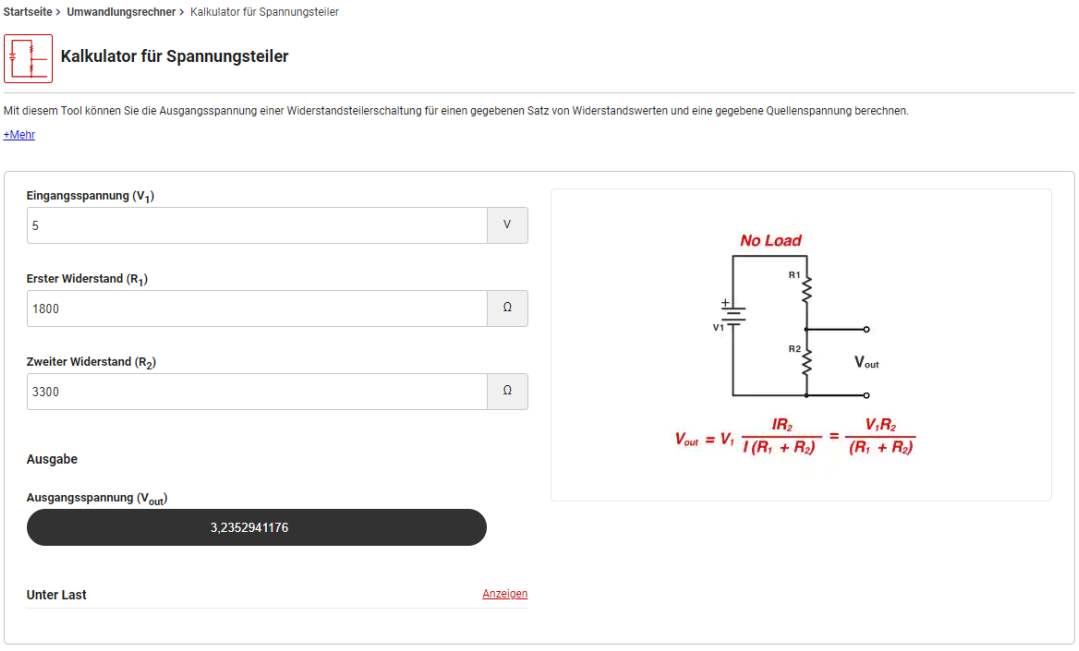
M5 Stack:
Input Voltage at the Pin:
3.3V
Voltage from the Sensor:
5V
Voltage Divider:



https://www.digikey.de/de/resources/conversion-calculators/conversion-calculator-voltage-divider
In Practice:

1. Resistor 1800 Ohm
2. Resistor 3300 Ohm

Because:
https://www.electronicsplanet.ch/Widerstand/Widerstandsreihe-E24.htm
]



7.3 Grove System

https://www.seeedstudio.com/category/Grove-c-1003.html
https://exp-tech.de/search?type=product&q=grove
https://www.distrelec.de/search?q=grove

7.4 Series Connection of Limit Switches

https://www.baumer.com/de/de/service-support/inbetriebnahme-montage/inbetriebnahme-und-montage-von-induktiven-sensoren/a/Know-how_Mounting_Inductive-sensors

7.5 Button Boards

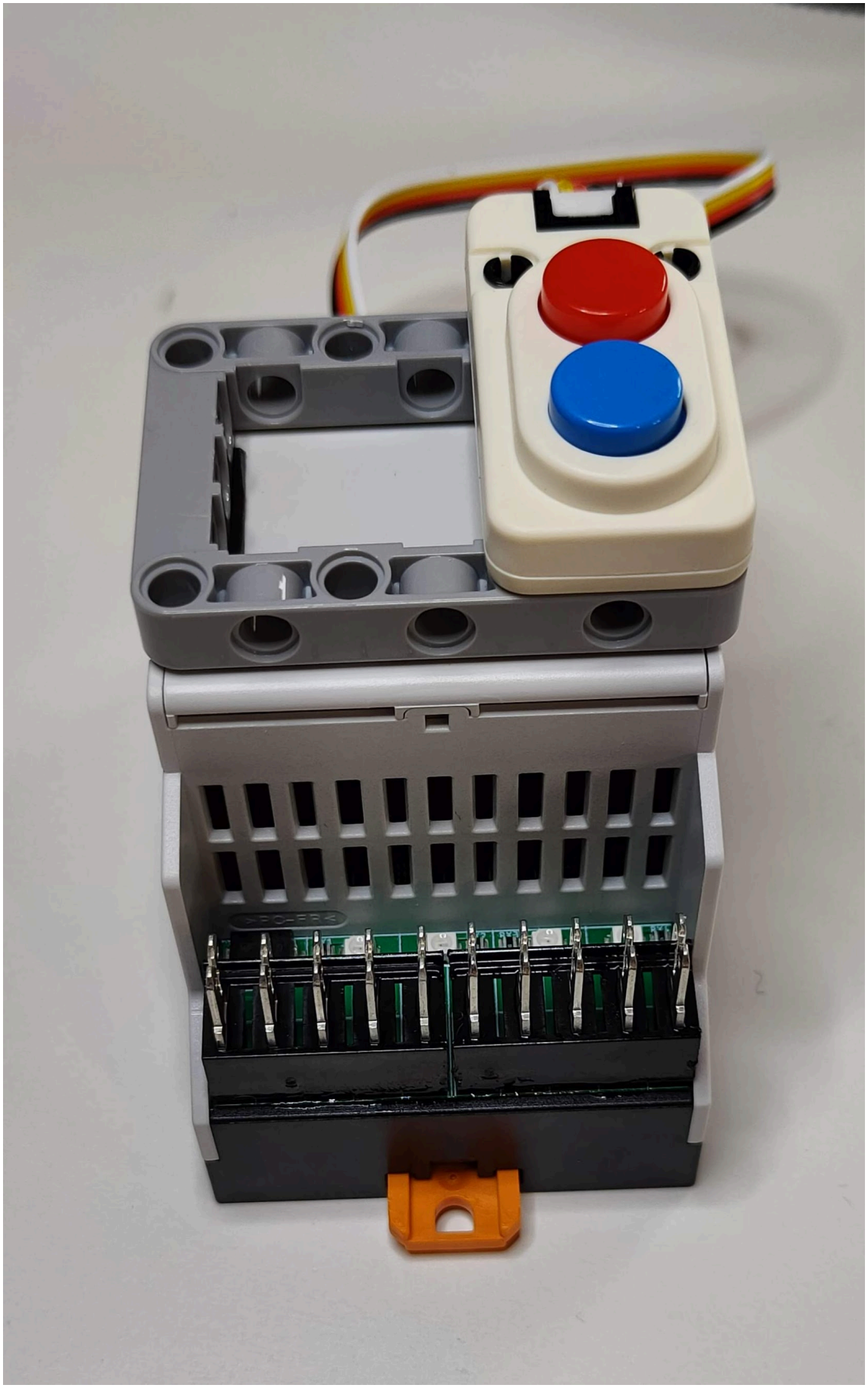
https://github.com/Meisterschulen-am-Ostbahnhof-Munchen/ButtonBoards
Multiplexers

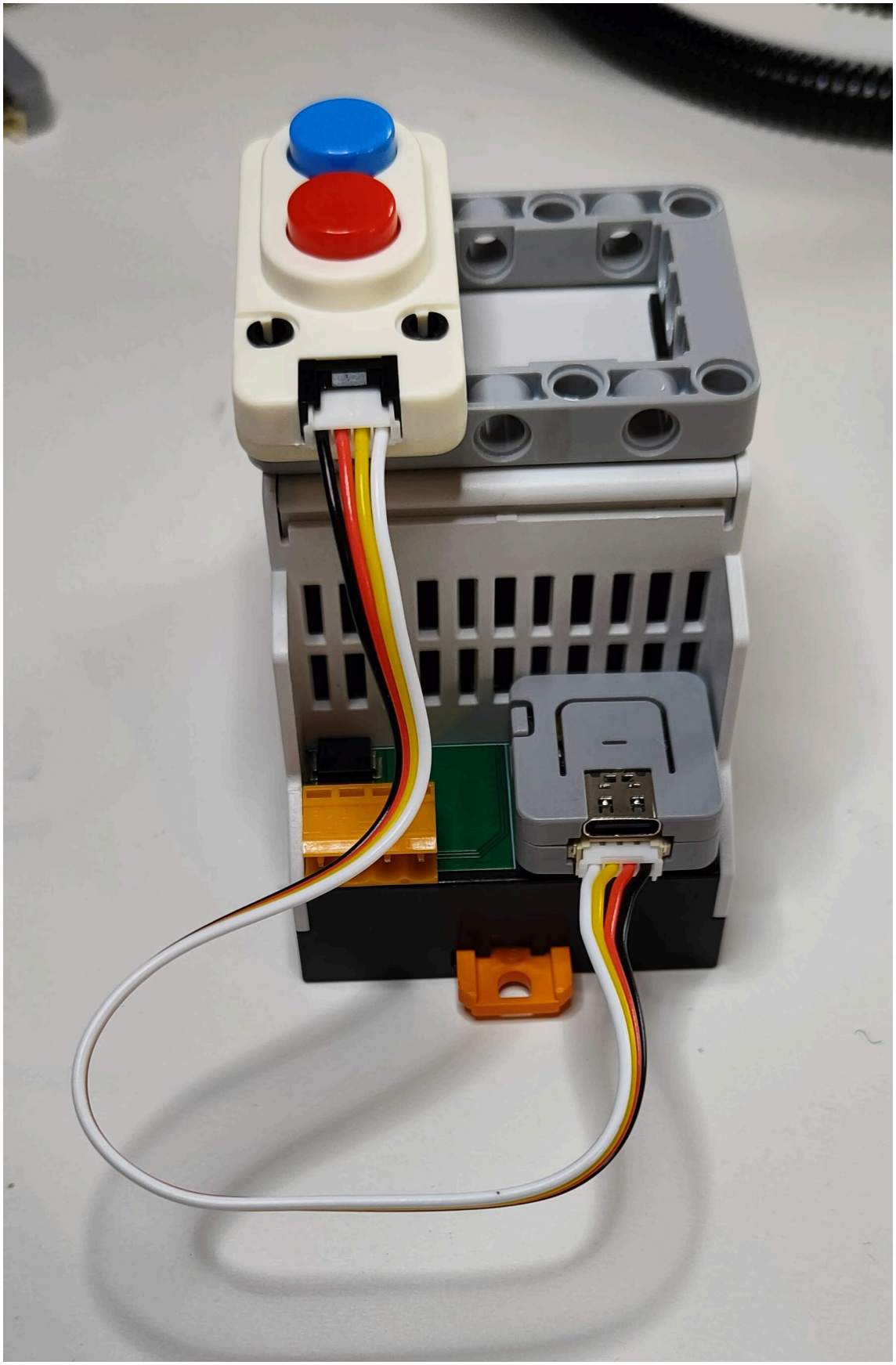
7.5.1 🌐 Related Topic Subpages on ms-muc-docs.de

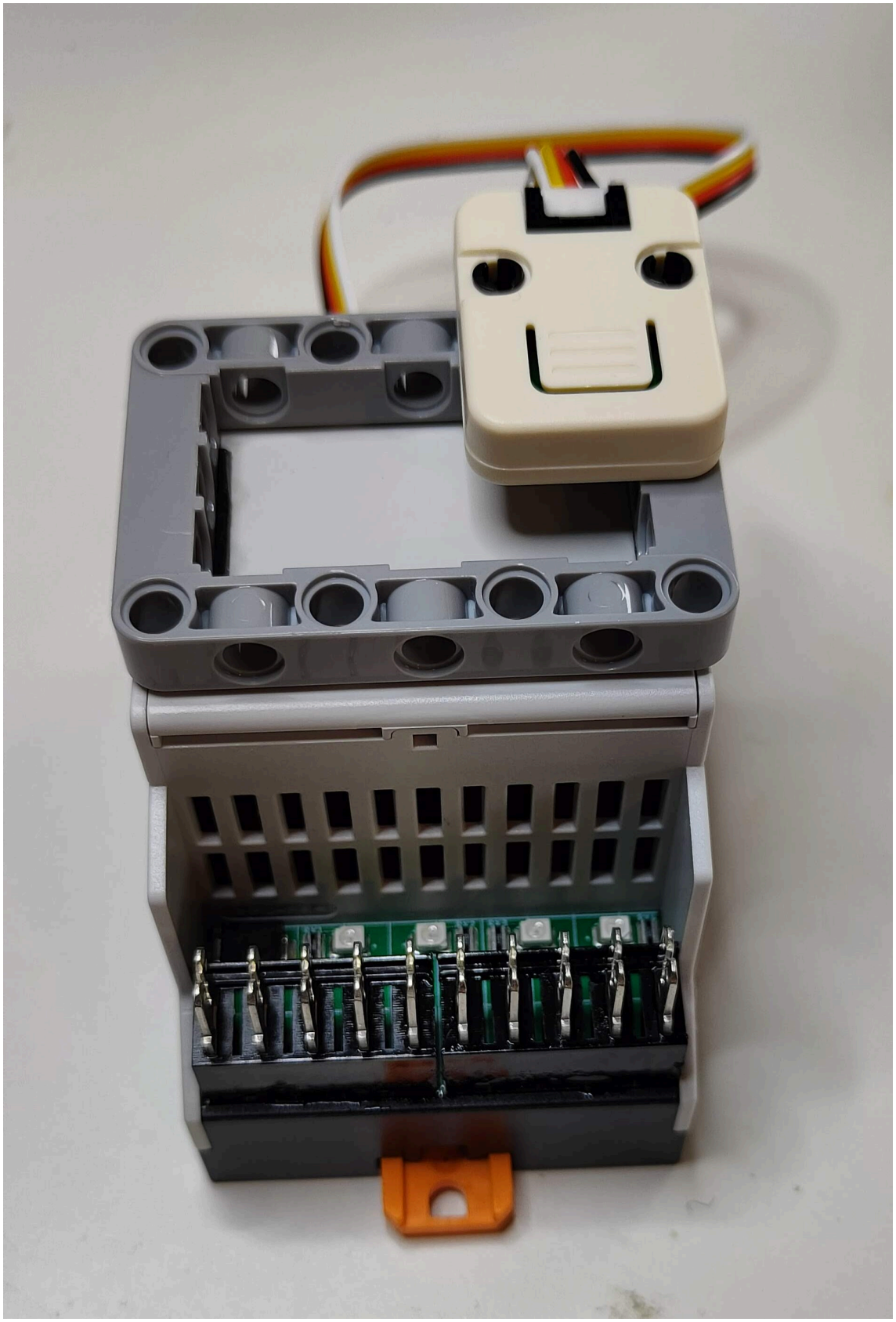
- 🌐 Total Resistance in Series & Parallel Circuits on ms-muc-docs.de
- 🌐 Loaded & Unloaded Voltage Dividers on ms-muc-docs.de
- 🌐 Bipolar transistor fundamentals at ms-muc-docs.de

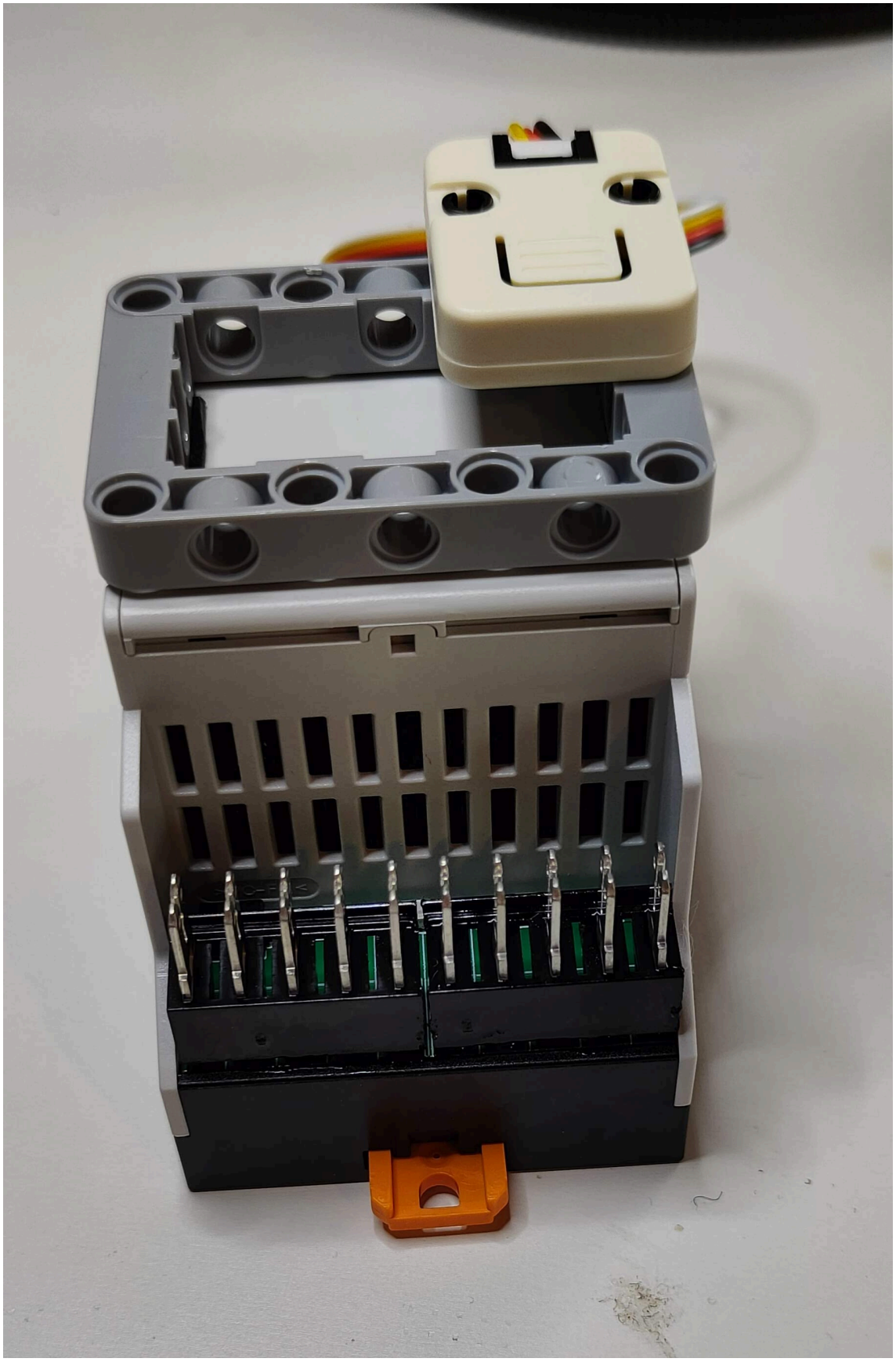
8. DIN Rail Moped Extensions

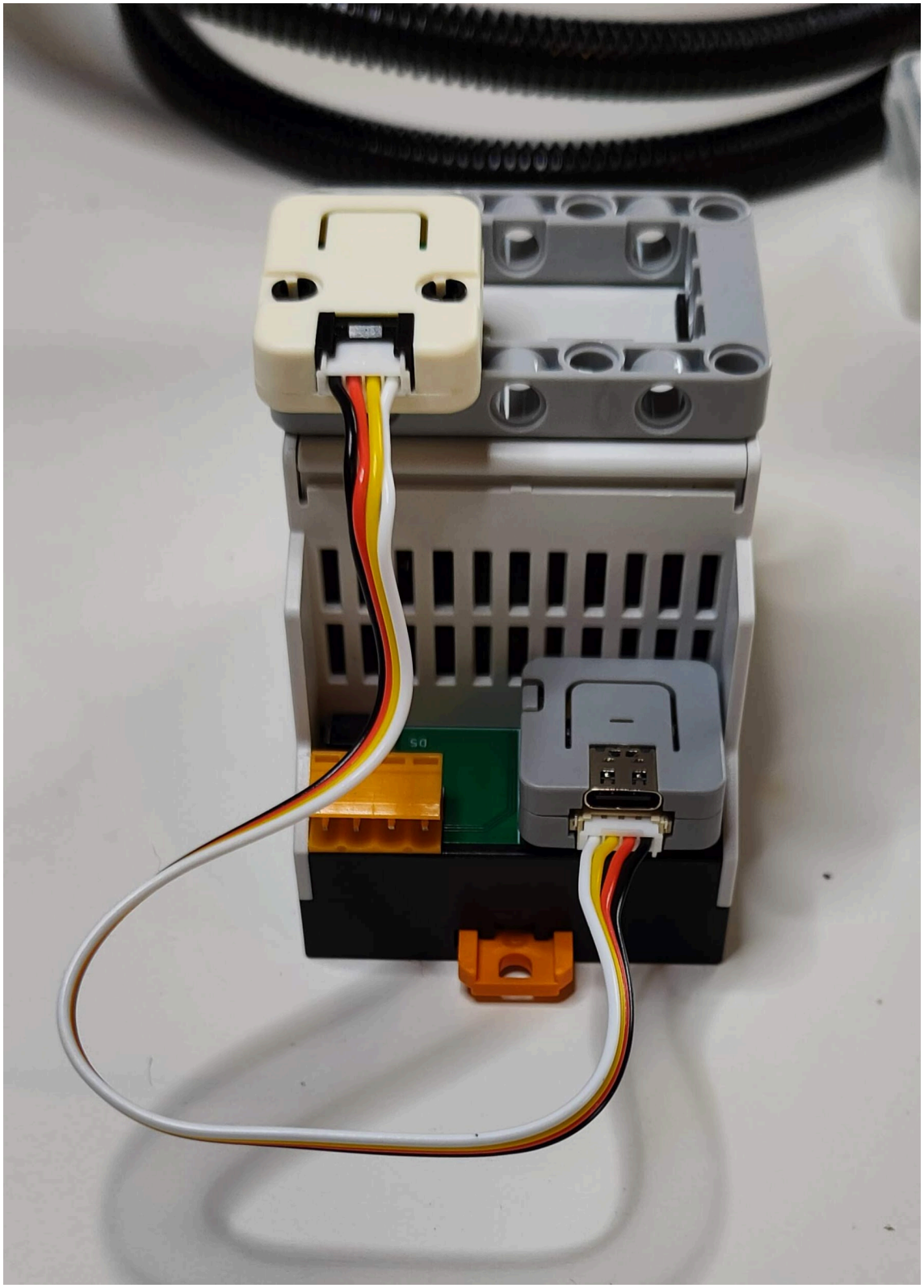
See also: https://install-isobus-environment-docs.readthedocs.io/de/latest/Atom-Erweiterungen.html

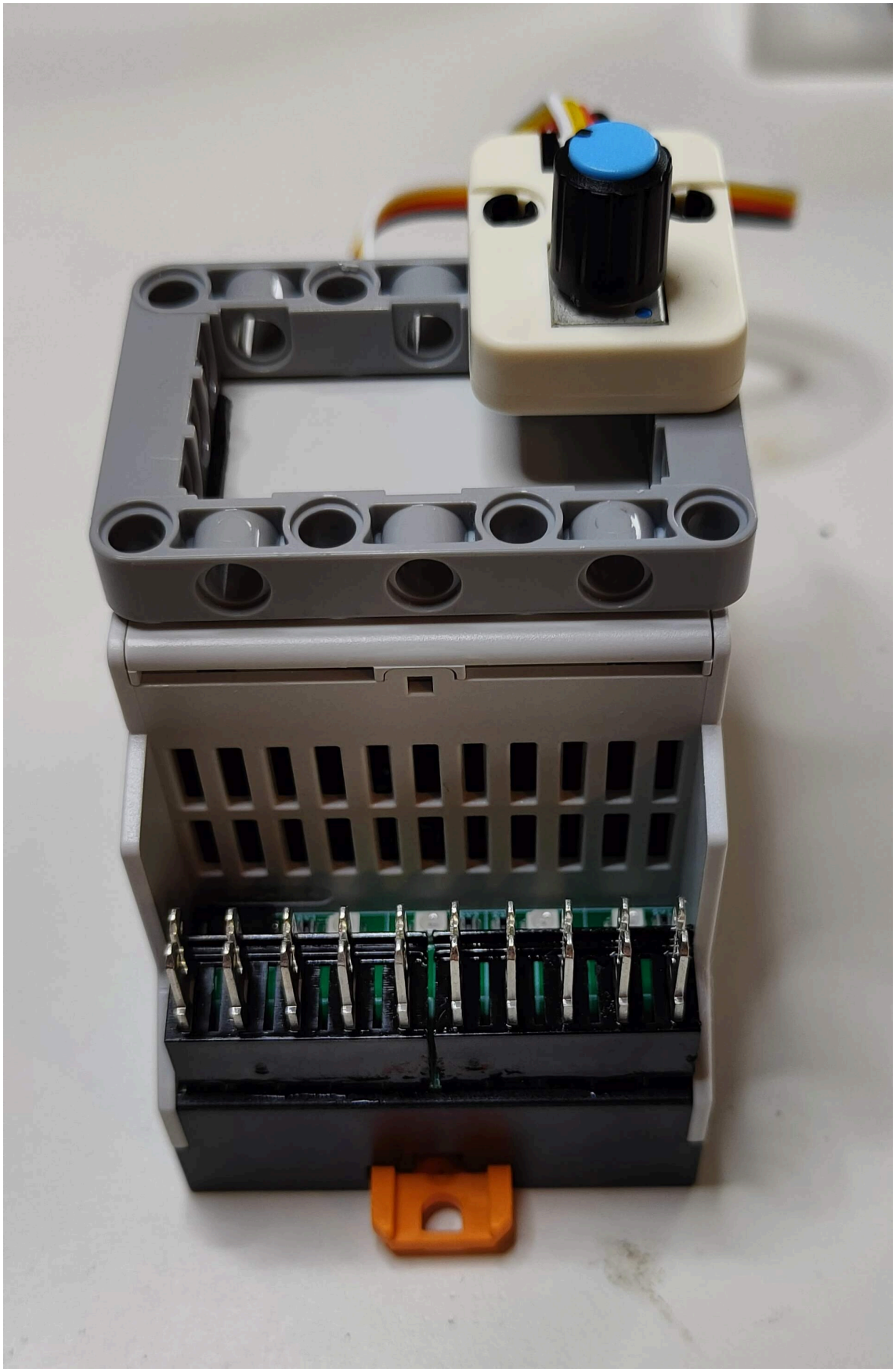


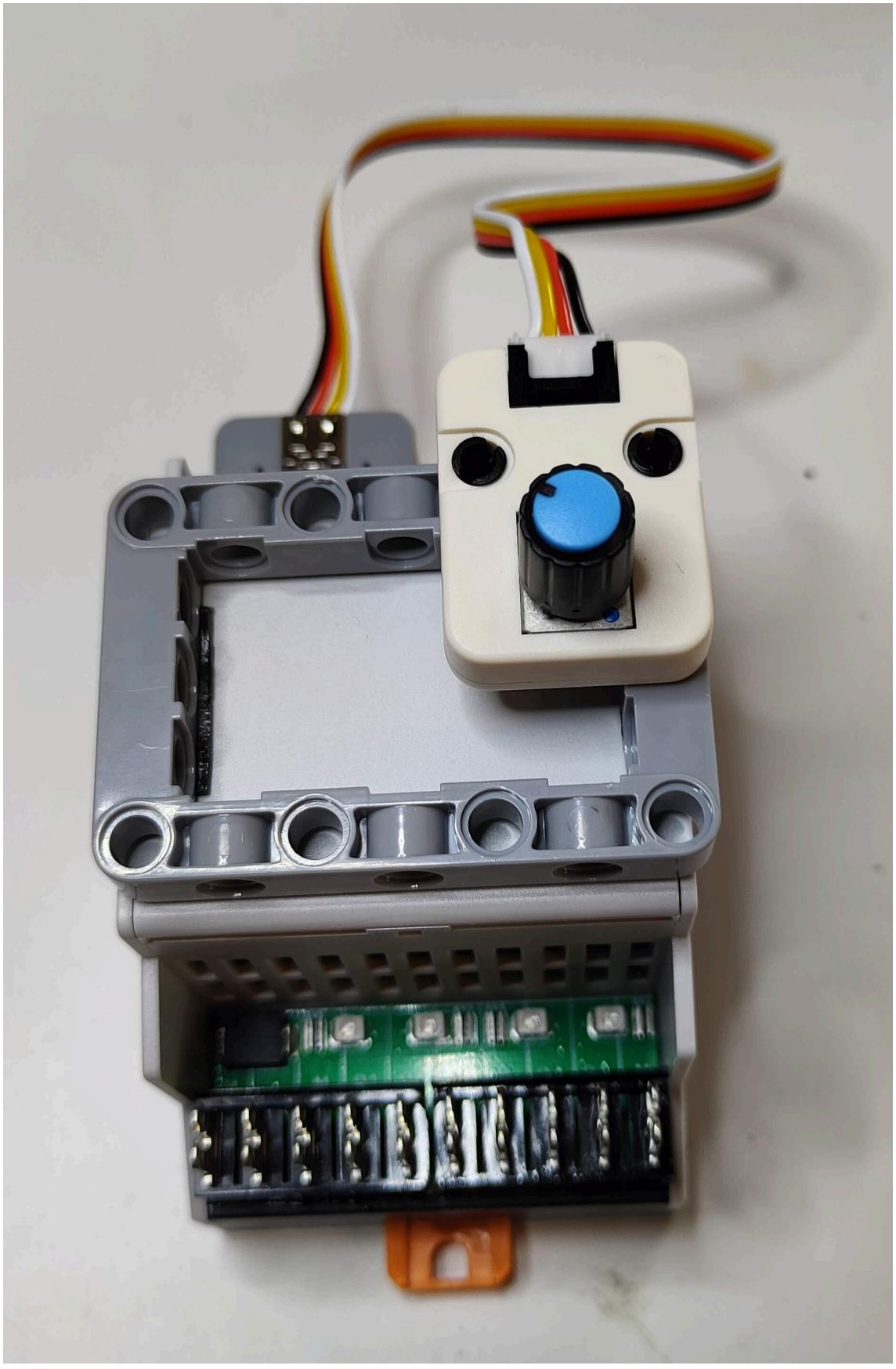


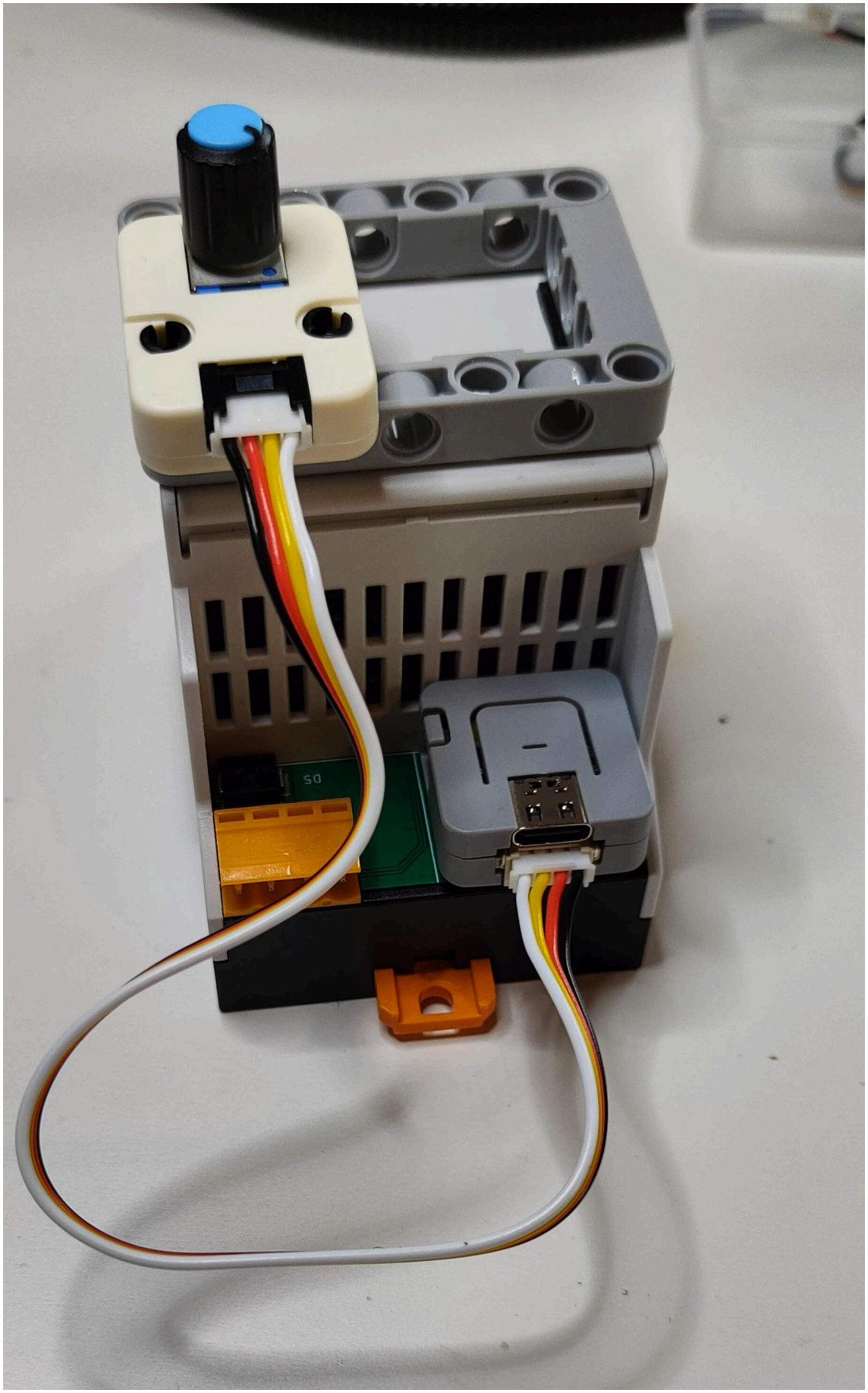












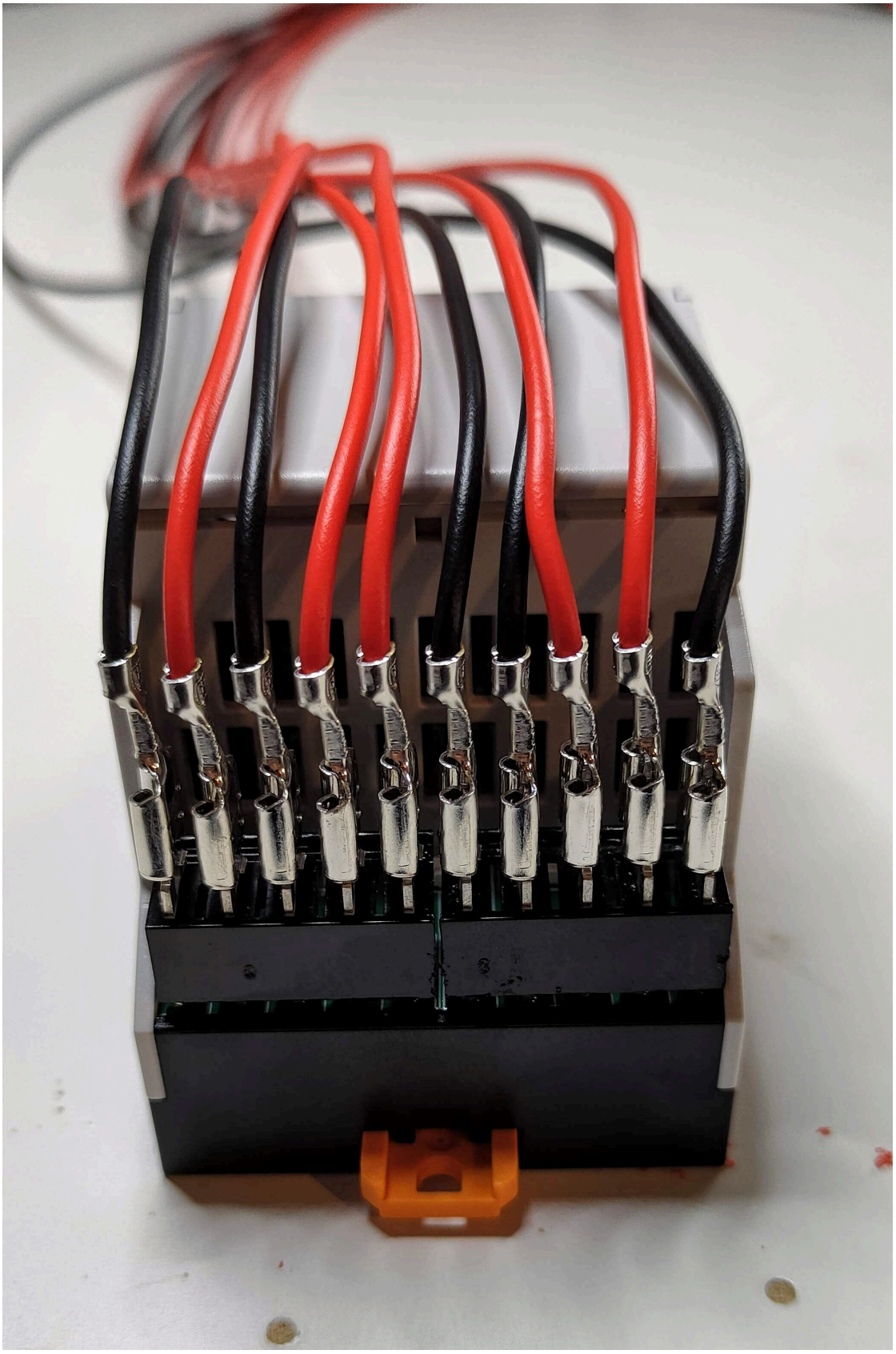
9. DIN Rail Moped Connectors

This photo series documents the correct wiring and connector types used for the “DIN Rail Moped” module.

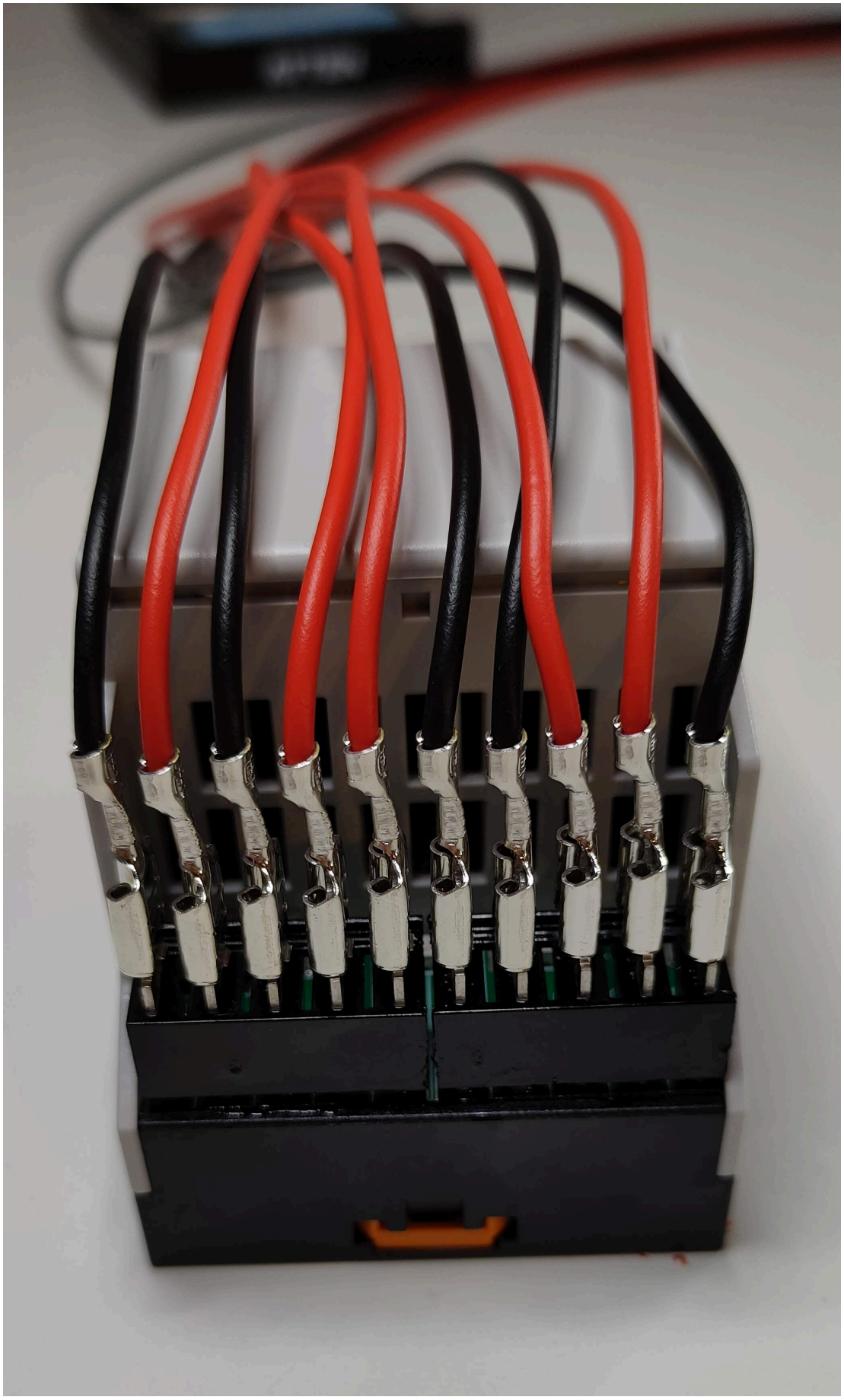
9.1 1. Basic Wiring and Connections

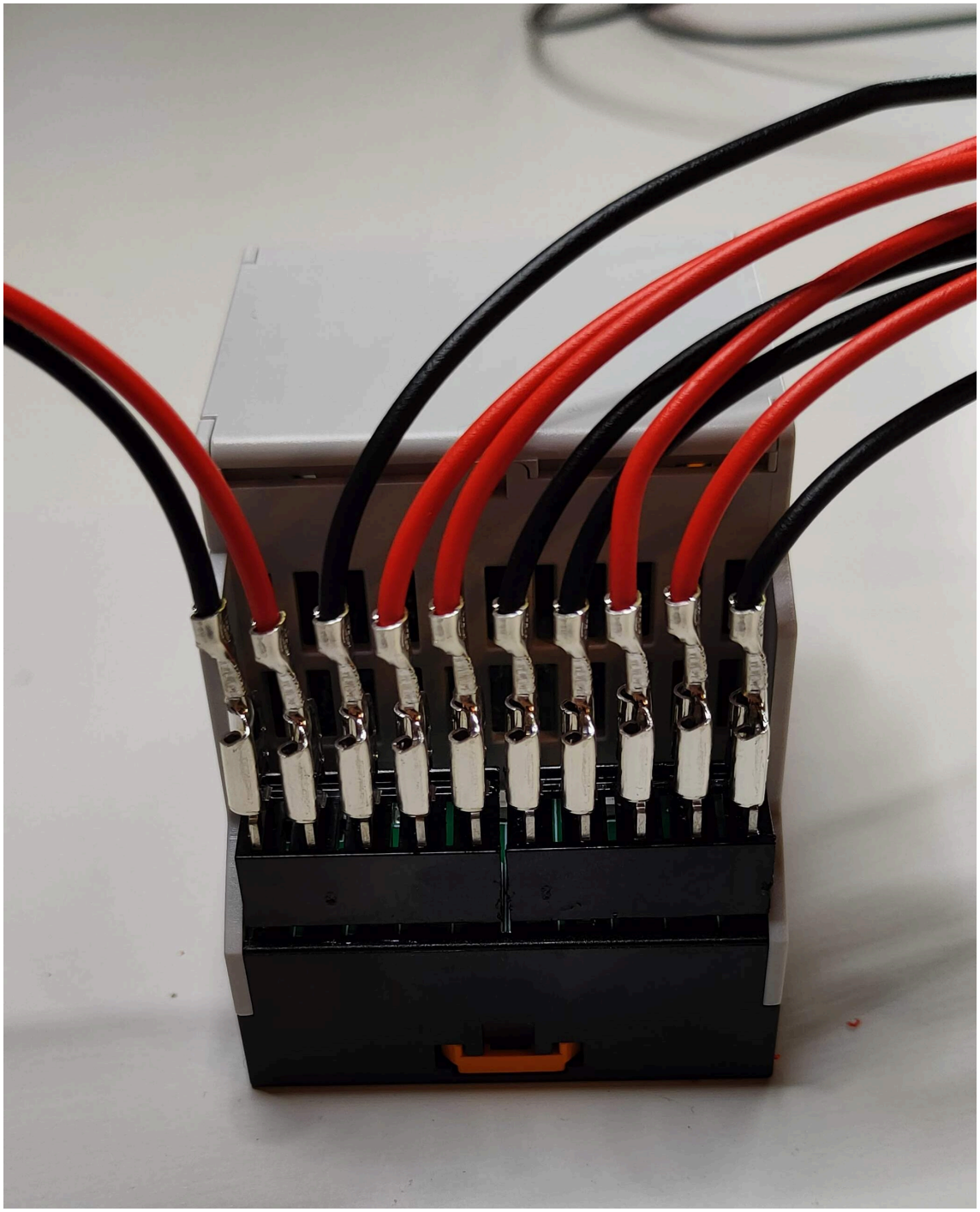
The module’s basic wiring on the DIN rail is done via red and black wires (power supply and ground), which are connected to the circuit board’s pin headers using spade connectors.

- **Connector Type:** 6.3 mm spade connectors are used as standard.



6.3 mm Spade Connectors





9.2 2. Insulation and Touch Protection

To prevent short circuits between the closely spaced contacts and to ensure touch safety, the spade connectors are insulated. Red and black insulating sleeves (or heat shrink tubing) completely enclose the crimp points and the rear part of the connectors.



9.3 3. Detail View: Special Signal Connections

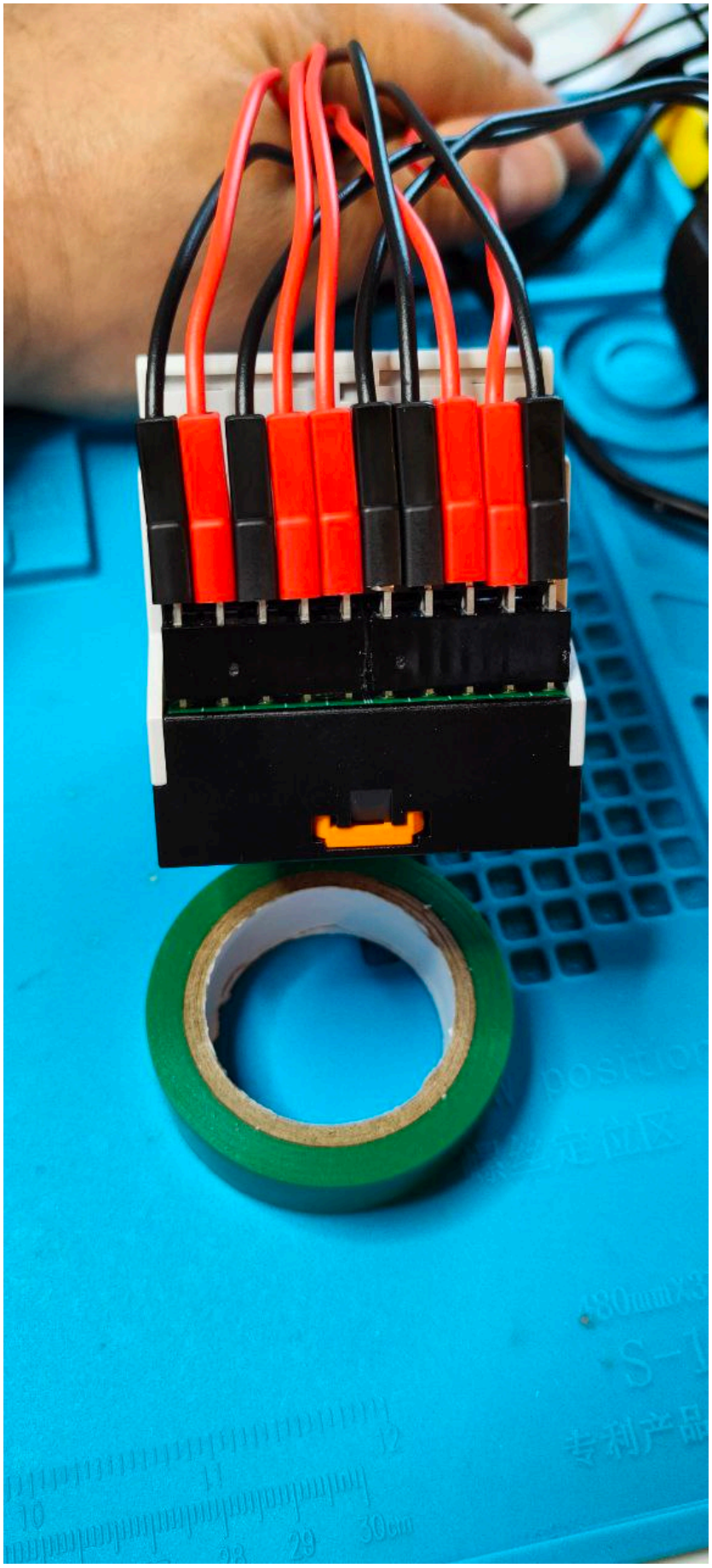
For specific connections or in confined spaces, different sizes are used. The image shows the use of **two 2.8 mm spade receptacles** mounted side-by-side on a terminal block. This is important for contacts that are not mechanically designed for the wider 6.3 mm receptacles.

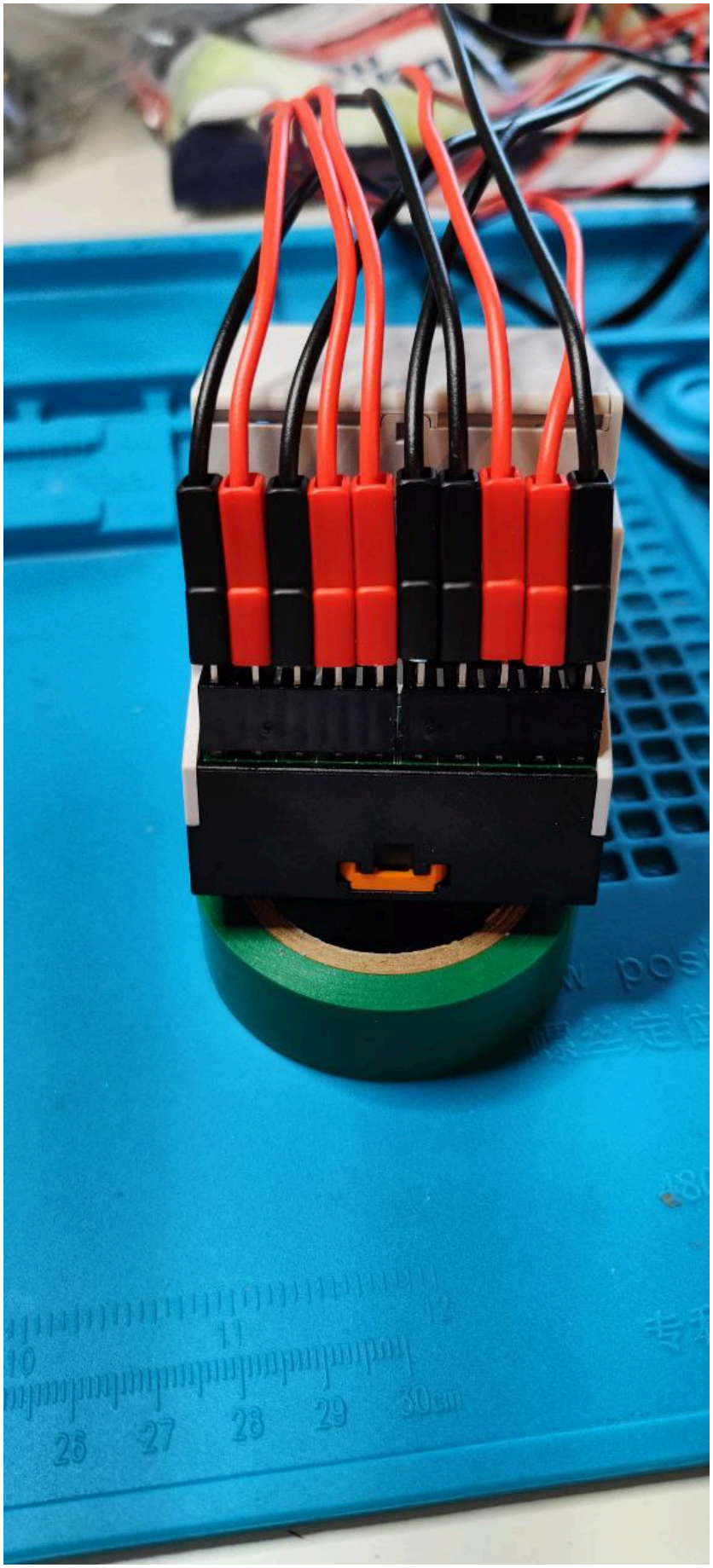


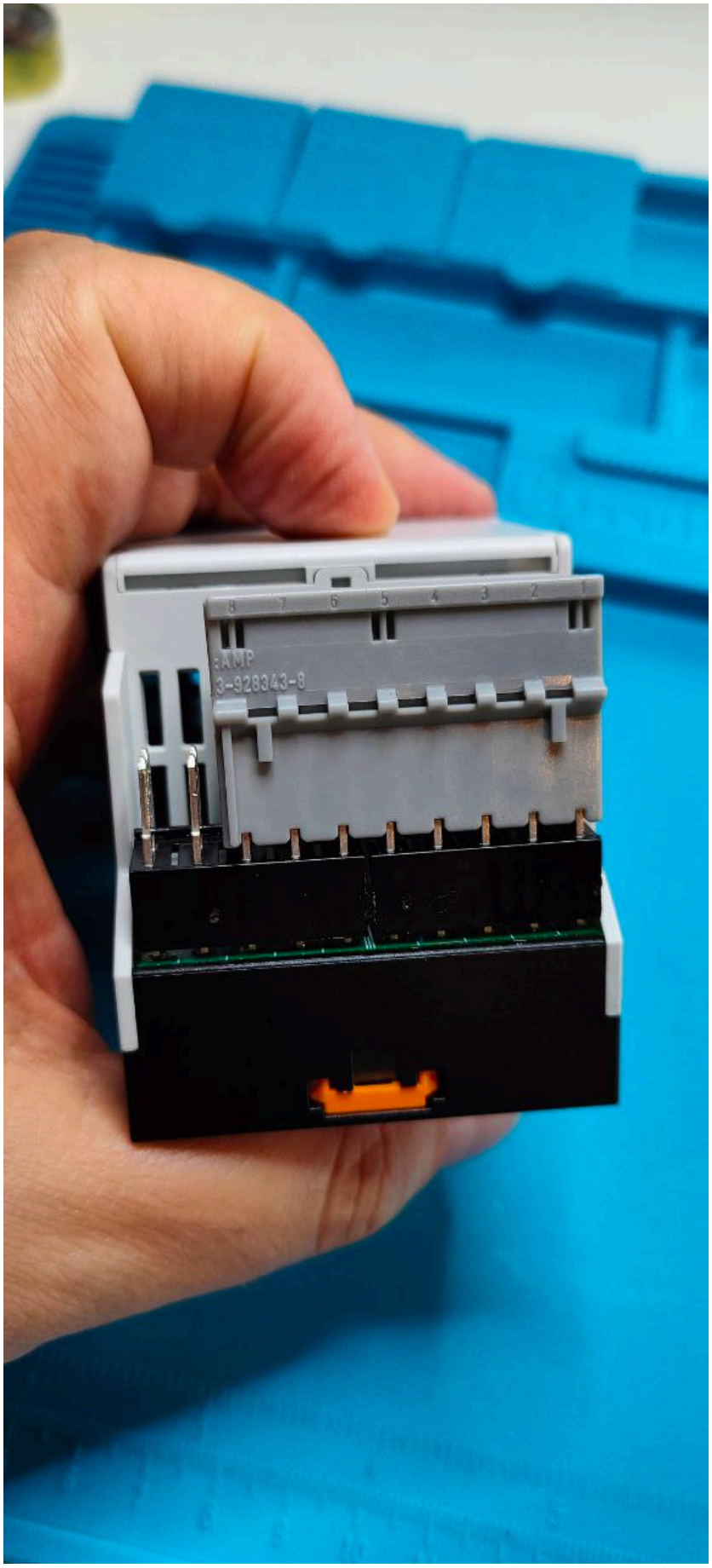
2x 2.8 mm spade receptacles

9.4 4. Connector Housing and Overall View

The pre-insulated wires are neatly routed. For a more modular connection, a gray AMP (TE Connectivity) connector housing with part number **3-328343-8** is used to bundle multiple contacts.

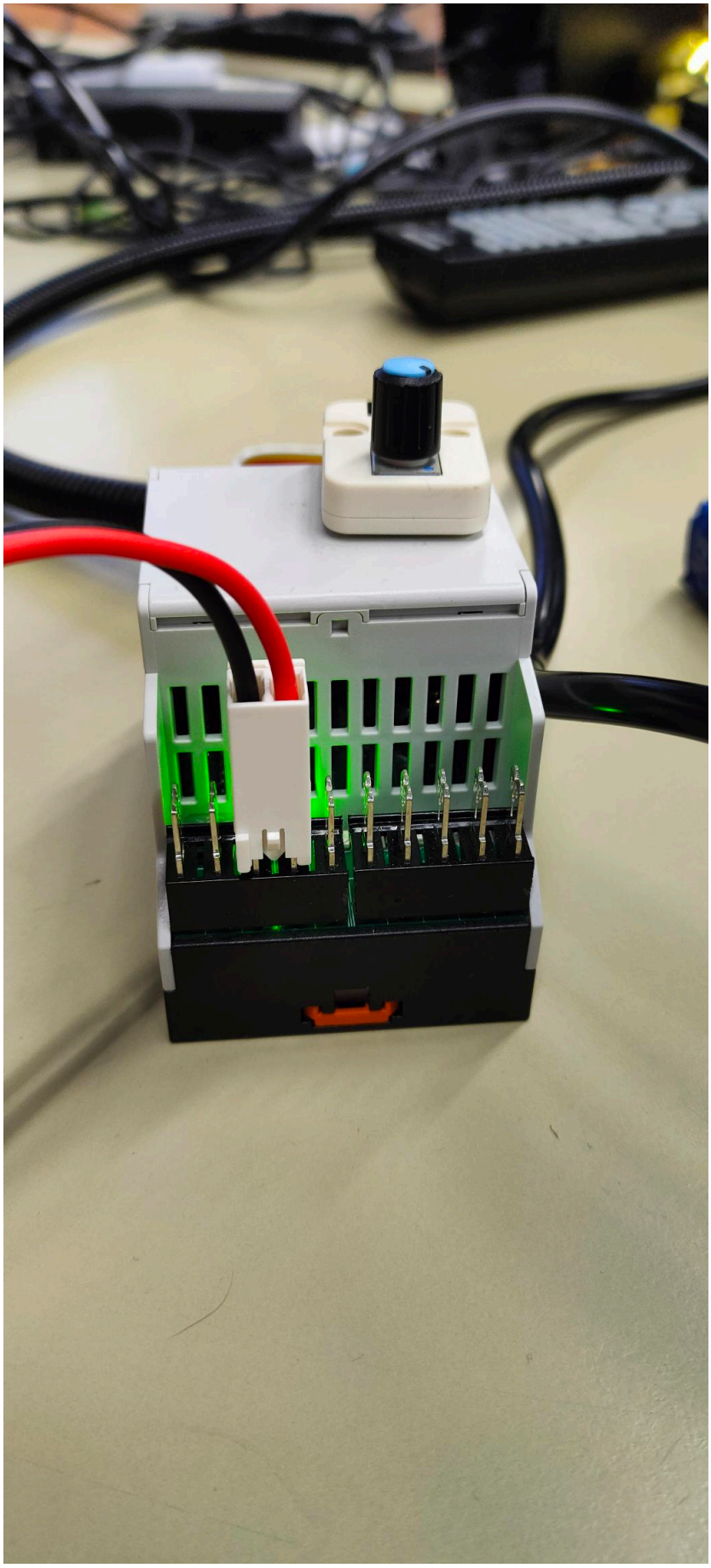





8 7 6 5 4 3 2 1
AMP
3-928343-8

9.5 5. Operating State

The fully wired module in active operation: The green LEDs inside the housing are illuminated, and the peripherals are connected.



10. DINRailMoped

10.1 🎧 Podcast

- DINRailMoped XL: ISOBUS ECU Development]

10.2 📂 Repository & Resources

The source code and hardware files are located in the GitHub repository: 📂 GitHub: DINRailMoped]

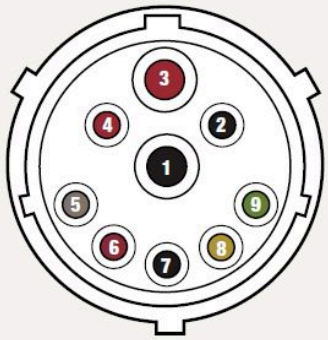
10.3 ⚡ Connection & Pinout

The DIN rail moped features fused outputs and flexible connection options.

- **Fuse Protection:** The individual outputs are protected by MOSFETs (see datasheet).
- **Power Supply:** The power supply requires an external fuse for line protection.

10.3.1 Pin Assignment

Pin	Bezeichnung	Farbe	Beschreibung	Spannung	Querschnitt/ Bemerkung
1	GND	schwarz	Masse	Masse	6 mm²
2	ECU_GND	schwarz	Masse Steuereinheit	Masse	2,5 mm²
3	PWR	rot	Leistungsstromversorgung	12 V	6 mm²
4	ECU_PWR	rot	Stromversorgung der Steuereinheit	12 V	2,5 mm²
5	TBC_DIS	-	Steuerung Terminierung		Brücke
6	TBC_PWR	rot	Stromversorgung für Terminierung	12 V	
7	TBC_RTN	schwarz	Masse für Terminierung	Masse	
8	CAN_H	gelb	Datenübertragung		verdrillt
9	CAN_L	grün	Datenübertragung		



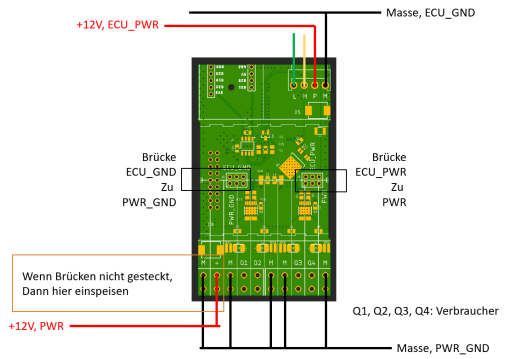
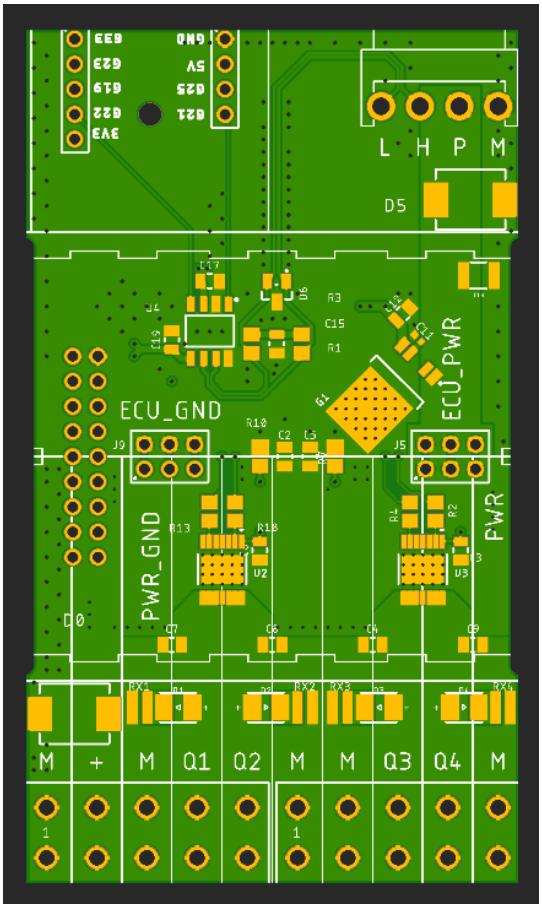
10.4 ⚙️ Configuration (4-pin vs. 6-pin)

The device can be operated in different ways depending on the application.

10.4.1 Operation with a 4-pin cable

For operation with a 4-pin cable, **two jumpers** must be installed.

View	Description

	**Position of the jumpers** Detailed view of the slots.
	**Schematic representation** Visualization of the jumper connection.

10.5 📸 Gallery & Mounting

The housing is designed for various mounting scenarios.

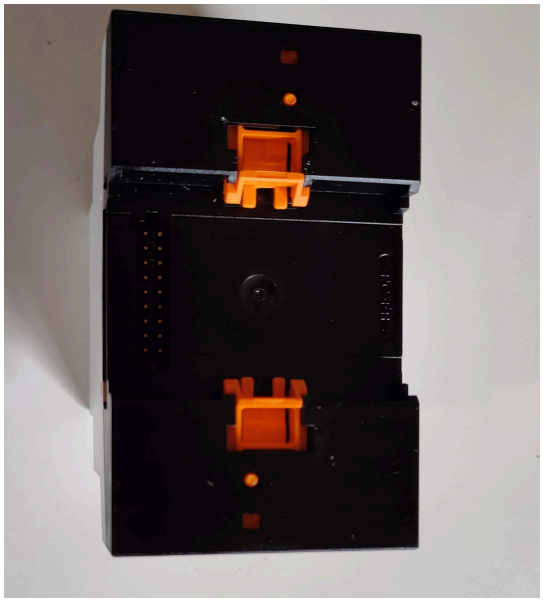
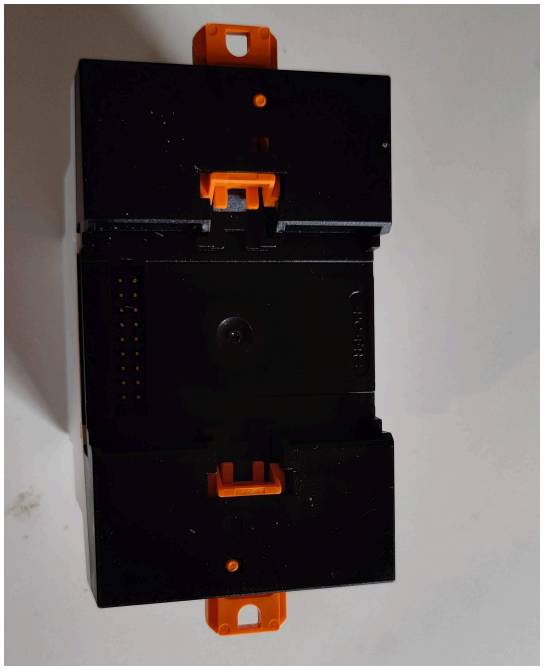
10.5.1 🔌 Hardware Views

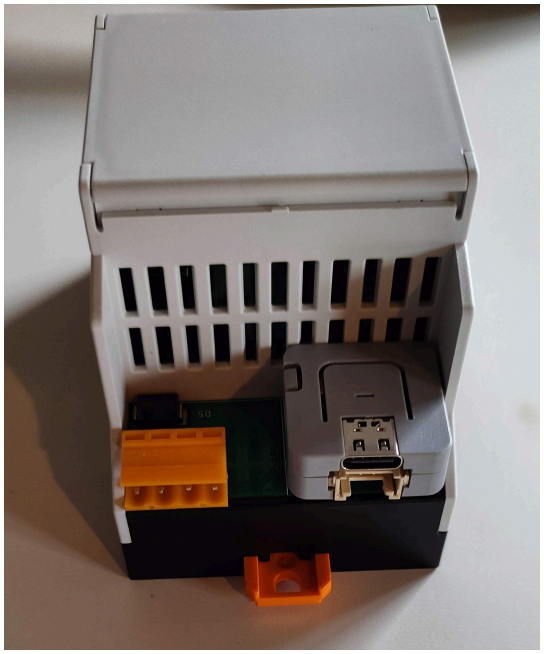
View	Description

	**PCB (Populated)**View of the components.
	**Case Open**Interior View.

	**Case Closed**Front View.

10.5.2 Mounting Options

DIN Rail	Wall Mount
	

–

10.5.3 🌐 Related topic subpages on ms-muc-docs.de

- 🌐 PCB design & assembly on ms-muc-docs.de

11. DIN Rail Moped XL

The **DIN Rail Moped XL** is the enhanced version of our open-source ECU. It was specifically developed for integration into ISOBUS systems and is ideally suited for demanding control tasks in agricultural machinery.

11.1 🌟 Highlights

- **ISOBUS Compatible:** Full support for **ISO 11783**.
- **IEC 61499:** Prepared for distributed control systems.
- **Open Source:** The hardware is fully disclosed and certified.
- **Expandable:** Thanks to its modular design and the use of the ESP32, the system can be flexibly adapted.

11.2 🎖️ Certification

The DIN Rail Moped XL is officially certified as open-source hardware.

- **UID:** DE000145
- **Certification Authority:** Open Source Hardware Association (OSHWA)



11.3 🛠️ Design & CAD

Hardware design is maintained in Autodesk Fusion 360. Here you can view the current 3D models and schematics:

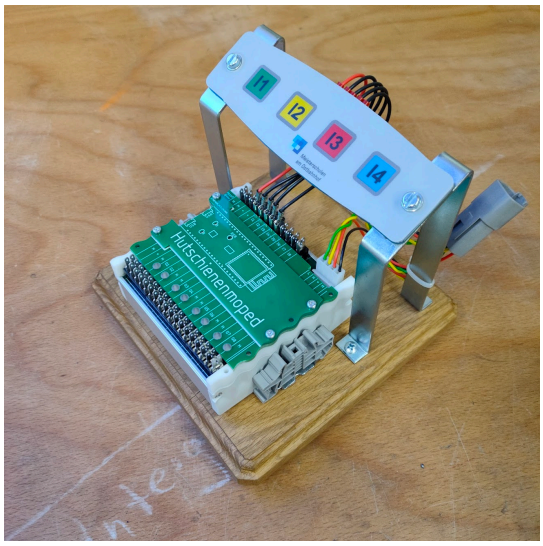
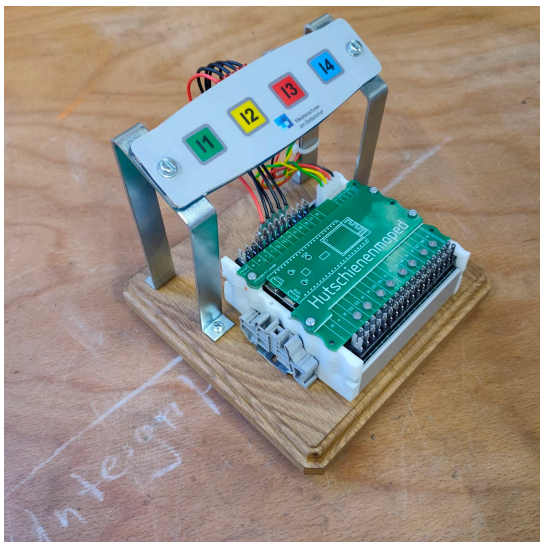
- 🔌 **View mainboard (PCB) in Fusion 360**
- 📦 **View complete assembly in Fusion 360**

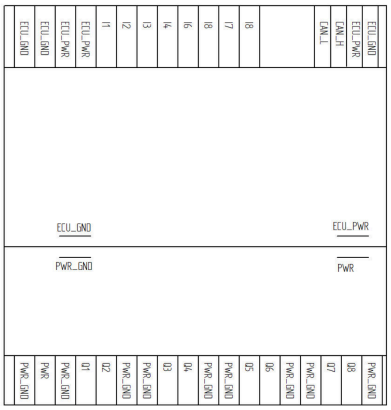
11.3.1 Repository

The source code and hardware files are located in the GitHub repository:
📂 GitHub: DINschienenmoped-XL

11.4 📸 Gallery

Here are some insights into the hardware:

View	Description
	**Assembled PCB (Top)** Shows the ESP32 and its connections.
	**Assembled PCB (Bottom)** Back side with trace layout.

	**Enclosure Assembly** The finished module in a DIN rail enclosure.

11.5 i Technical Details

- **Controller:** ESP32 (M5Stack Atom / Atom Lite compatible)
- **Power Supply:** 12V (vehicle electrical system)
- **Interfaces:**
- ISOBUS (CAN)
- Wi-Fi / Bluetooth (via ESP32)
- Expansion ports for sensors/actuators

11.5.1 🌐 Related topics on ms-muc-docs.de

- 🌐 ESP32 & ESP32-S3 DevKit on ms-muc-docs.de
- 🌐 PCB Design & Assembly on ms-muc-docs.de

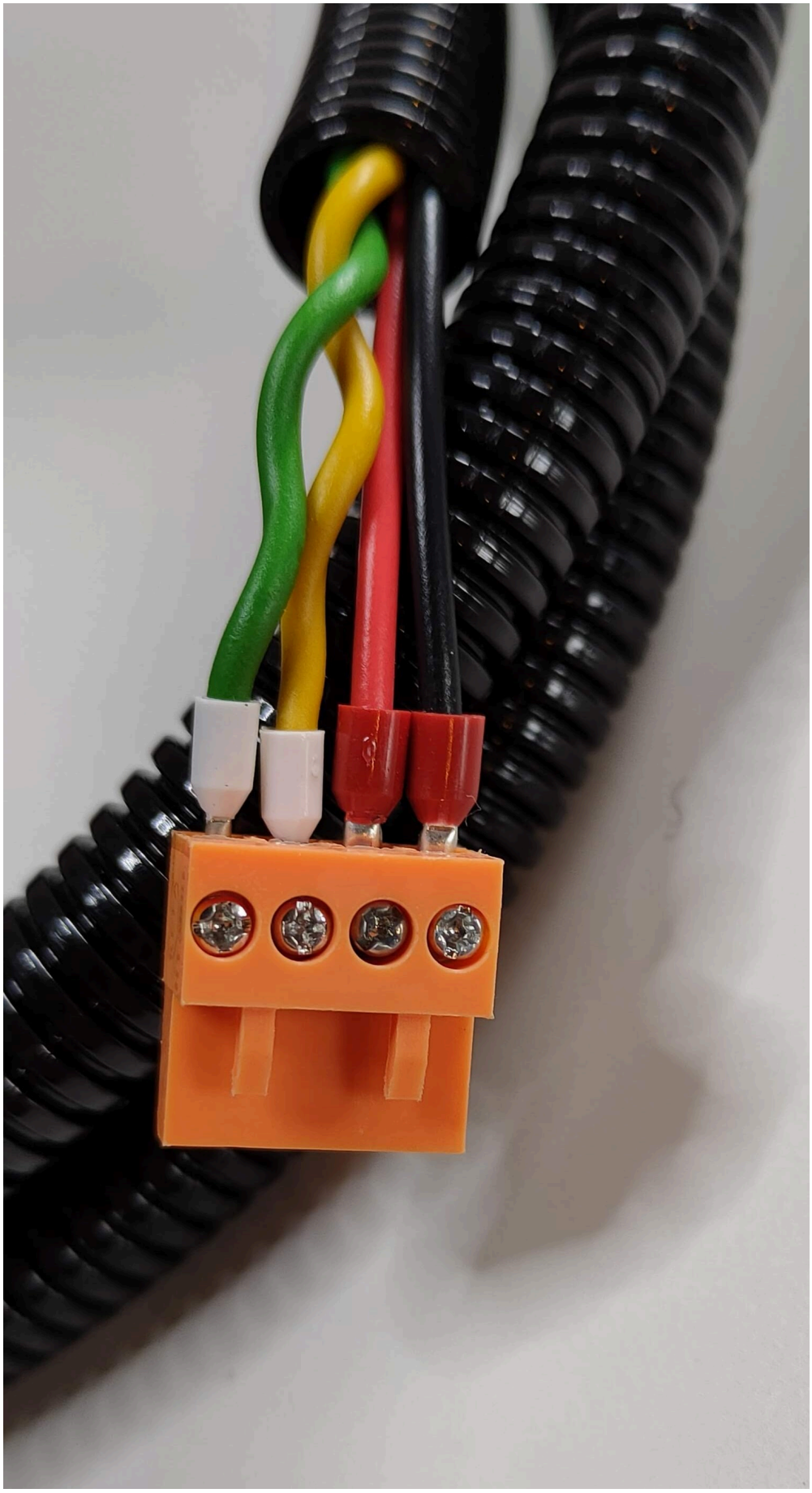
12. ISOBUS cable, small






DEUTS

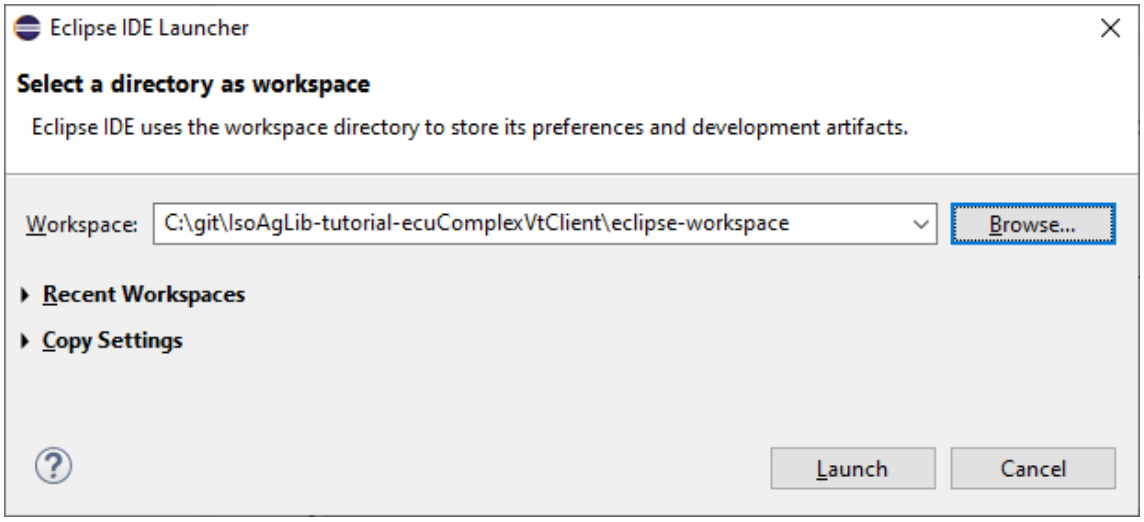

DEUTSCH



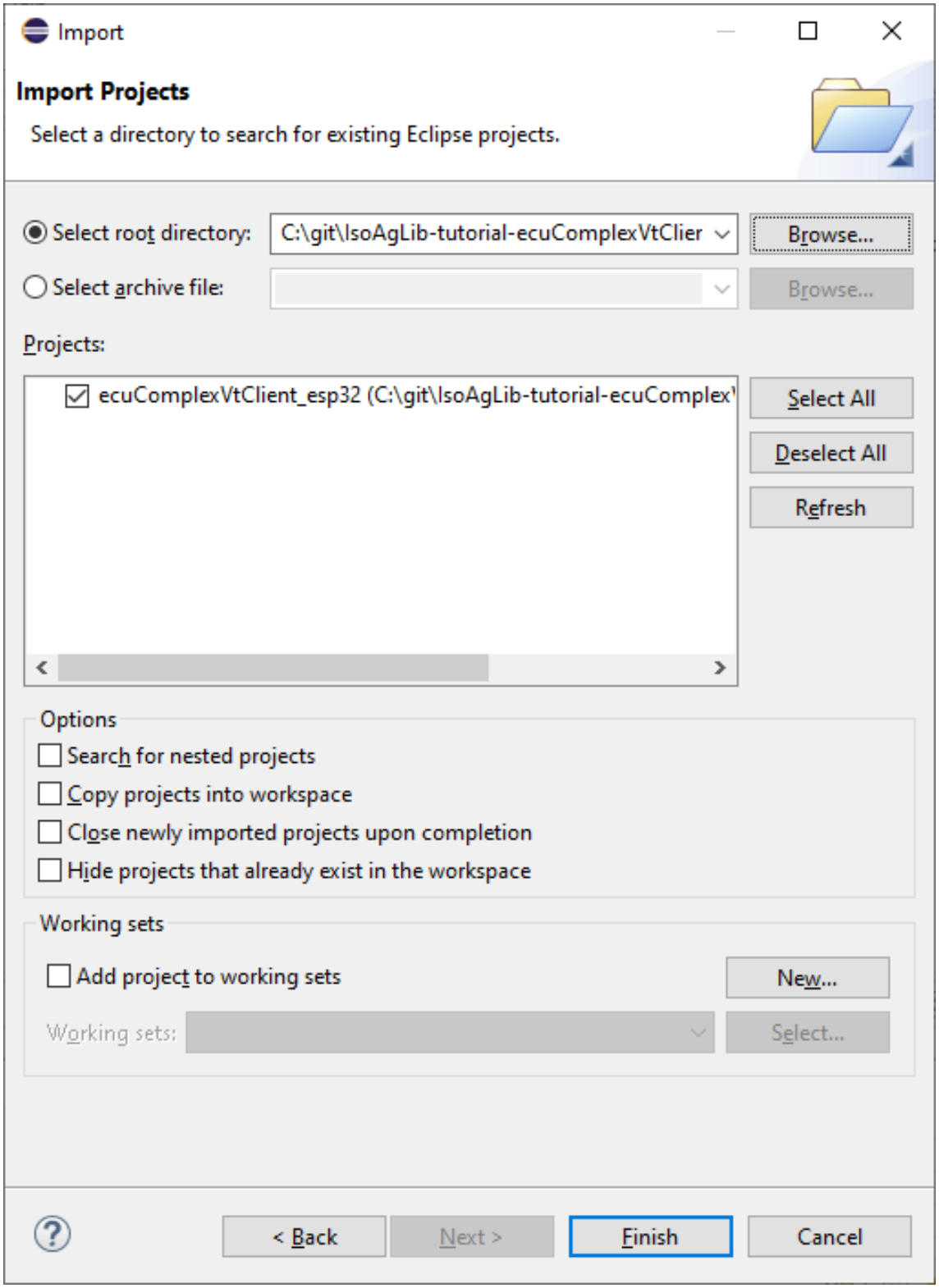
13. 📦 Installation

14. IsoAgLib-ISOMAN

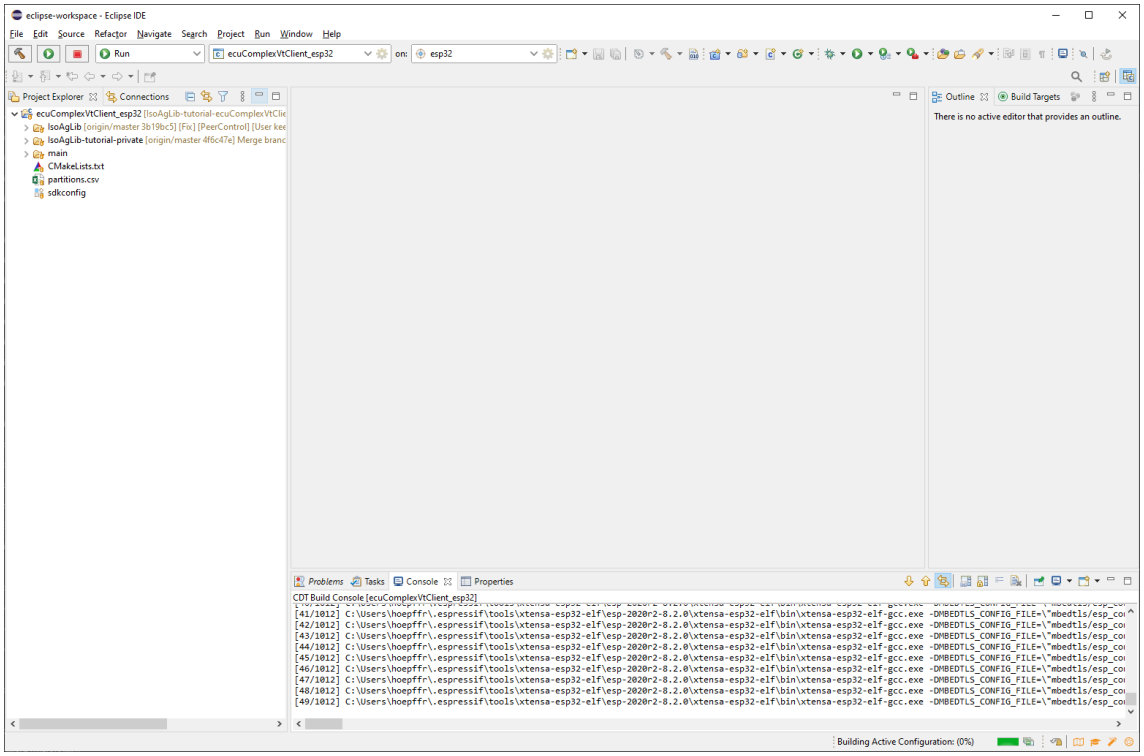
For this experiment, the atom must be mounted on the socket. Secure it carefully with the screw. ... (This is plastic, so 500 Nm of torque isn't needed >`_ Open the workspace C:\git\IsoAgLib-ISOMAN\eclipse-workspace



Import the project as before (you can do this now >`_



Create a launch target,
Download and configure the ESP-IDF (as before >`_
Then press the hammer,
then press Play:

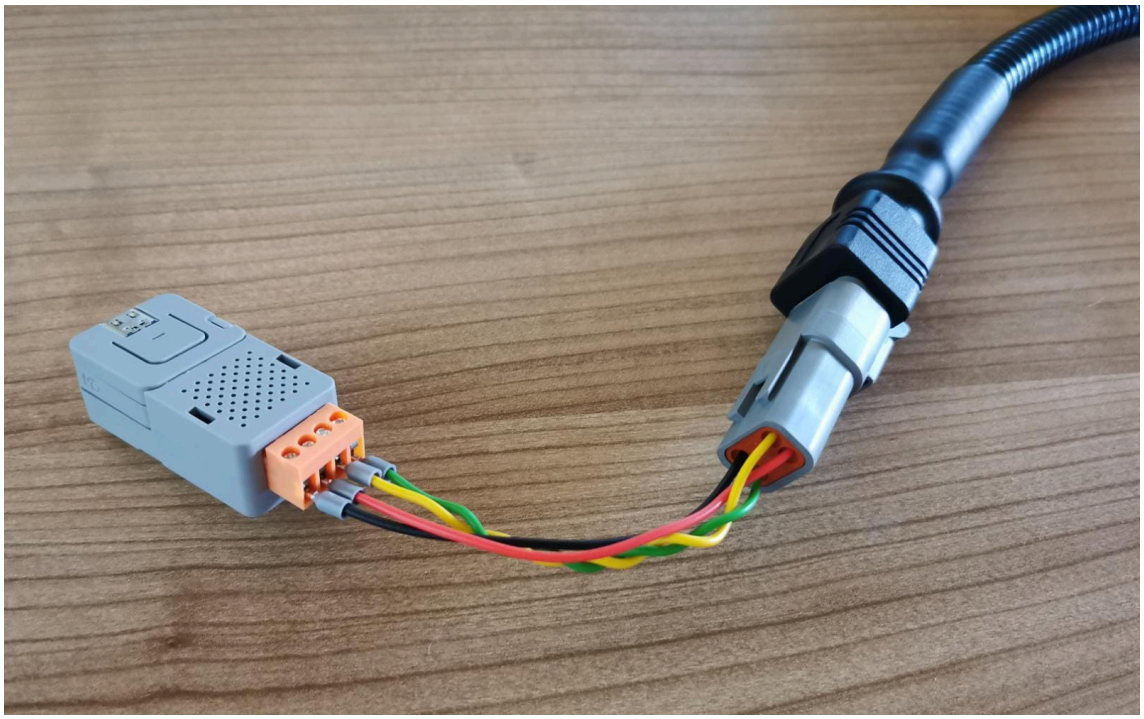


Now,

Get on the Fendt tractor and connect the device and see what happens.

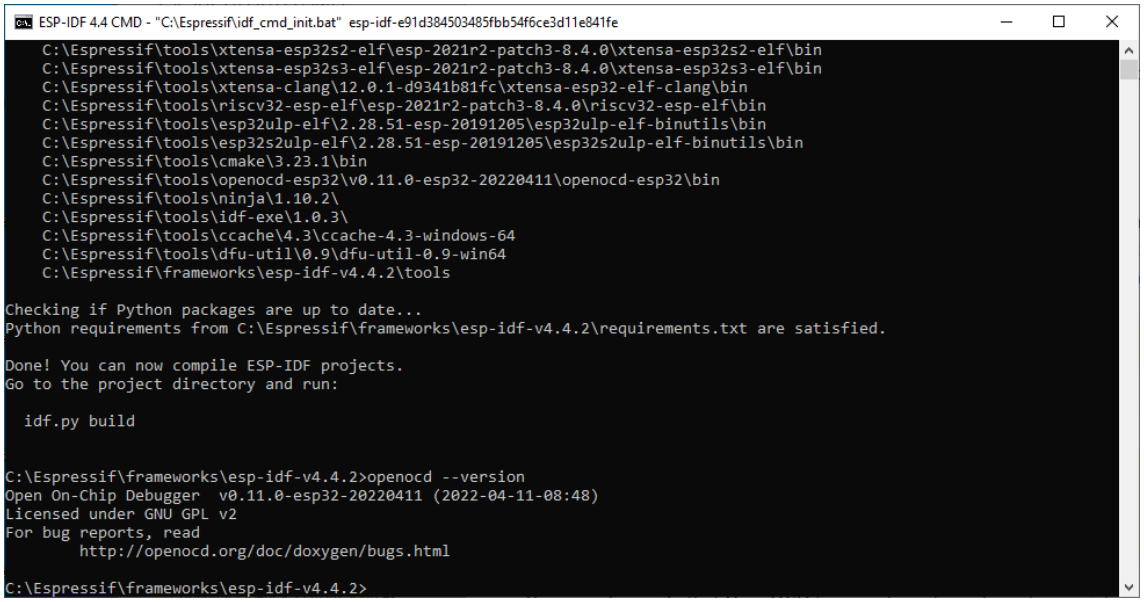
You do NOT need to connect the USB cable to the Fendt tractor; the Atom is powered by the 12V system.

Like this:

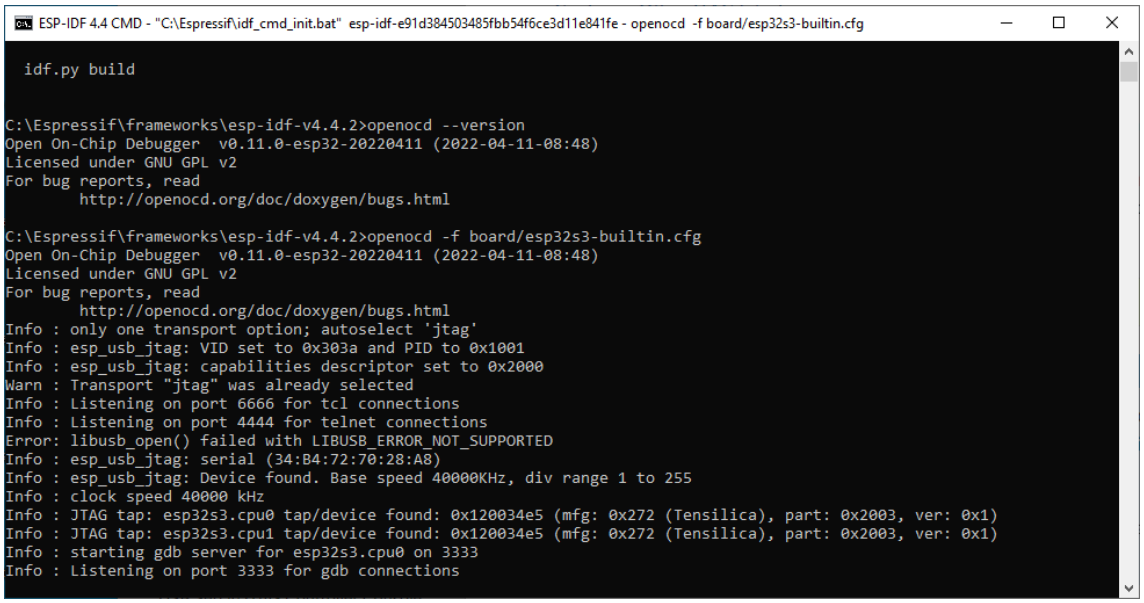


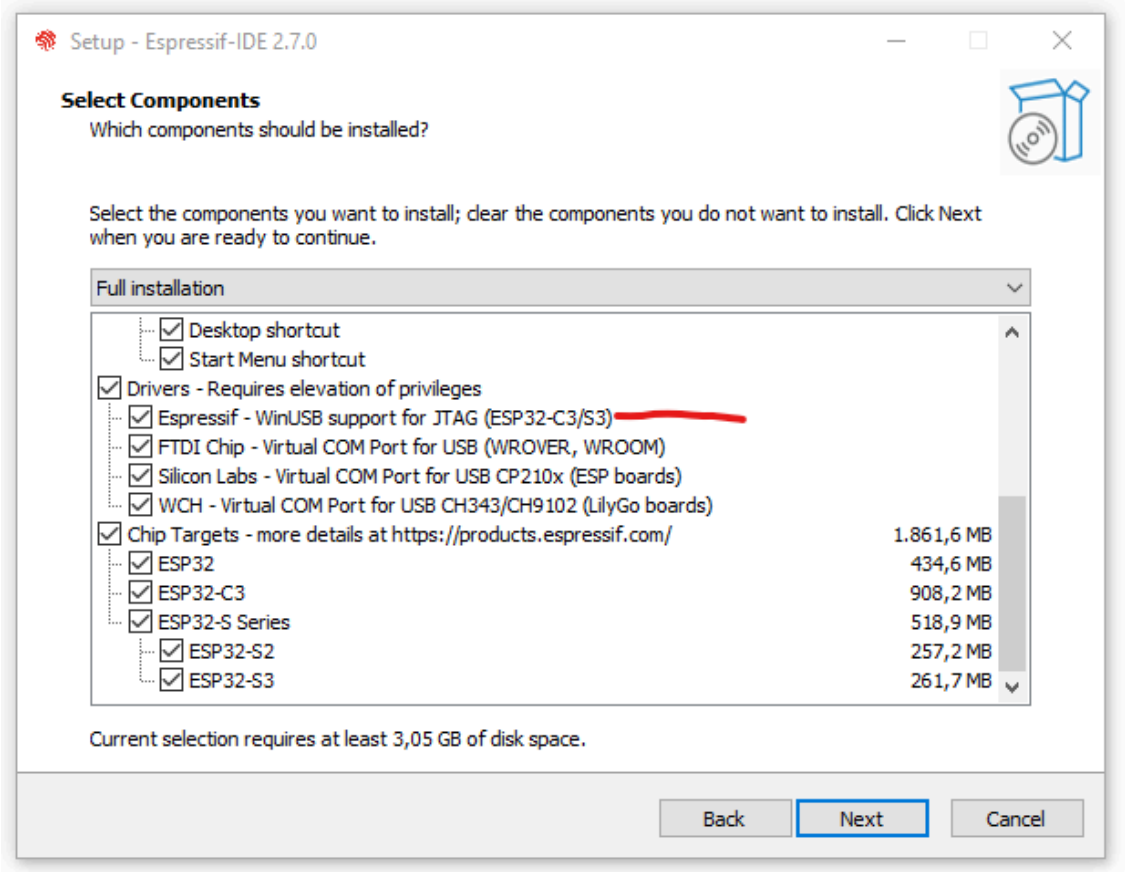
15. JTAG-S3-USB

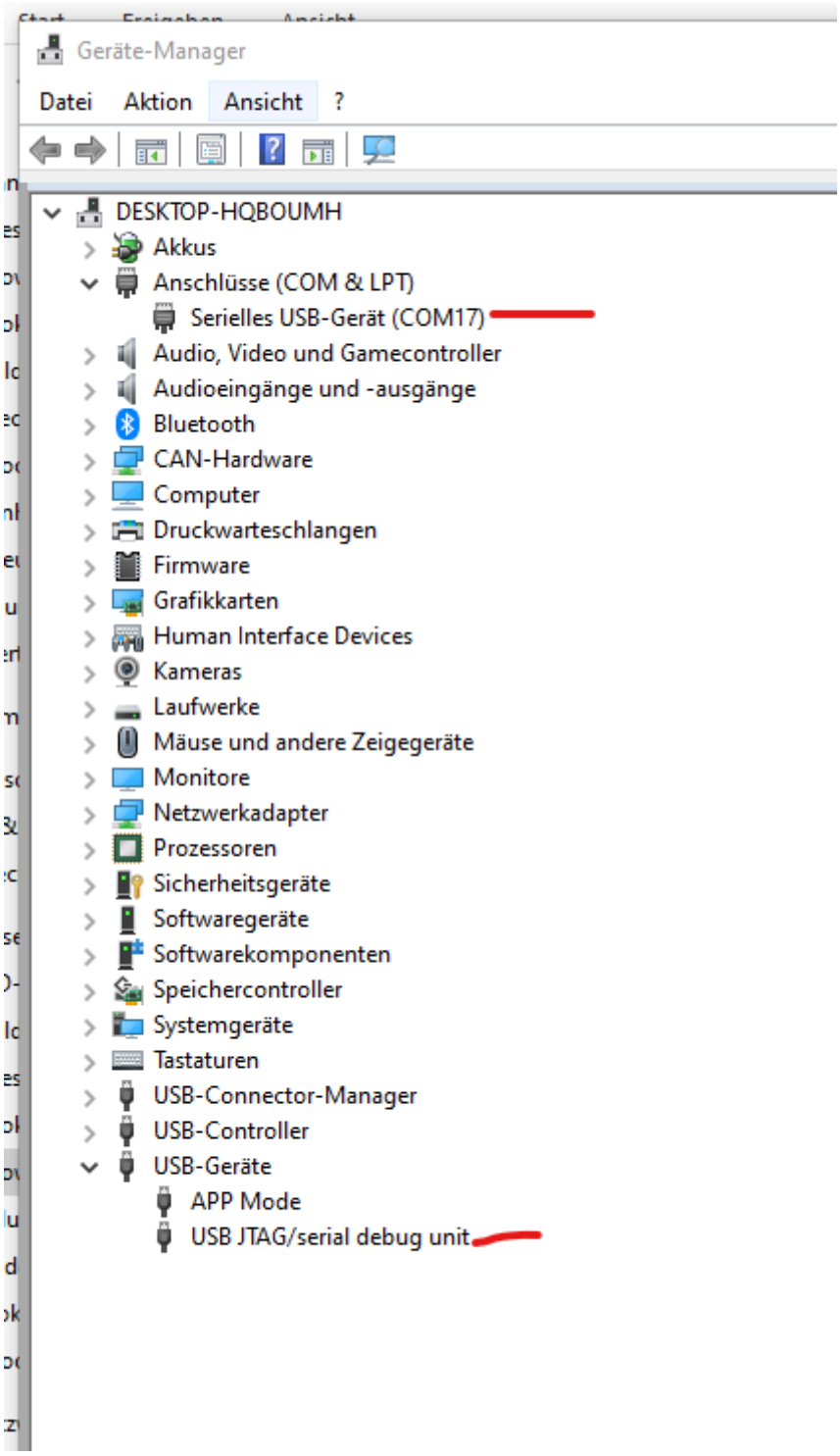
https://docs.espressif.com/projects/esp-idf/en/latest/esp32s3/api-guides/jtag-debugging/index.html
openocd –version

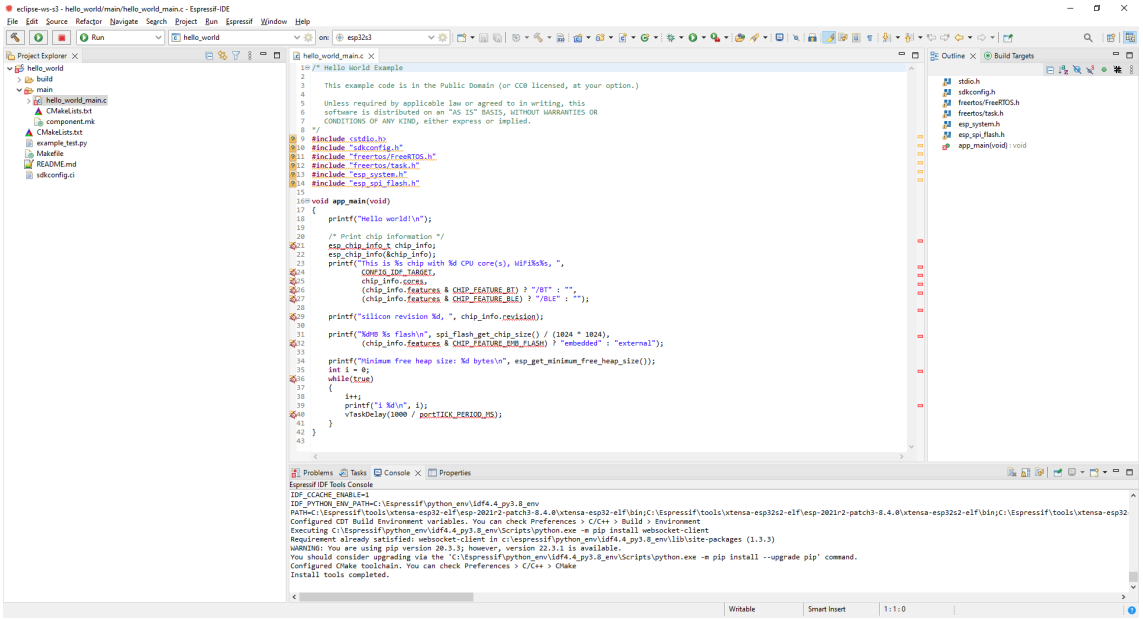


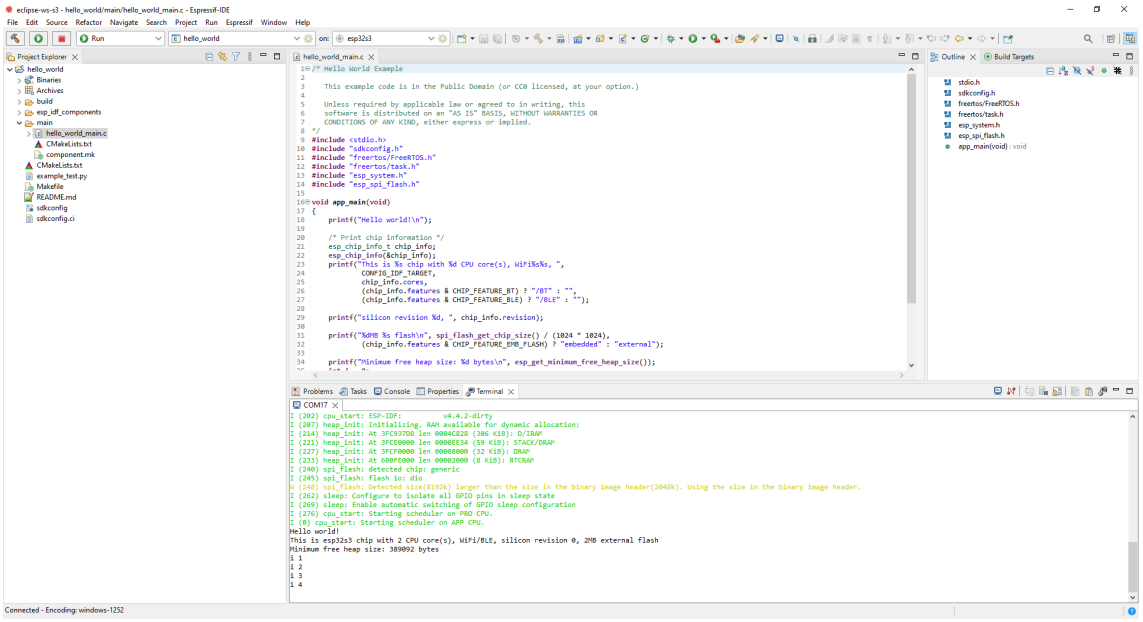
openocd -f board/esp32s3-builtin.cfg –> appears correct openocd -f board/esp32s3-bridge.cfg –> appears incorrect

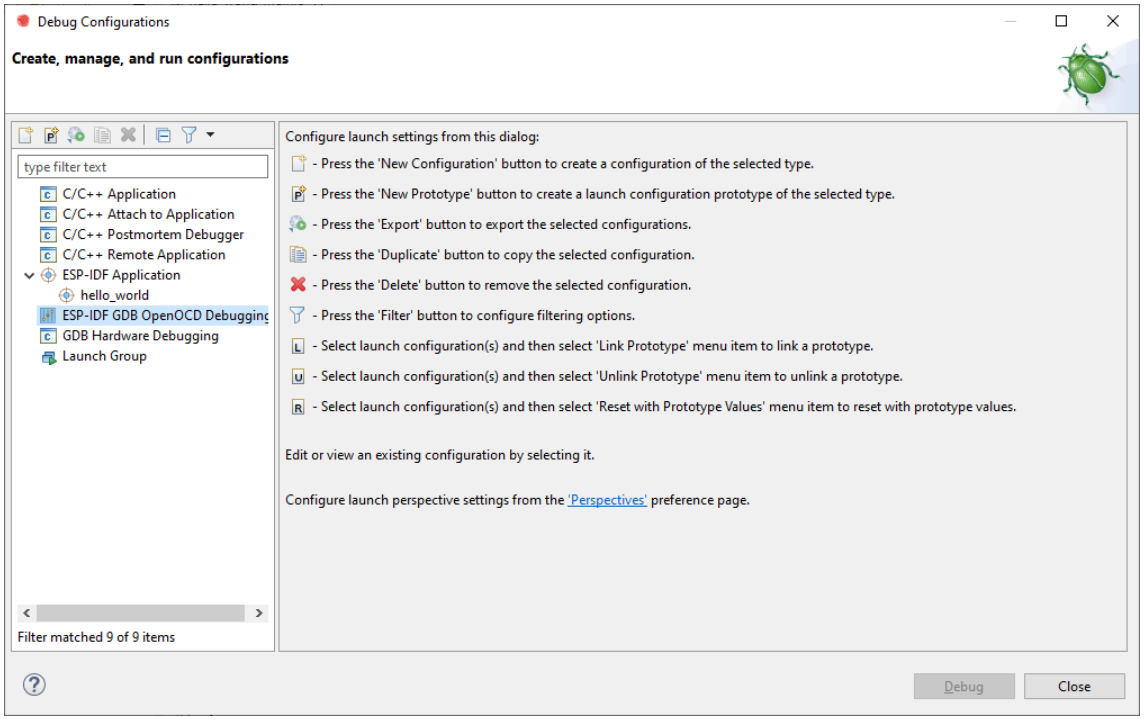


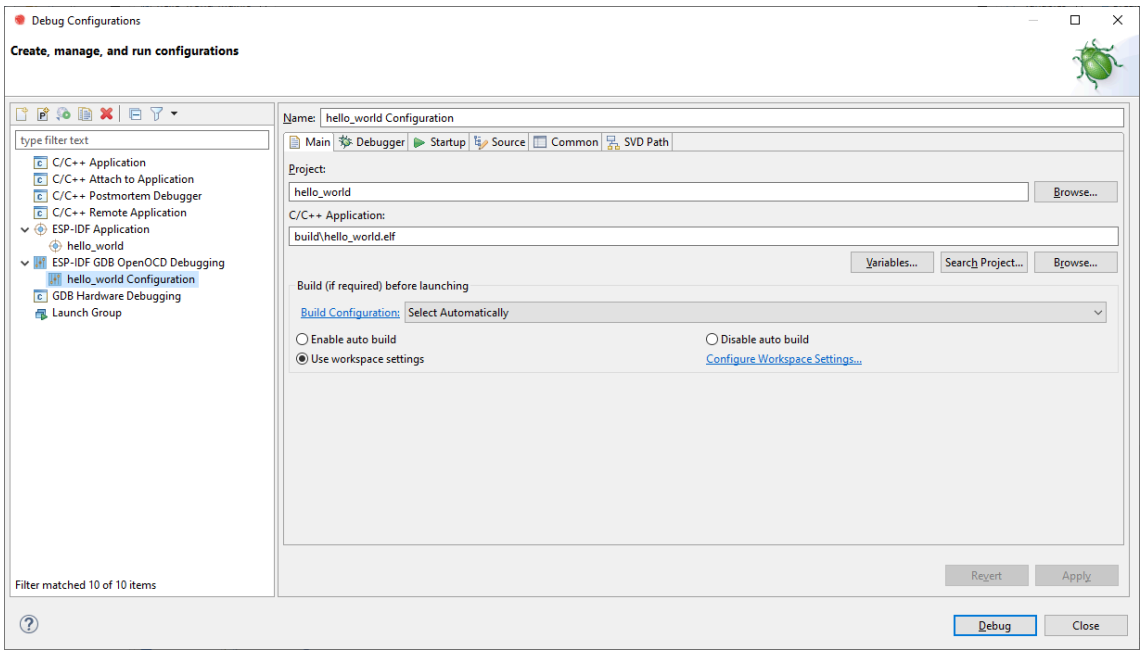

Setup - Espressif-IDE 2.7.0
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Full installation
Desktop shortcut
Start Menu shortcut
Drivers - Requires elevation of privileges
Espressif - WinUSB support for JTAG (ESP32-C3/S3)
FTDI Chip - Virtual COM Port for USB (WROVER, WROOM)
Silicon Labs - Virtual COM Port for USB CP210x (ESP boards)
WCH - Virtual COM Port for USB CH343/CH9102 (LilyGo boards)
Chip Targets - more details at https://products.espressif.com/ 1.861,6 MB
ESP32 434,6 MB
ESP32-C3 908,2 MB
ESP32-S Series 518,9 MB
ESP32-S2 257,2 MB
ESP32-S3 261,7 MB
Current selection requires at least 3,05 GB of disk space.
Back
Next
Cancel


Geräte-Manager
Datei
Aktion
Ansicht
?
DESKTOP-HQBOUMH
Akkus
Anschlüsse (COM & LPT)
Serielles USB-Gerät (COM17)
Audio, Video und Gamecontroller
Audioeingänge und -ausgänge
Bluetooth
CAN-Hardware
Computer
Druckwarteschlangen
Firmware
Grafikkarten
Human Interface Devices
Kameras
Laufwerke
Mäuse und andere Zeigegeräte
Monitore
Netzwerkadapter
Prozessoren
Sicherheitsgeräte
Softwaregeräte
Softwarekomponenten
Speichercontroller
Systemgeräte
Tastaturen
USB-Connector-Manager
USB-Controller
USB-Geräte
APP Mode
USB JTAG/serial debug unit

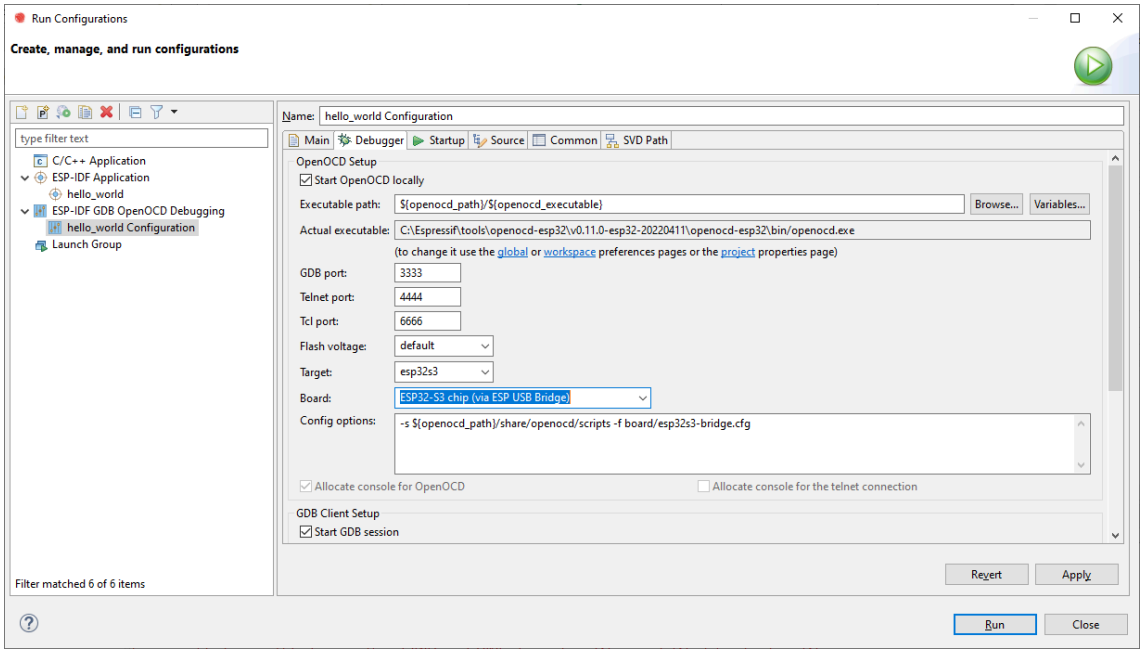

eclipse-ws-s3 - hello_world/main/hello_world_main.c - Espressif-IDE
File Edit Source Refactor Navigate Search Project Run Espressif Window Help
Project Explorer
hello_world
build
main
hello_world_main.c
CMakeLists.txt
component.mk
CMakeLists.txt
example_test.py
Makefile
README.md
sdkconfig.ci
hello_world_main.c
/* Hello World Example
This example code is in the Public Domain (or CC0 licensed, at your option.)
Unless required by applicable law or agreed to in writing, this
software is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied.
*/
#include <stdio.h>
#include "sdkconfig.h"
#include "freertos/FreeRTOS.h"
#include "freertos/task.h"
#include "esp_system.h"
#include "esp_spi_flash.h"
void app_main(void)
{
printf("Hello world!\n");
/* Print chip information */
esp_chip_info_t chip_info;
esp_chip_info(&chip_info);
printf("This is %s chip with %d CPU core(s), WiFi%s%s, ",
CONFIG_IDF_TARGET,
chip_info.cores,
(chip_info.features & CHIP_FEATURE_BT) ? "/BT" : "",
(chip_info.features & CHIP_FEATURE_BLE) ? "/BLE" : "");
printf("silicon revision %d, ", chip_info.revision);
printf("%dMB %s flash\n", spi_flash_get_chip_size() / (1024 * 1024),
(chip_info.features & CHIP_FEATURE_EMB_FLASH) ? "embedded" : "external");
printf("Minimum free heap size: %d bytes\n", esp_get_minimum_free_heap_size());
int i = 0;
while(true)
{
i++;
printf("i %d\n", i);
vTaskDelay(1000 / portTICK_PERIOD_MS);
}
}
Outline
Build Targets
stdio.h
sdkconfig.h
freertos/FreeRTOS.h
freertos/task.h
esp_system.h
esp_spi_flash.h
app_main(void) : void
Problems Tasks Console Properties
Espressif IDF Tools Console
IDF_CCACHE_ENABLE=1
IDF_PYTHON_ENV_PATH=C:\Espressif\python_env\idf4.4_py3.8_env
Configured CDT Build Environment variables. You can check Preferences > C/C++ > Build > Environment
Executing C:\Espressif\python_env\idf4.4_py3.8_env\Scripts\python.exe -m pip install websocket-client
Requirement already satisfied: websocket-client in c:\espressif\python_env\idf4.4_py3.8_env\lib\site-packages (1.3.3)
WARNING: You are using pip version 20.3.3; however, version 22.3.1 is available.
You should consider upgrading via the 'C:\Espressif\python_env\idf4.4_py3.8_env\Scripts\python.exe -m pip install --upgrade pip' command.
Configured CMake toolchain. You can check Preferences > C/C++ > CMake
Install tools completed.
Writable
Smart Insert
1:1:0


eclipse-ws-s3 - hello_world/main/hello_world_main.c - Espressif-IDE
File Edit Source Refactor Navigate Search Project Run Espressif Window Help
Project Explorer
hello_world
Binaries
Archives
build
esp_idf_components
main
hello_world_main.c
CMakeLists.txt
component.mk
CMakeLists.txt
example_test.py
Makefile
README.md
sdkconfig
sdkconfig.ci
hello_world_main.c
/* Hello World Example
This example code is in the Public Domain (or CC0 licensed, at your option.)
Unless required by applicable law or agreed to in writing, this
software is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied.
*/
#include <stdio.h>
#include "sdkconfig.h"
#include "freertos/FreeRTOS.h"
#include "freertos/task.h"
#include "esp_system.h"
#include "esp_spi_flash.h"
void app_main(void)
{
printf("Hello world!\n");
/* Print chip information */
esp_chip_info_t chip_info;
esp_chip_info(&chip_info);
printf("This is %s chip with %d CPU core(s), WiFi%s%s, ",
CONFIG_IDF_TARGET,
chip_info.cores,
(chip_info.features & CHIP_FEATURE_BT) ? "/BT" : "",
(chip_info.features & CHIP_FEATURE_BLE) ? "/BLE" : "");
printf("silicon revision %d, ", chip_info.revision);
printf("%dMB %s flash\n", spi_flash_get_chip_size() / (1024 * 1024),
(chip_info.features & CHIP_FEATURE_EMB_FLASH) ? "embedded" : "external");
printf("Minimum free heap size: %d bytes\n", esp_get_minimum_free_heap_size());
Outline
Build Targets
stdio.h
sdkconfig.h
freertos/FreeRTOS.h
freertos/task.h
esp_system.h
esp_spi_flash.h
app_main(void) : void
Problems Tasks Console Properties Terminal
COM17
I (202) cpu_start: ESP-IDF: v4.4.2-dirty
I (207) heap_init: Initializing. RAM available for dynamic allocation:
I (214) heap_init: At 3FC937D8 len 000AC828 (306 KiB): D/IRAM
I (221) heap_init: At 3FCE0000 len 0000EE34 (59 KiB): STACK/DRAM
I (227) heap_init: At 3FCF0000 len 00008000 (32 KiB): DRAM
I (233) heap_init: At 600FE000 len 00002000 (8 KiB): RTCRAM
I (240) spi_flash: detected chip: generic
I (245) spi_flash: flash io: dio
W (248) spi_flash: Detected size(8192k) larger than the size in the binary image header(2048k). Using the size in the binary image header.
I (262) sleep: Configure to isolate all GPIO pins in sleep state
I (269) sleep: Enable automatic switching of GPIO sleep configuration
I (276) cpu_start: Starting scheduler on PRO CPU.
I (0) cpu_start: Starting scheduler on APP CPU.
Hello world!
This is esp32s3 chip with 2 CPU core(s), WiFi/BLE, silicon revision 0, 2MB external flash
Minimum free heap size: 389092 bytes
i 1
i 2
i 3
i 4
Connected - Encoding: windows-1252

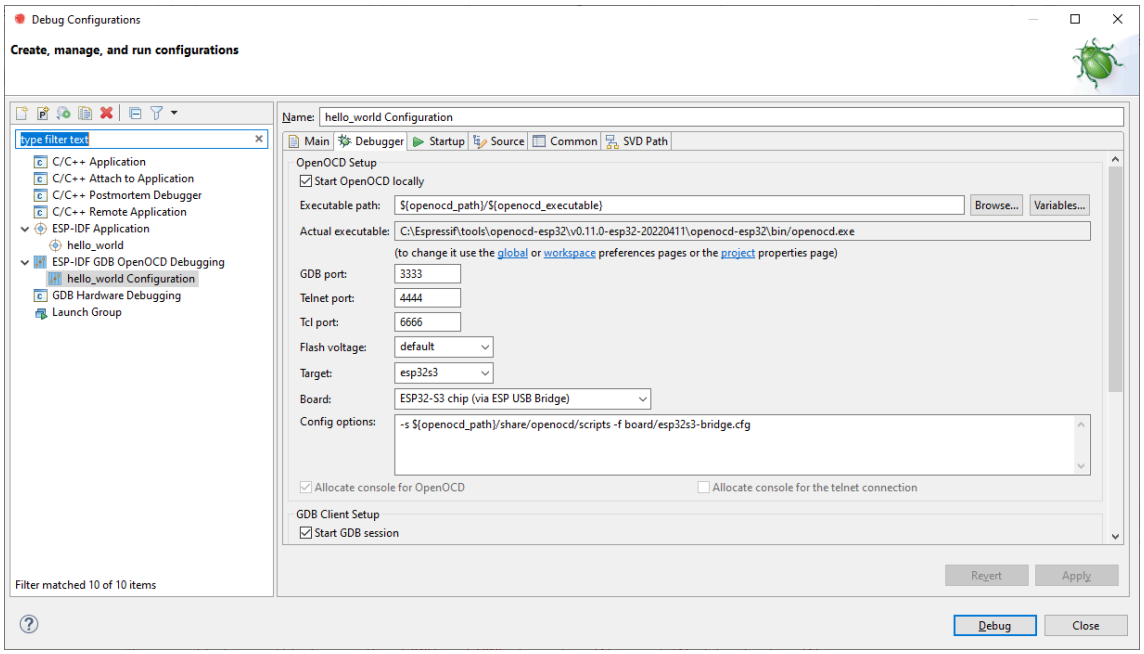




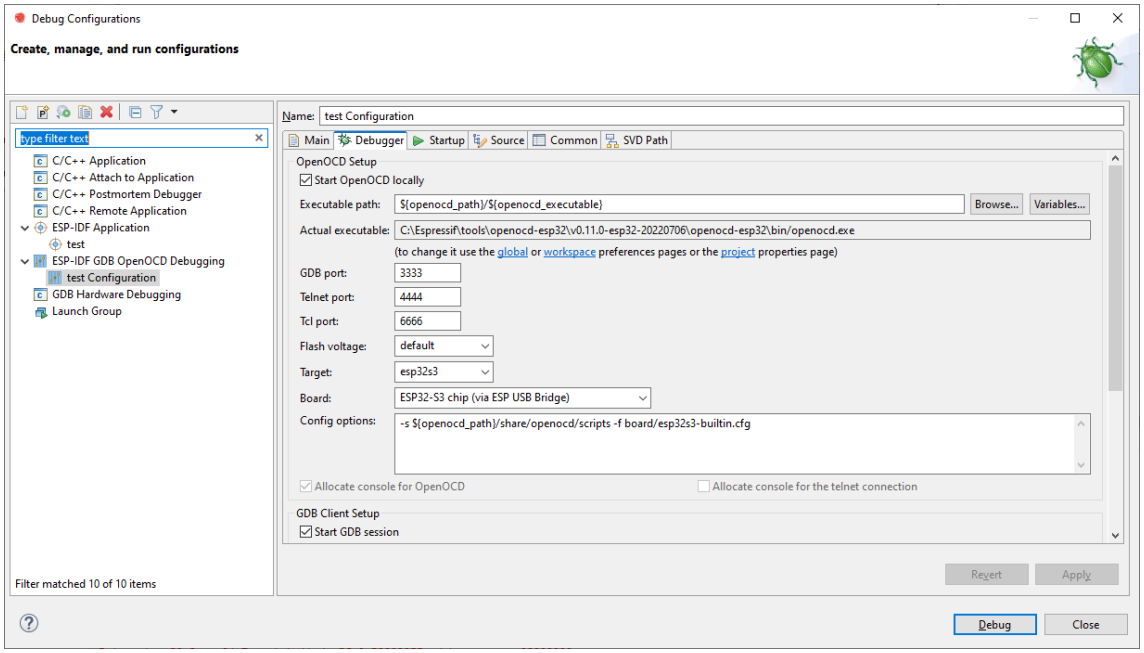
Then it won't work; the following setting must be changed. will be:



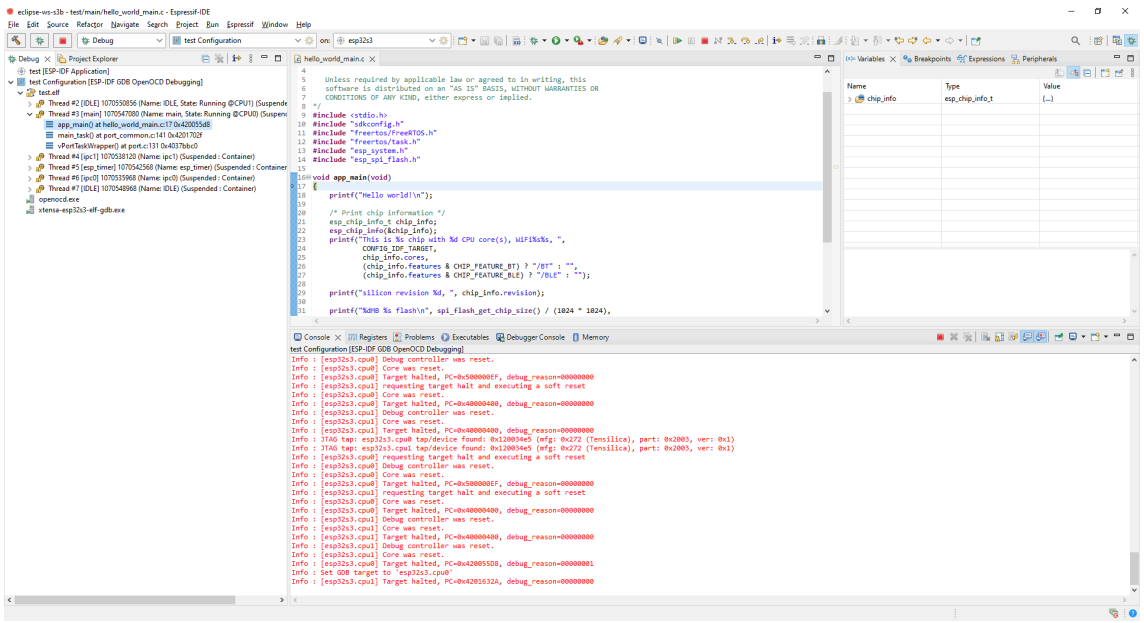
ESP32-S3 chip (via ESP USB Bridge)



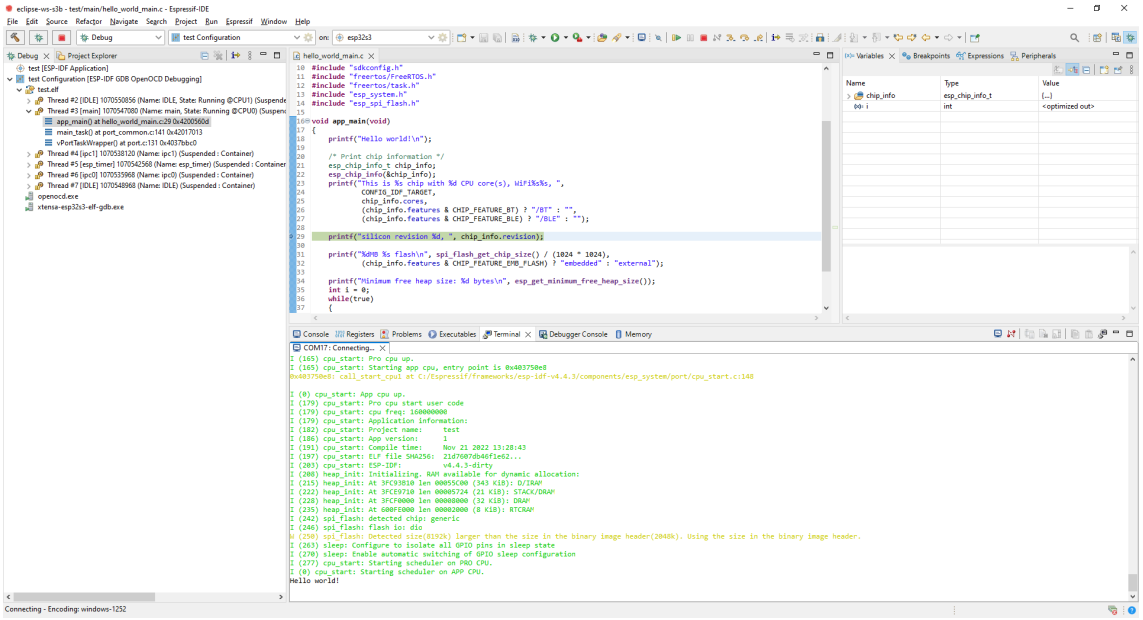
and then change “bridge” to “builtin”:



works:



Here too, the console can run in parallel with debugging,
via a USB cable,
cool:



15.1 🌐 Relevant topic subpages on ms-muc-docs.de

- 🌐 ESP32 & ESP32-S3 DevKit on ms-muc-docs.de

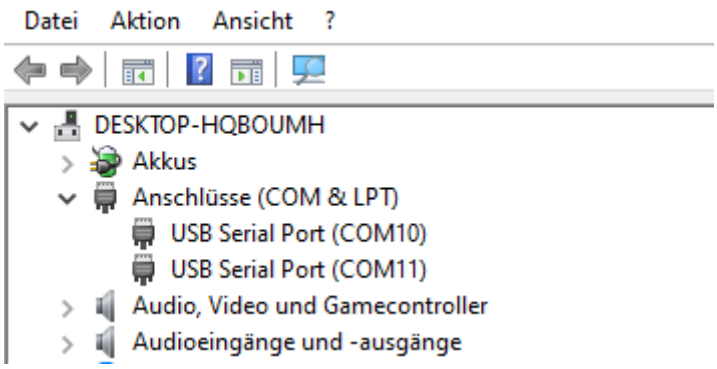
16. JTAG

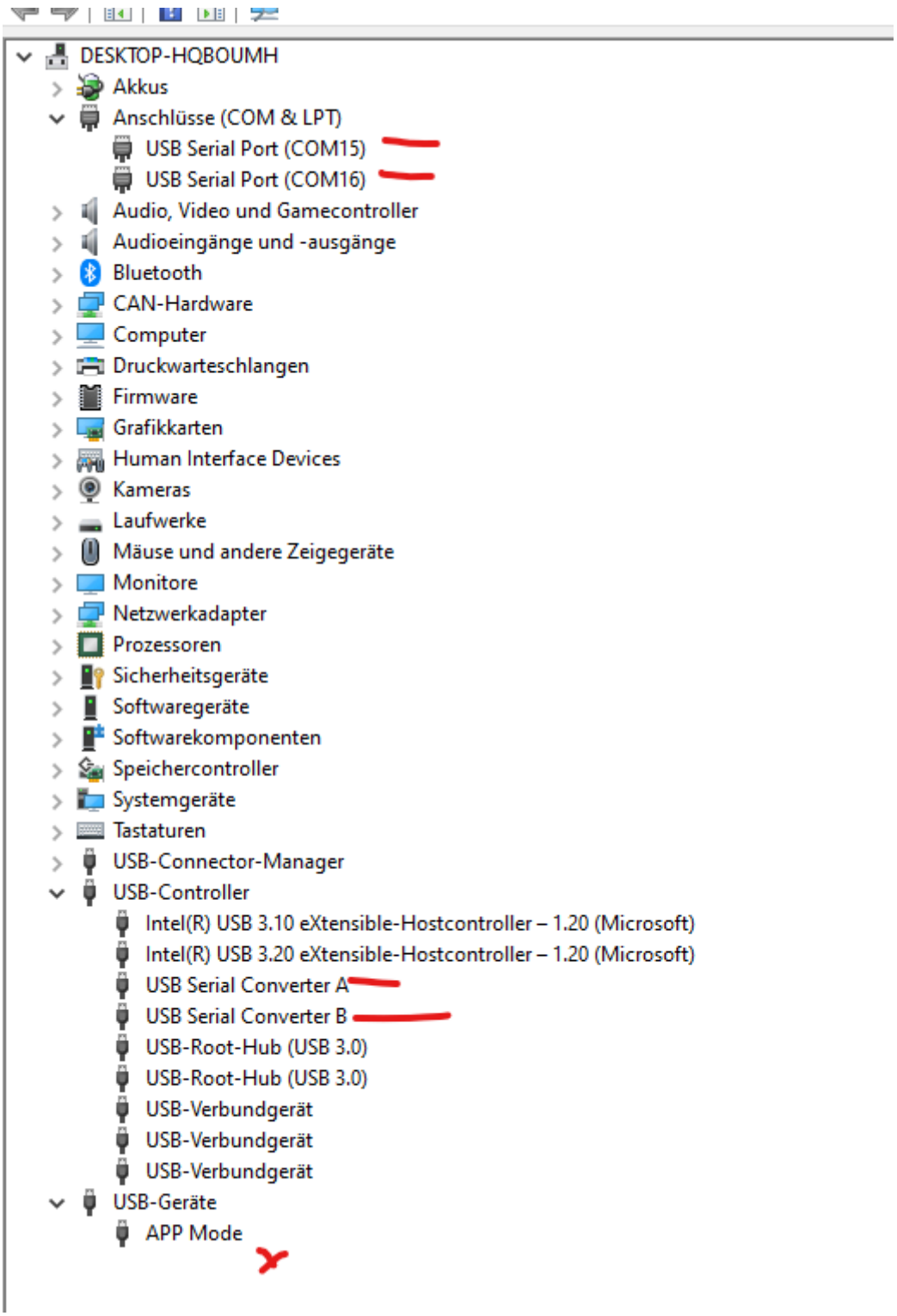
16.1 JTAG Step 01

Now we'll set up JTAG. https://docs.espressif.com/projects/esp-idf/en/v4.4.3/esp32/api-guides/jtag-debugging/configure-ft2232h-jtag.html https://docs.espressif.com/projects/espressif-esp-iot-solution/en/latest/hw-reference/ESP-Prog_guide.html https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/jtag-debugging/index.html

16.1.1 Switching the USB interface with ZADIG

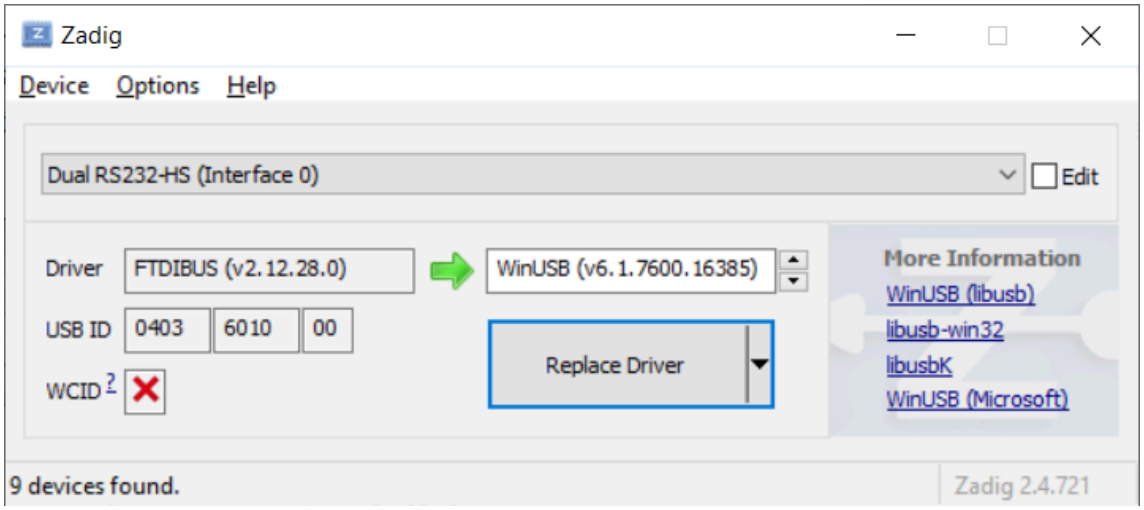
16.1.1.1 Before



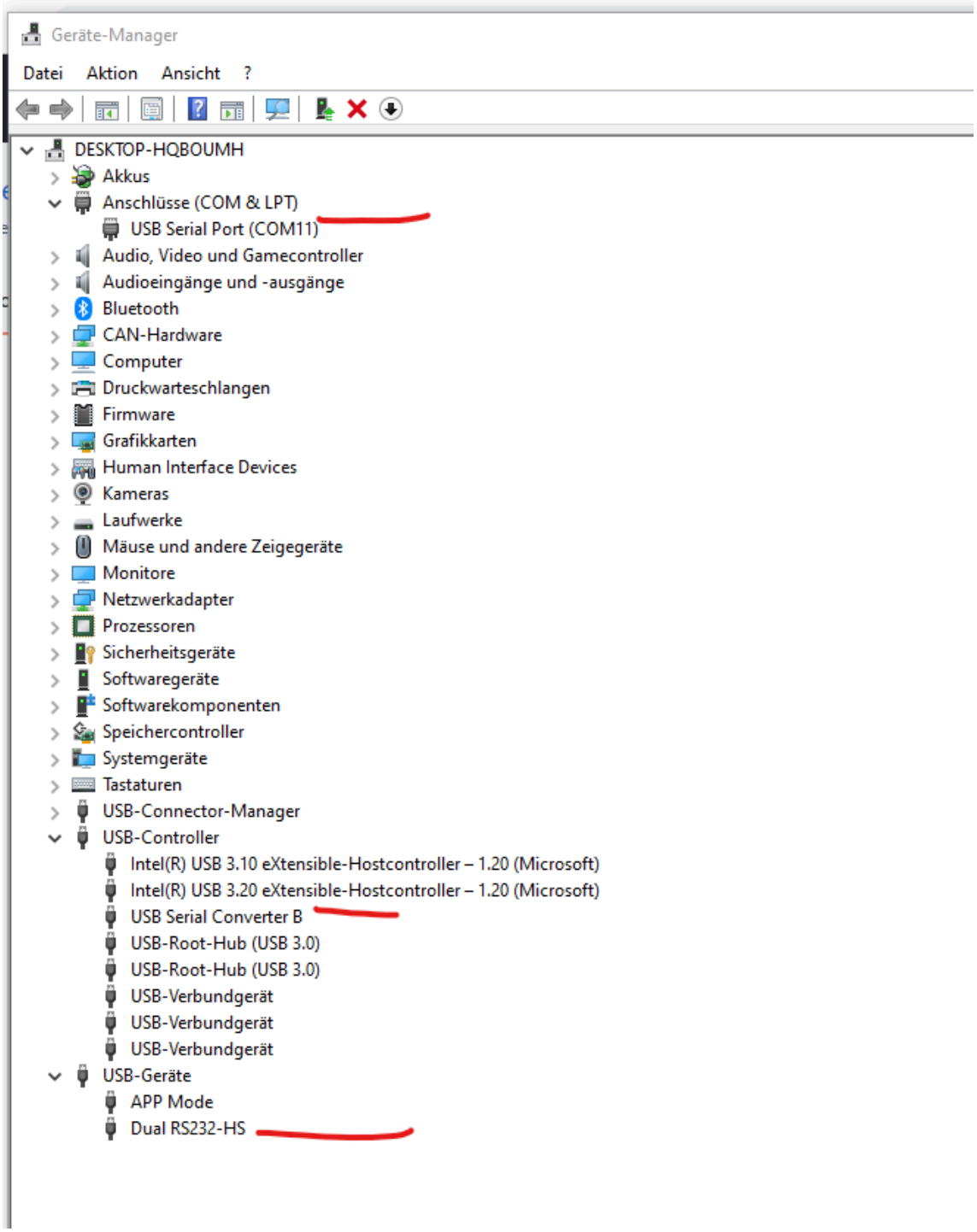


16.1.1.2 Calling ZADIG

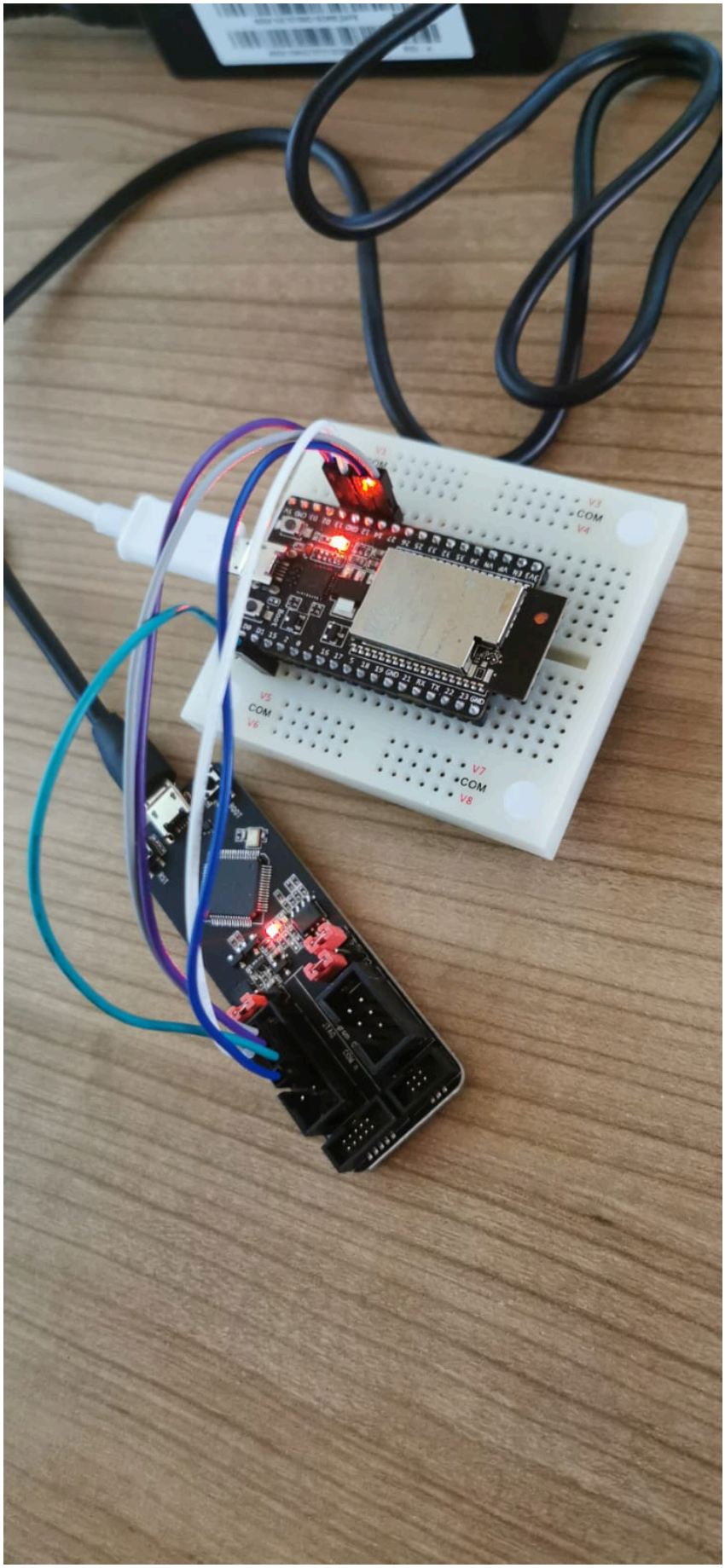
See also: https://docs.espressif.com/projects/esp-idf/en/v4.4.3/esp32/api-guides/jtag-debugging/configure-ft2232h-jtag.html#windows


Zadig
Device
Options
Help
Dual RS232-HS (Interface 0)
Edit
Driver
FTDIBUS (v2.12.28.0)
WinUSB (v6.1.7600.16385)
USB ID
0403
6010
00
WCID
Replace Driver
More Information
WinUSB (libusb)
libusb-win32
libusbK
WinUSB (Microsoft)
9 devices found.
Zadig 2.4.721

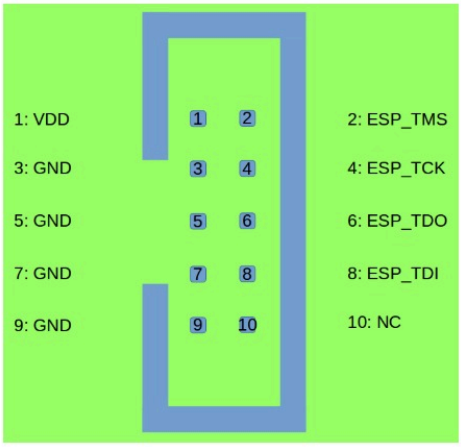
16.1.1.3 After



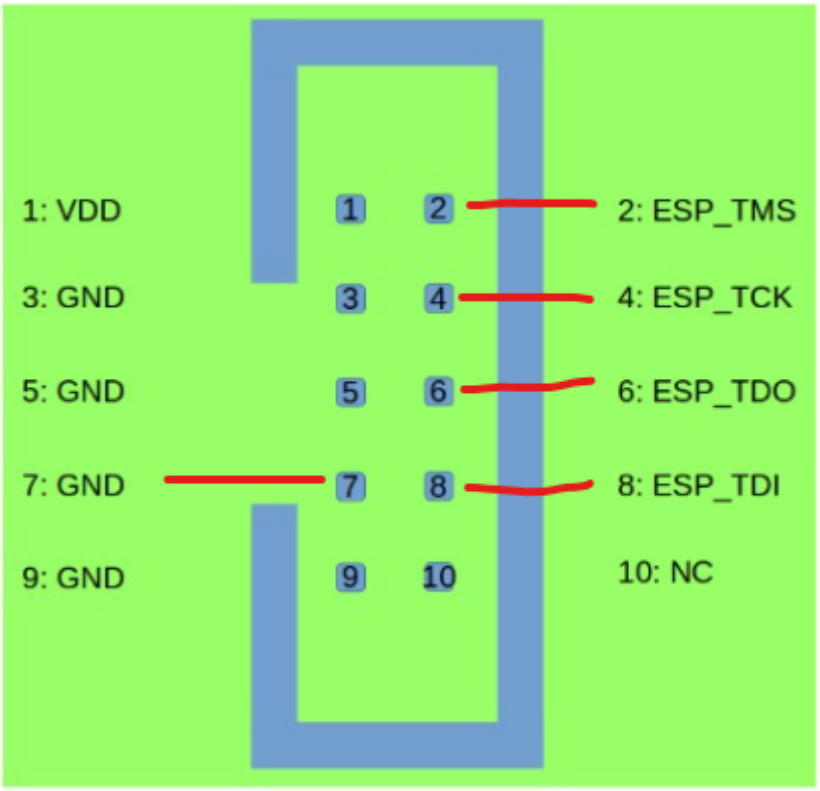
Note: This process must be repeated if you connect the ESP-PROG to a different interface. Therefore, ideally, always use the same USB ports.


V3
COM
V4
V5
COM
V6
V7
COM
V8

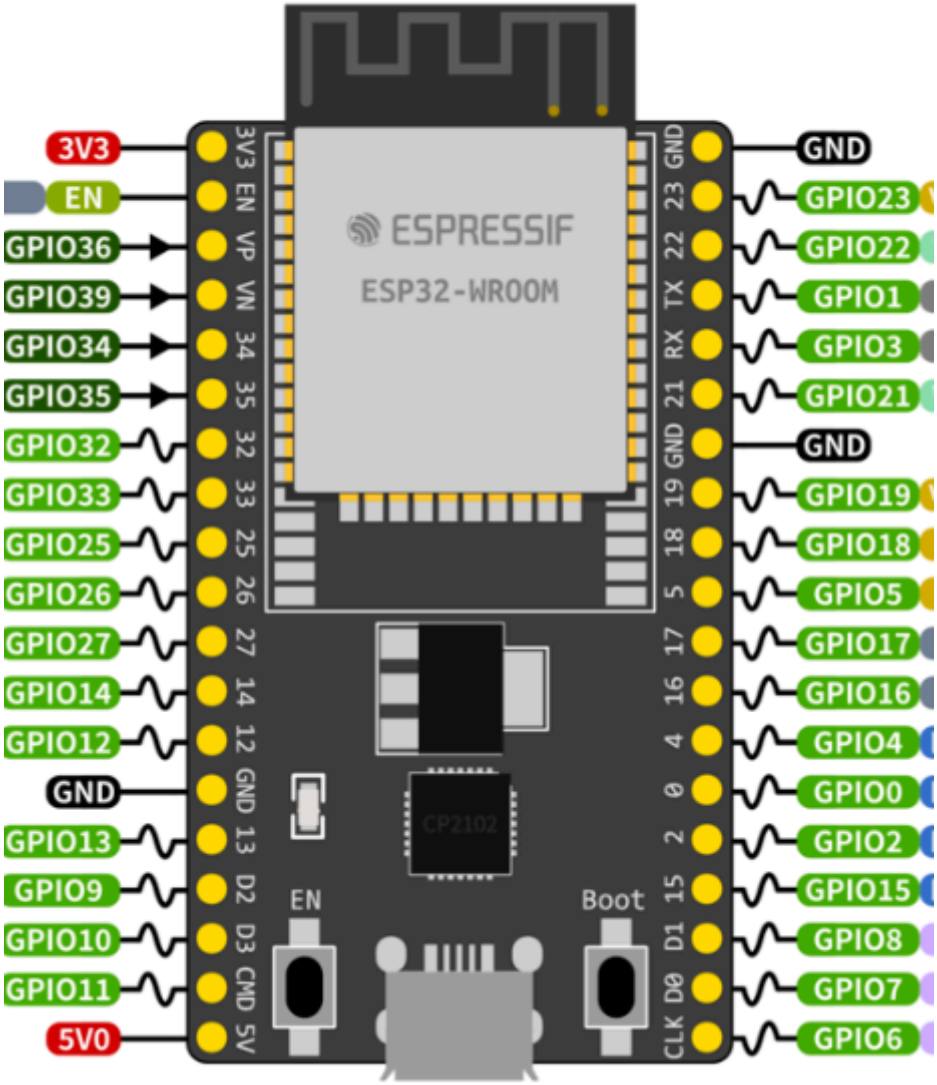
- **JTAG Interface**
  The design for the JTAG interface on the user board should follow the reference provided in the figure below.

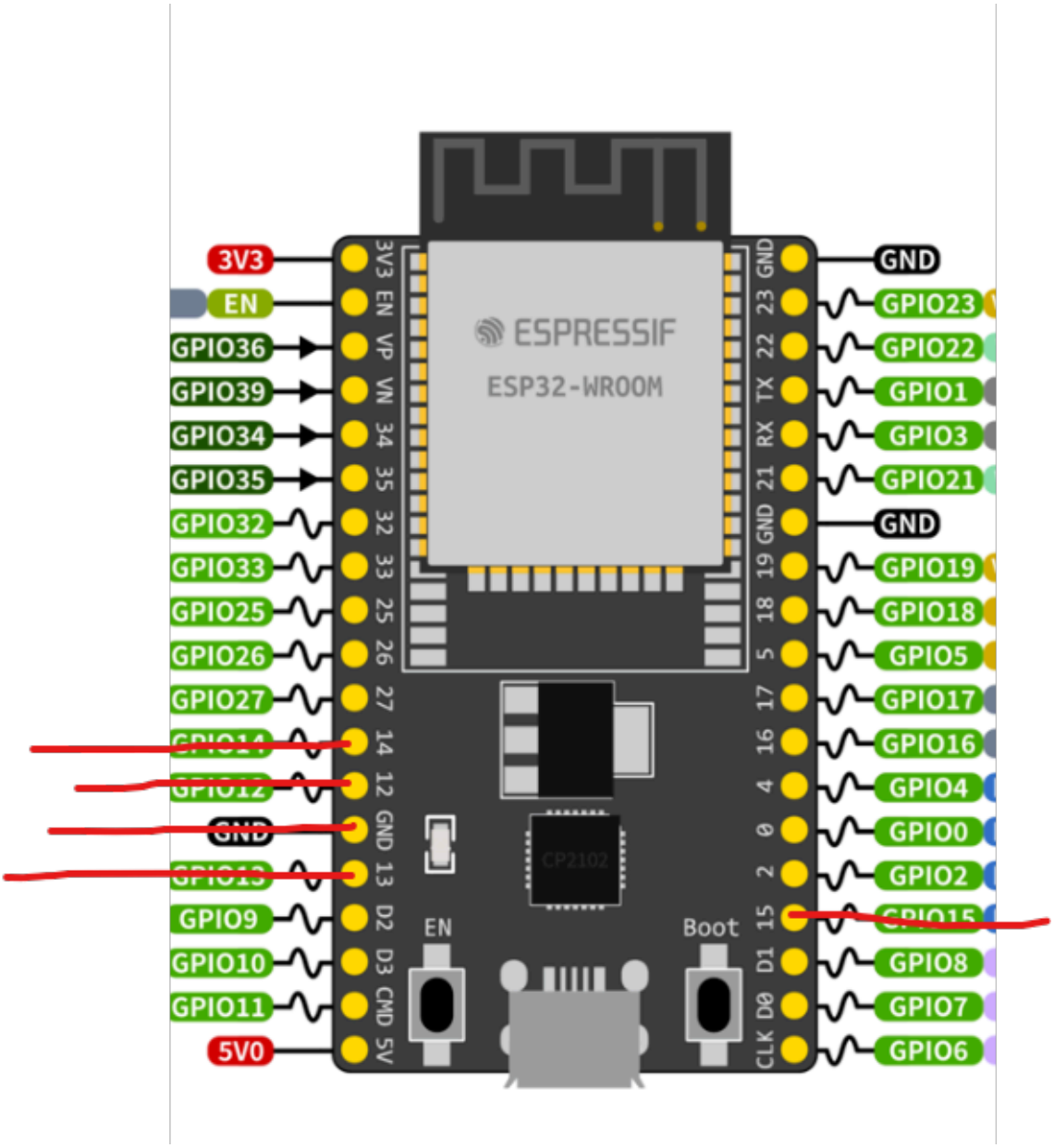


These pins are required at a minimum:



	ESP32 Pin	JTAG Signal
1	CHIP_PU	TRST_N
2	MTDO / GPIO15	TDO
3	MTDI / GPIO12	TDI
4	MTCK / GPIO13	TCK
5	MTMS / GPIO14	TMS
6	GND	GND

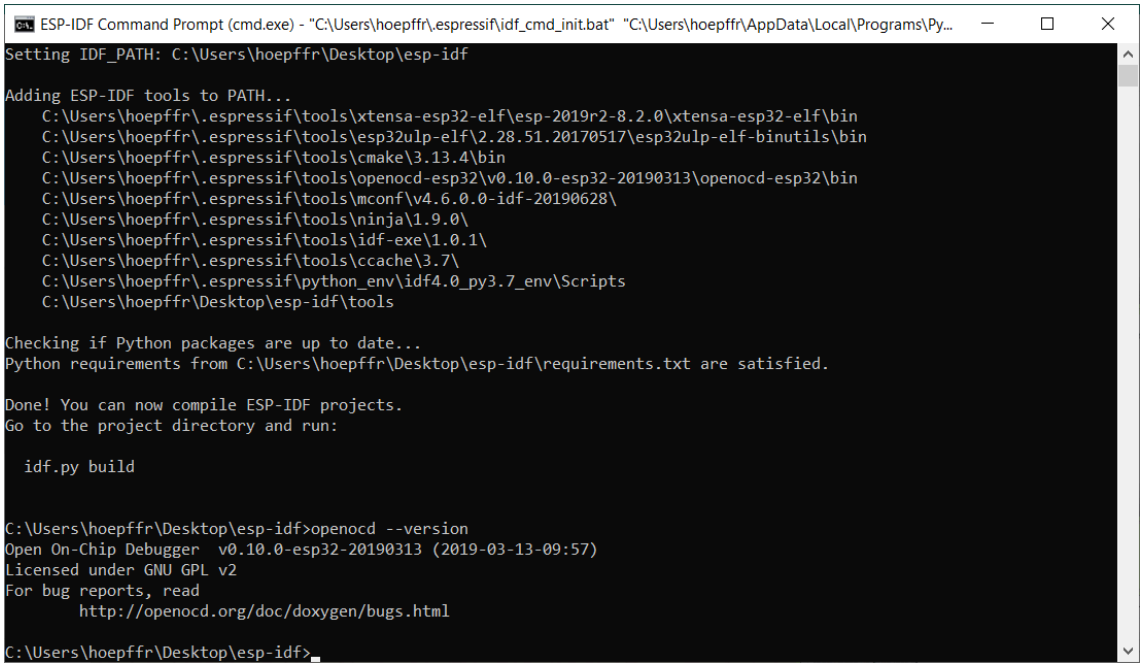

3V3
EN
GPIO36
GPIO39
GPIO34
GPIO35
GPIO32
GPIO33
GPIO25
GPIO26
GPIO27
GPIO14
GPIO12
GND
GPIO13
GPIO9
GPIO10
GPIO11
5V0
3V3 EN VP VN 34 35 32 33 25 26 27 14 12 GND 13 D2 D3 CMD 5V
GND
GPIO23
GPIO22
GPIO1
GPIO3
GPIO21
GND
GPIO19
GPIO18
GPIO5
GPIO17
GPIO16
GPIO4
GPIO0
GPIO2
GPIO15
GPIO8
GPIO7
GPIO6
GND 23 22 TX RX 21 GND 19 18 5 17 16 4 0 2 15 D1 D0 CLK
ESPRESSIF
ESP32-WROOM
CP2102
EN
Boot



https://docs.espressif.com/projects/esp-idf/en/latest/api-guides/jtag-debugging/configure-other-jtag.html

16.2 JTAG Step 2

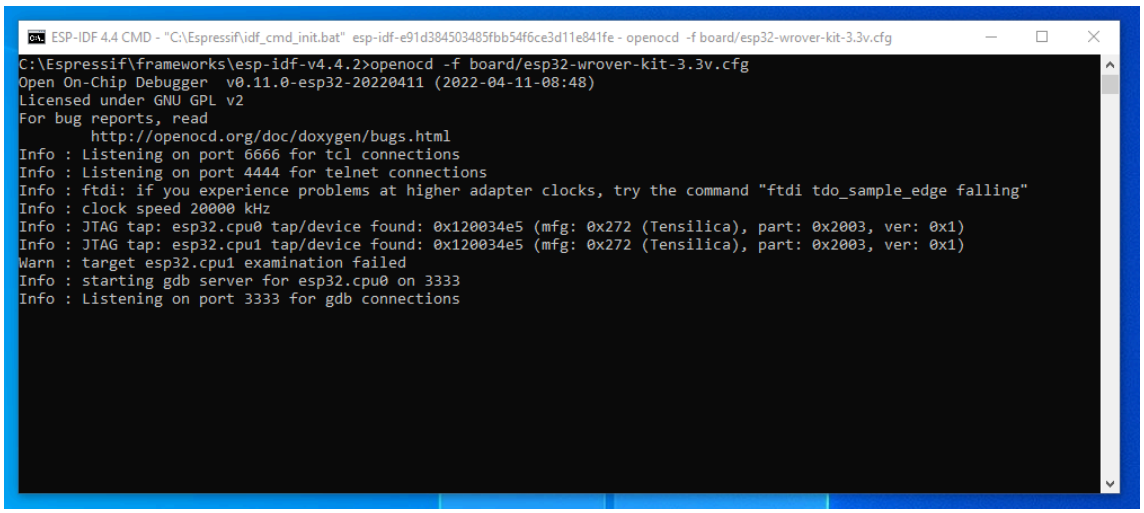
openocd –version



16.3 JTAG Step 3

https://docs.espressif.com/projects/esp-idf/en/v4.4.3/esp32/api-guides/jtag-debugging/index.html#run-openocd

openocd -f board/esp32-wrover-kit-3.3v.cfg



ATTENTION!!! Starting the GDB server from the command line is only necessary if you are also debugging from the command line.

–> OpenOCD is started automatically from within the ESP-IDF.

16.4 JTAG Step_04

At this point, the documentation is hopelessly outdated.

If you are using the Espressif IDE, these links are no longer valid.

https://docs.espressif.com/projects/esp-idf/en/latest/api-guides/jtag-debugging/debugging-examples.html#jtag-debugging-examples-eclipse

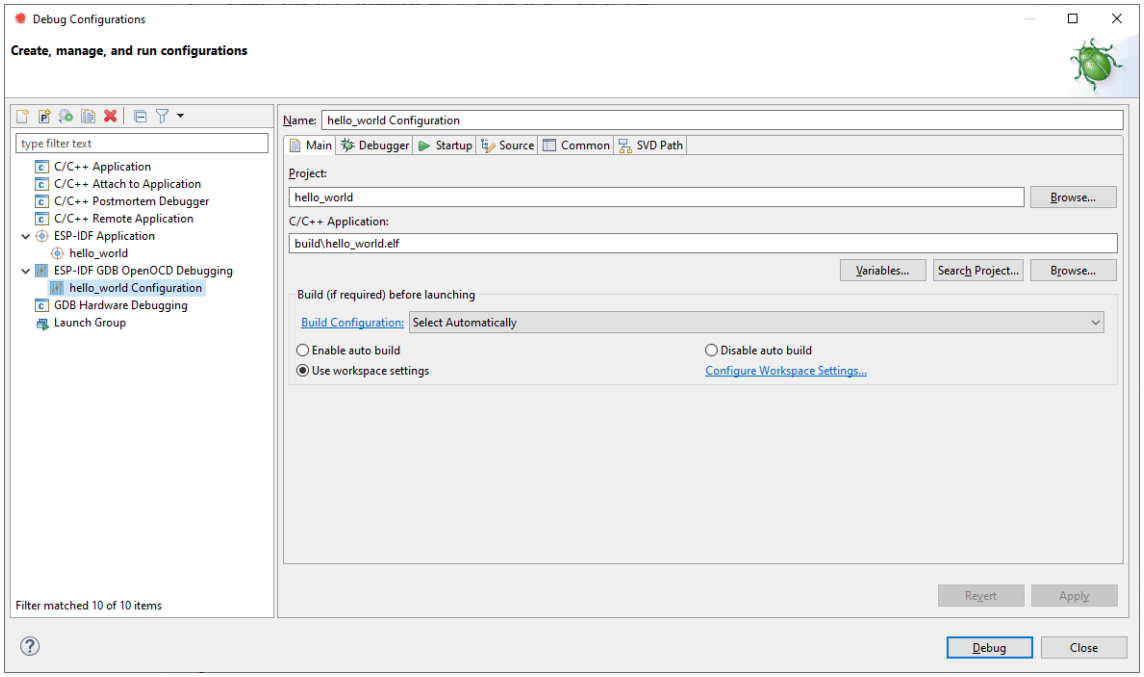
https://docs.espressif.com/projects/esp-idf/en/latest/api-guides/jtag-debugging/using-debugger.html#jtag-debugging-using-debugger-eclipse

I was able to debug immediately as follows:

1. Select only ESP-IDF GDB..., then click on the



icon in the upper left,
and start it.
No settings need to be changed.



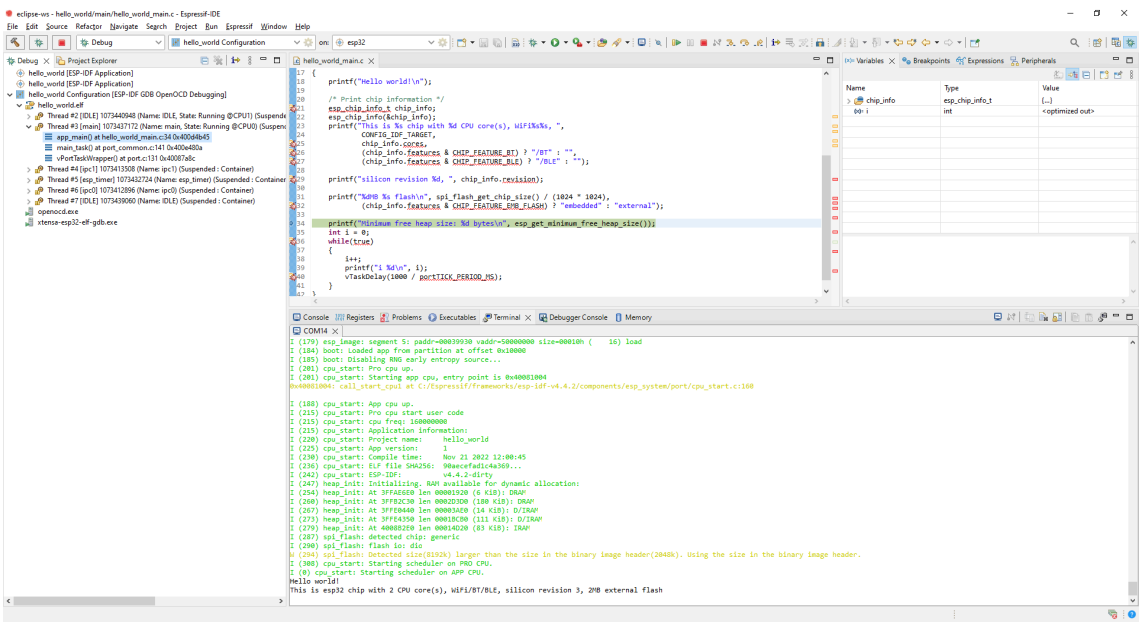
16.5 JTAG Step_05

Not applicable

16.6 JTAG Step_06

Not applicable

16.7 JTAG Step_07



And yes, you can run the console alongside the debugging session,
but the console must be open before the debugging session starts.

17. Joystick

Now try controlling the ISOBUS machine you just created using the AUX joystick. Try this with both the tractor monitor and the auxiliary screen. Note your findings; we will discuss them later in class.

18. Hardware Cost Calculation

Our ECU is based on the ATOM from M5 Stack. Schematic: https://github.com/Meisterschulen-am-Ostbahnhof-Munchen/ISOBUS_Hardware] Bill of materials: see above. The schematic was created in EAGLE; you will receive it as is.

> As a student or teacher, you are entitled to an equivalent of the premium version of Autodesk EAGLE for non-commercial teaching purposes.
>
> Activate a free, individual educational license via the Education Community at TODO: Link

Here is the cost calculation:

1. Cost calculation excluding additional costs:

| Item No. | Description | Quantity | Unit Price | Total Price | Link | | — | — | — | — | — | — | — | | USB cable to Type-C USB 2.0 black | 1 | | €0.99 | €0.99 | | C008 | ATOM Lite ESP32 | 1 | $5.95 | €5.10 | €5.10 | atom-lite-esp32 | | A077 | ATOMIC Proto Kit | 1 | $3.50 | €3.00 | €3.00 | atomic-proto-kit | | R-78E5.0-0.5 | DC-DC CONVERTER 5V 0.5A | 1 | | €2.29 | €2.29 | 919-R-78E5.0-0.5 | | TMK325ABJ476MM-T | 1210 25VDC 47uF 20% X5R | 1 | | €1.05 | €1.05 | 963-TMK325ABJ476MM-T | | TLE9250VSJ | CAN IC Interface | 1 | | €0.889 | €0.889 | 726-TLE9250VSJXUMA1 | | SS12L R3 | 1A 20V Schottky | 1 | | €0.39 | €0.39 | 821-SS12LR3 | | 824501161 | 400W Bidir. 16V TVS | 1 | | €0.214 | €0.214 | 824501161 | | | circuit board | 1 | | 0.2235€ | 0.2235€ | Atomic_V1.0_2020-10-15.zip | | | 60.4R | 2 | | €0.005 | €0.010 | CRCW040260R4FKED | | | 47nF | 1 | | €0.008 | €0.016 | TMF105B7473KVHF | | | 100nF | 2 | | €0.006 | €0.012 | GCM155R71C104KA55D | | 824098024 or PESD2CAN | WE-TVS_SOT23-3L | 1 | | €0.076 | €0.076 | PESD2CAN215 |

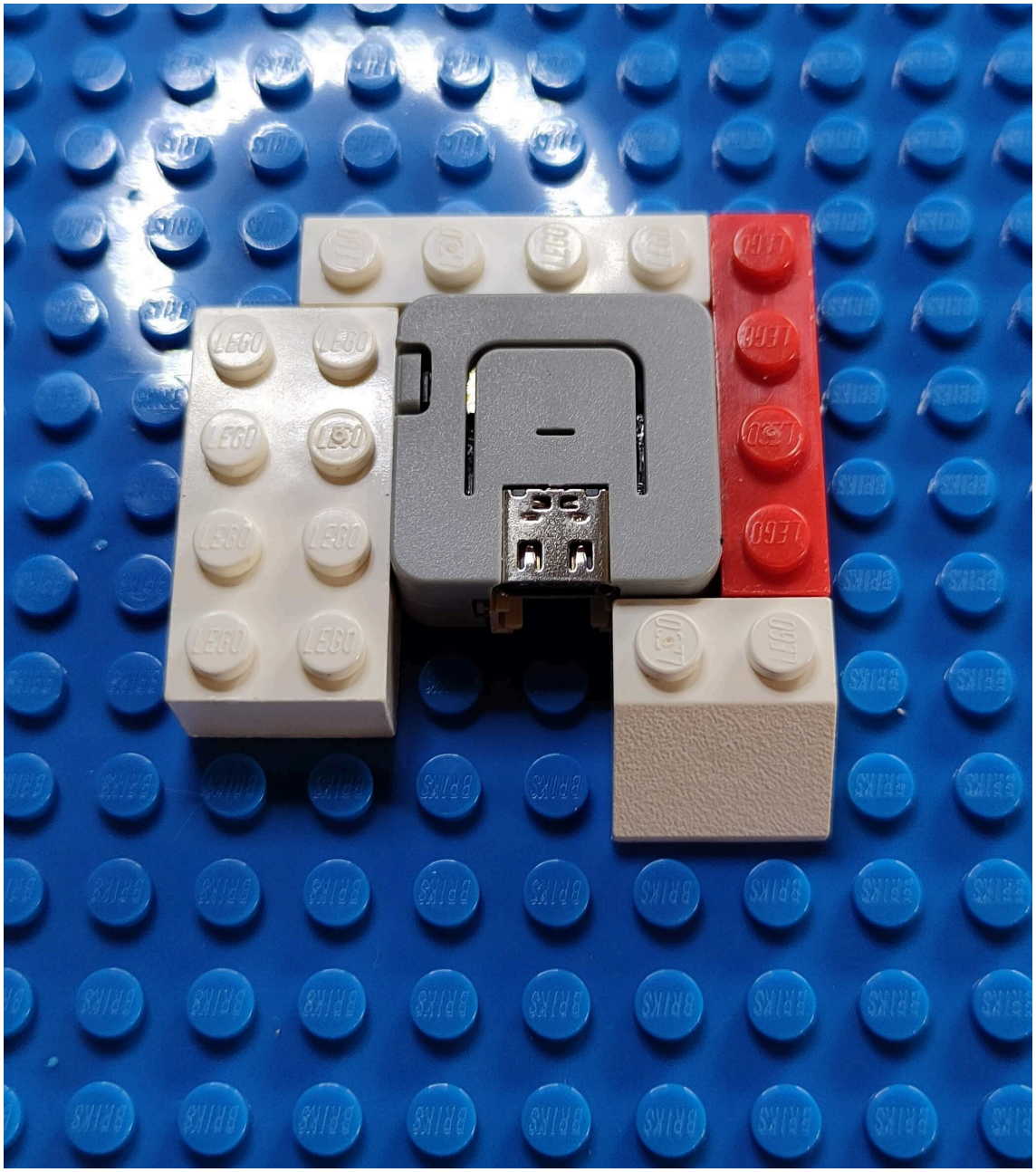
–

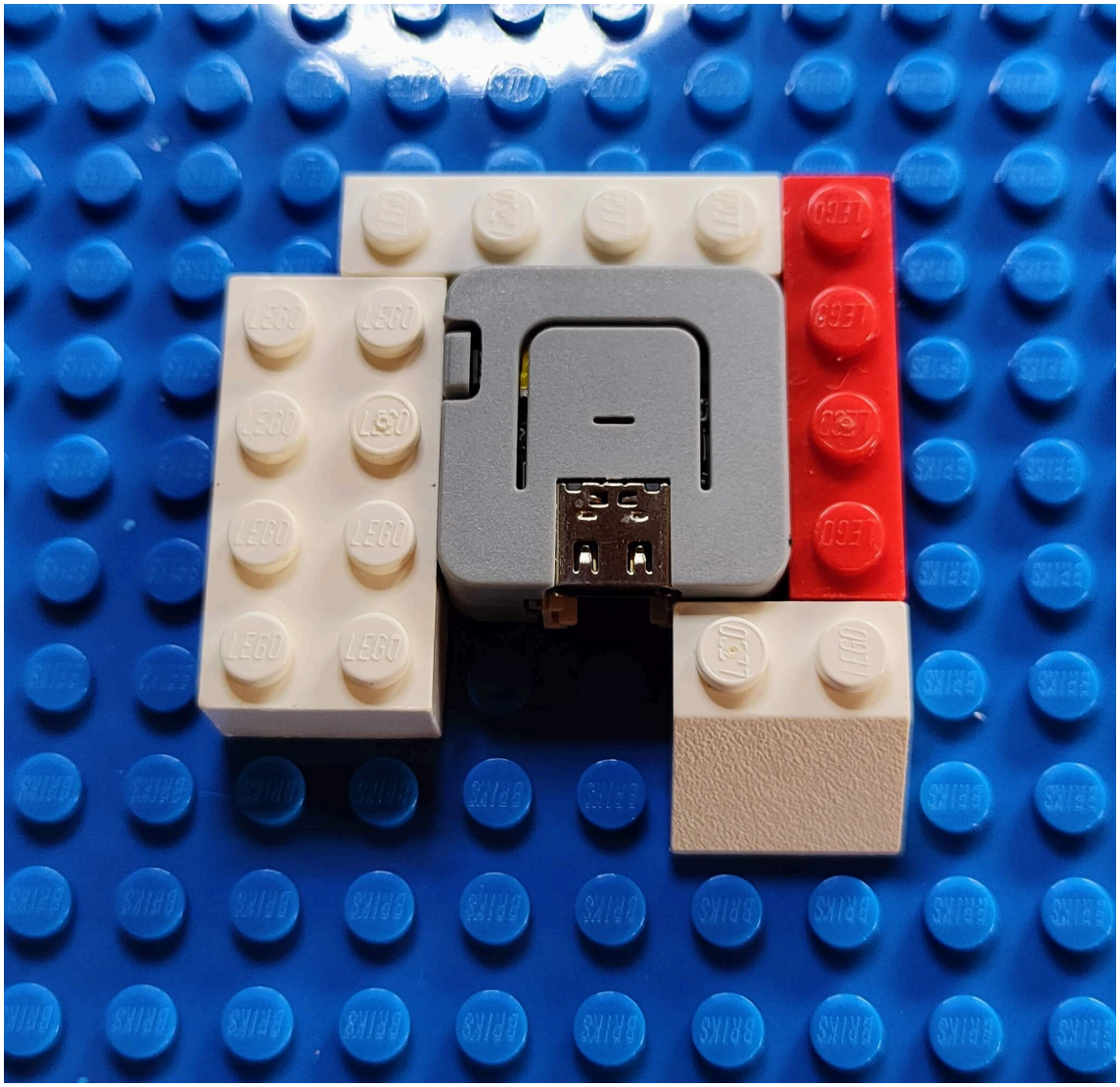
18.1 🌐 Related topic subpages on ms-muc-docs.de

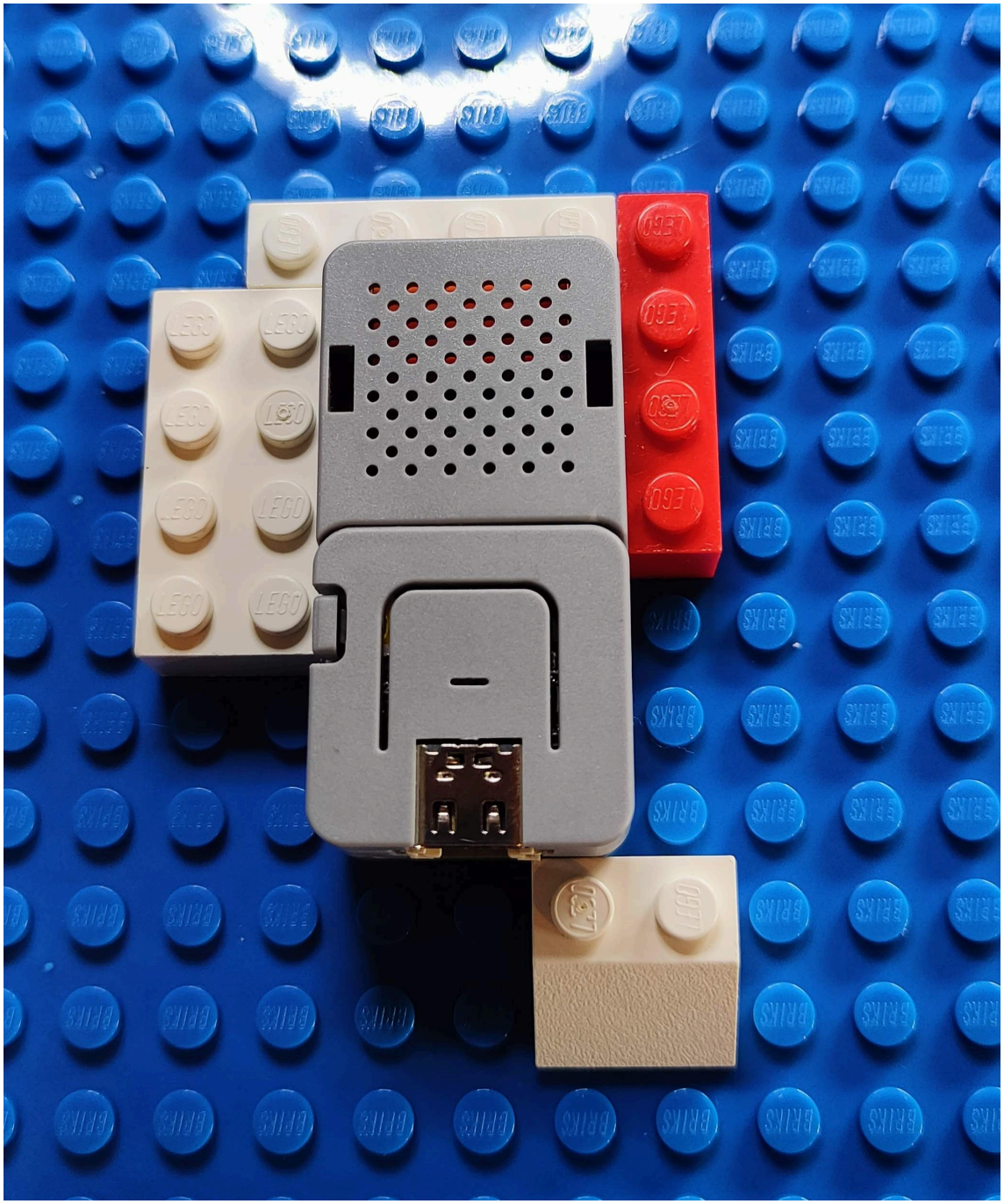
- 🌐 ESP32 & ESP32-S3 DevKit on ms-muc-docs.de

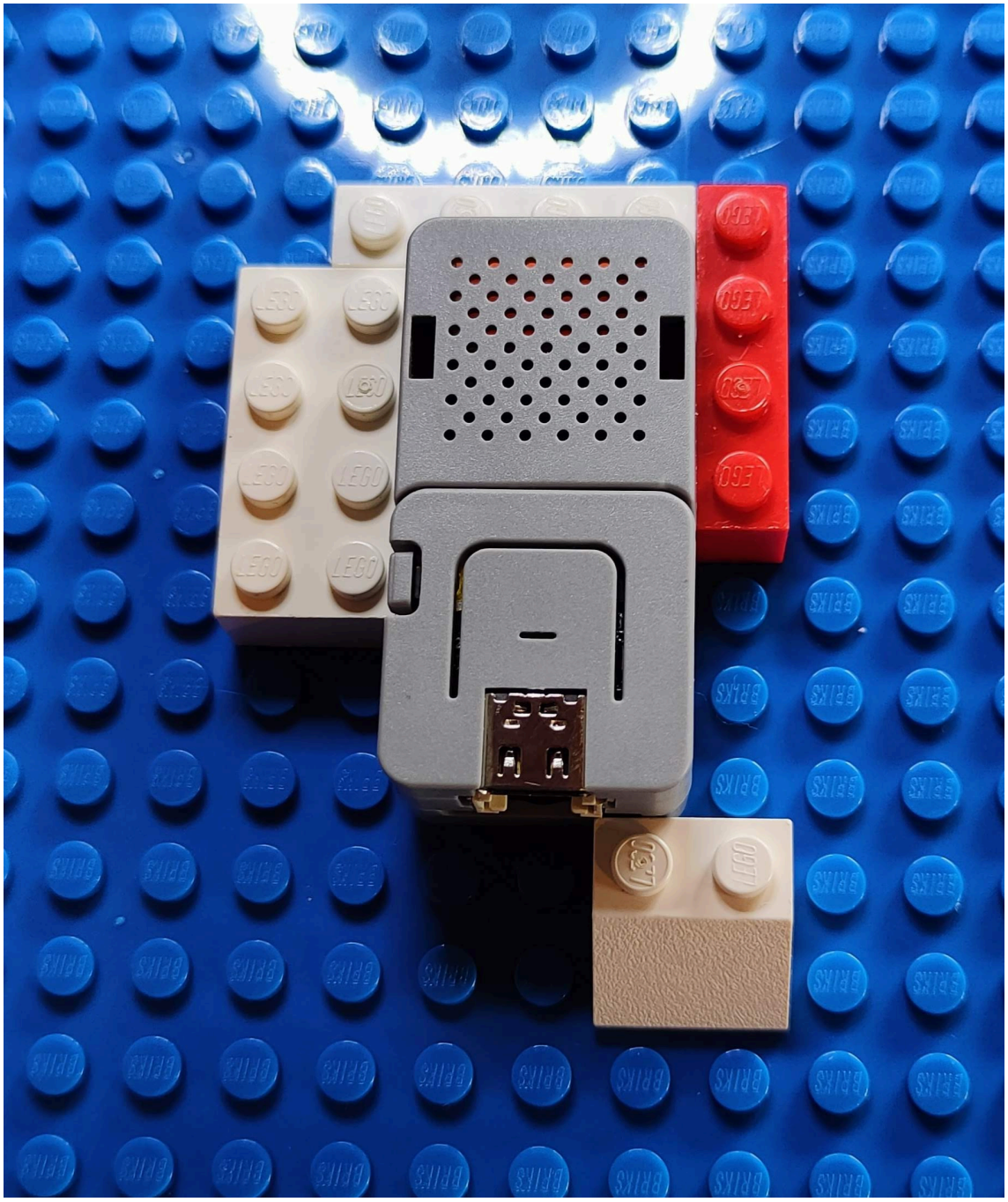
19. Lego

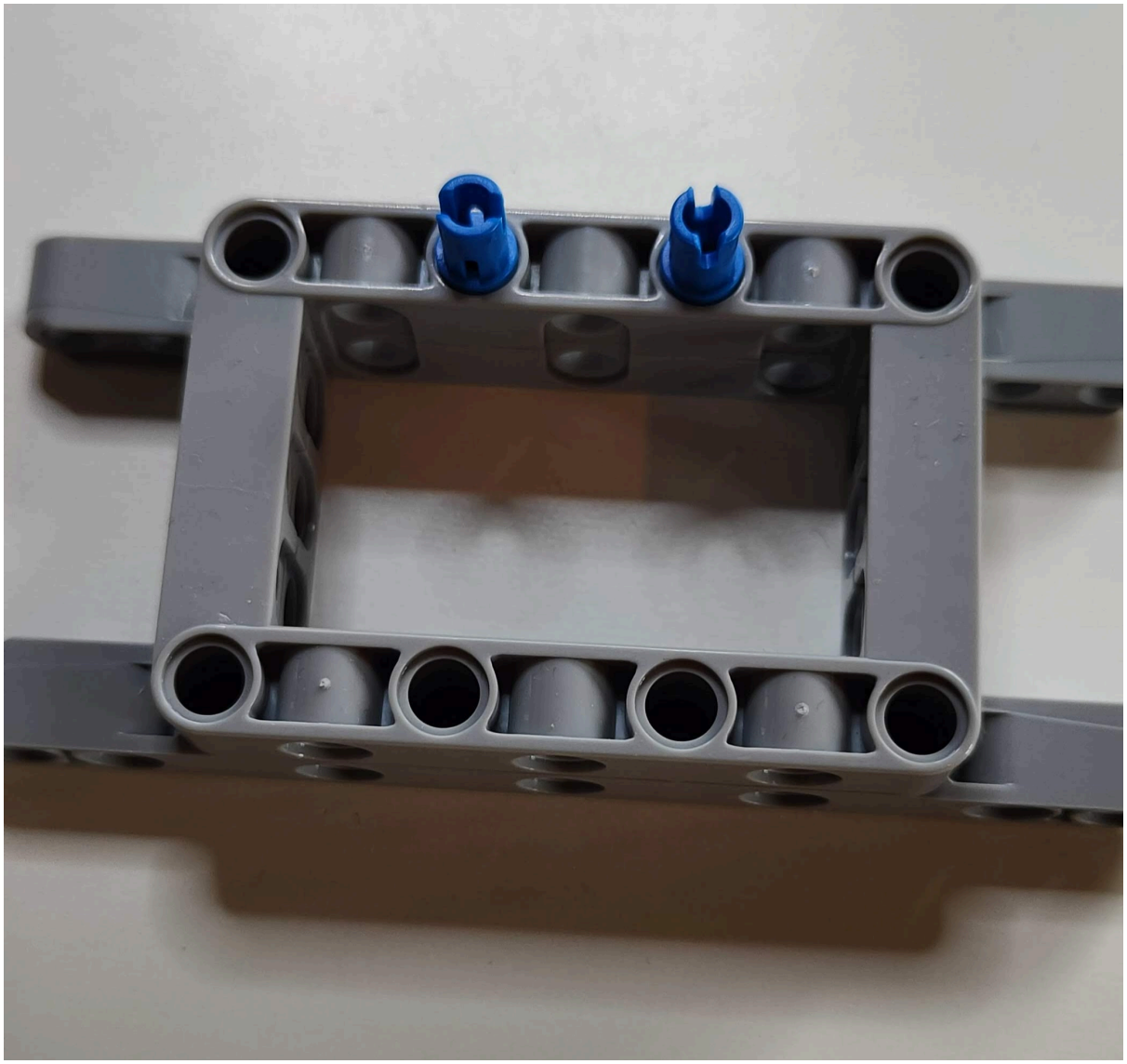
<https://www.1000steine.de/de/gemeinschaft/forum/?entry=1&id=442294 <https://youtu.be/z4hOIl2aygs

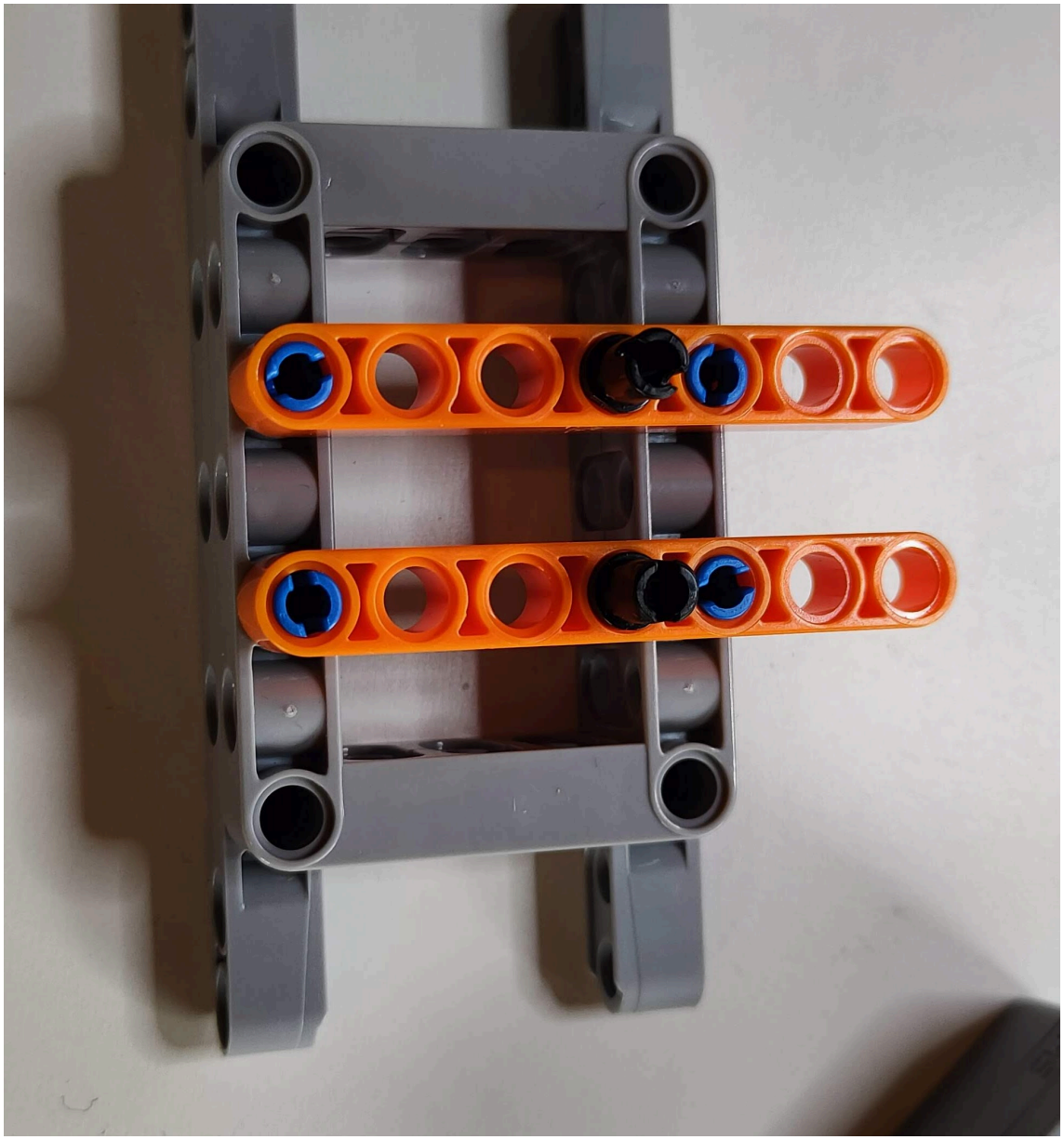




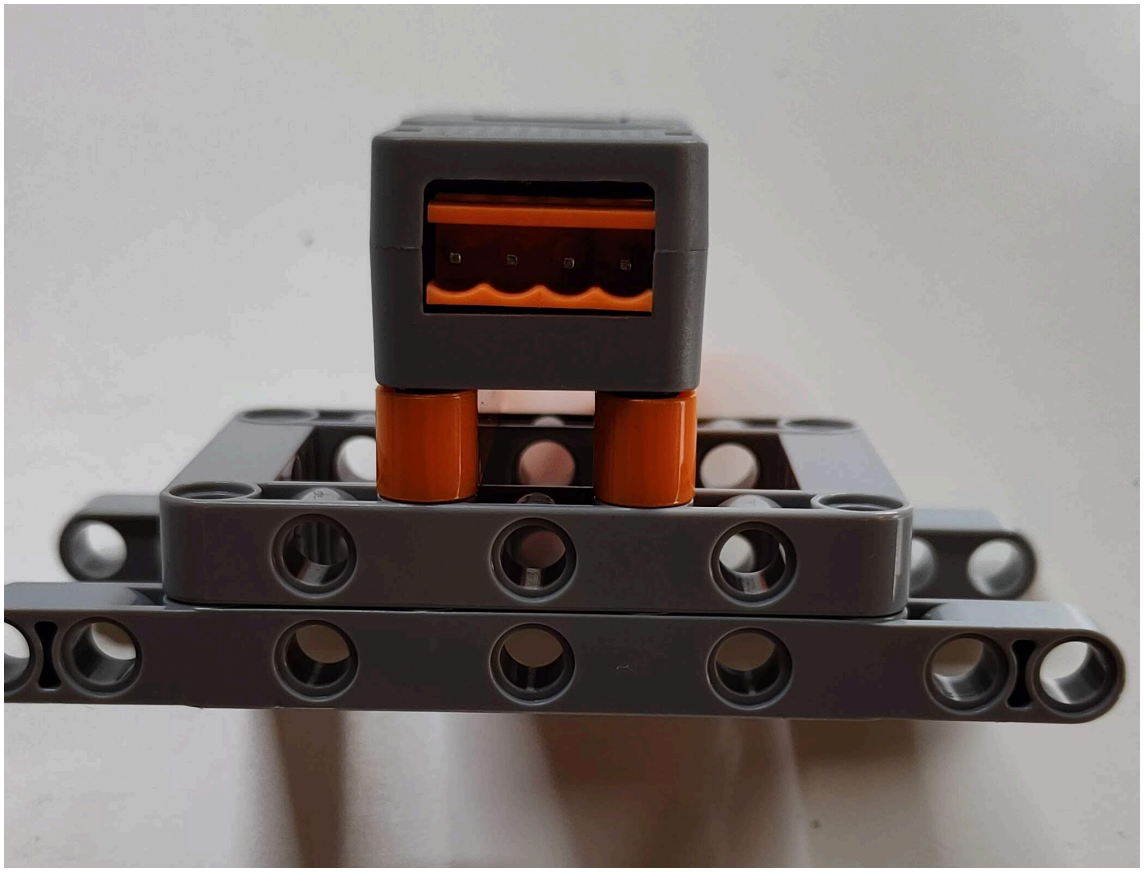


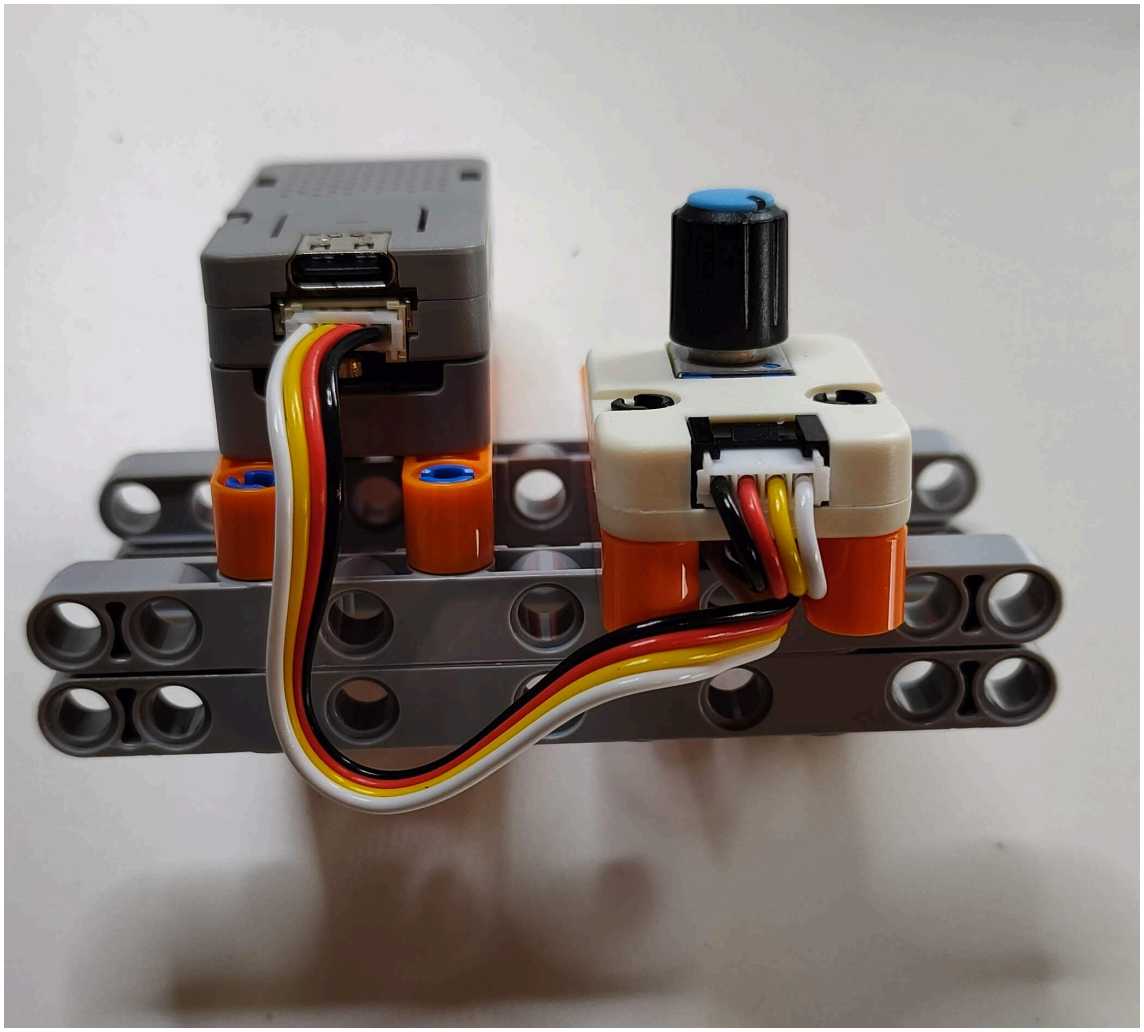


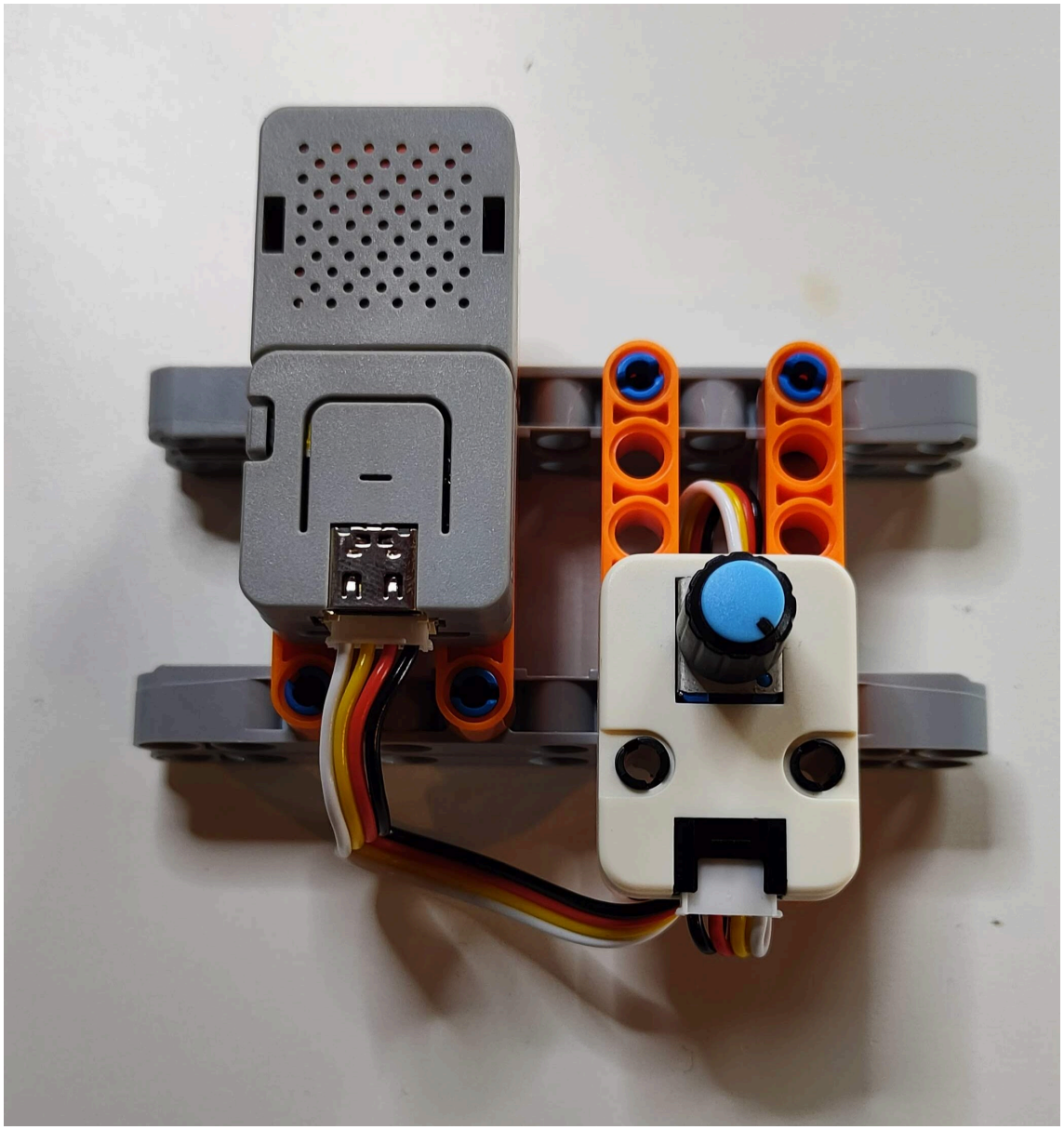


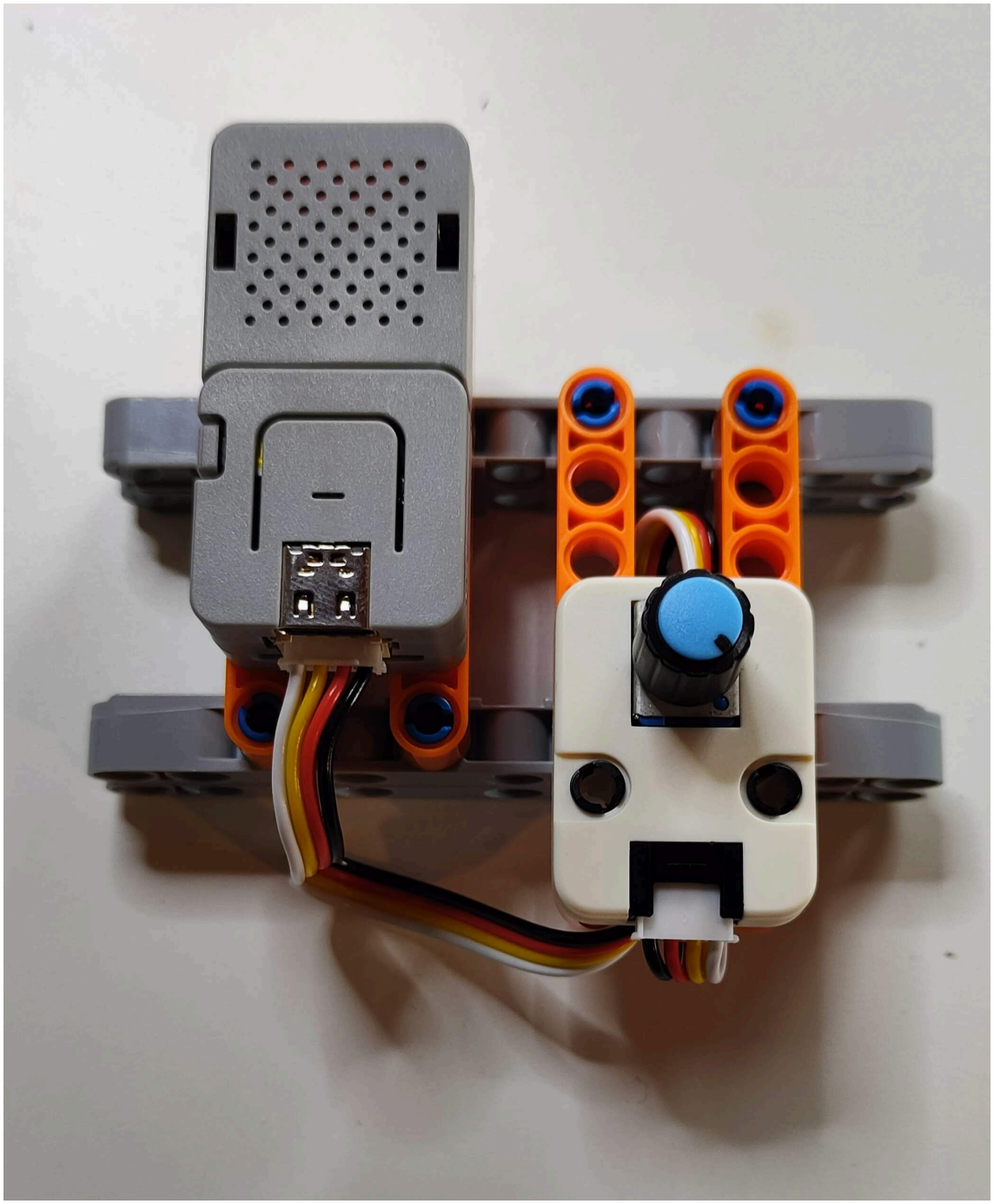


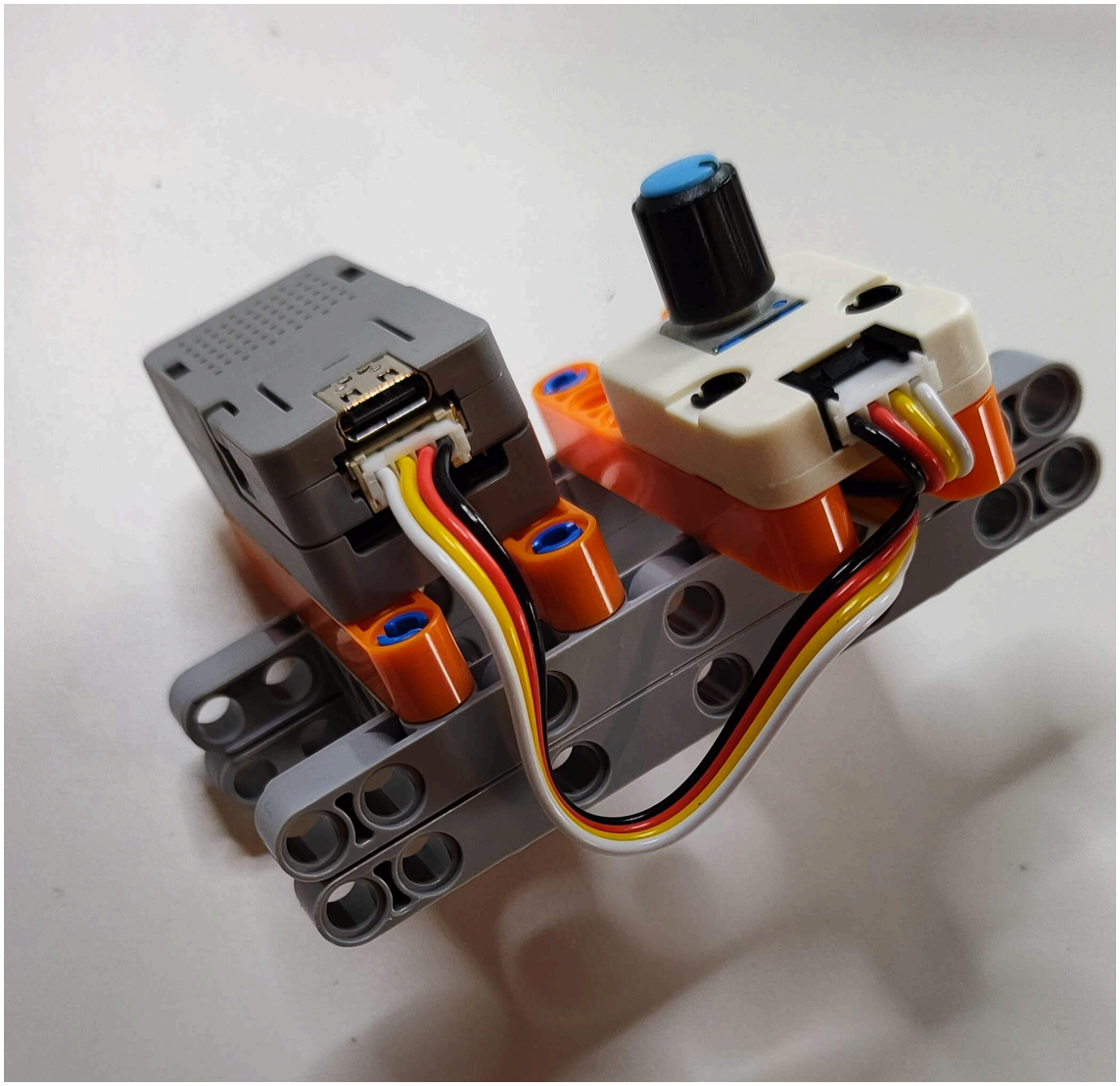




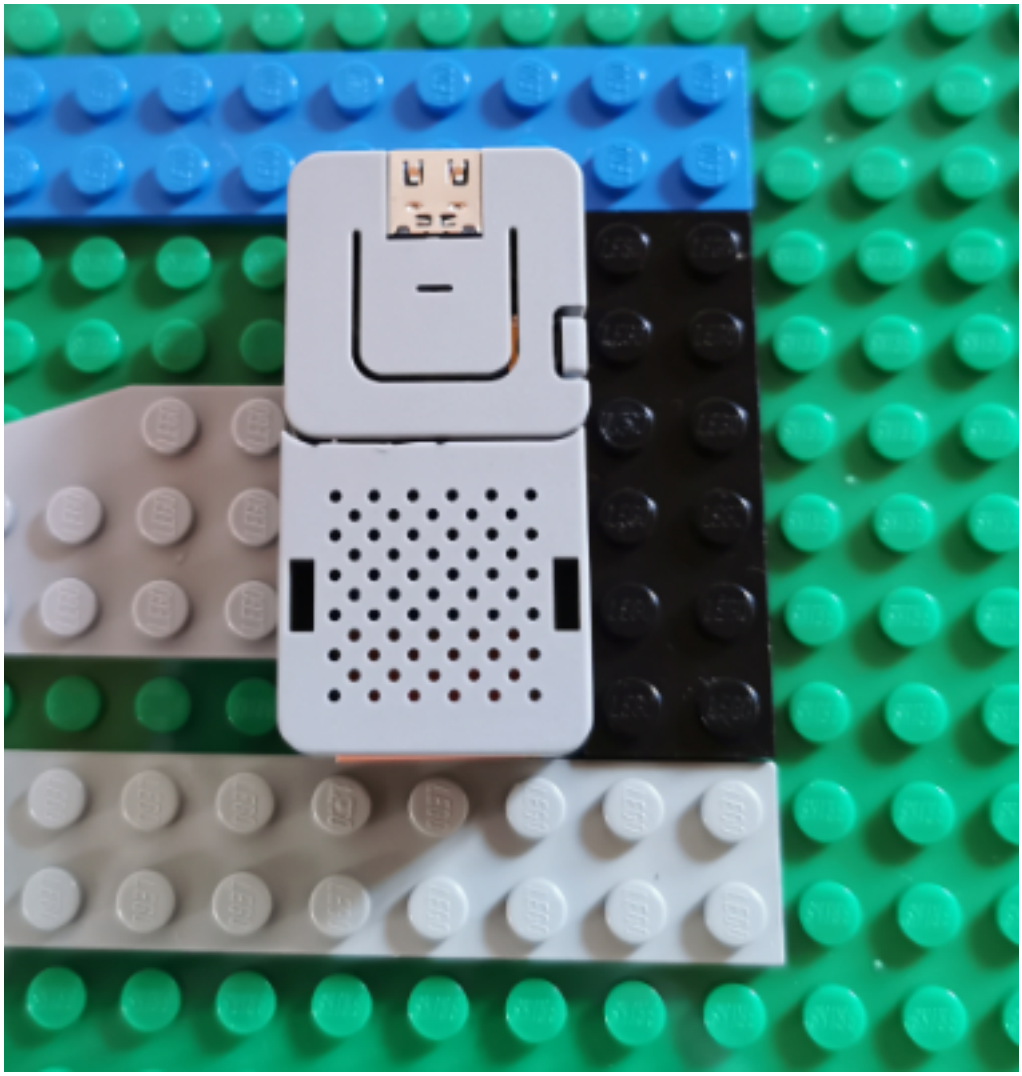








qzms docs000013qz



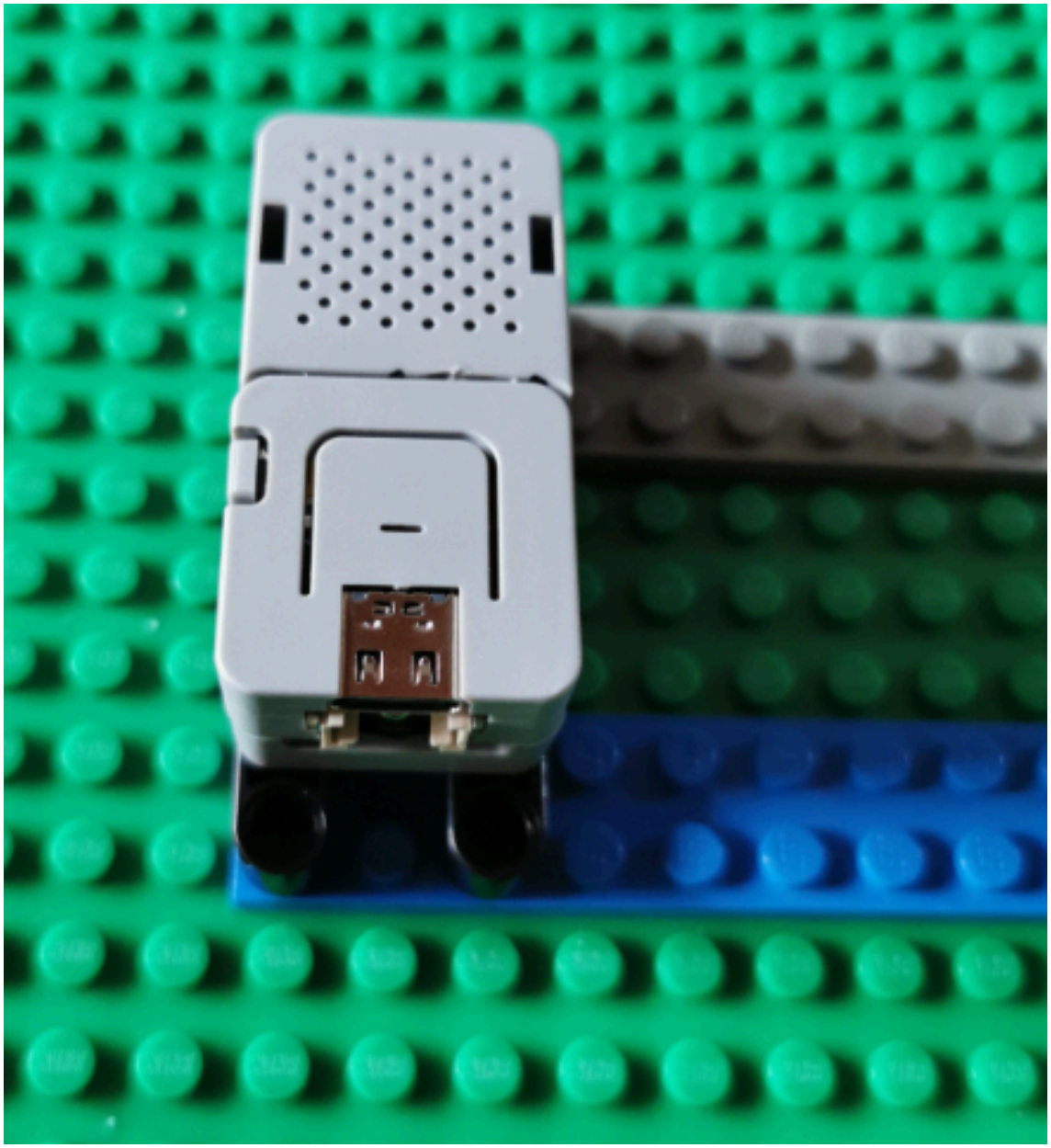

30 Pro
O CAMERA



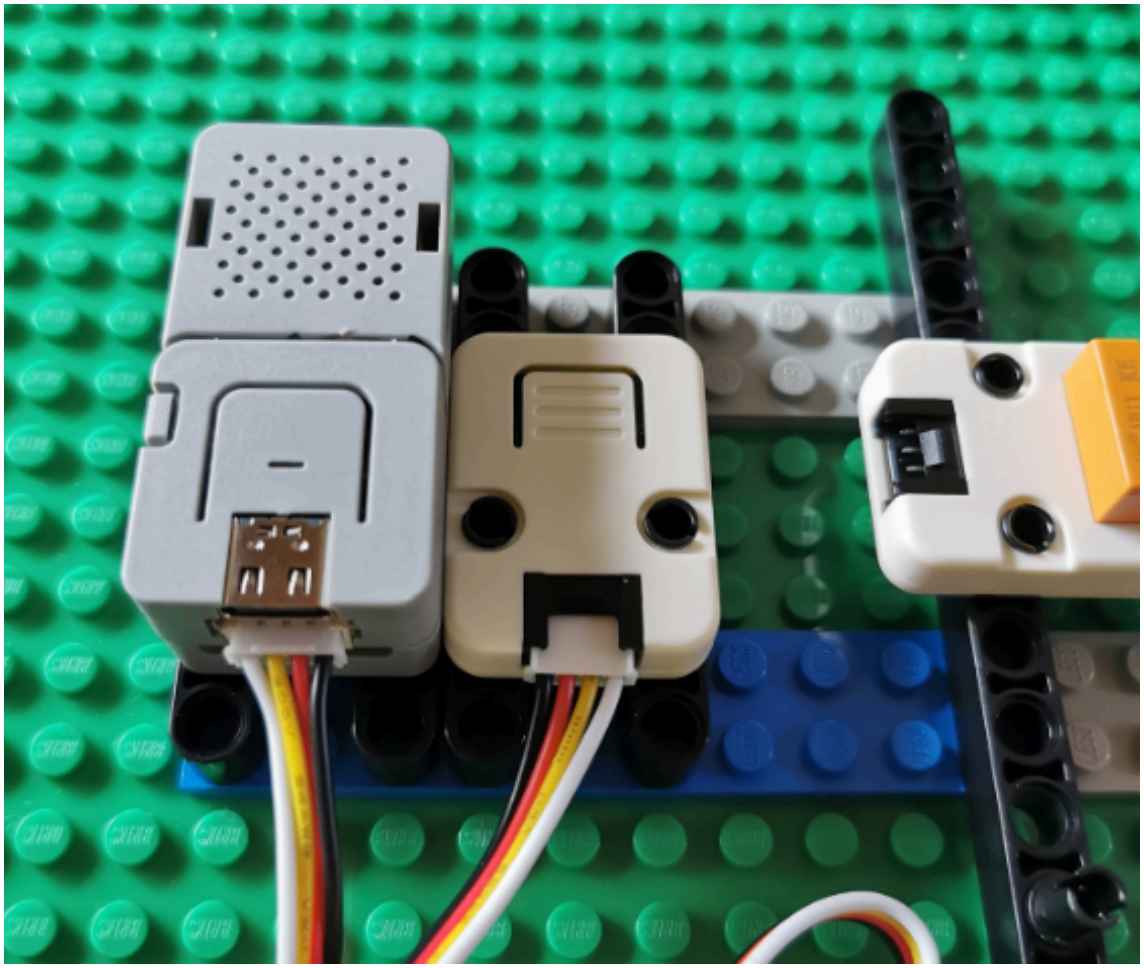


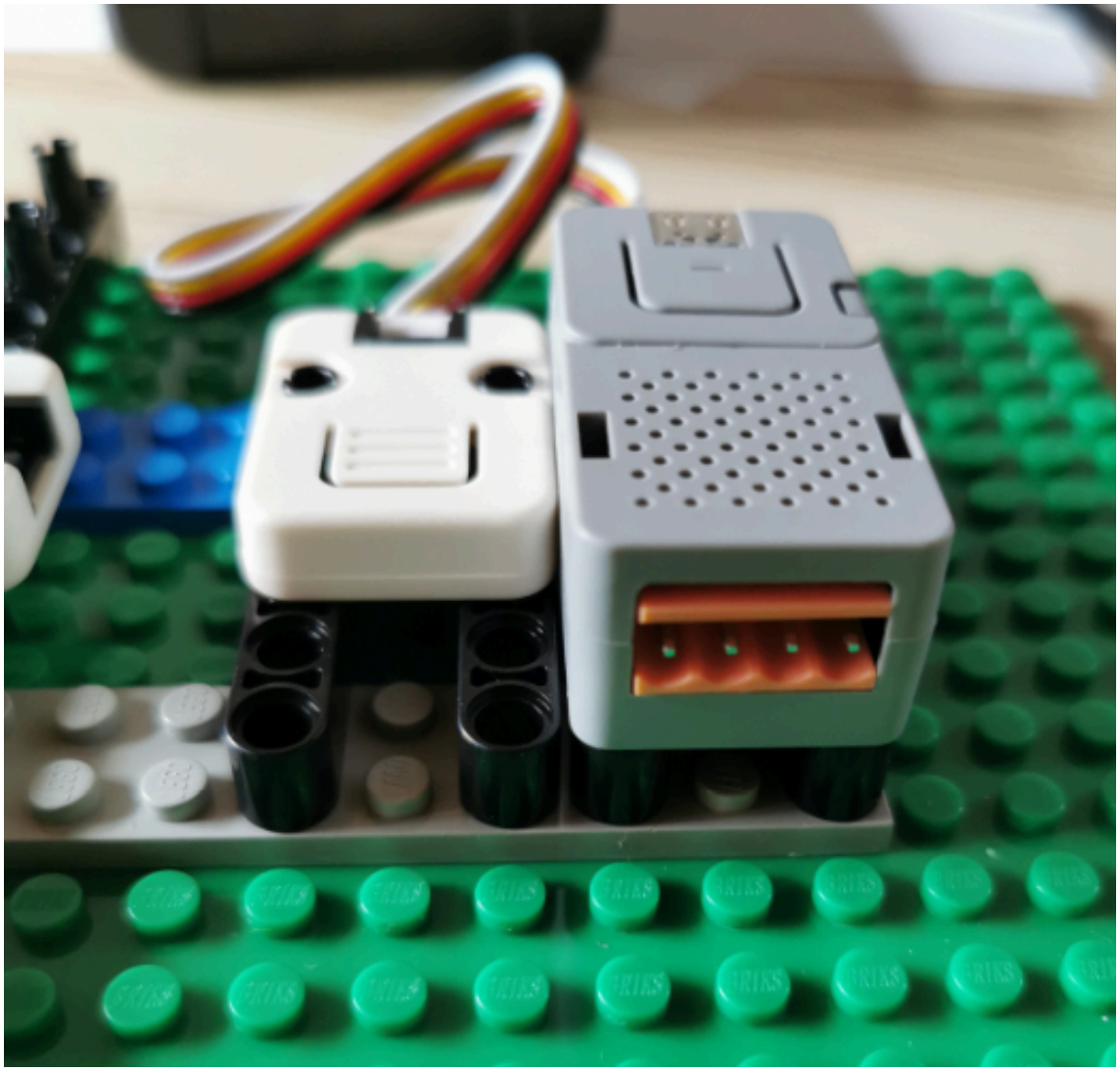




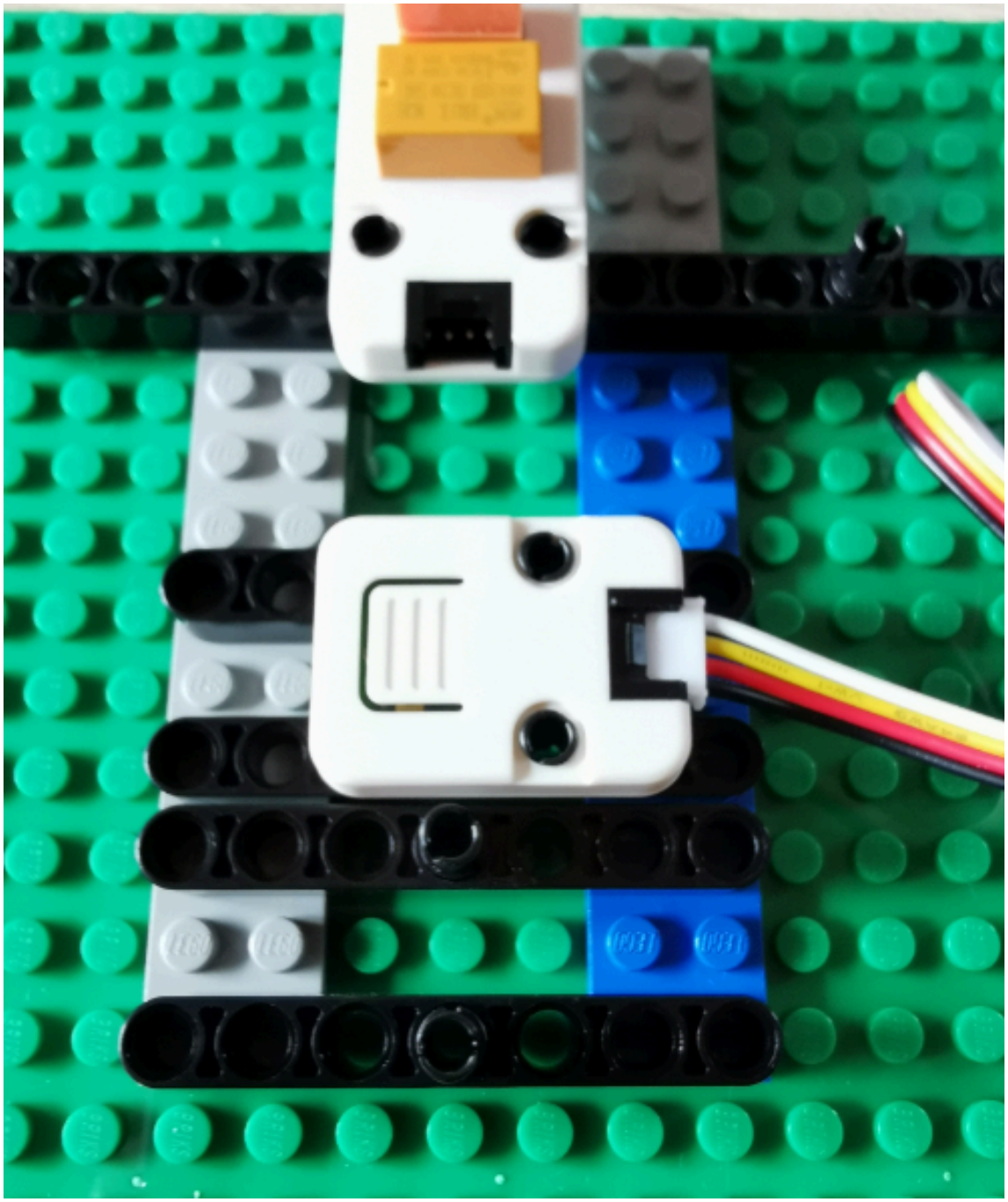


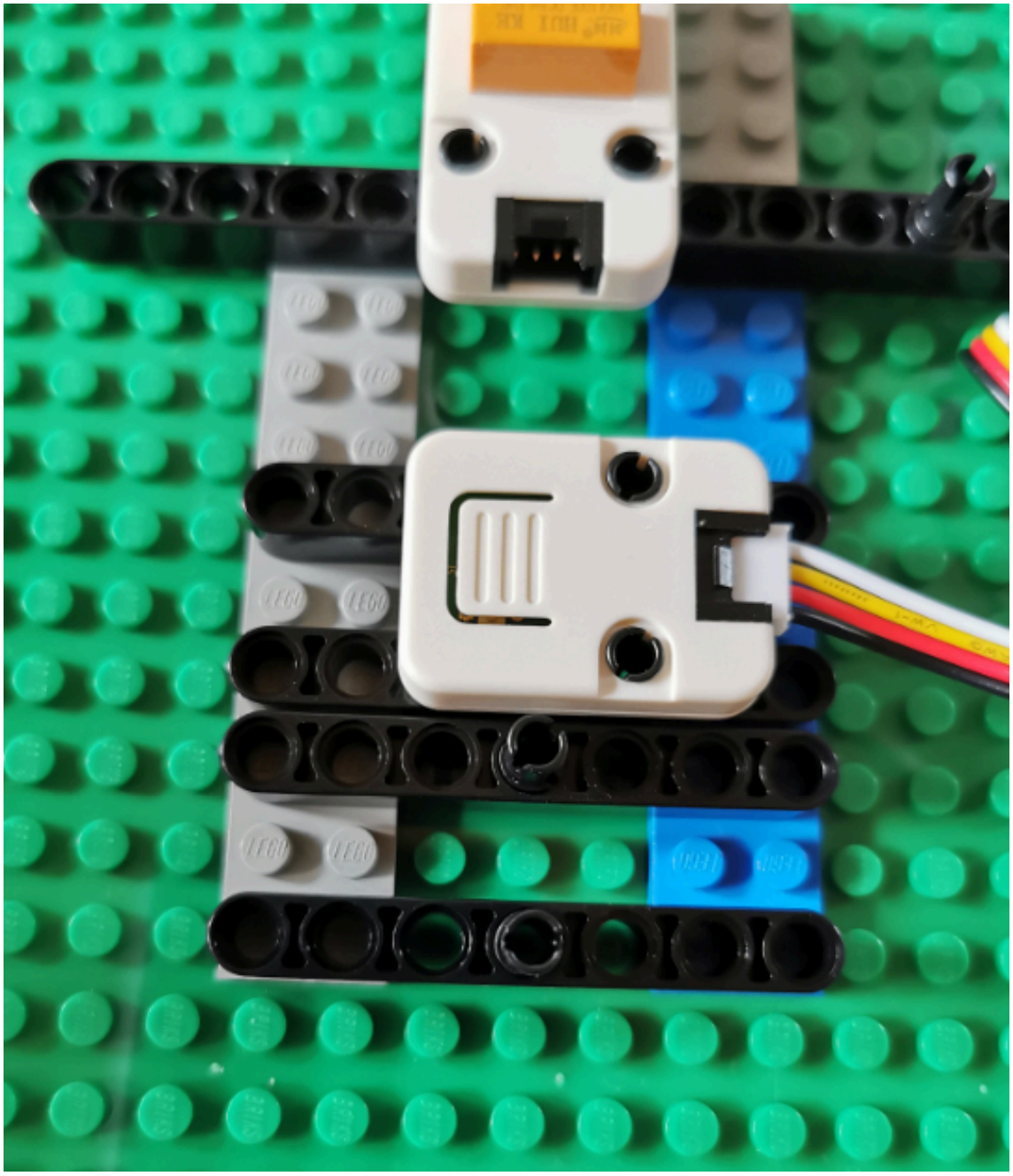

M5











19.1 🎧 Podcast

- Eclipse 4diac FORTE: Understanding IEC 61499 – The LEGO® Building Blocks for Your Industry 4.0 Control System
- DIN EN 61499-1: The Digital LEGO Building Blocks for Flexible Automation and a Smart Future
- DIN EN 61499-1: The LEGO Bricks for Flexible and Event-Driven Industrial Control Systems
- EventFBs according to IEC 61499: LEGO Bricks of Automation – How Event Function Blocks Work
- IEC 61499-1: The Function Block – LEGO Principle for Modern Control Systems in Agricultural and Construction Machinery

---

19.1.1 🌐 Related topic subpages on ms-muc-docs.de

- 🌐 Eclipse 4diac IDE & Color Reference on ms-muc-docs.de

20. Optional

21. Package



21.1 🎧 Podcast

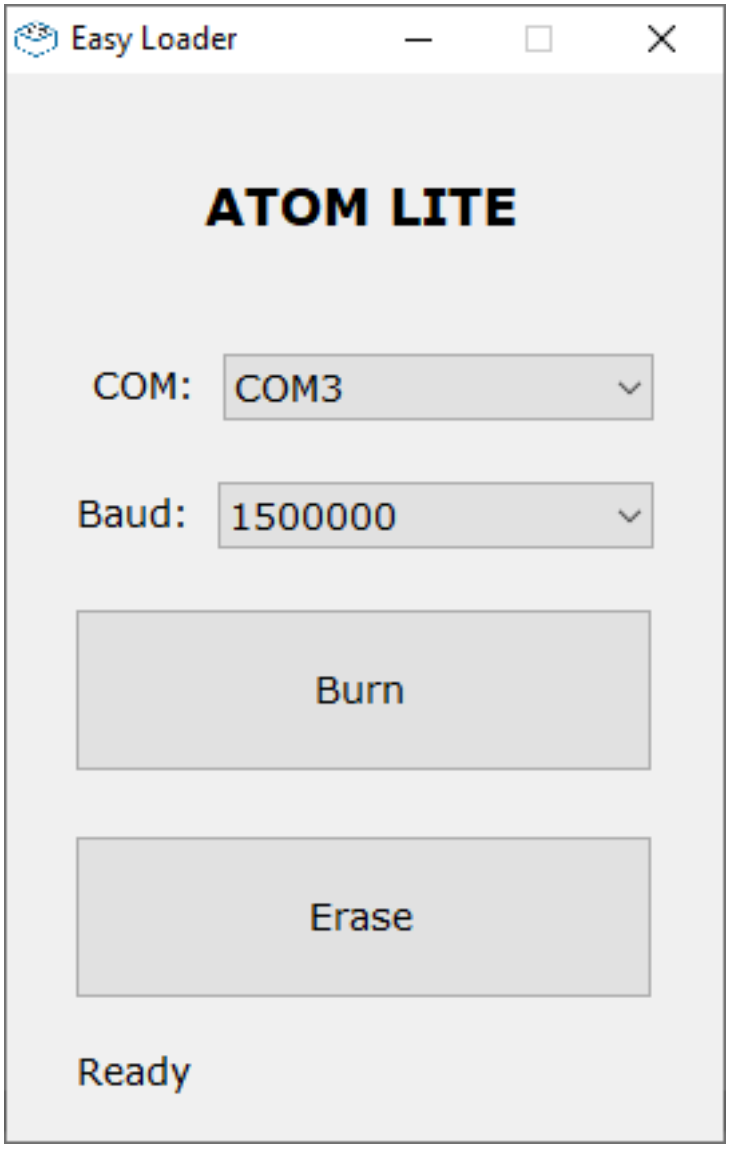
- Powerhouses in Action: The Secret of Hydraulic Cylinders – From Excavators to High-Tech Machines

21.2 📺 Video

- Chip Becomes a Powerhouse

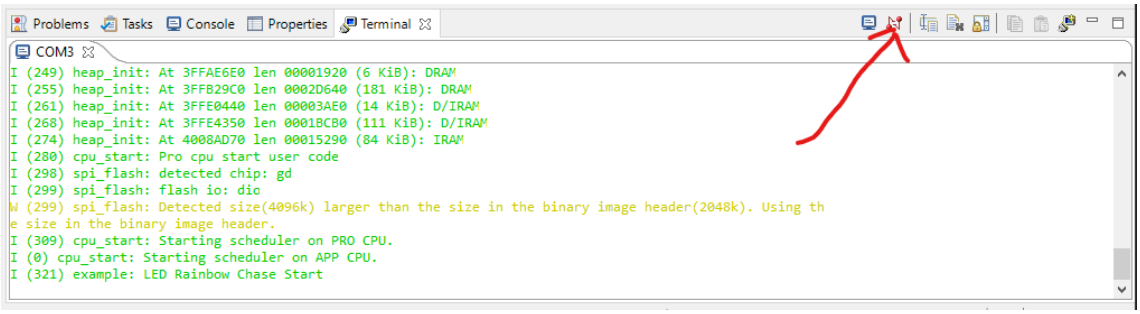
22. Self-Test

Open the file M:\Agricultural Machinery Mechanics\Lessons\SL\esp-idf\Atom-SelfTest\EasyLoader_ATOM _LITE_FactoryTest.exe



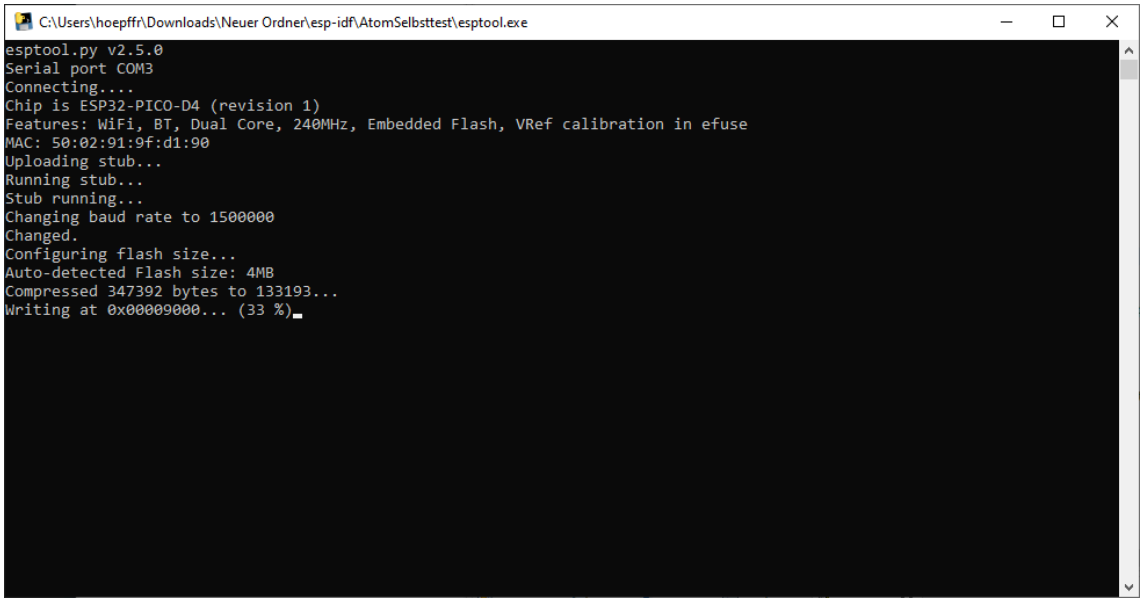
With Erase, you can always reset the ATOM to factory settings. With BURN, you load a self-test program.

If you are still connected to the terminal, press here:



and try BURN or ERASE again.

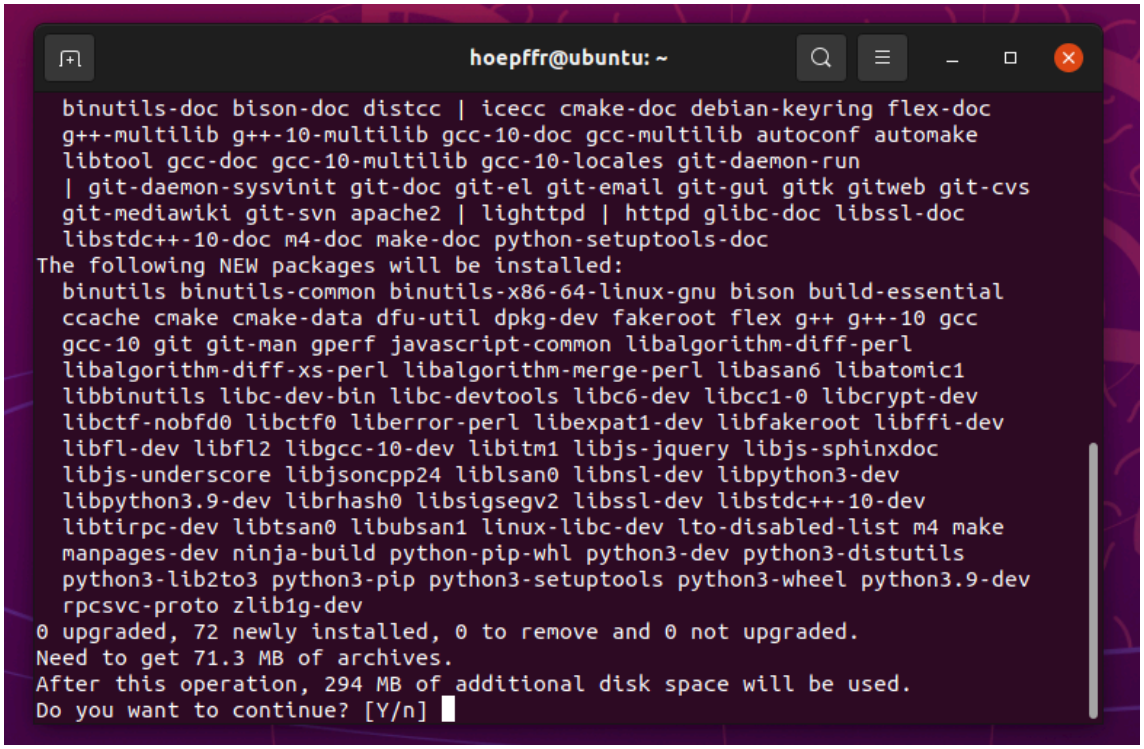
It should look like this:



To perform the self-test, the ATOM's button must be pressed.

23. Setting up (LINUX)

https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/linux-setup.html sudo apt install git wget flex bison gperf python3 python3-pip python3-setuptools cmake ninja-build ccache libffi-dev libssl-dev dfu-util libusb-1.0-0



Confirm with Yes.

Now download ESP-IDF.

mkdir -p /esp cd /esp git clone –recursive https://github.com/espressif/esp-idf.git

ESP-IDF will be downloaded to this folder: `~/esp/esp-idf`

Now navigate to the folder using `cd ~/esp/esp-idf`.

Make the install file executable using `chmod +x install.sh`. Then run the file.

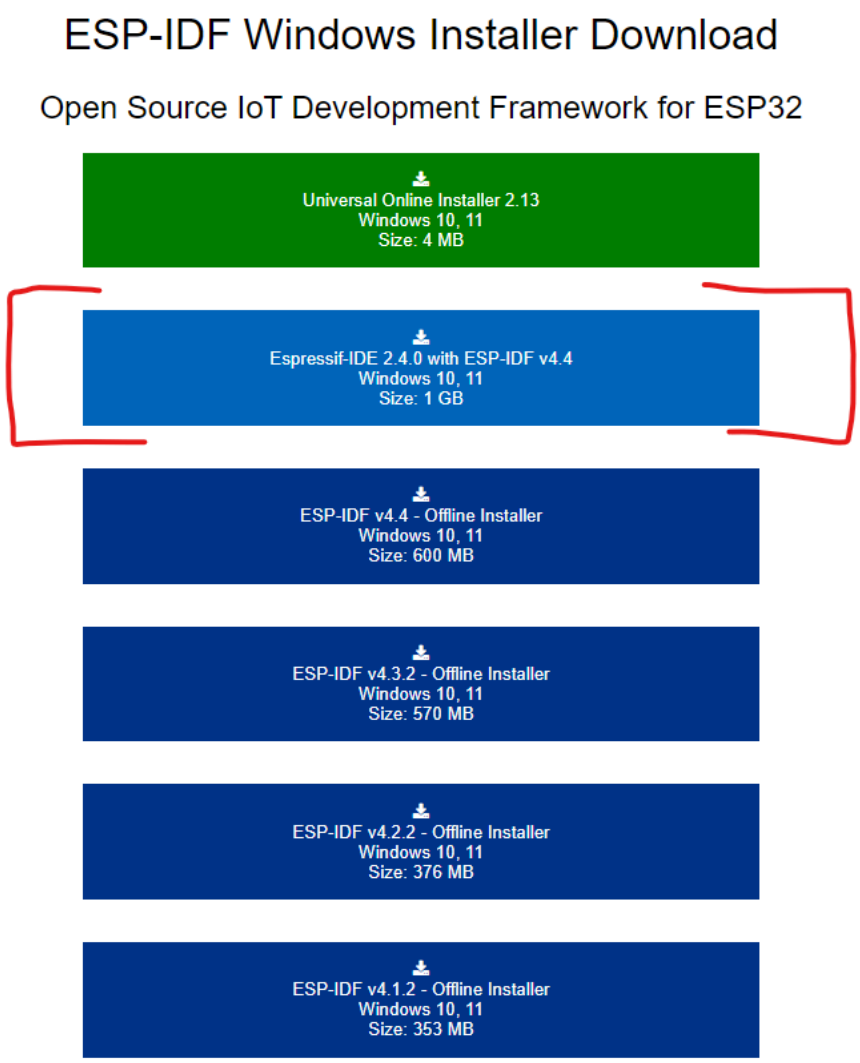
./install.sh`

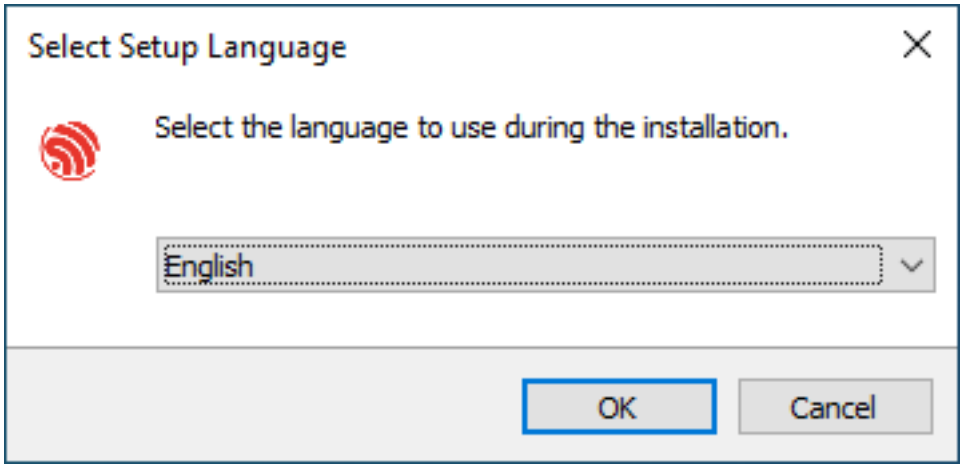
Done.

24. Setting-up-II

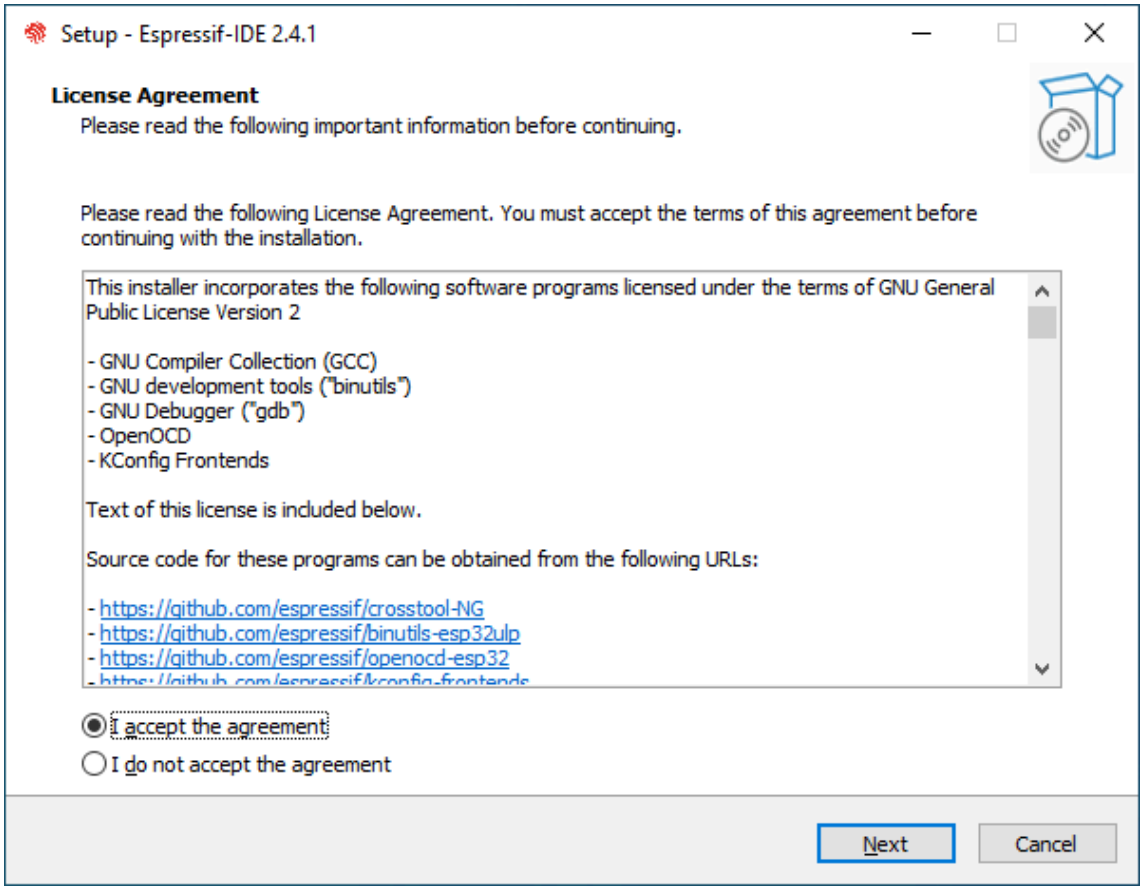
This setup will be available from around January 2022. <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/index.html <https://dl.espressif.com/dl/esp-idf/?idf=4.4 Espressif-IDE 2.4.0 with ESP-IDF v4.4 Windows 10, 11 Size: 1GB

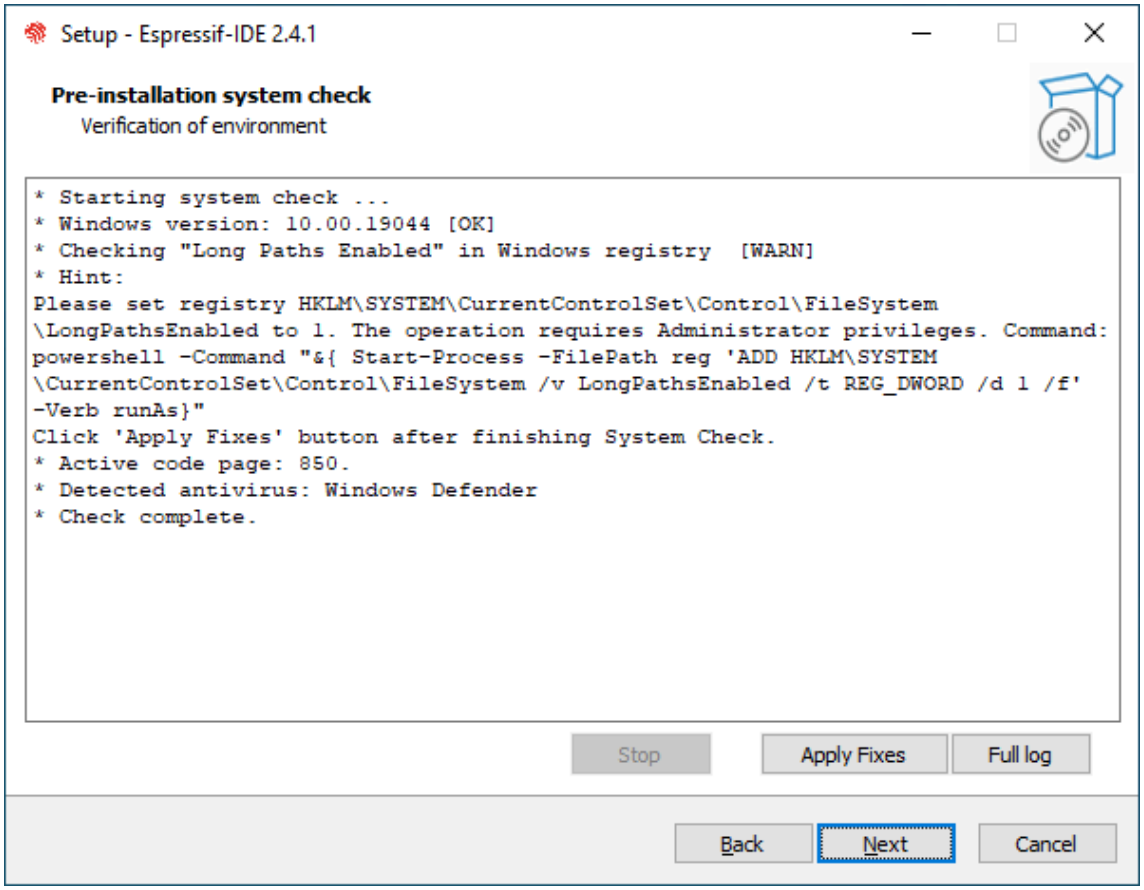
espressif-ide-setup-espressif-ide-2.4.1-with-esp-idf-4.4.exe



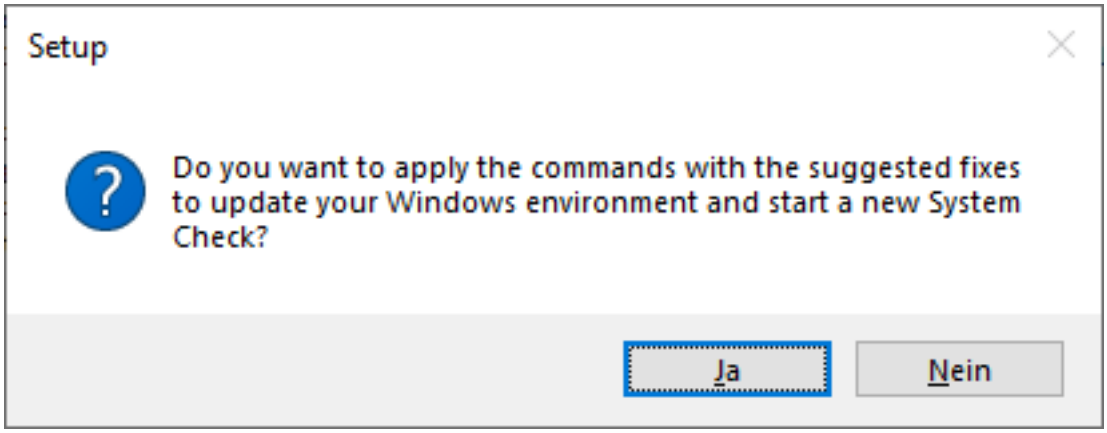

Select Setup Language
Select the language to use during the installation.
English
OK
Cancel

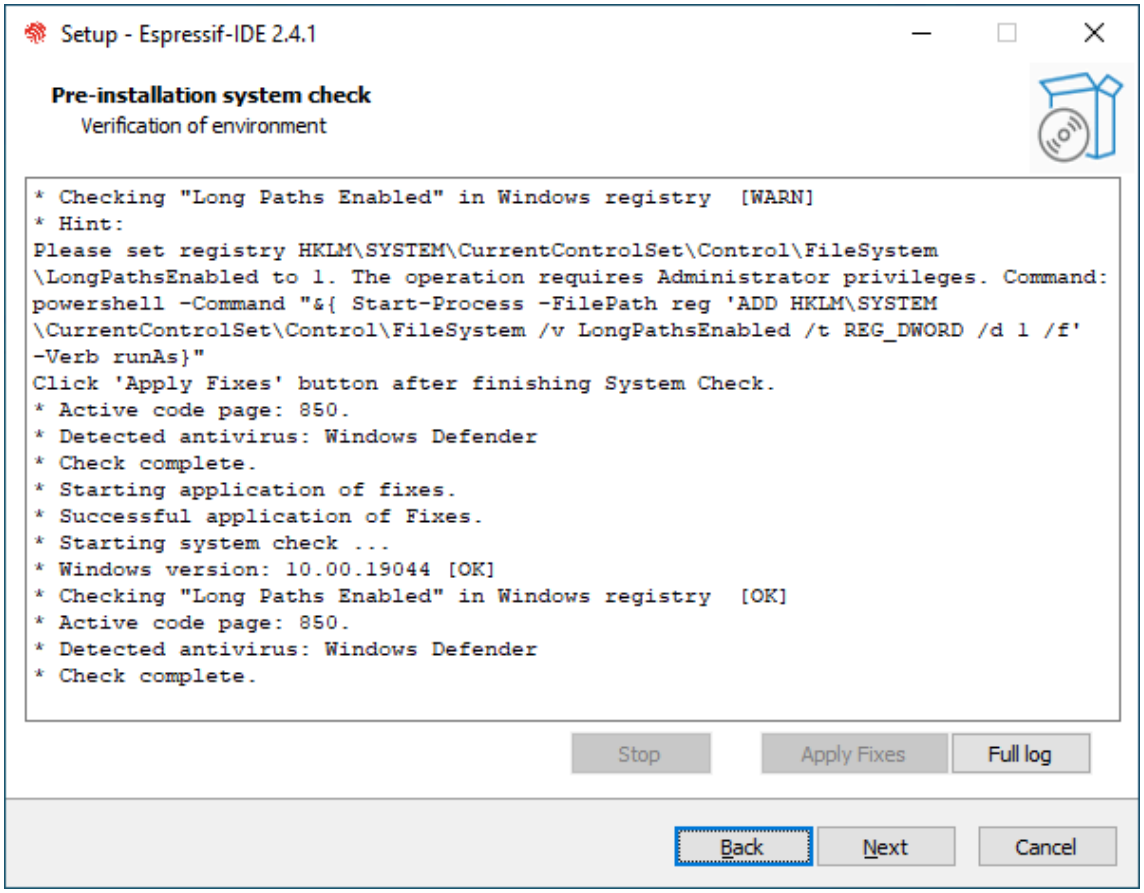
accept:

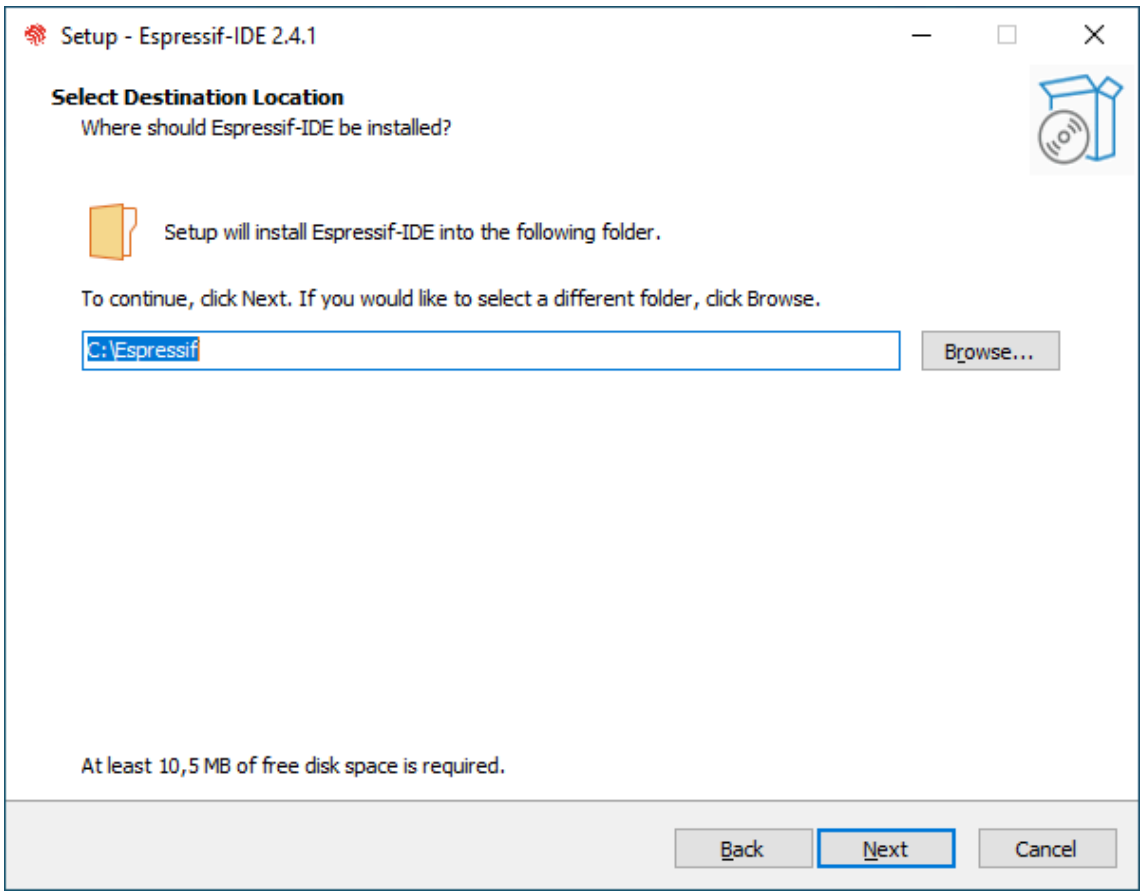

Setup - Espressif-IDE 2.4.1
License Agreement
Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
This installer incorporates the following software programs licensed under the terms of GNU General Public License Version 2
- GNU Compiler Collection (GCC)
- GNU development tools ("binutils")
- GNU Debugger ("gdb")
- OpenOCD
- KConfig Frontends
Text of this license is included below.
Source code for these programs can be obtained from the following URLs:
- https://github.com/espressif/crosstool-NG
- https://github.com/espressif/binutils-esp32ulp
- https://github.com/espressif/openocd-esp32
I accept the agreement
I do not accept the agreement
Next
Cancel

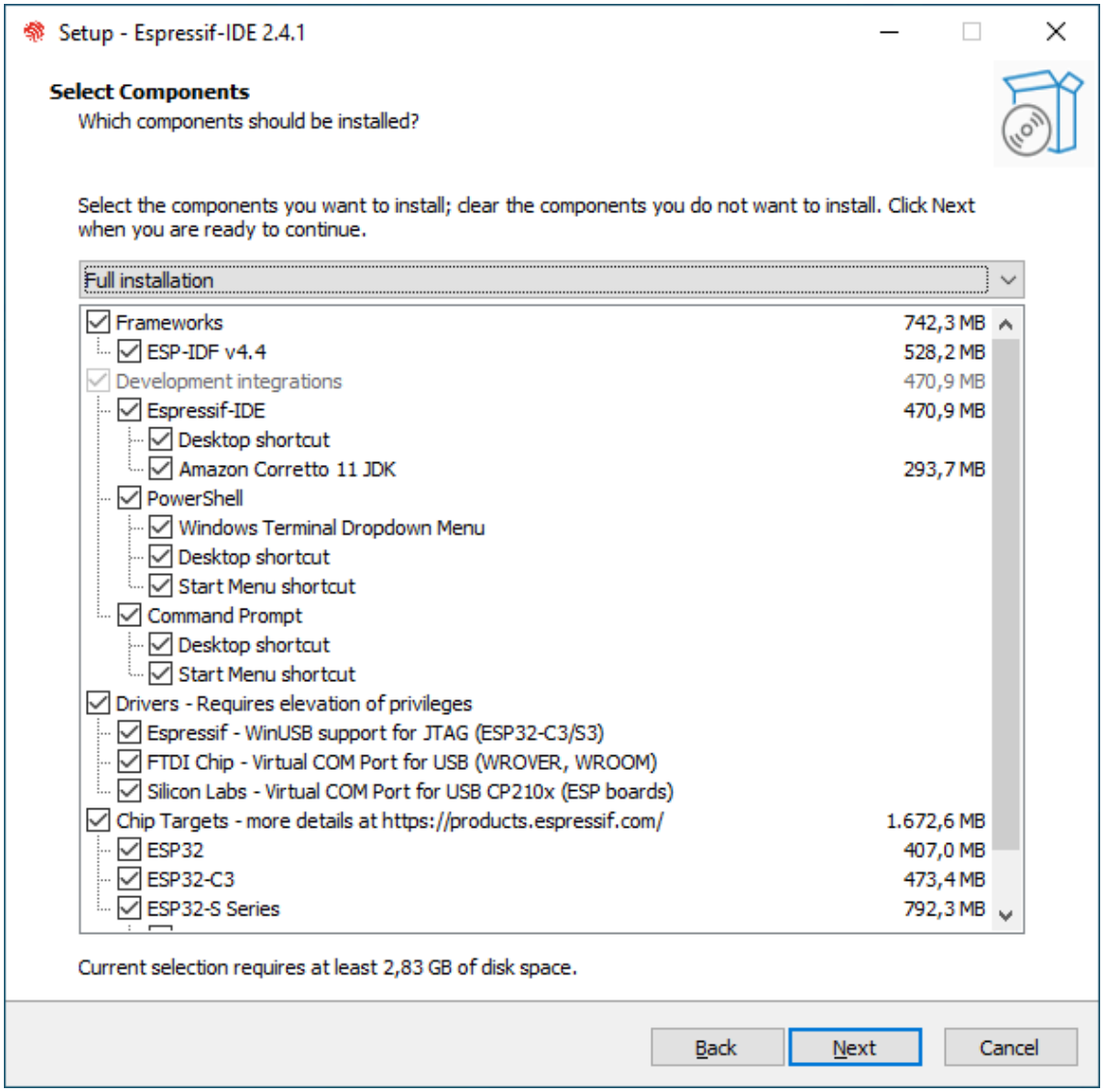

Setup - Espressif-IDE 2.4.1
Pre-installation system check
Verification of environment
* Starting system check ...
* Windows version: 10.00.19044 [OK]
* Checking "Long Paths Enabled" in Windows registry [WARN]
* Hint:
Please set registry HKLM\SYSTEM\CurrentControlSet\Control\FileSystem
\LongPathsEnabled to 1. The operation requires Administrator privileges. Command:
powershell -Command "&{ Start-Process -FilePath reg 'ADD HKLM\SYSTEM
\CurrentControlSet\Control\FileSystem /v LongPathsEnabled /t REG_DWORD /d 1 /f'
-Verb runAs}"
Click 'Apply Fixes' button after finishing System Check.
* Active code page: 850.
* Detected antivirus: Windows Defender
* Check complete.
Stop
Apply Fixes
Full log
Back
Next
Cancel

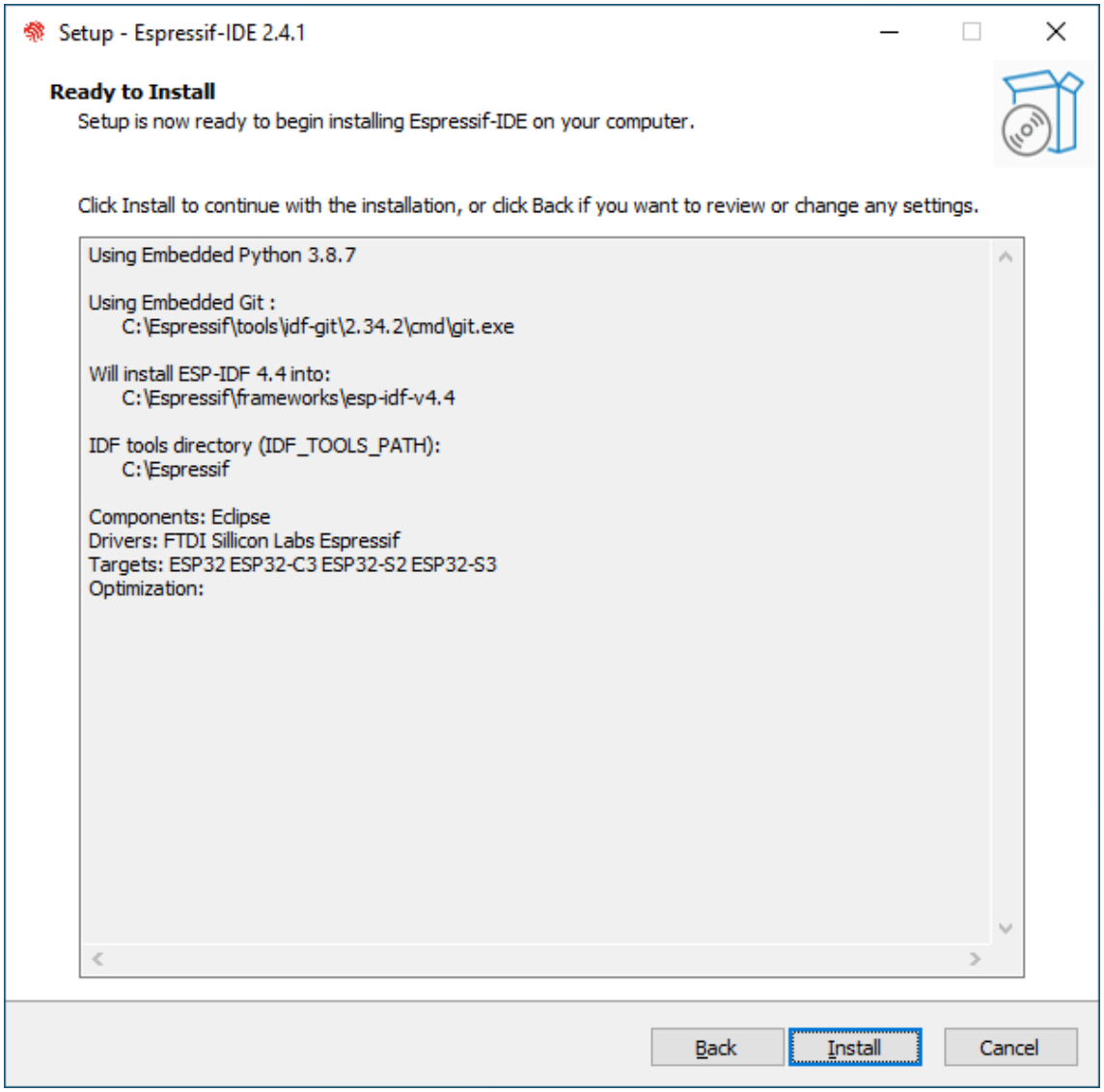
Apply Fixes

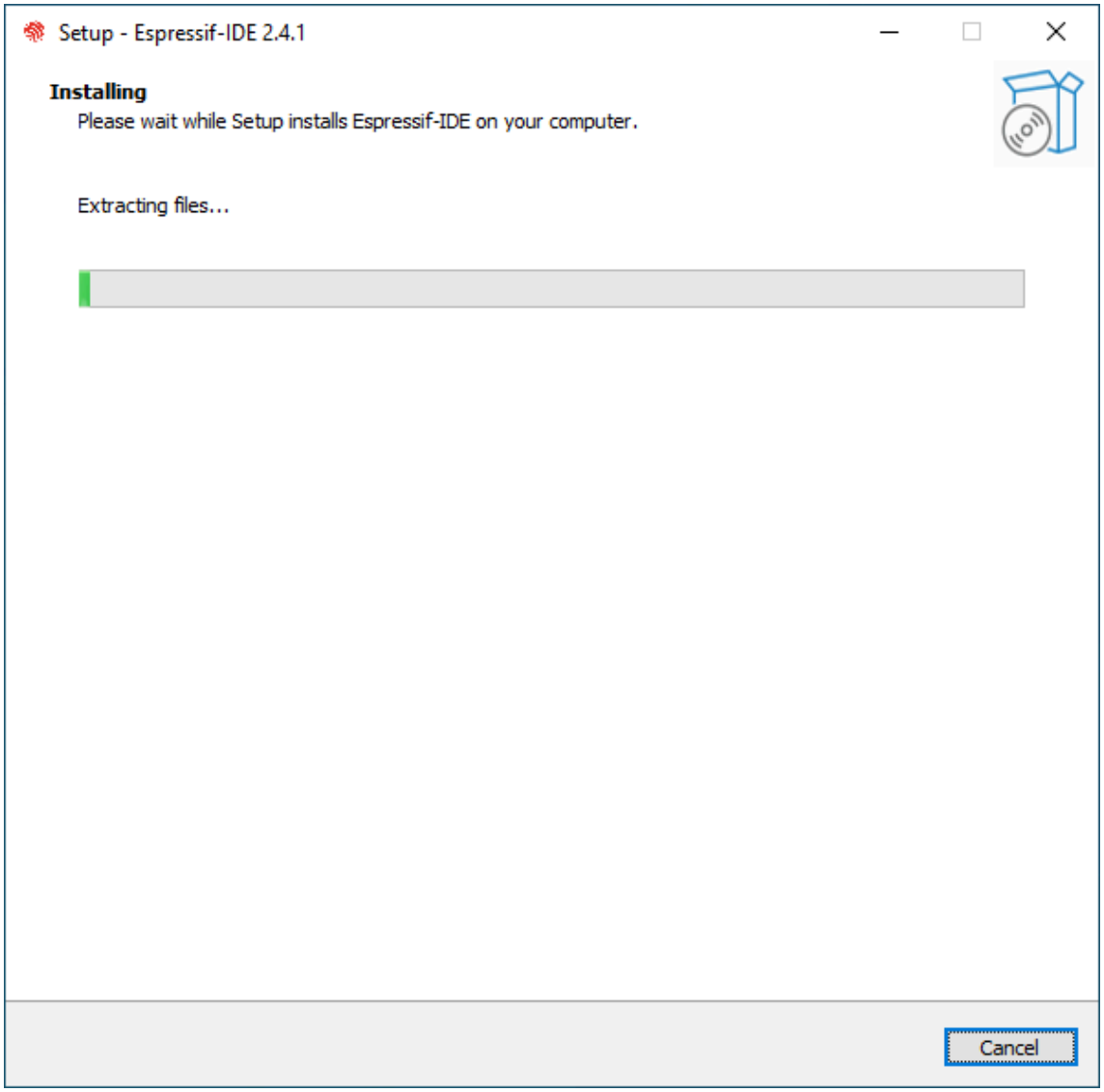

Setup
Do you want to apply the commands with the suggested fixes to update your Windows environment and start a new System Check?
Ja
Nein

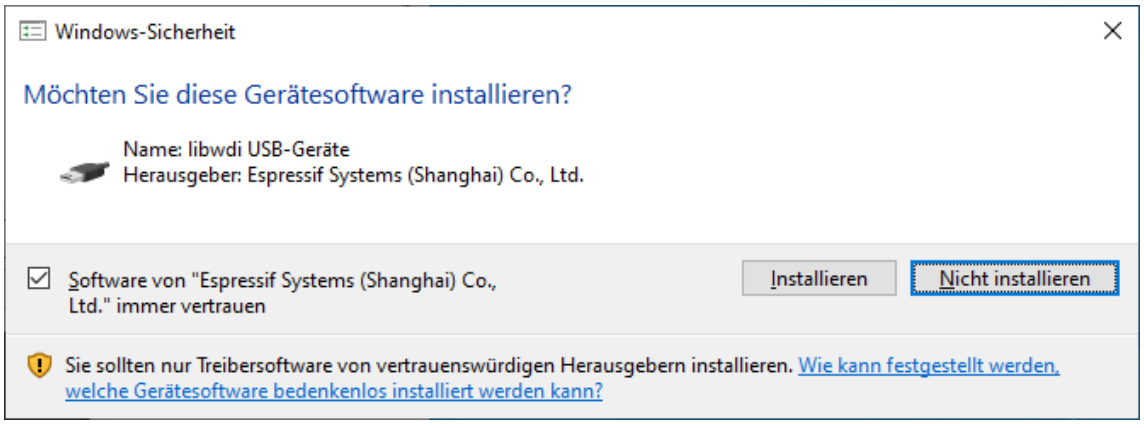

Setup - Espressif-IDE 2.4.1
Pre-installation system check
Verification of environment
* Checking "Long Paths Enabled" in Windows registry [WARN]
* Hint:
Please set registry HKLM\SYSTEM\CurrentControlSet\Control\FileSystem
\LongPathsEnabled to 1. The operation requires Administrator privileges. Command:
powershell -Command "&{ Start-Process -FilePath reg 'ADD HKLM\SYSTEM
\CurrentControlSet\Control\FileSystem /v LongPathsEnabled /t REG_DWORD /d 1 /f'
-Verb runAs}"
Click 'Apply Fixes' button after finishing System Check.
* Active code page: 850.
* Detected antivirus: Windows Defender
* Check complete.
* Starting application of fixes.
* Successful application of Fixes.
* Starting system check ...
* Windows version: 10.00.19044 [OK]
* Checking "Long Paths Enabled" in Windows registry [OK]
* Active code page: 850.
* Detected antivirus: Windows Defender
* Check complete.
Stop
Apply Fixes
Full log
Back
Next
Cancel

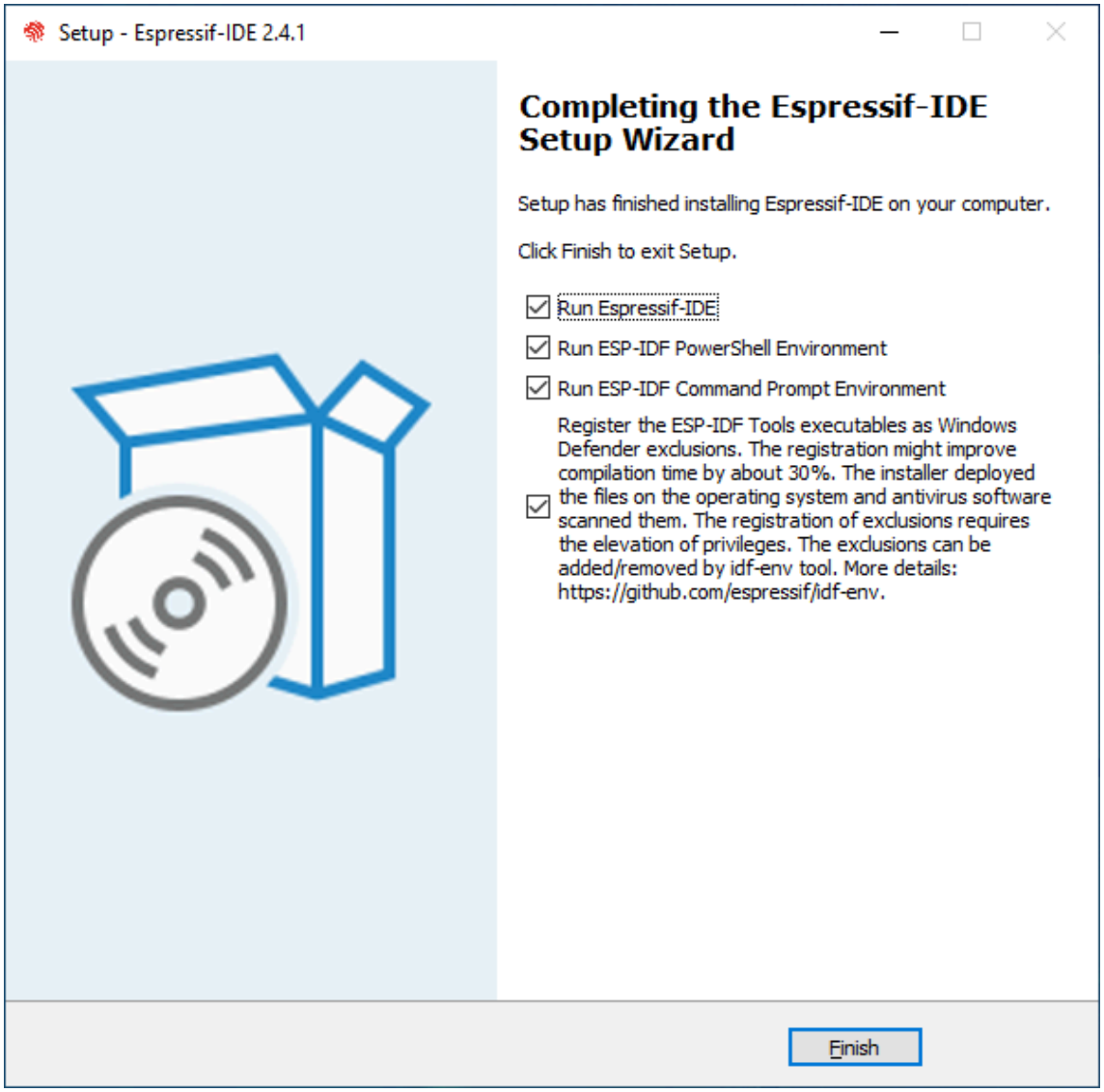

Setup - Espressif-IDE 2.4.1
Select Destination Location
Where should Espressif-IDE be installed?
Setup will install Espressif-IDE into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Espressif
Browse...
At least 10,5 MB of free disk space is required.
Back
Next
Cancel

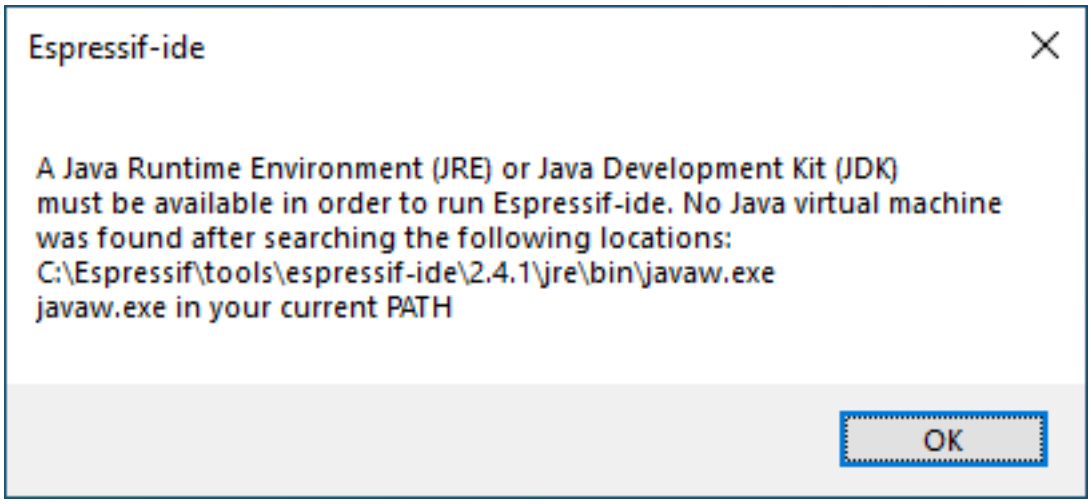

Setup - Espressif-IDE 2.4.1
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Full installation
Frameworks 742,3 MB
ESP-IDF v4.4 528,2 MB
Development integrations 470,9 MB
Espressif-IDE 470,9 MB
Desktop shortcut
Amazon Corretto 11 JDK 293,7 MB
PowerShell
Windows Terminal Dropdown Menu
Desktop shortcut
Start Menu shortcut
Command Prompt
Desktop shortcut
Start Menu shortcut
Drivers - Requires elevation of privileges
Espressif - WinUSB support for JTAG (ESP32-C3/S3)
FTDI Chip - Virtual COM Port for USB (WROVER, WROOM)
Silicon Labs - Virtual COM Port for USB CP210x (ESP boards)
Chip Targets - more details at https://products.espressif.com/ 1.672,6 MB
ESP32 407,0 MB
ESP32-C3 473,4 MB
ESP32-S Series 792,3 MB
Current selection requires at least 2,83 GB of disk space.
Back
Next
Cancel


Setup - Espressif-IDE 2.4.1
Ready to Install
Setup is now ready to begin installing Espressif-IDE on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Using Embedded Python 3.8.7
Using Embedded Git :
C:\Espressif\tools\idf-git\2.34.2\cmd\git.exe
Will install ESP-IDF 4.4 into:
C:\Espressif\frameworks\esp-idf-v4.4
IDF tools directory (IDF_TOOLS_PATH):
C:\Espressif
Components: Eclipse
Drivers: FTDI Sillicon Labs Espressif
Targets: ESP32 ESP32-C3 ESP32-S2 ESP32-S3
Optimization:
Back
Install
Cancel

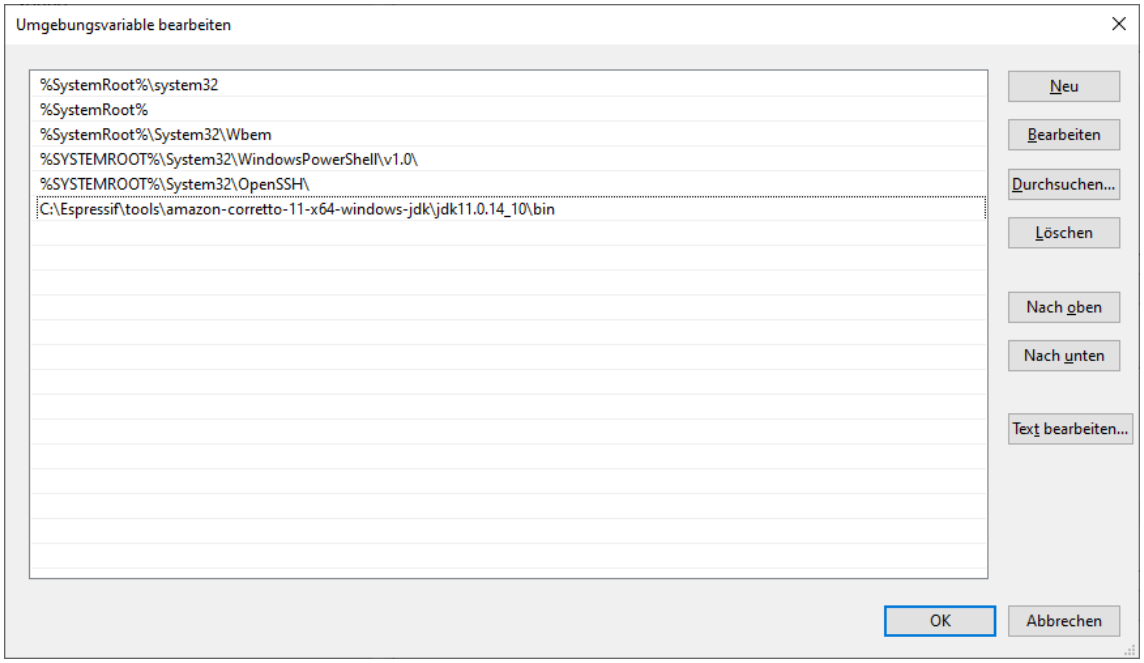

Setup - Espressif-IDE 2.4.1
Installing
Please wait while Setup installs Espressif-IDE on your computer.
Extracting files...
Cancel


Windows-Sicherheit
Möchten Sie diese Gerätesoftware installieren?
Name: libwdi USB-Geräte
Herausgeber: Espressif Systems (Shanghai) Co., Ltd.
Software von "Espressif Systems (Shanghai) Co., Ltd." immer vertrauen
Installieren
Nicht installieren
Sie sollten nur Treibersoftware von vertrauenswürdigen Herausgebern installieren. Wie kann festgestellt werden, welche Gerätesoftware bedenkenlos installiert werden kann?

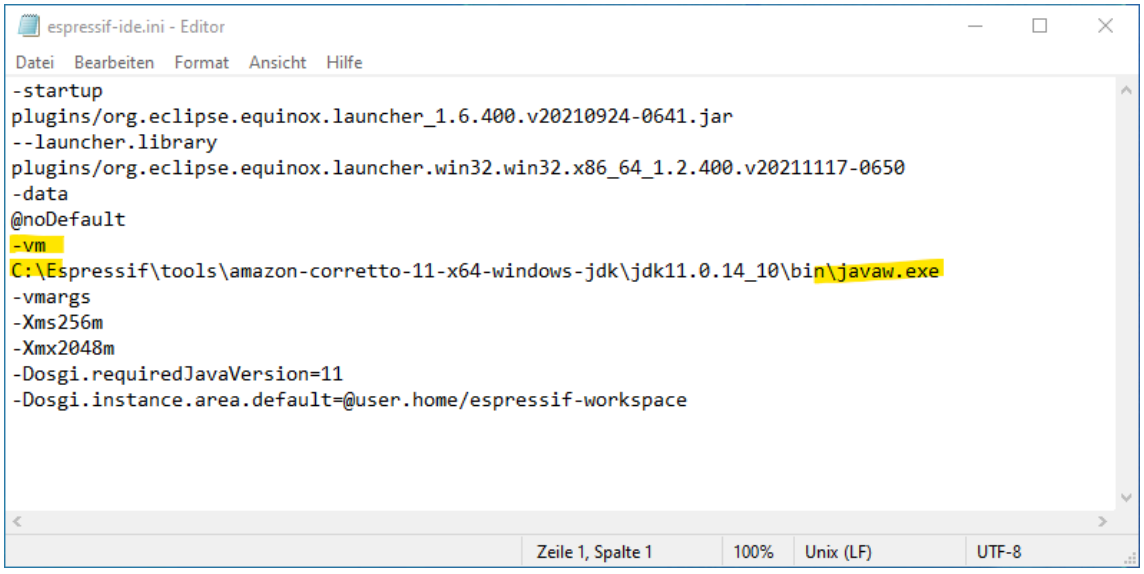

Setup - Espressif-IDE 2.4.1
Completing the Espressif-IDE Setup Wizard
Setup has finished installing Espressif-IDE on your computer.
Click Finish to exit Setup.
Run Espressif-IDE
Run ESP-IDF PowerShell Environment
Run ESP-IDF Command Prompt Environment
Register the ESP-IDF Tools executables as Windows Defender exclusions. The registration might improve compilation time by about 30%. The installer deployed the files on the operating system and antivirus software scanned them. The registration of exclusions requires the elevation of privileges. The exclusions can be added/removed by idf-env tool. More details: https://github.com/espressif/idf-env.
Finish


Espressif-ide
A Java Runtime Environment (JRE) or Java Development Kit (JDK) must be available in order to run Espressif-ide. No Java virtual machine was found after searching the following locations:
C:\Espressif\tools\espressif-ide\2.4.1\jre\bin\javaw.exe
javaw.exe in your current PATH
OK

Path:


Umgebungsvariable bearbeiten
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Espressif\tools\amazon-corretto-11-x64-windows-jdk\jdk11.0.14_10\bin
Neu
Bearbeiten
Durchsuchen...
Löschen
Nach oben
Nach unten
Text bearbeiten...
OK
Abbrechen

OR:


espressif-ide.ini - Editor
Datei Bearbeiten Format Ansicht Hilfe
-startup
plugins/org.eclipse.equinox.launcher_1.6.400.v20210924-0641.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.2.400.v20211117-0650
-data
@noDefault
-vm
C:\Espressif\tools\amazon-corretto-11-x64-windows-jdk\jdk11.0.14_10\bin\javaw.exe
-vmargs
-Xms256m
-Xmx2048m
-Dosgi.requiredJavaVersion=11
-Dosgi.instance.area.default=@user.home/espressif-workspace
Zeile 1, Spalte 1
100%
Unix (LF)
UTF-8

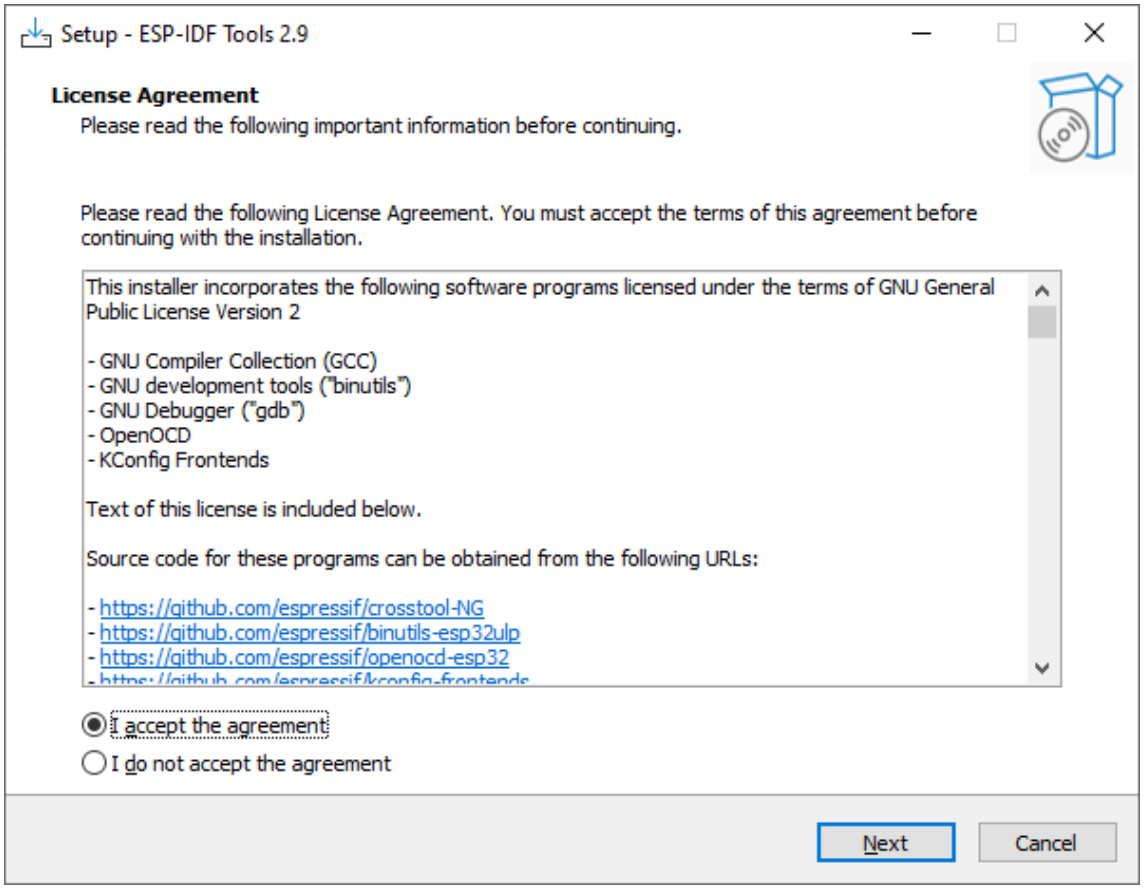
25. Installation Setup

25.1 Step 1

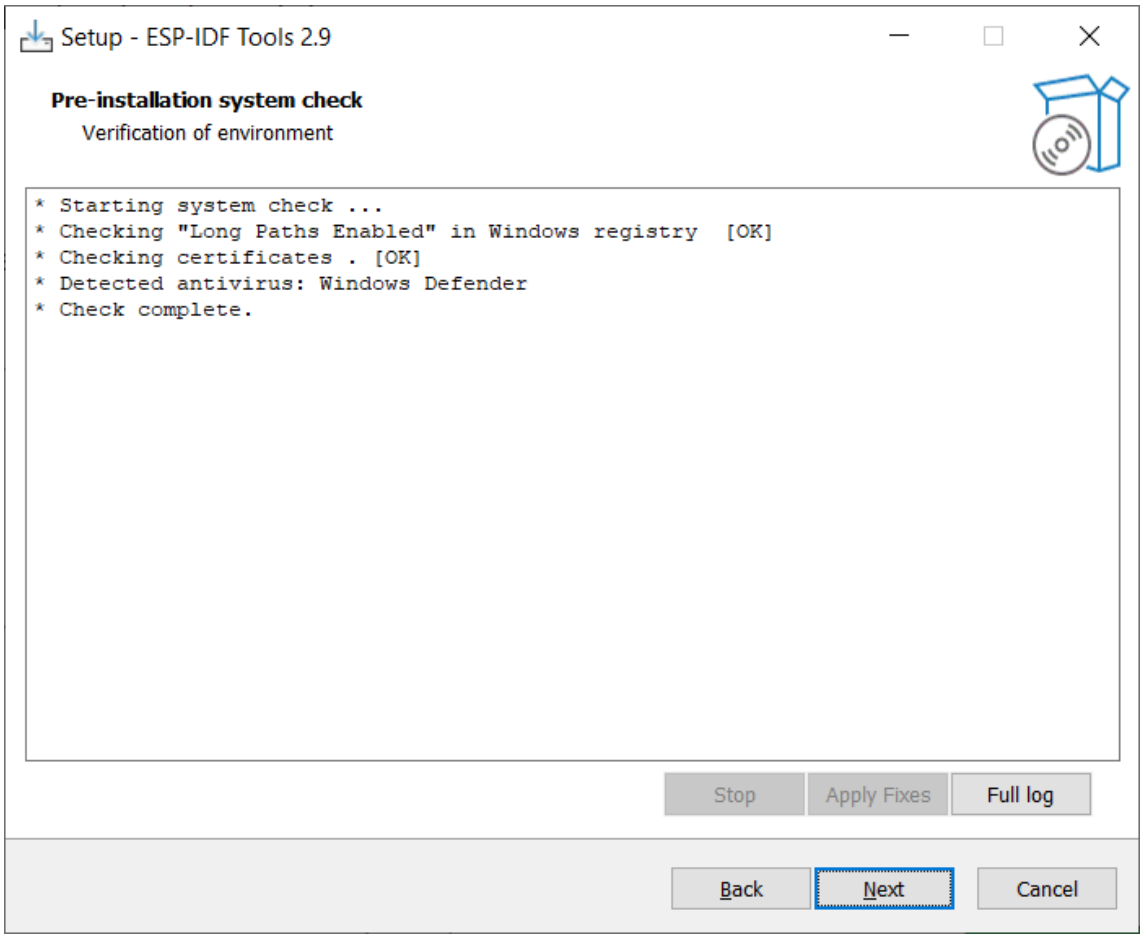
Background information can be found here: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/ Please check beforehand that your username under C:\Users is without umlauts and spaces. Otherwise, you will encounter serious problems.

Download the ESP-IDF Tools.
https://dl.espressif.com/dl/esp-idf/?idf=4.4
https://dl.espressif.com/dl/esp-idf-tools-setup-online-2.9.exe
https://dl.espressif.com/dl/esp-idf-tools-setup-offline-2.10.exe

25.2 Step 2

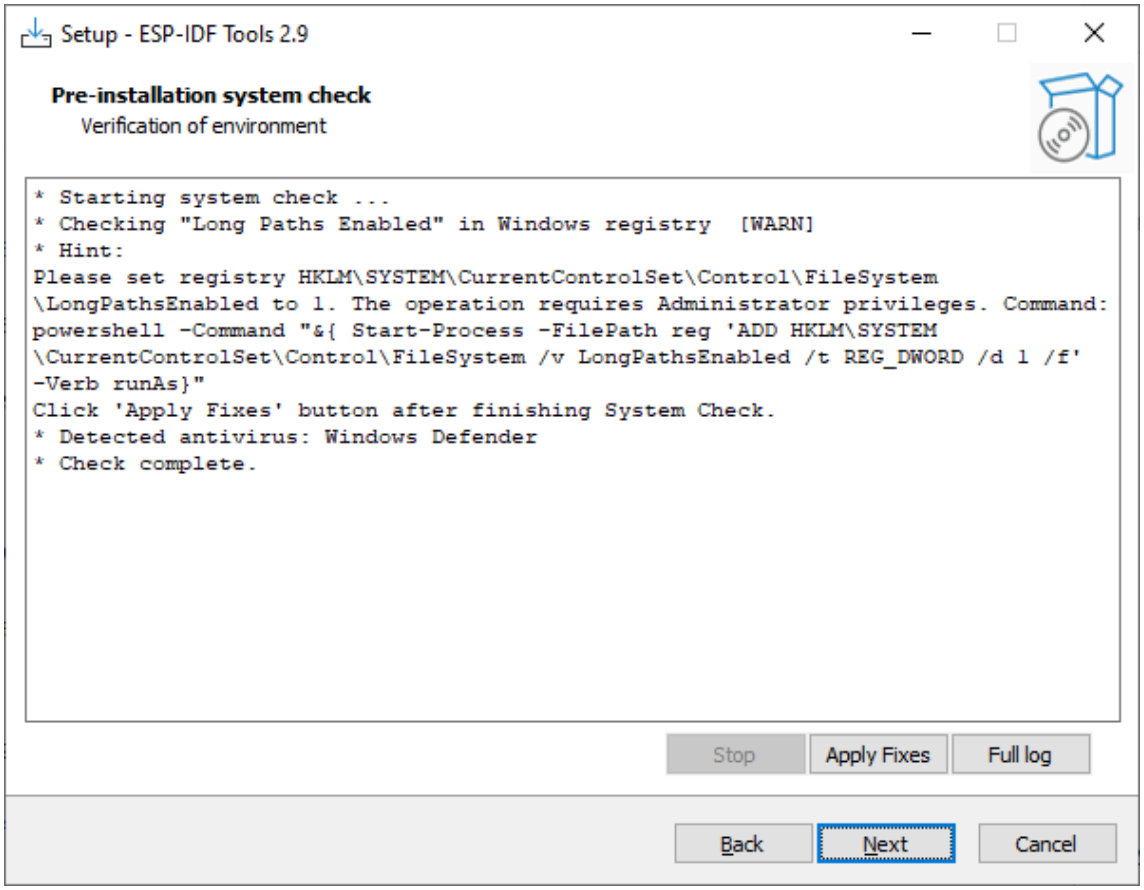


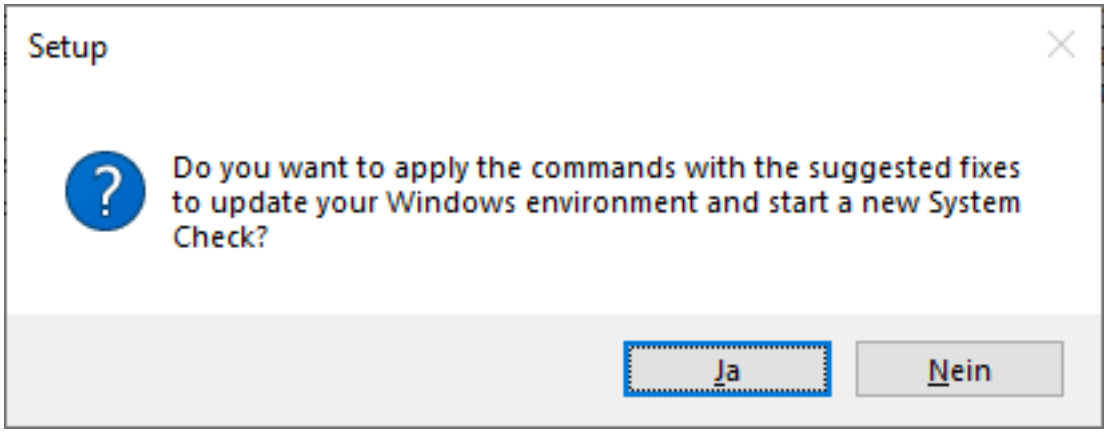
25.3 Step 3



If it looks like this, that's great.

If not:

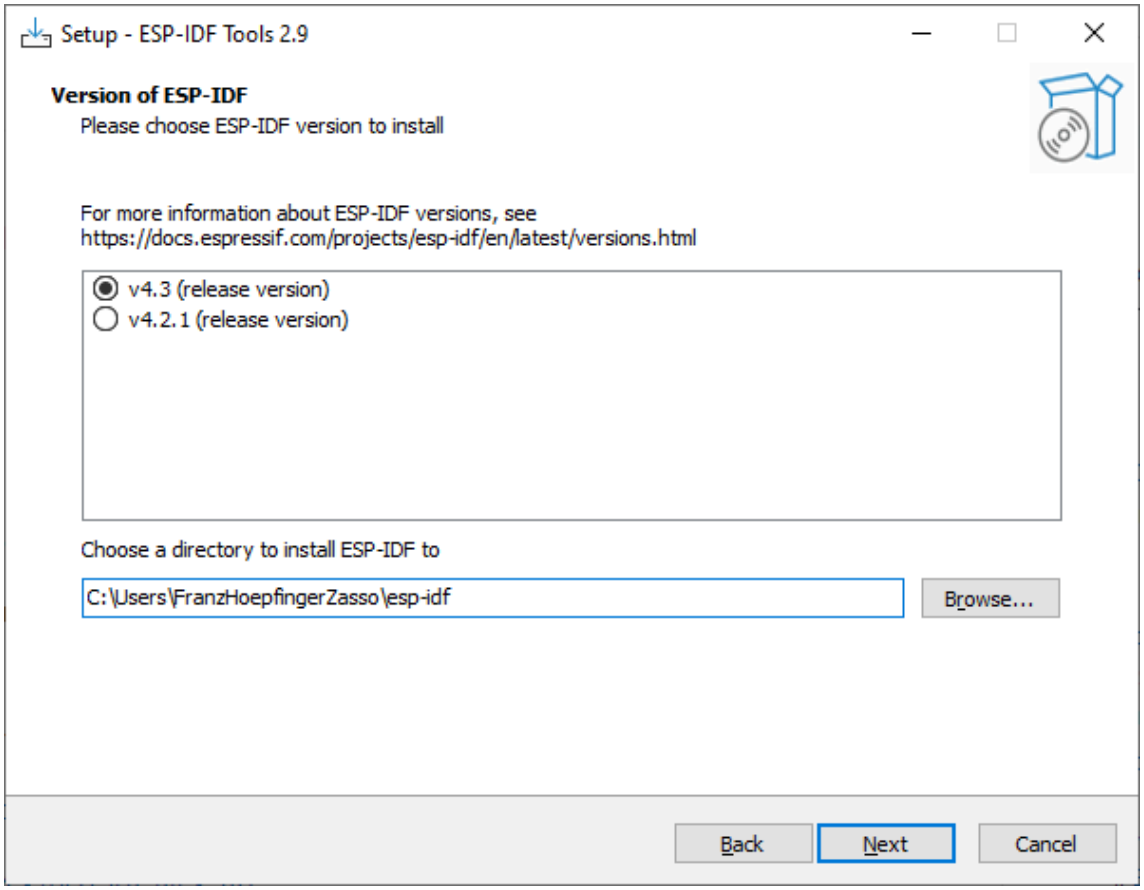




Please click “Apply Fixes”
and then yes.

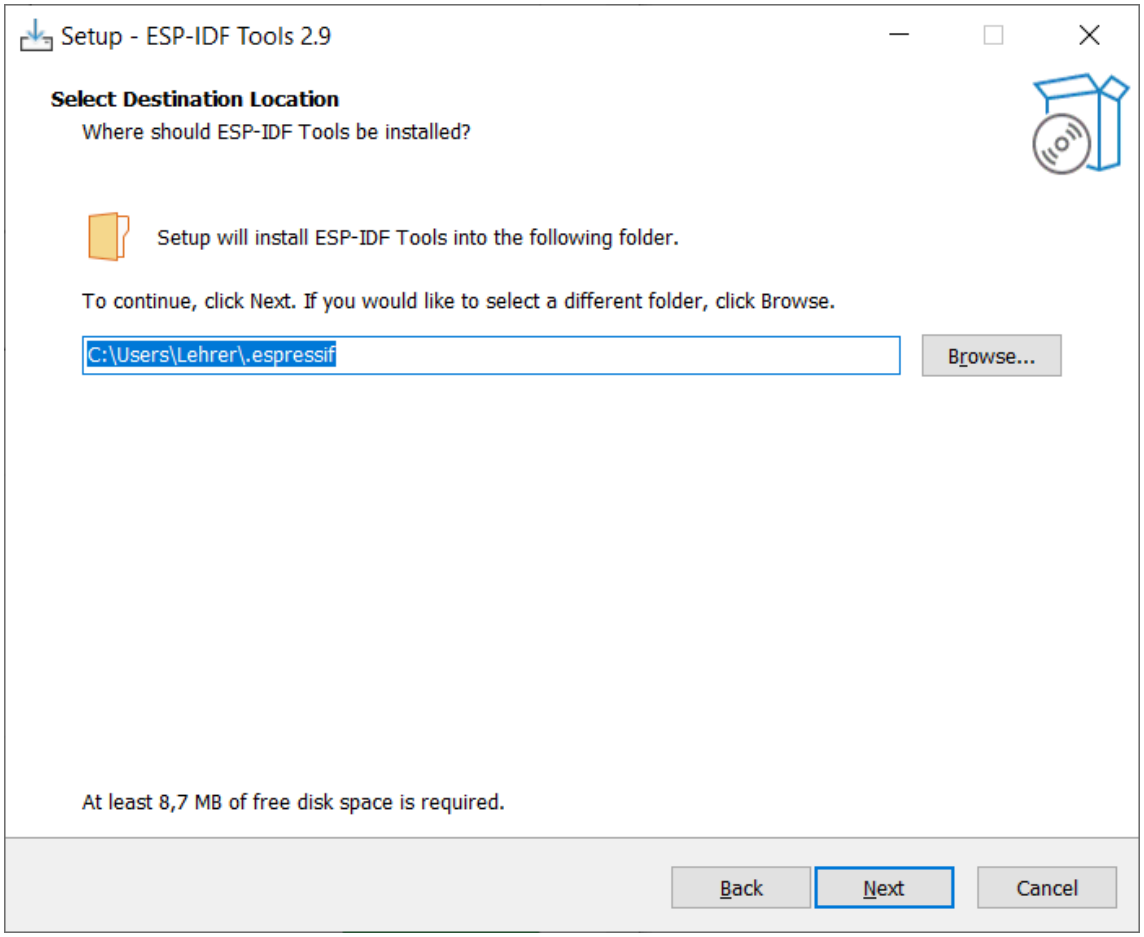
25.4 Step 4

Select 4.3
The installer will suggest “desktop” here, but I don’t like that!

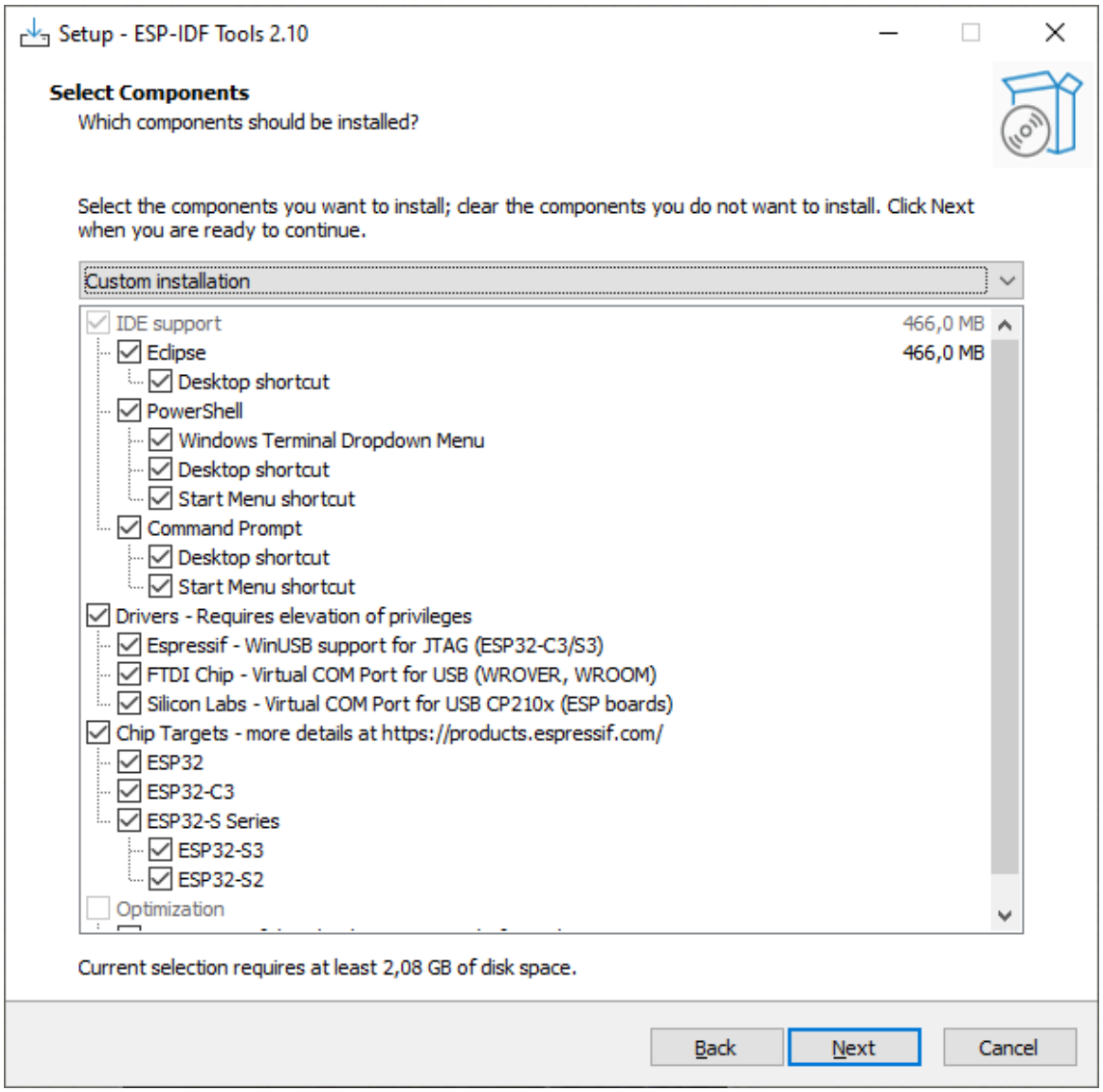


25.5 Step 5

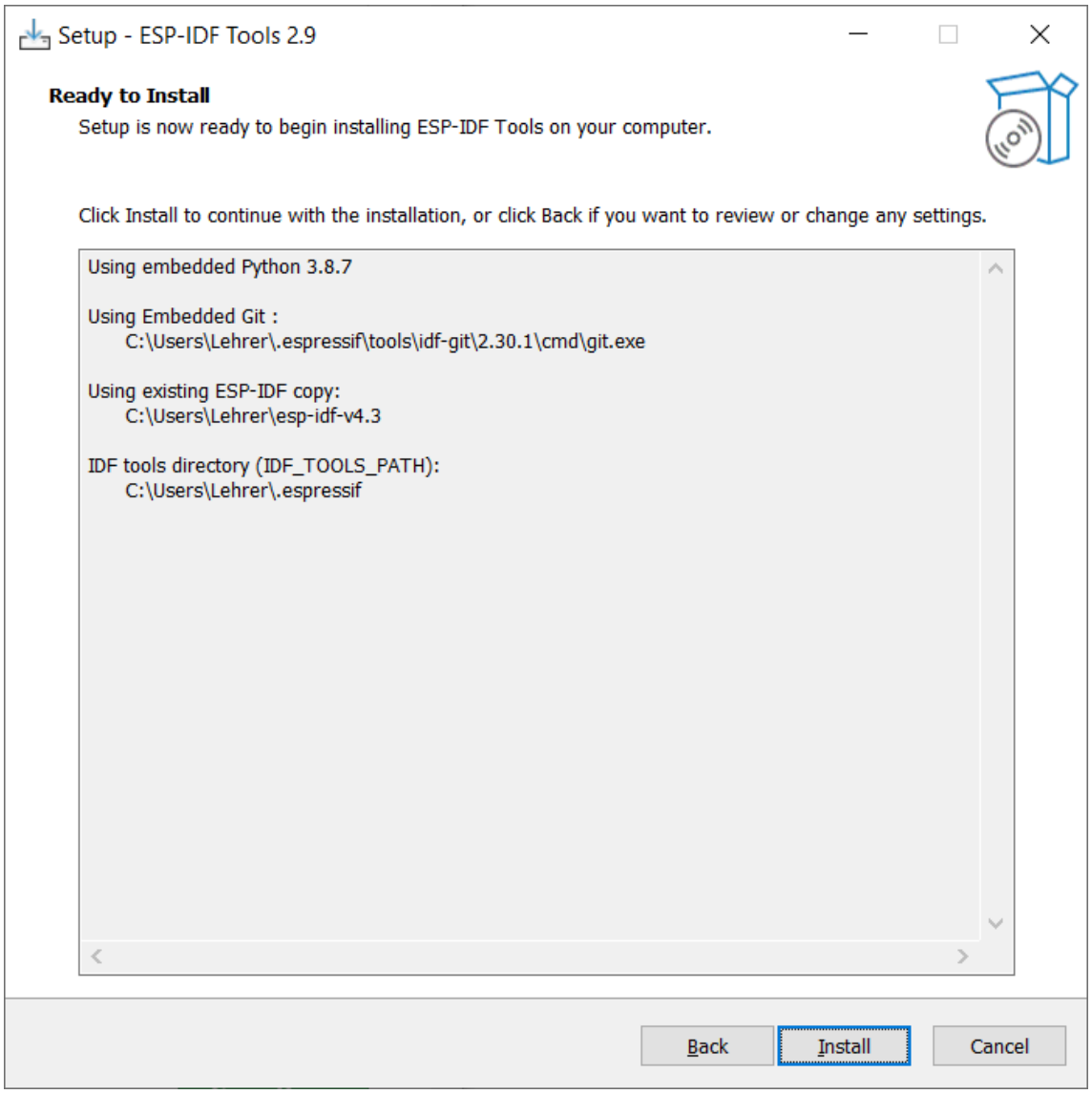
C:\Users\hoepffr\.espressif
This folder must also not contain any umlauts or spaces.


Setup - ESP-IDF Tools 2.9
Select Destination Location
Where should ESP-IDF Tools be installed?
Setup will install ESP-IDF Tools into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Users\Lehrer\.espressif
Browse...
At least 8,7 MB of free disk space is required.
Back
Next
Cancel

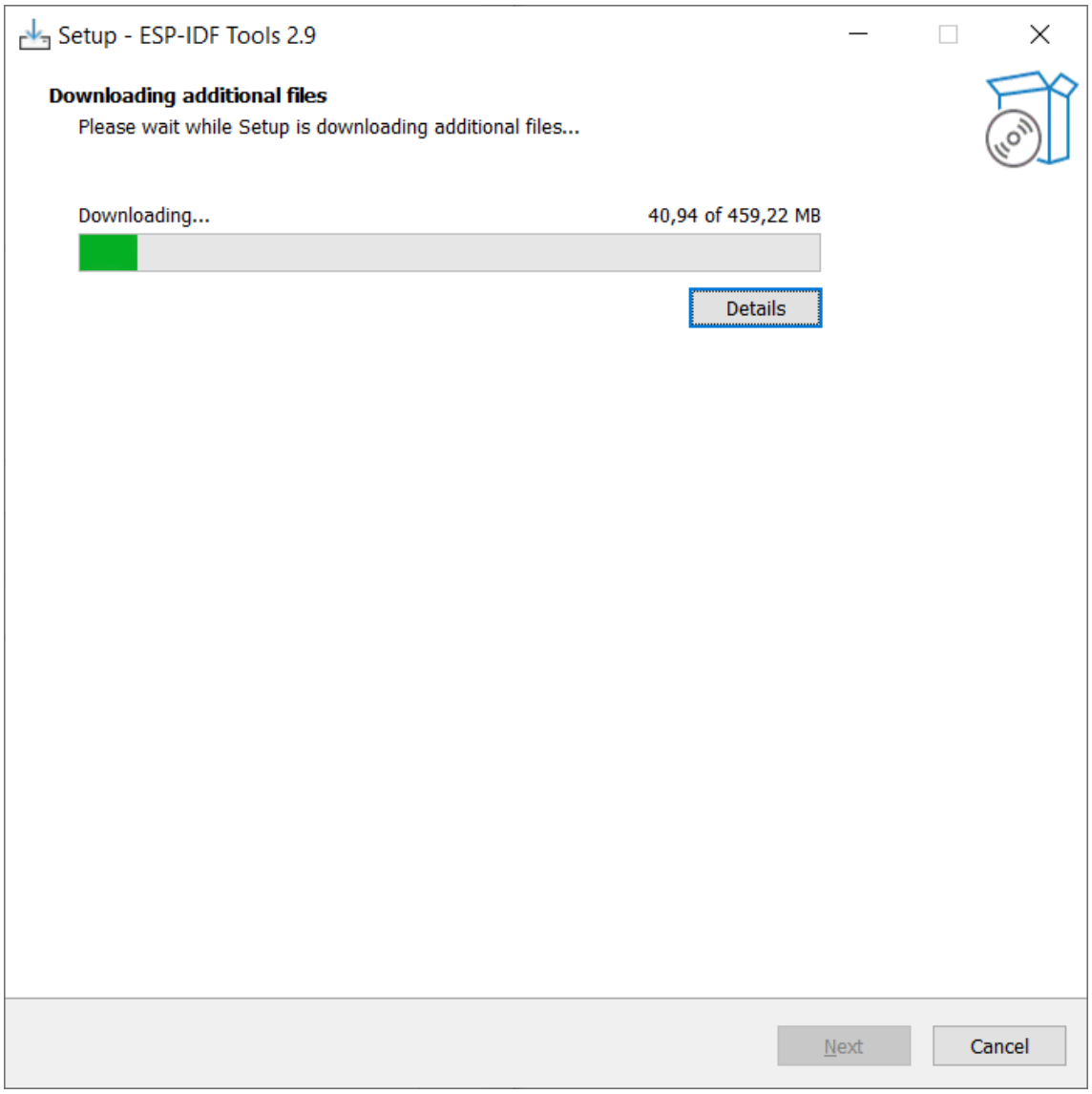
25.6 Step 6


Setup - ESP-IDF Tools 2.10
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Custom installation
IDE support 466,0 MB
Eclipse 466,0 MB
Desktop shortcut
PowerShell
Windows Terminal Dropdown Menu
Desktop shortcut
Start Menu shortcut
Command Prompt
Desktop shortcut
Start Menu shortcut
Drivers - Requires elevation of privileges
Espressif - WinUSB support for JTAG (ESP32-C3/S3)
FTDI Chip - Virtual COM Port for USB (WROVER, WROOM)
Silicon Labs - Virtual COM Port for USB CP210x (ESP boards)
Chip Targets - more details at https://products.espressif.com/
ESP32
ESP32-C3
ESP32-S Series
ESP32-S3
ESP32-S2
Optimization
Current selection requires at least 2,08 GB of disk space.
Back
Next
Cancel

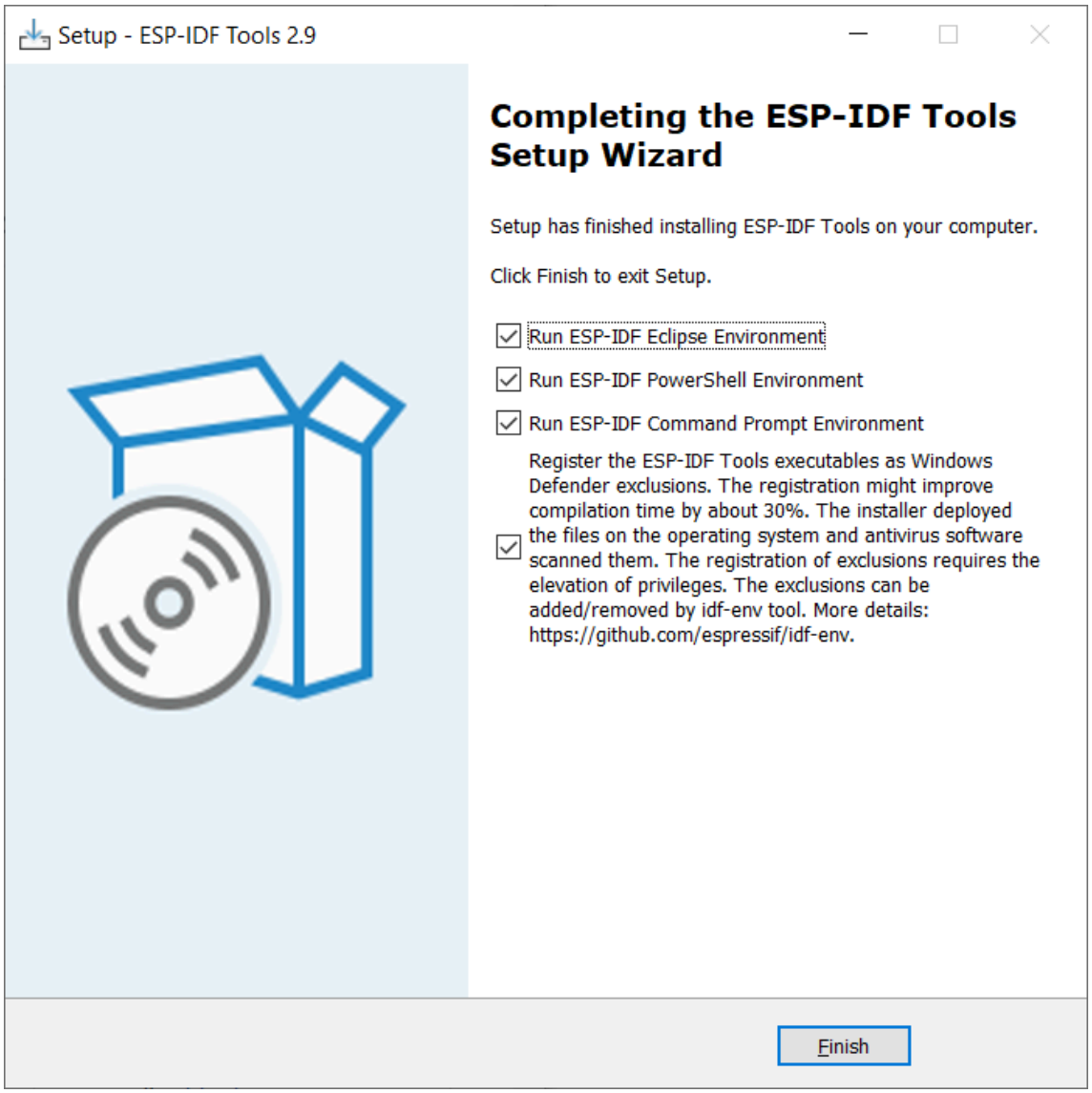
25.7 Step 7


Setup - ESP-IDF Tools 2.9
Ready to Install
Setup is now ready to begin installing ESP-IDF Tools on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Using embedded Python 3.8.7
Using Embedded Git :
C:\Users\Lehrer\.espressif\tools\idf-git\2.30.1\cmd\git.exe
Using existing ESP-IDF copy:
C:\Users\Lehrer\esp-idf-v4.3
IDF tools directory (IDF_TOOLS_PATH):
C:\Users\Lehrer\.espressif
Back
Install
Cancel

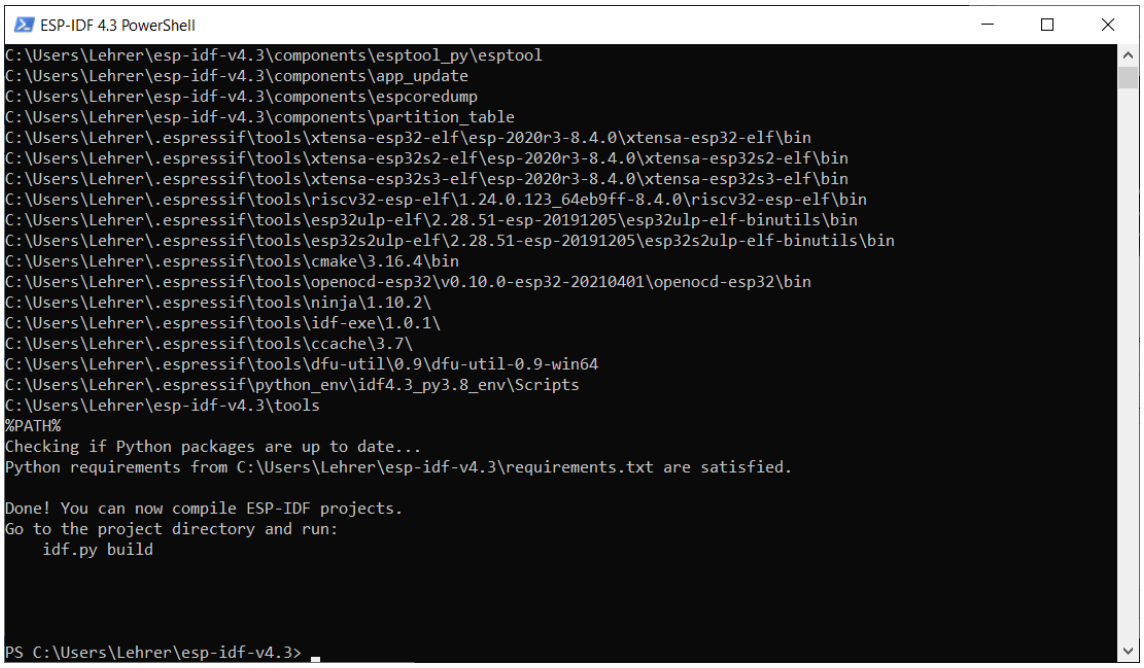
25.8 Step 8

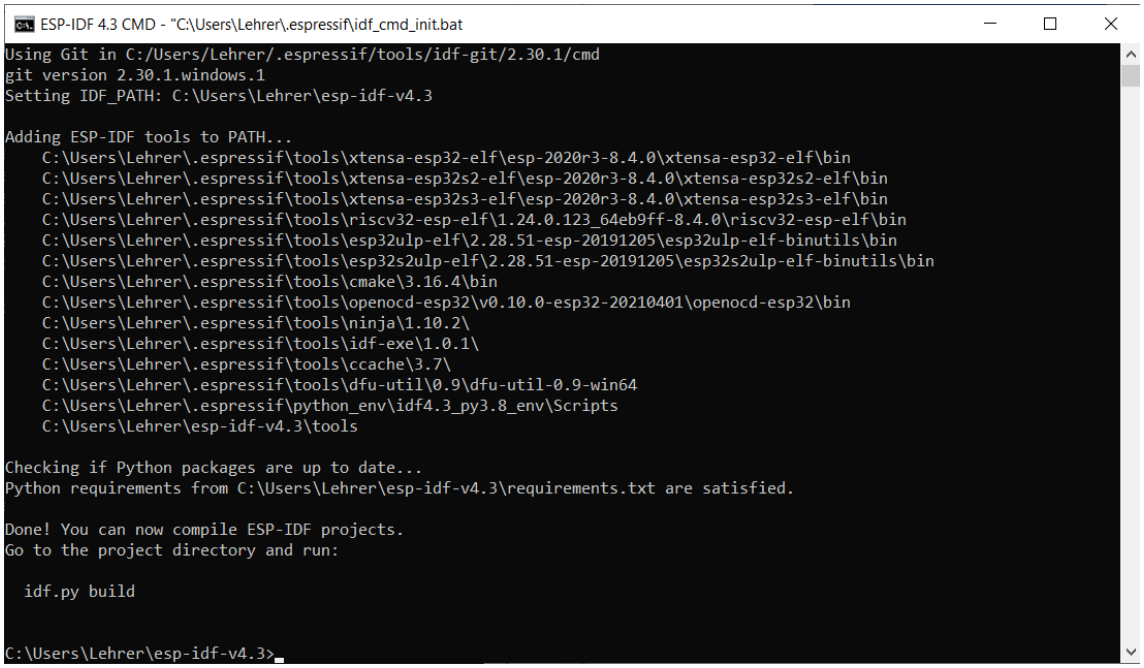

Setup - ESP-IDF Tools 2.9
Downloading additional files
Please wait while Setup is downloading additional files...
Downloading...
40,94 of 459,22 MB
Details
Next
Cancel

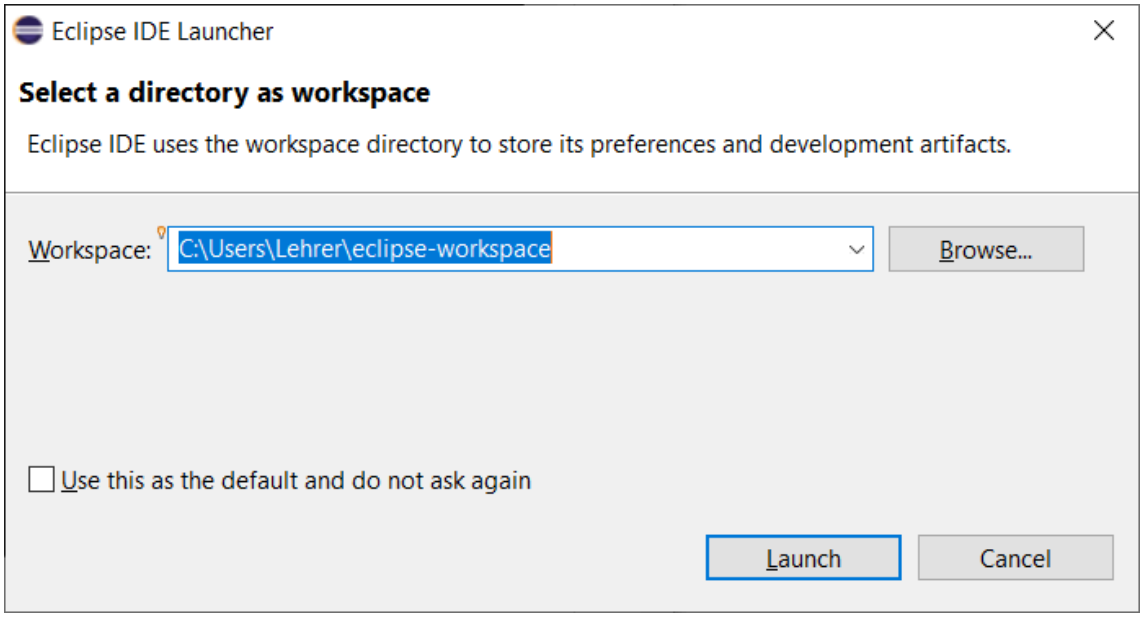
25.9 Step 9


Setup - ESP-IDF Tools 2.9
Completing the ESP-IDF Tools Setup Wizard
Setup has finished installing ESP-IDF Tools on your computer.
Click Finish to exit Setup.
Run ESP-IDF Eclipse Environment
Run ESP-IDF PowerShell Environment
Run ESP-IDF Command Prompt Environment
Register the ESP-IDF Tools executables as Windows Defender exclusions. The registration might improve compilation time by about 30%. The installer deployed the files on the operating system and antivirus software scanned them. The registration of exclusions requires the elevation of privileges. The exclusions can be added/removed by idf-env tool. More details: https://github.com/espressif/idf-env.
Finish

25.10 Step 10

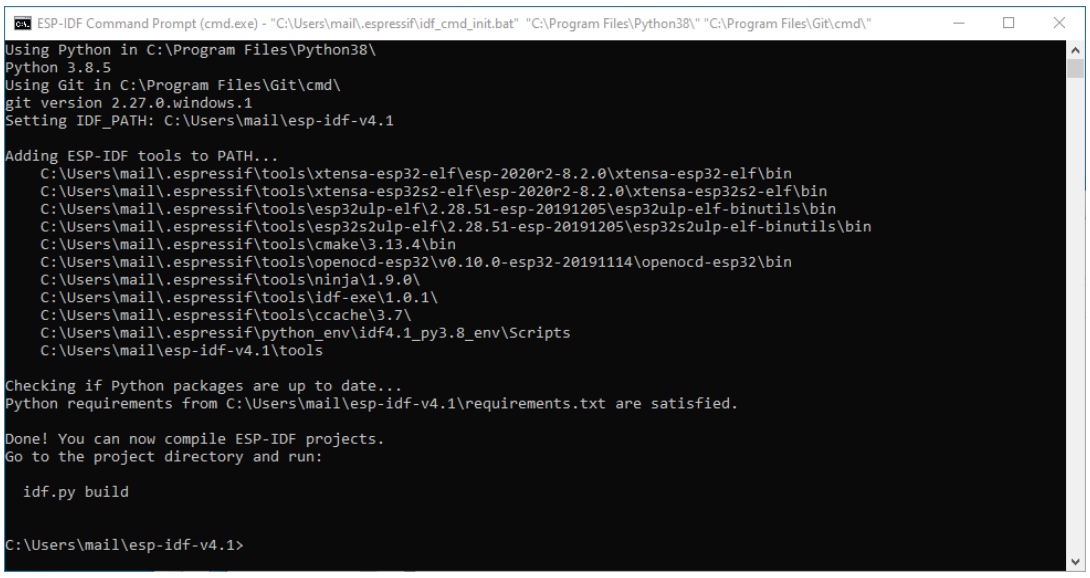

ESP-IDF 4.3 PowerShell
C:\Users\Lehrer\esp-idf-v4.3\components\esptool_py\esptool
C:\Users\Lehrer\esp-idf-v4.3\components\app_update
C:\Users\Lehrer\esp-idf-v4.3\components\espcoredump
C:\Users\Lehrer\esp-idf-v4.3\components\partition_table
C:\Users\Lehrer\.espressif\tools\xtensa-esp32-elf\esp-2020r3-8.4.0\xtensa-esp32-elf\bin
C:\Users\Lehrer\.espressif\tools\xtensa-esp32s2-elf\esp-2020r3-8.4.0\xtensa-esp32s2-elf\bin
C:\Users\Lehrer\.espressif\tools\xtensa-esp32s3-elf\esp-2020r3-8.4.0\xtensa-esp32s3-elf\bin
C:\Users\Lehrer\.espressif\tools\riscv32-esp-elf\1.24.0.123_64eb9ff-8.4.0\riscv32-esp-elf\bin
C:\Users\Lehrer\.espressif\tools\esp32ulp-elf\2.28.51-esp-20191205\esp32ulp-elf-binutils\bin
C:\Users\Lehrer\.espressif\tools\esp32s2ulp-elf\2.28.51-esp-20191205\esp32s2ulp-elf-binutils\bin
C:\Users\Lehrer\.espressif\tools\cmake\3.16.4\bin
C:\Users\Lehrer\.espressif\tools\openocd-esp32\v0.10.0-esp32-20210401\openocd-esp32\bin
C:\Users\Lehrer\.espressif\tools\ninja\1.10.2\
C:\Users\Lehrer\.espressif\tools\idf-exe\1.0.1\
C:\Users\Lehrer\.espressif\tools\ccache\3.7\
C:\Users\Lehrer\.espressif\tools\dfu-util\0.9\dfu-util-0.9-win64
C:\Users\Lehrer\.espressif\python_env\idf4.3_py3.8_env\Scripts
C:\Users\Lehrer\esp-idf-v4.3\tools
%PATH%
Checking if Python packages are up to date...
Python requirements from C:\Users\Lehrer\esp-idf-v4.3\requirements.txt are satisfied.
Done! You can now compile ESP-IDF projects.
Go to the project directory and run:
idf.py build
PS C:\Users\Lehrer\esp-idf-v4.3>


ESP-IDF 4.3 CMD - "C:\Users\Lehrer\.espressif\idf_cmd_init.bat
Using Git in C:/Users/Lehrer/.espressif/tools/idf-git/2.30.1/cmd
git version 2.30.1.windows.1
Setting IDF_PATH: C:\Users\Lehrer\esp-idf-v4.3
Adding ESP-IDF tools to PATH...
C:\Users\Lehrer\.espressif\tools\xtensa-esp32-elf\esp-2020r3-8.4.0\xtensa-esp32-elf\bin
C:\Users\Lehrer\.espressif\tools\xtensa-esp32s2-elf\esp-2020r3-8.4.0\xtensa-esp32s2-elf\bin
C:\Users\Lehrer\.espressif\tools\xtensa-esp32s3-elf\esp-2020r3-8.4.0\xtensa-esp32s3-elf\bin
C:\Users\Lehrer\.espressif\tools\riscv32-esp-elf\1.24.0.123_64eb9ff-8.4.0\riscv32-esp-elf\bin
C:\Users\Lehrer\.espressif\tools\esp32ulp-elf\2.28.51-esp-20191205\esp32ulp-elf-binutils\bin
C:\Users\Lehrer\.espressif\tools\esp32s2ulp-elf\2.28.51-esp-20191205\esp32s2ulp-elf-binutils\bin
C:\Users\Lehrer\.espressif\tools\cmake\3.16.4\bin
C:\Users\Lehrer\.espressif\tools\openocd-esp32\v0.10.0-esp32-20210401\openocd-esp32\bin
C:\Users\Lehrer\.espressif\tools\ninja\1.10.2\
C:\Users\Lehrer\.espressif\tools\idf-exe\1.0.1\
C:\Users\Lehrer\.espressif\tools\ccache\3.7\
C:\Users\Lehrer\.espressif\tools\dfu-util\0.9\dfu-util-0.9-win64
C:\Users\Lehrer\.espressif\python_env\idf4.3_py3.8_env\Scripts
C:\Users\Lehrer\esp-idf-v4.3\tools
Checking if Python packages are up to date...
Python requirements from C:\Users\Lehrer\esp-idf-v4.3\requirements.txt are satisfied.
Done! You can now compile ESP-IDF projects.
Go to the project directory and run:
idf.py build
C:\Users\Lehrer\esp-idf-v4.3>



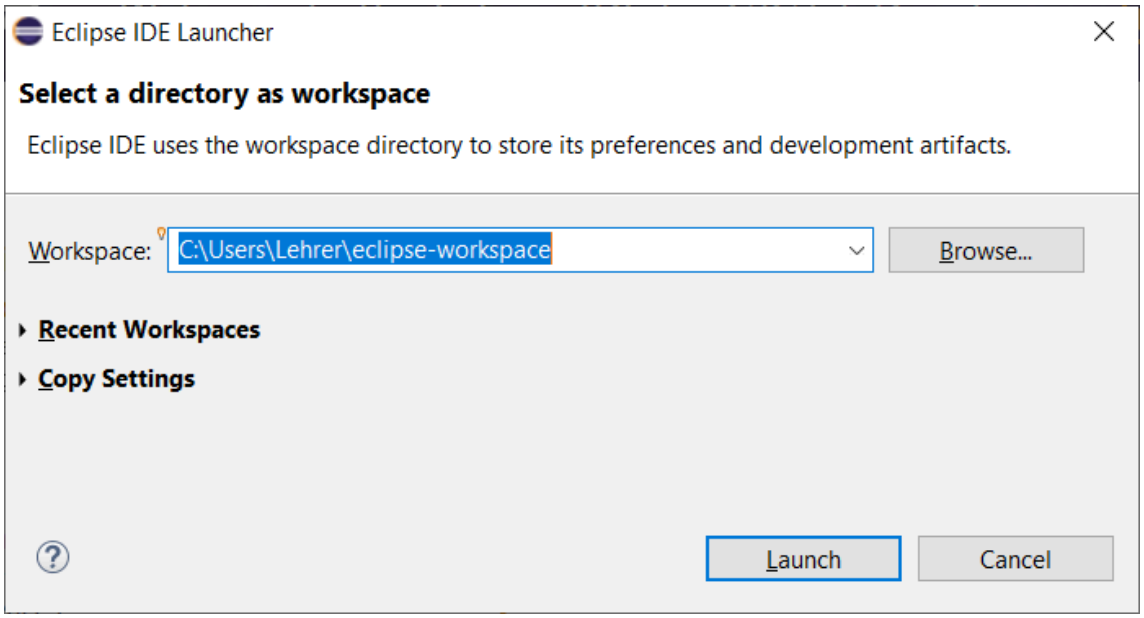
25.11 Step 11: (Information only)

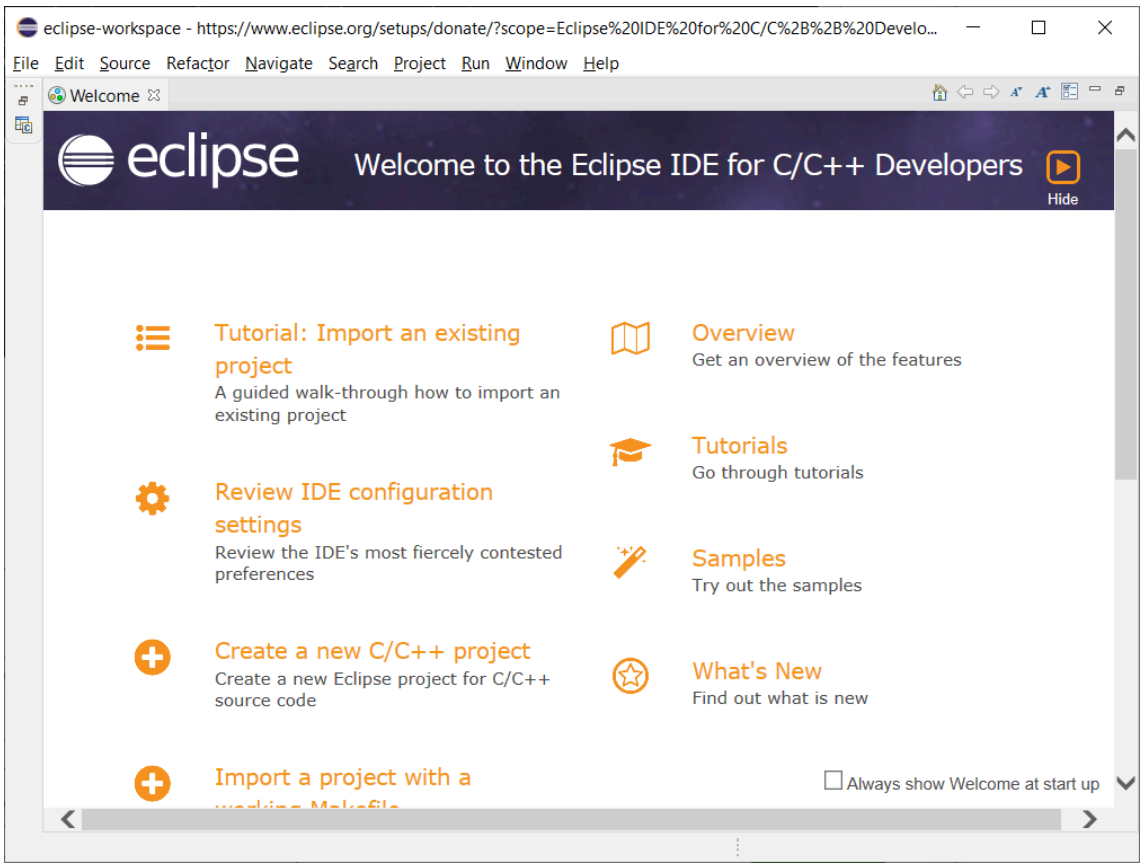
idf.py menuconfig
idf.py build
https://docs.espressif.com/projects/esp-idf/en/latest/get-started/index.html#step-6-connect-your-device
https://docs.espressif.com/projects/esp-idf/en/latest/get-started/establish-serial-connection.html
https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers
https://www.ftdichip.com/Drivers/VCP.htm
https://docs.espressif.com/projects/espressif-esp-iot-solution/en/latest/hw-reference/ESP-Prog_guide.html
idf.py flash
idf.py monitor
So, you could already work from the command line here.



25.12 Step 12

Now open Eclipse and create a new workspace.





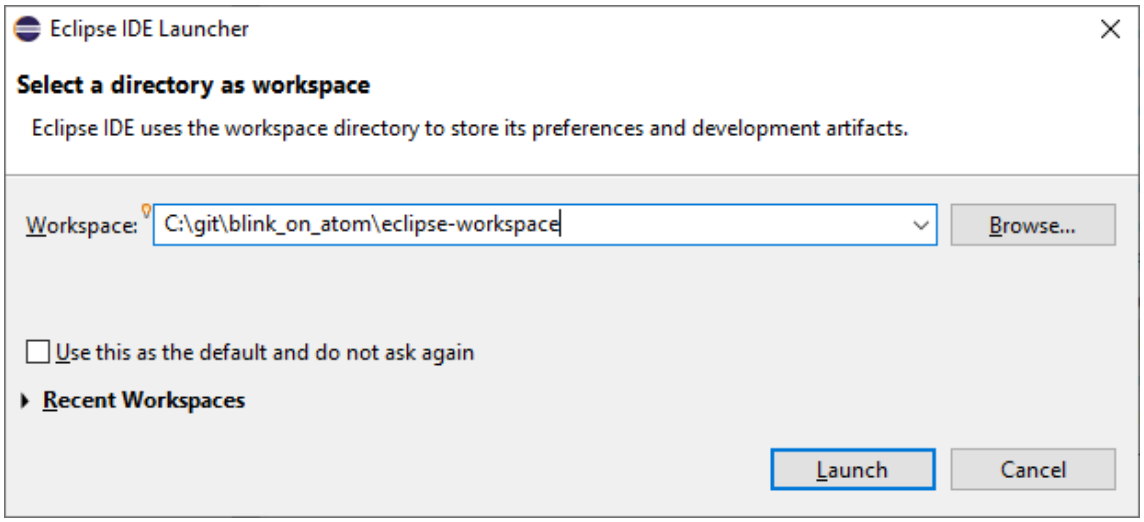
Done.

You don't need to install Java, Python, or Git beforehand.

26. blink_on_atom

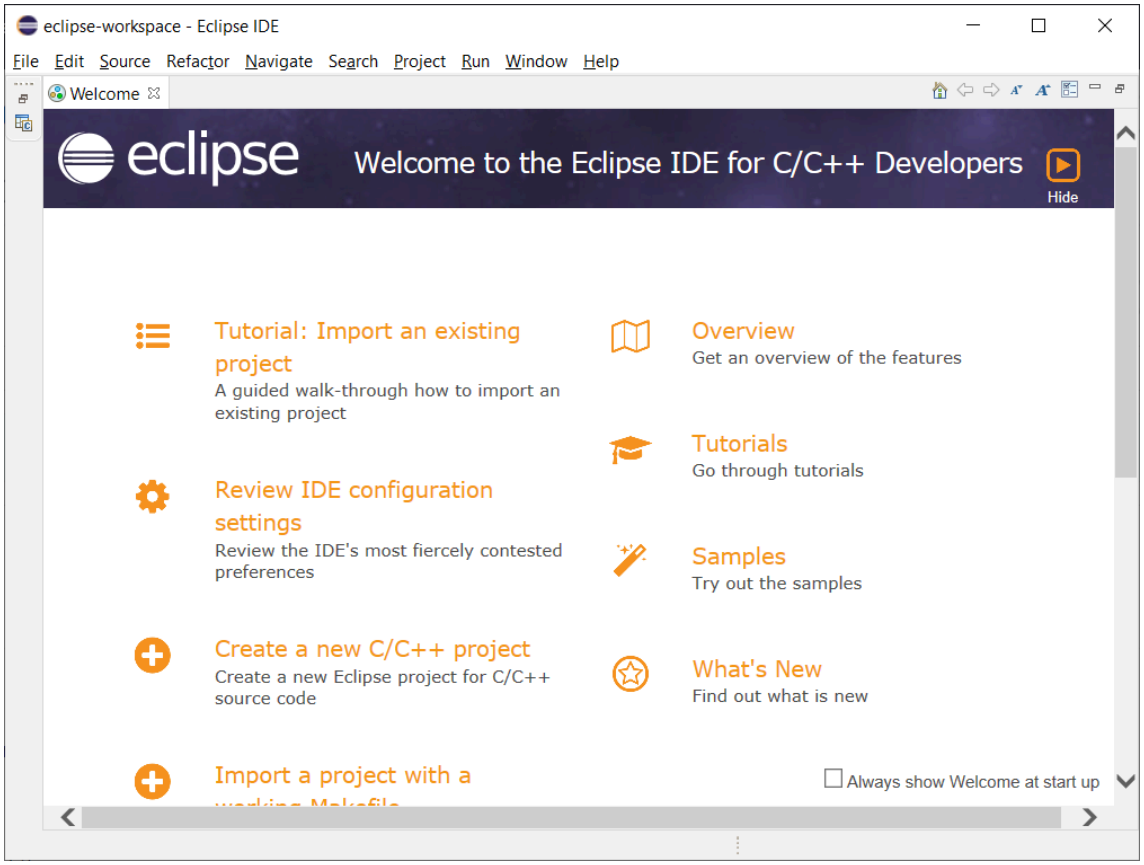
26.1 Step 14

Now you can actually start Eclipse: For your first project, use this workspace: C:\git\blink_on_atom\eclipse-workspace (unless you checked out to D:\git earlier)

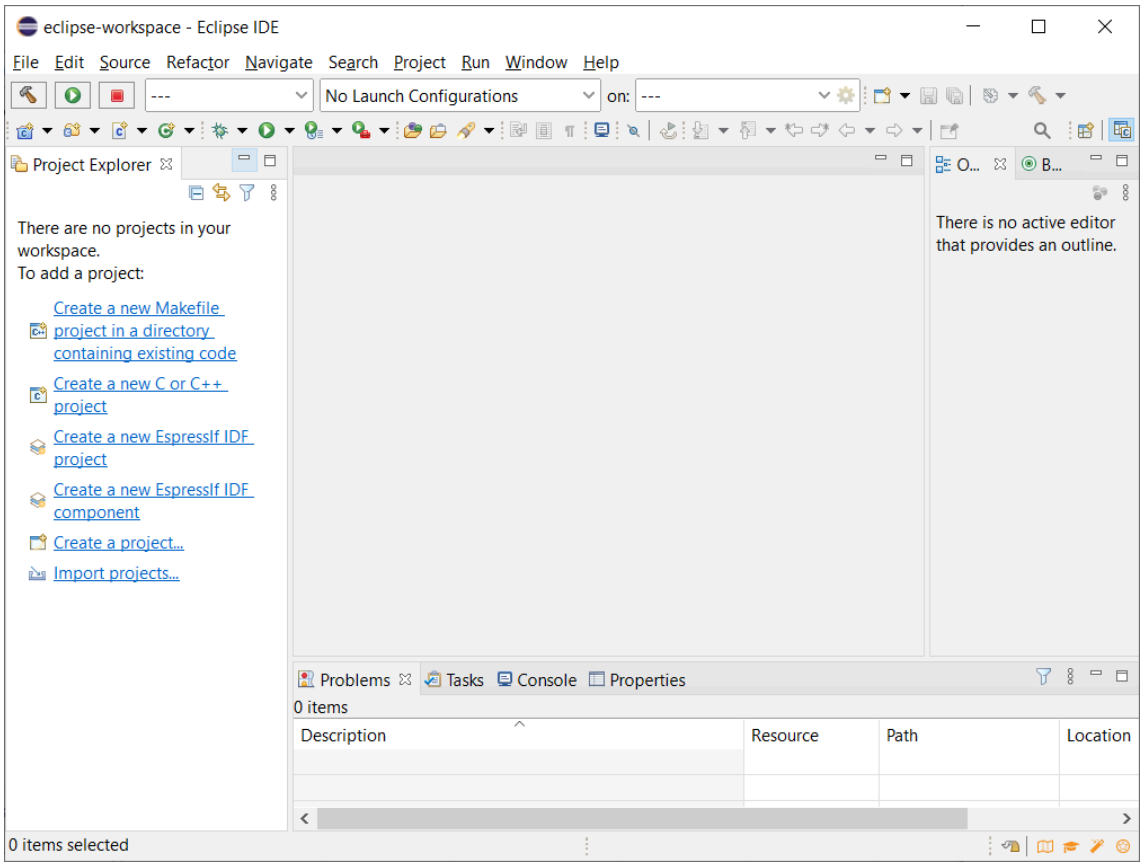


26.2 Step 15

Workbench –> Hide (top right)

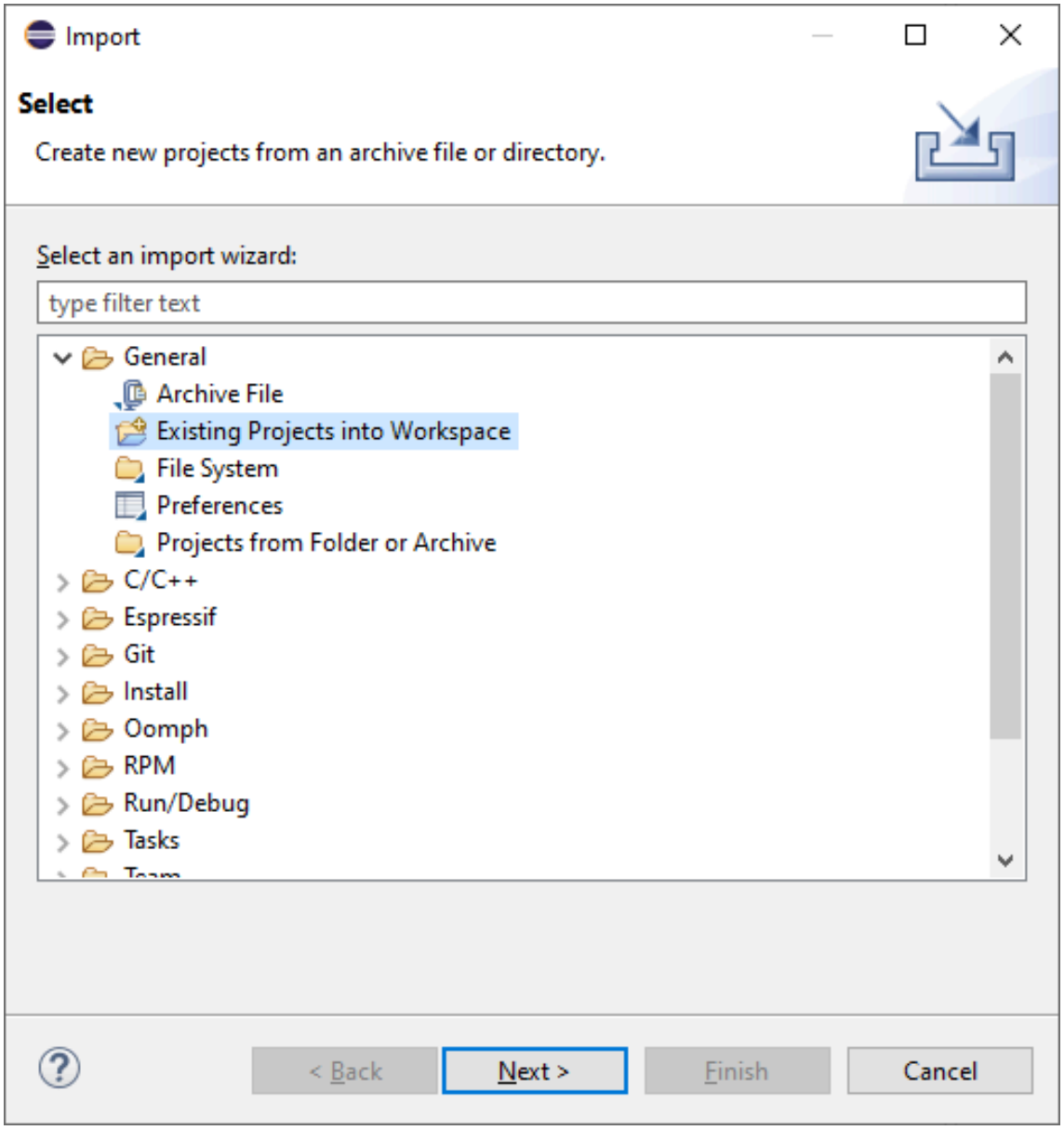

eclipse-workspace - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Welcome
eclipse
Welcome to the Eclipse IDE for C/C++ Developers
Hide
Tutorial: Import an existing project
A guided walk-through how to import an existing project
Review IDE configuration settings
Review the IDE's most fiercely contested preferences
Create a new C/C++ project
Create a new Eclipse project for C/C++ source code
Import a project with a
Overview
Get an overview of the features
Tutorials
Go through tutorials
Samples
Try out the samples
What's New
Find out what is new
Always show Welcome at start up

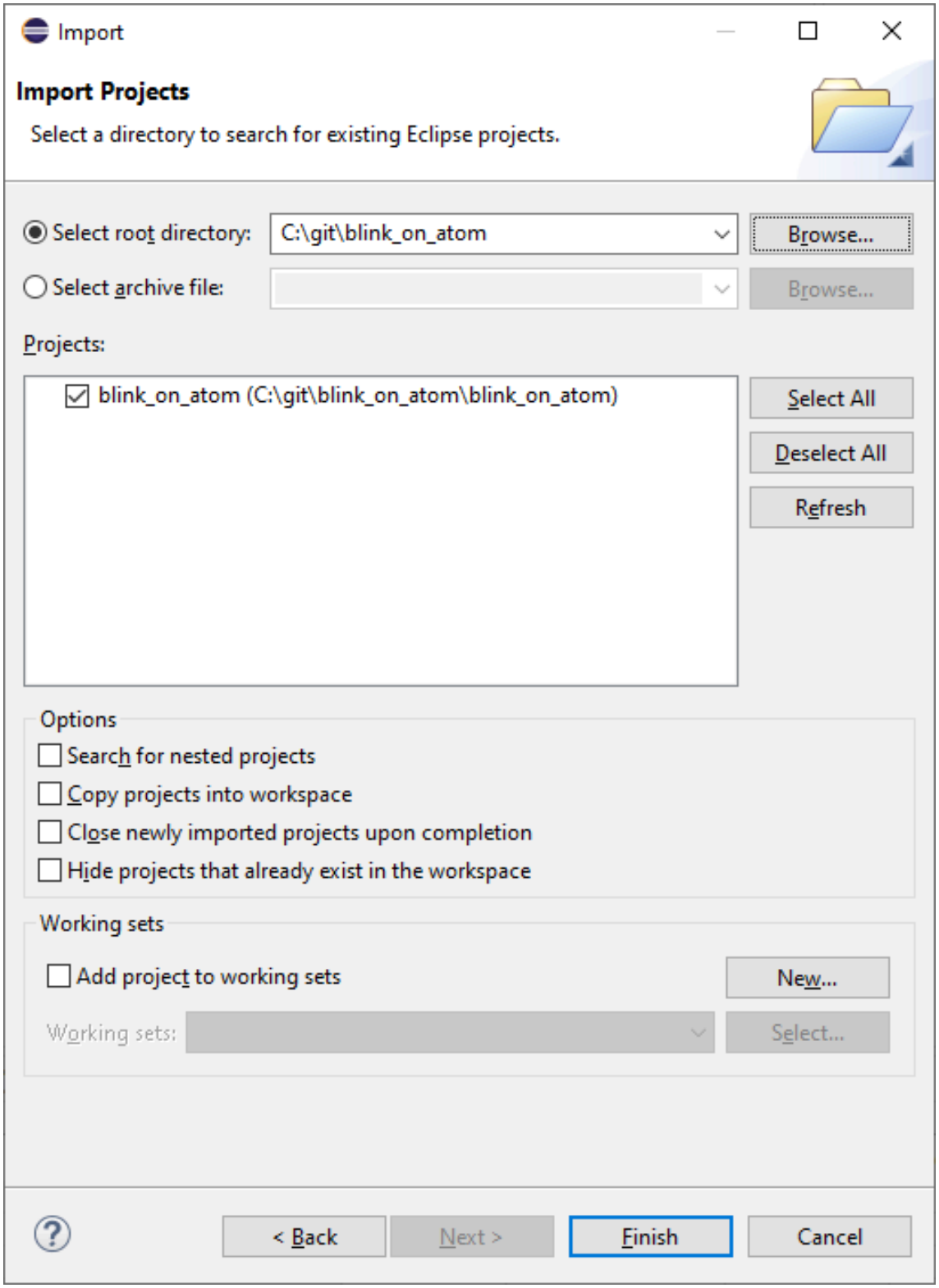
26.3 Step 16


eclipse-workspace - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
No Launch Configurations
on:
Project Explorer
There are no projects in your workspace.
To add a project:
Create a new Makefile project in a directory containing existing code
Create a new C or C++ project
Create a new EspressIf IDF project
Create a new EspressIf IDF component
Create a project...
Import projects...
There is no active editor that provides an outline.
Problems
Tasks
Console
Properties
0 items
Description
Resource
Path
Location
0 items selected

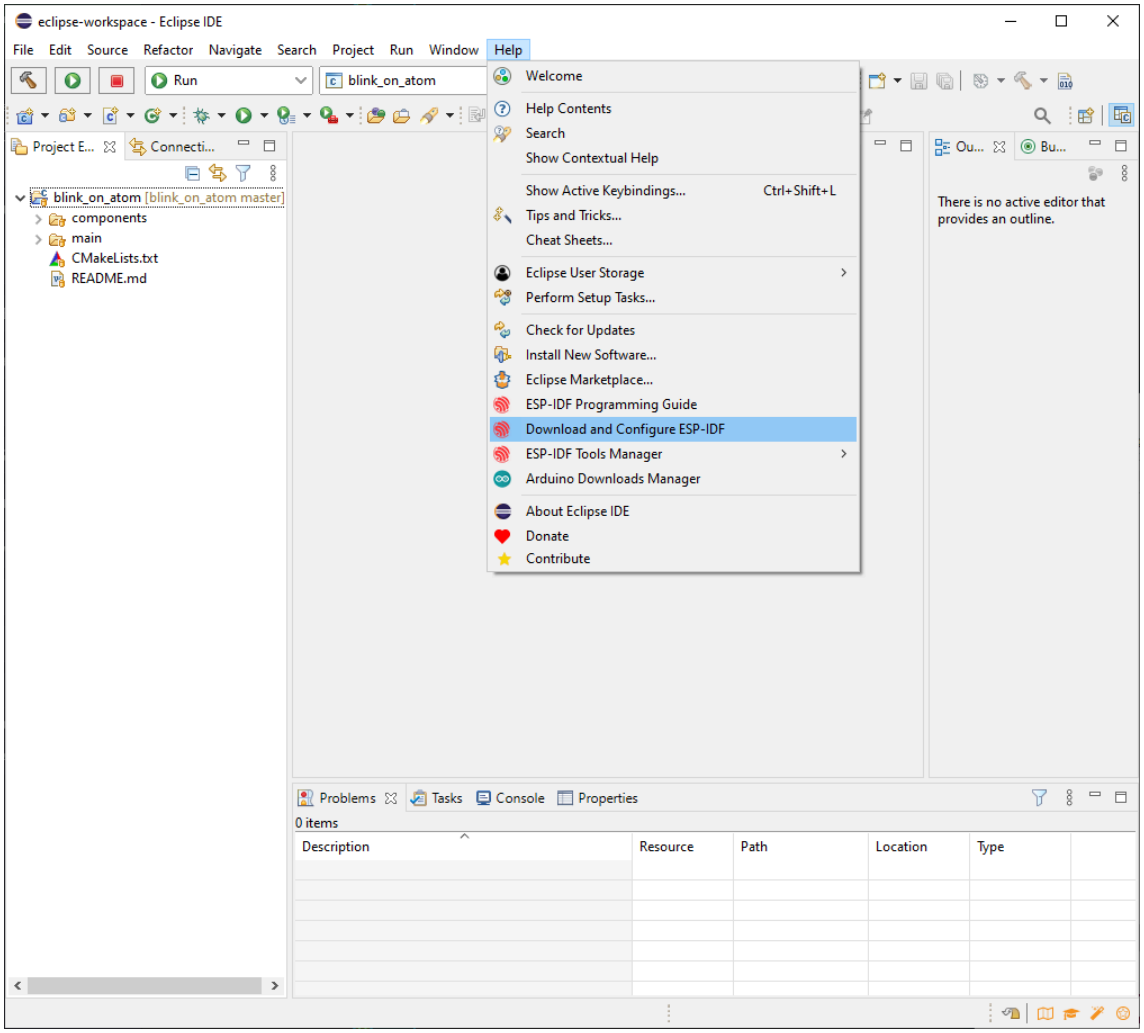
26.4 Step 17

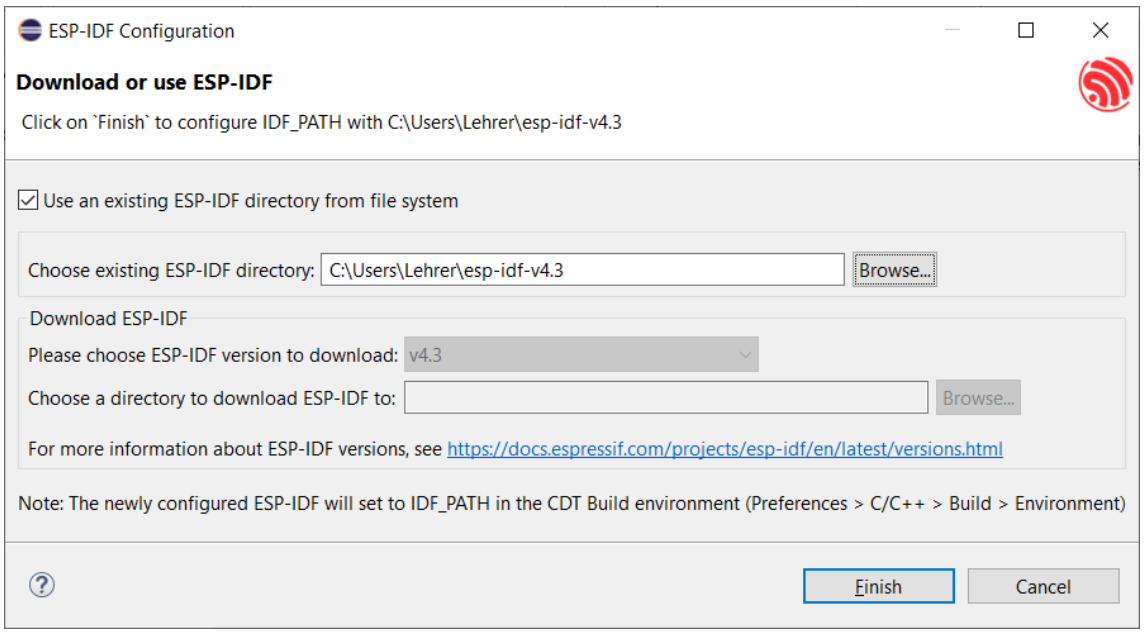
Now go to Import Projects
(You will repeat this process for other projects later!!! Remember this!!!)

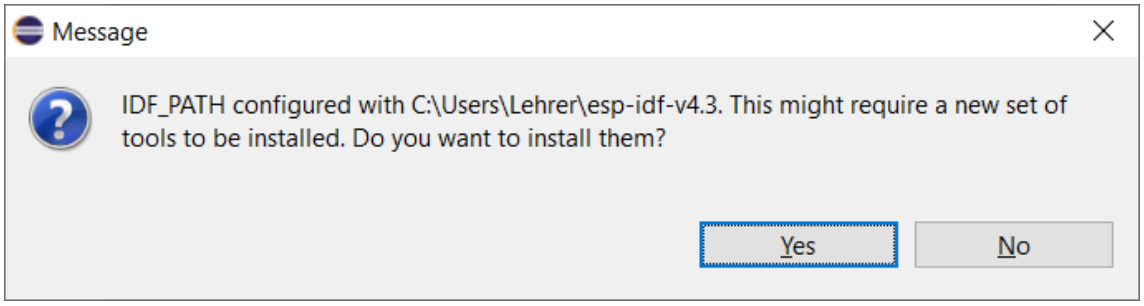




ALWAYS When you open a new workspace (that you've never opened on this PC before) ALWAYS see: <https://github.com/espressif/idf-eclipse-plugin/blob/master/README.md#installing-esp-idf-tools

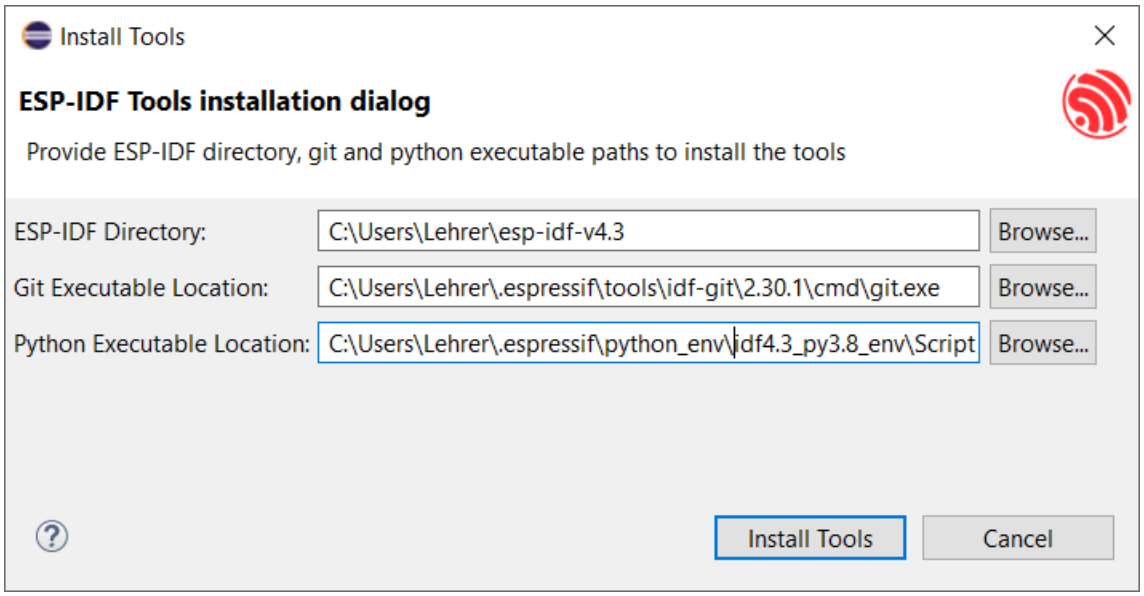

eclipse-workspace - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Run
blink_on_atom
Welcome
Help Contents
Search
Show Contextual Help
Show Active Keybindings... Ctrl+Shift+L
Tips and Tricks...
Cheat Sheets...
Eclipse User Storage
Perform Setup Tasks...
Check for Updates
Install New Software...
Eclipse Marketplace...
ESP-IDF Programming Guide
Download and Configure ESP-IDF
ESP-IDF Tools Manager
Arduino Downloads Manager
About Eclipse IDE
Donate
Contribute
Project E...
Connecti...
blink_on_atom [blink_on_atom master]
components
main
CMakeLists.txt
README.md
There is no active editor that provides an outline.
Problems
Tasks
Console
Properties
0 items
Description
Resource
Path
Location
Type


ESP-IDF Configuration
Download or use ESP-IDF
Click on `Finish` to configure IDF_PATH with C:\Users\Lehrer\esp-idf-v4.3
Use an existing ESP-IDF directory from file system
Choose existing ESP-IDF directory: C:\Users\Lehrer\esp-idf-v4.3
Browse...
Download ESP-IDF
Please choose ESP-IDF version to download: v4.3
Choose a directory to download ESP-IDF to:
Browse...
For more information about ESP-IDF versions, see https://docs.espressif.com/projects/esp-idf/en/latest/versions.html
Note: The newly configured ESP-IDF will set to IDF_PATH in the CDT Build environment (Preferences > C/C++ > Build > Environment)
Finish
Cancel



YES

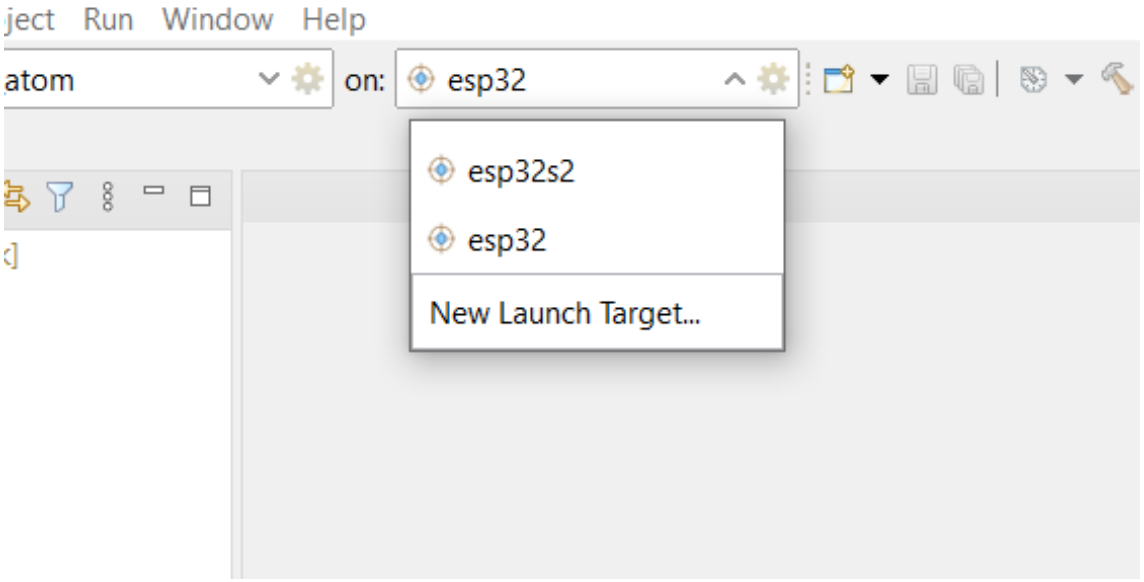
26.5 Step 18

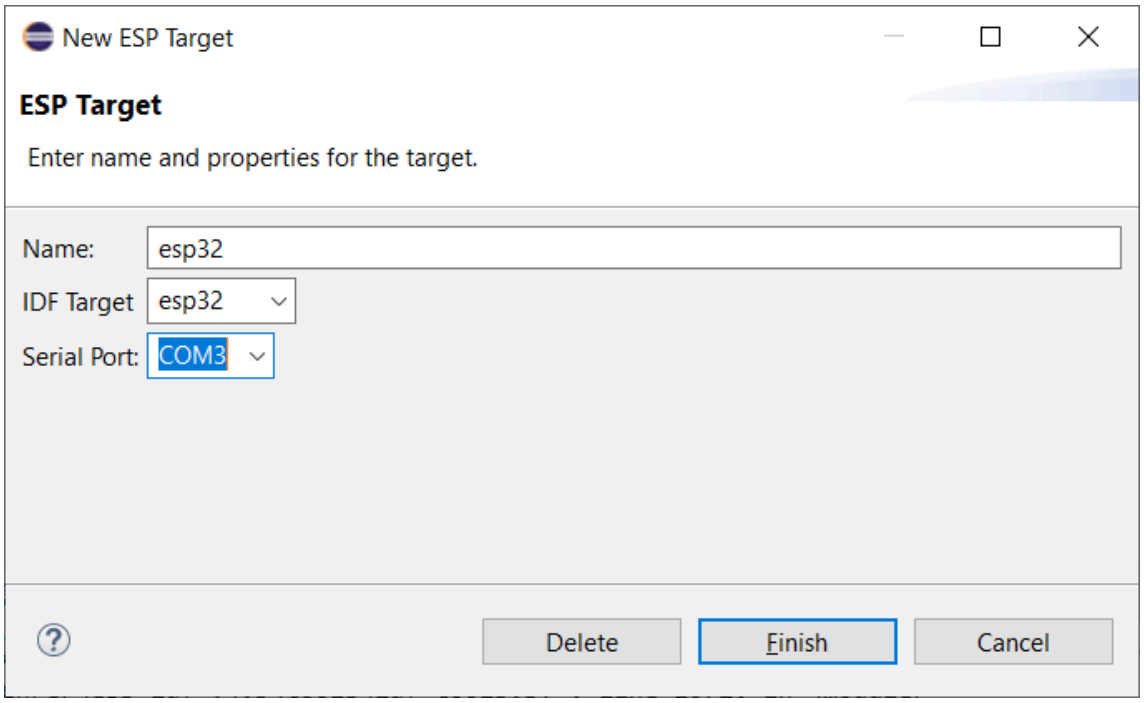


C:\Users\Teacher\esp-idf-v4.3
C:\Users\Teacher.espressif\tools\idf-git\2.30.1\cmd\git.exe
C:\Users\Teacher.espressif\python_env\idf4.3_py3.8_env\Scripts\python.exe

26.6 Step 20

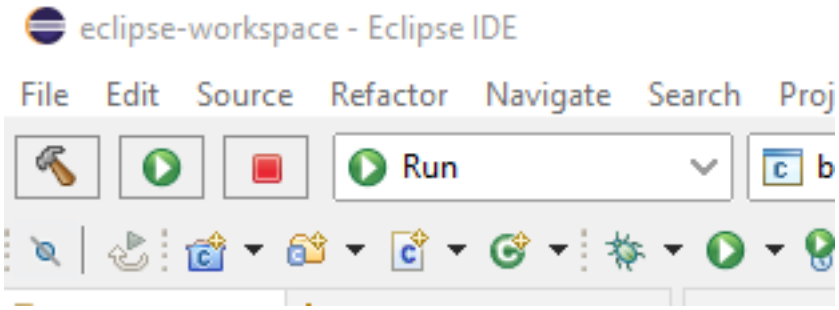
<https://github.com/espressif/idf-eclipse-plugin/blob/master/README.md#configuring-launch-target



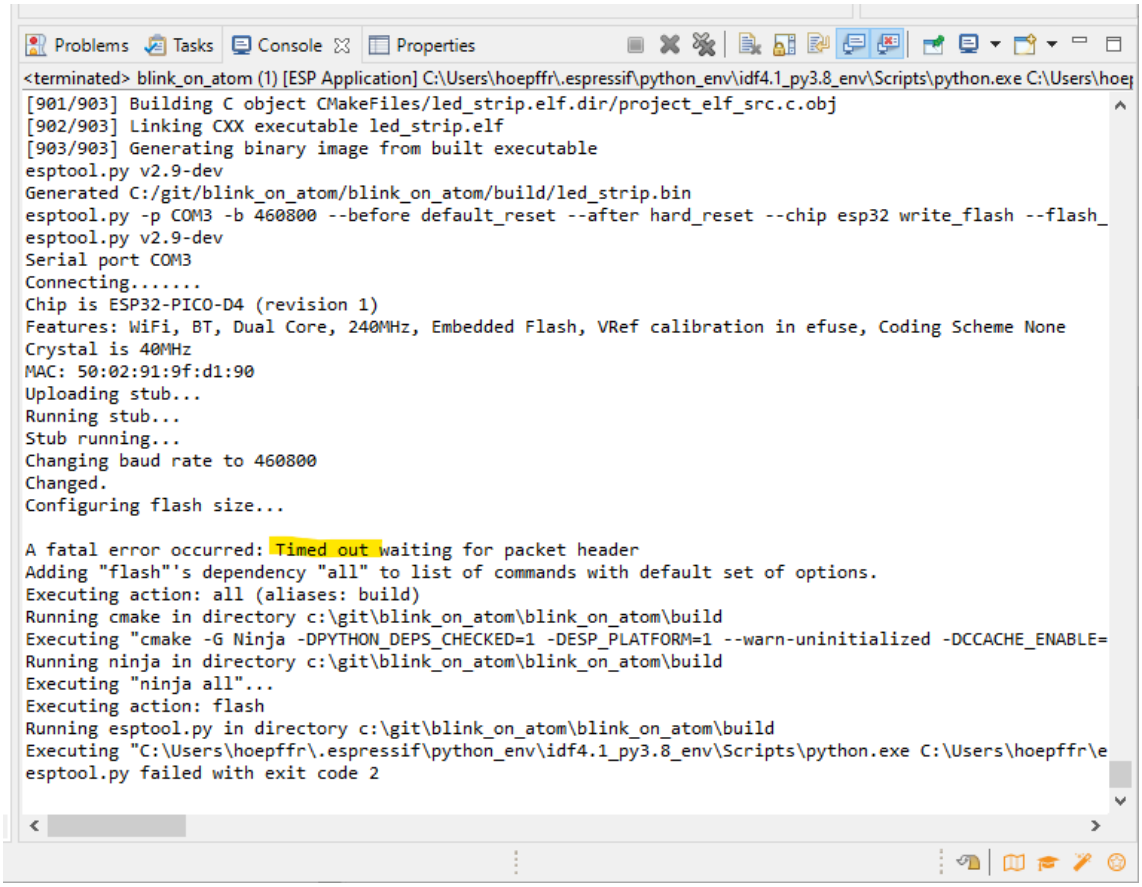


26.7 Step 21

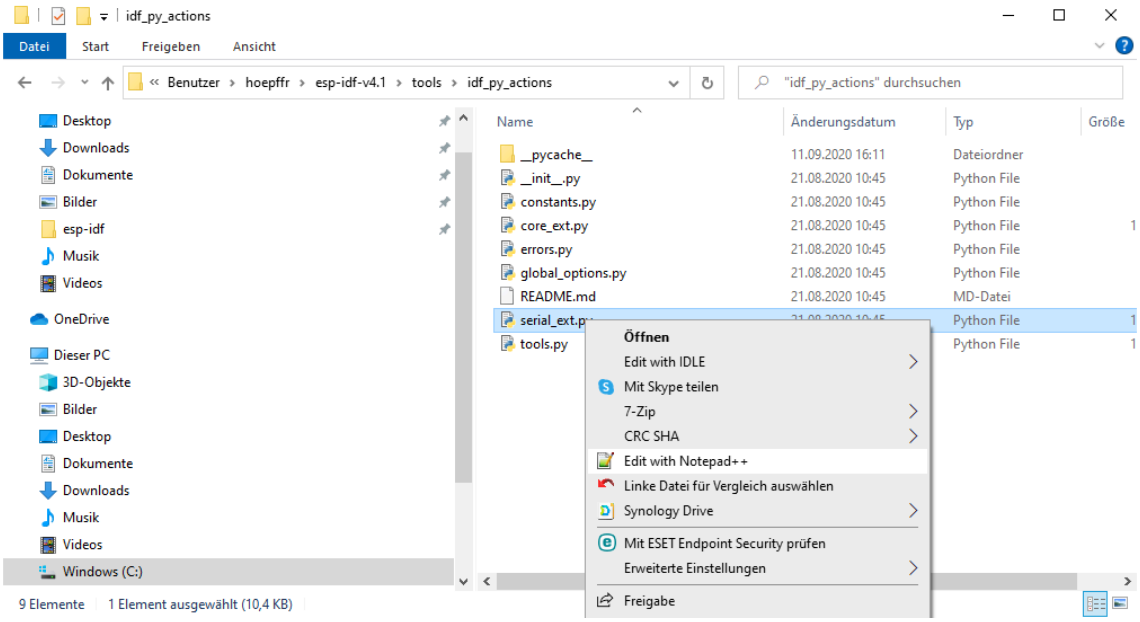
First press the hammer,
then the “Play” button. Button.



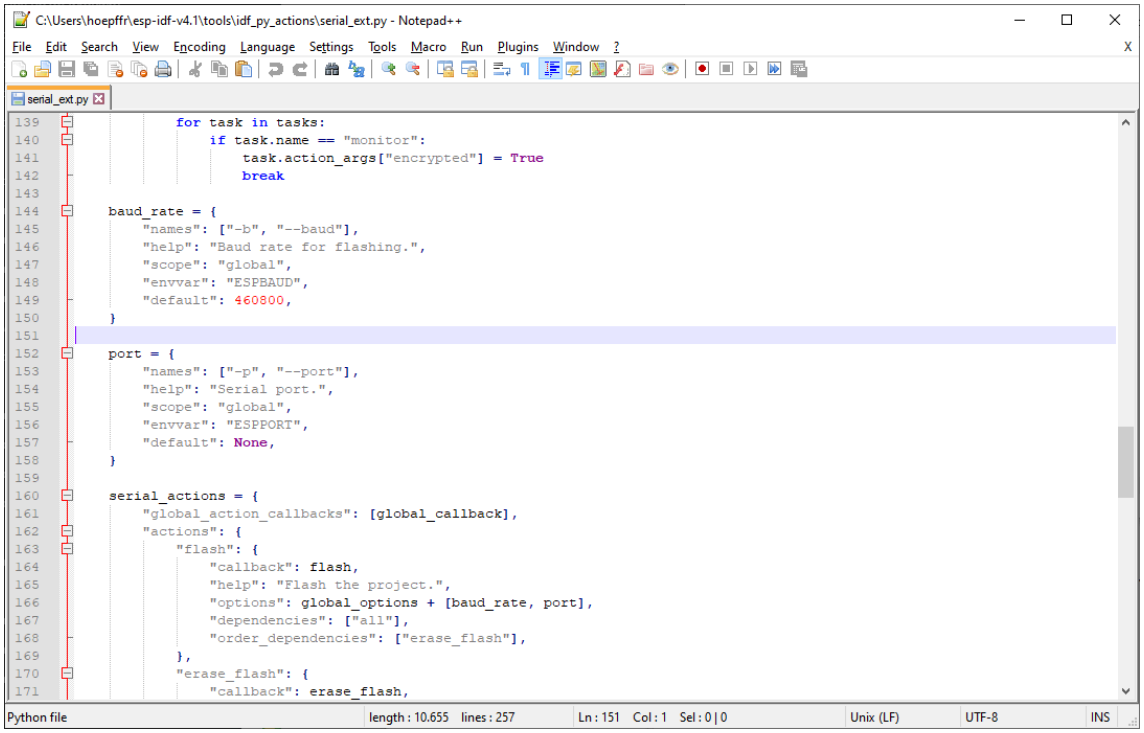
If the following appears:



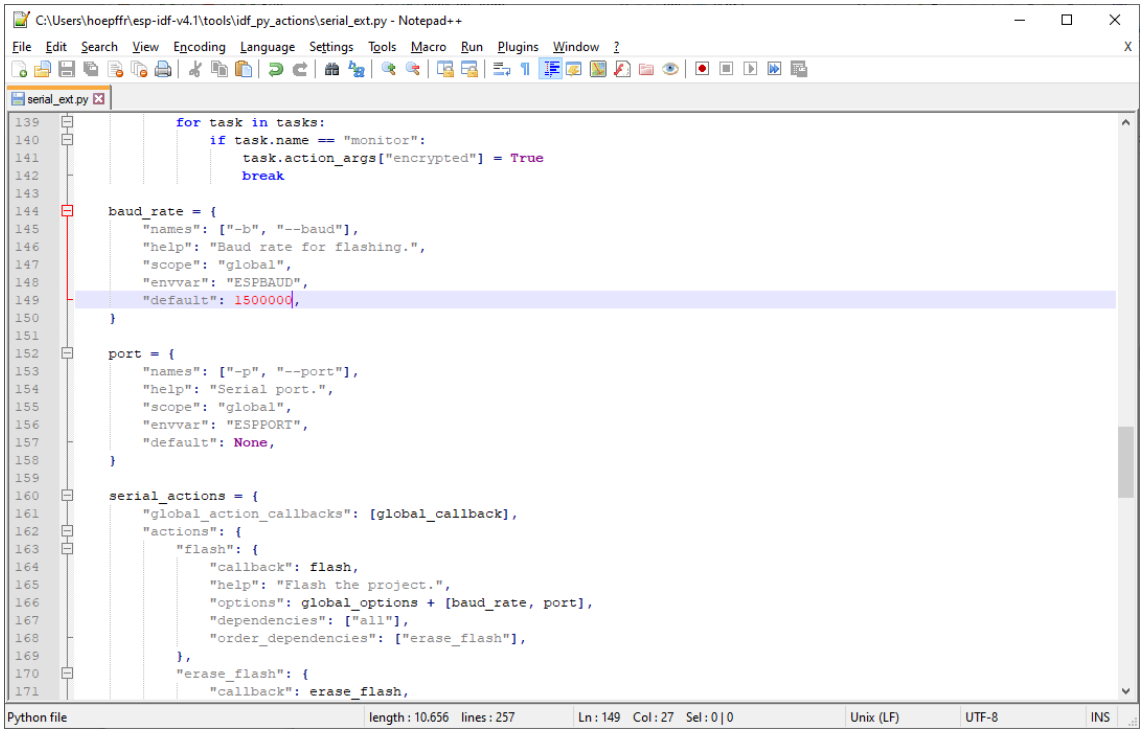
then open the file



C:\Users\hoepffr\esp-idf-v4.1\tools\idf_py_actions\serial_ext.py

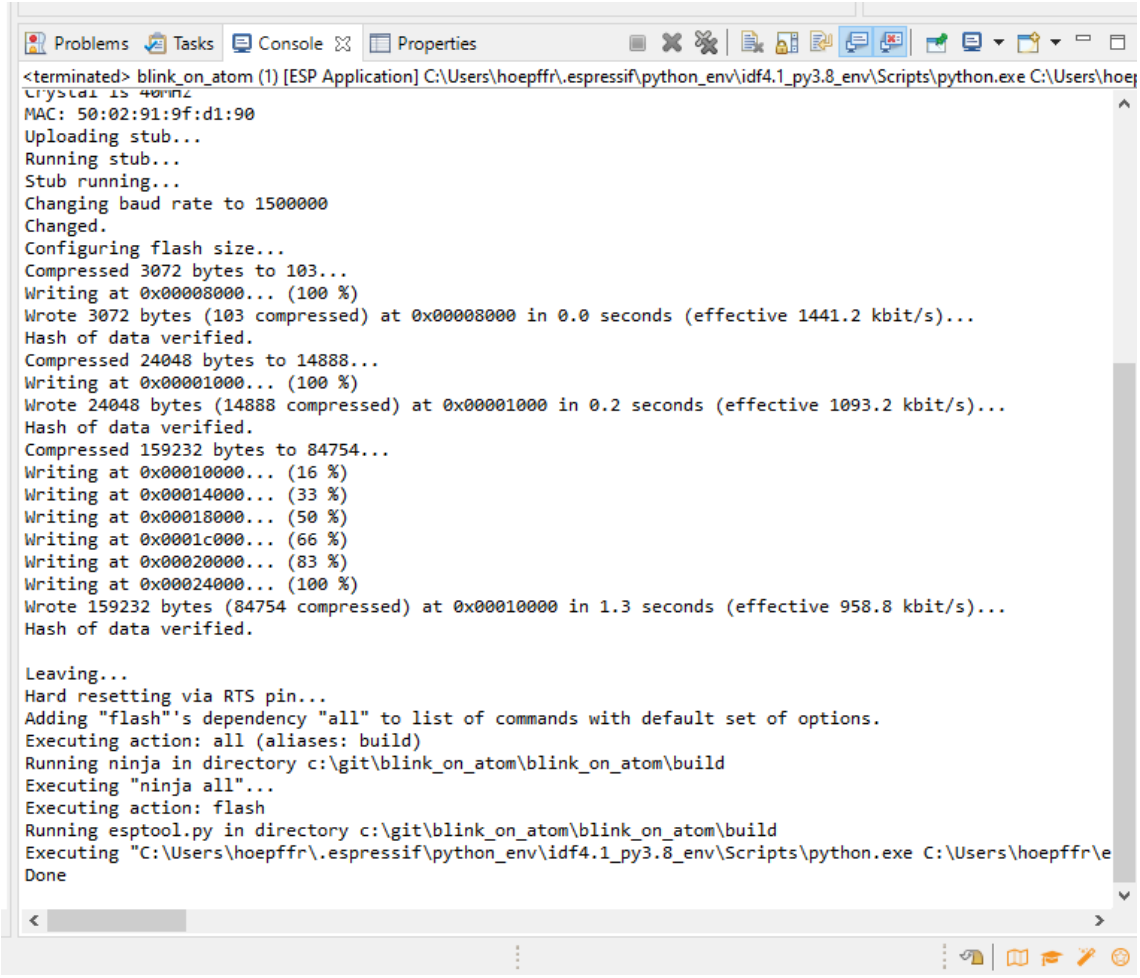


and change the number 460800 to 1500000



Don’t forget to SAVE!

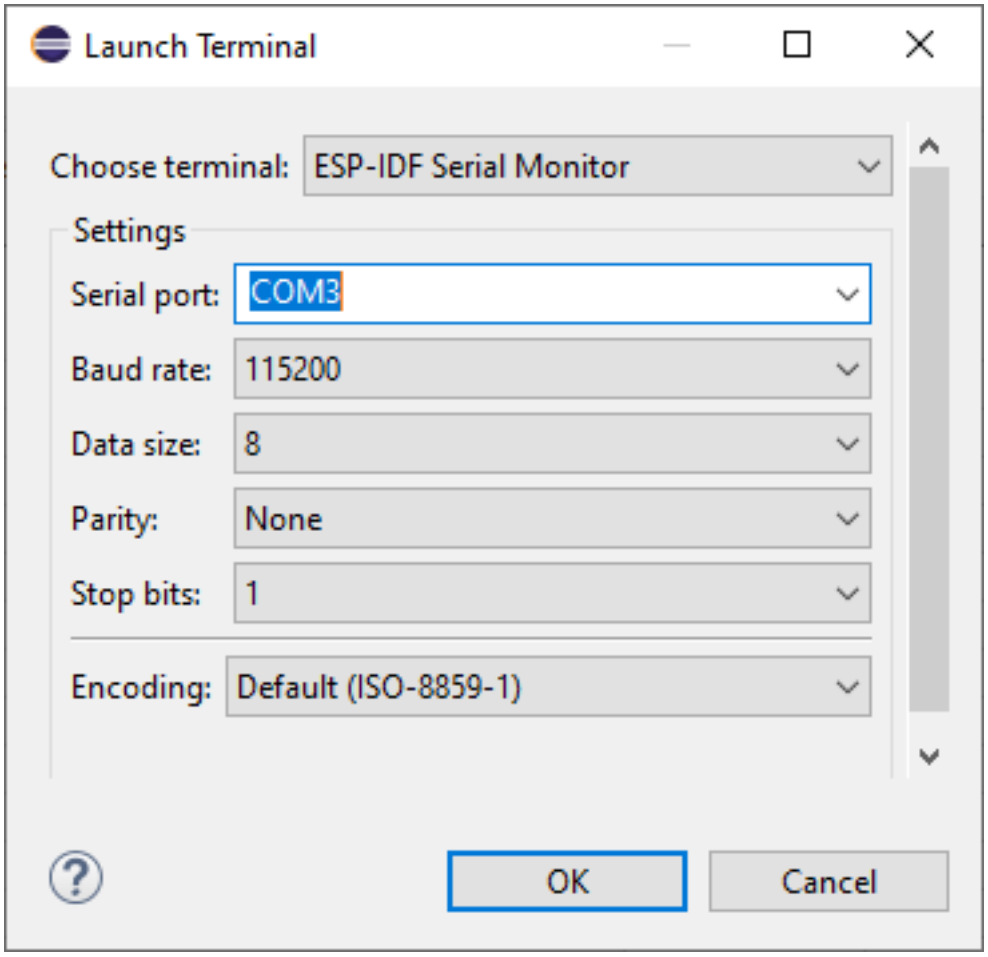
Now press the PLAY button again:

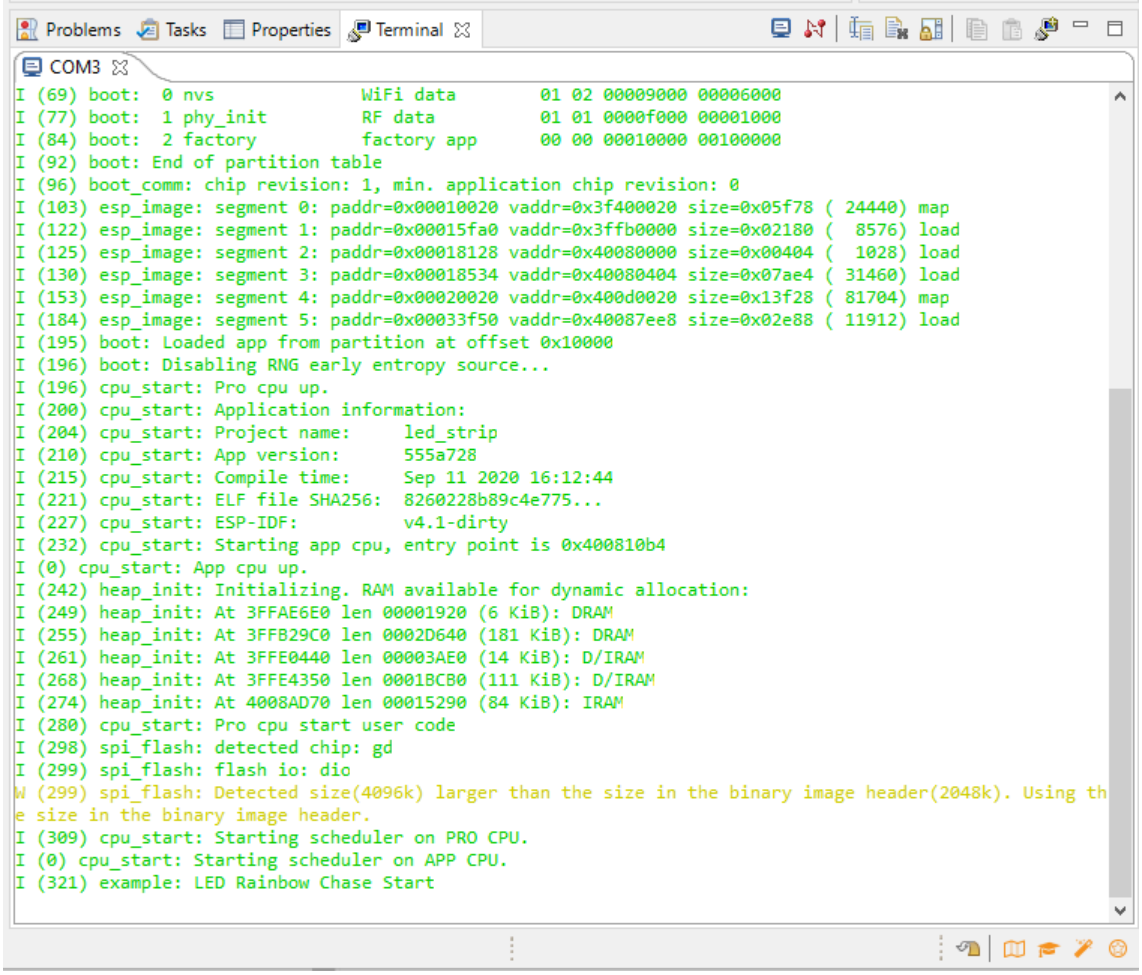


Now it should look like this,
and the ATOM should be blinking.
Familiarize yourself with the program.

26.8 Step 22

https://github.com/espressif/idf-eclipse-plugin/blob/master/README.md#viewing-serial-output


Launch Terminal
Choose terminal:
ESP-IDF Serial Monitor
Settings
Serial port:
COM3
Baud rate:
115200
Data size:
8
Parity:
None
Stop bits:
1
Encoding:
Default (ISO-8859-1)
OK
Cancel



26.9 Modifying the Program

For example, change the CHASE_SPEED to 10ms,
and see what happens.
Make these changes a few more times to familiarize yourself with the program.

27. cci_EasyExample

You can now do everything: Get cci_EasyExample using GitKraken, Open the workspace C:\git\cci_EasyExample\eclipse-workspace Import the project, Set the ESP-IDF version,

Set the build target,

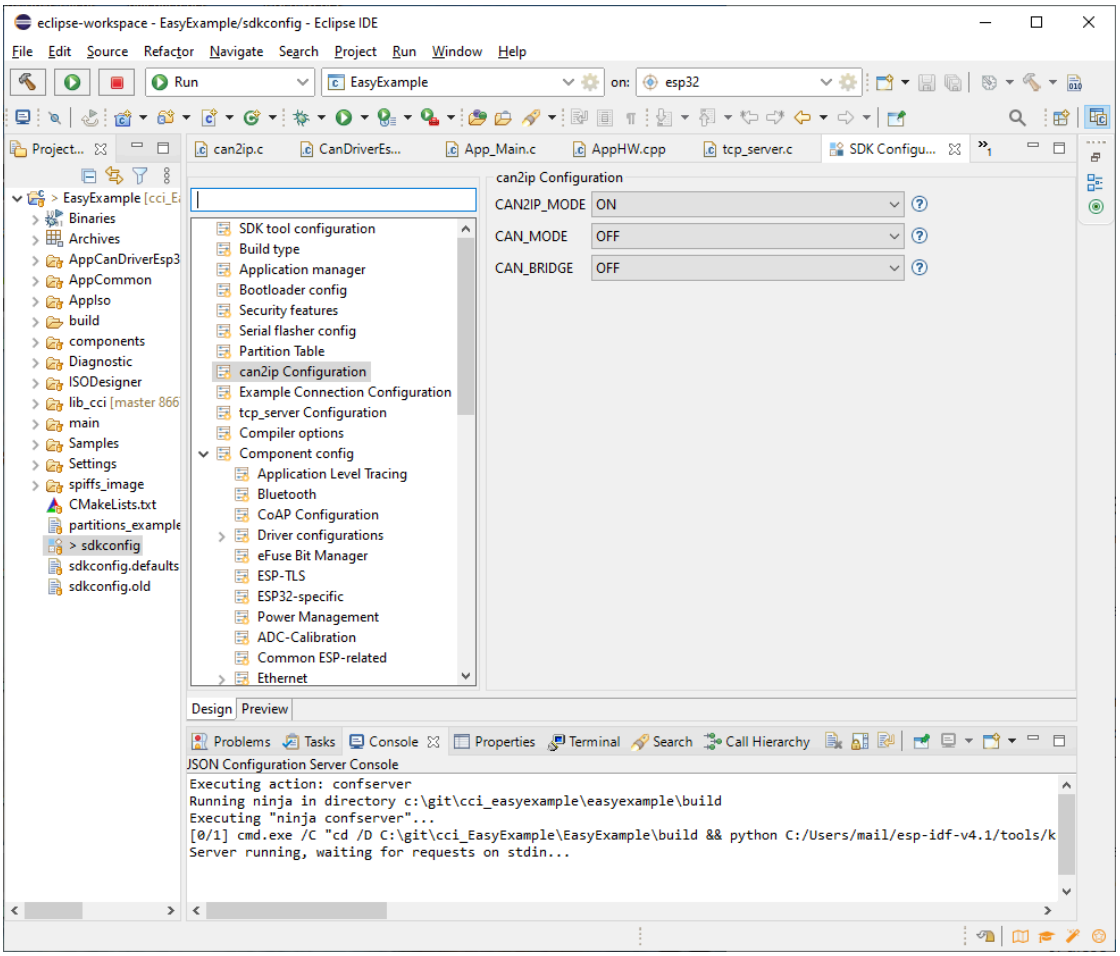
Compile everything and test it on the Traktor or ISOBUS test station.

The pools for ISO Designer can be found at: C:\git\cci_EasyExample\EasyExample\ISODesigner\MyWorkspace1

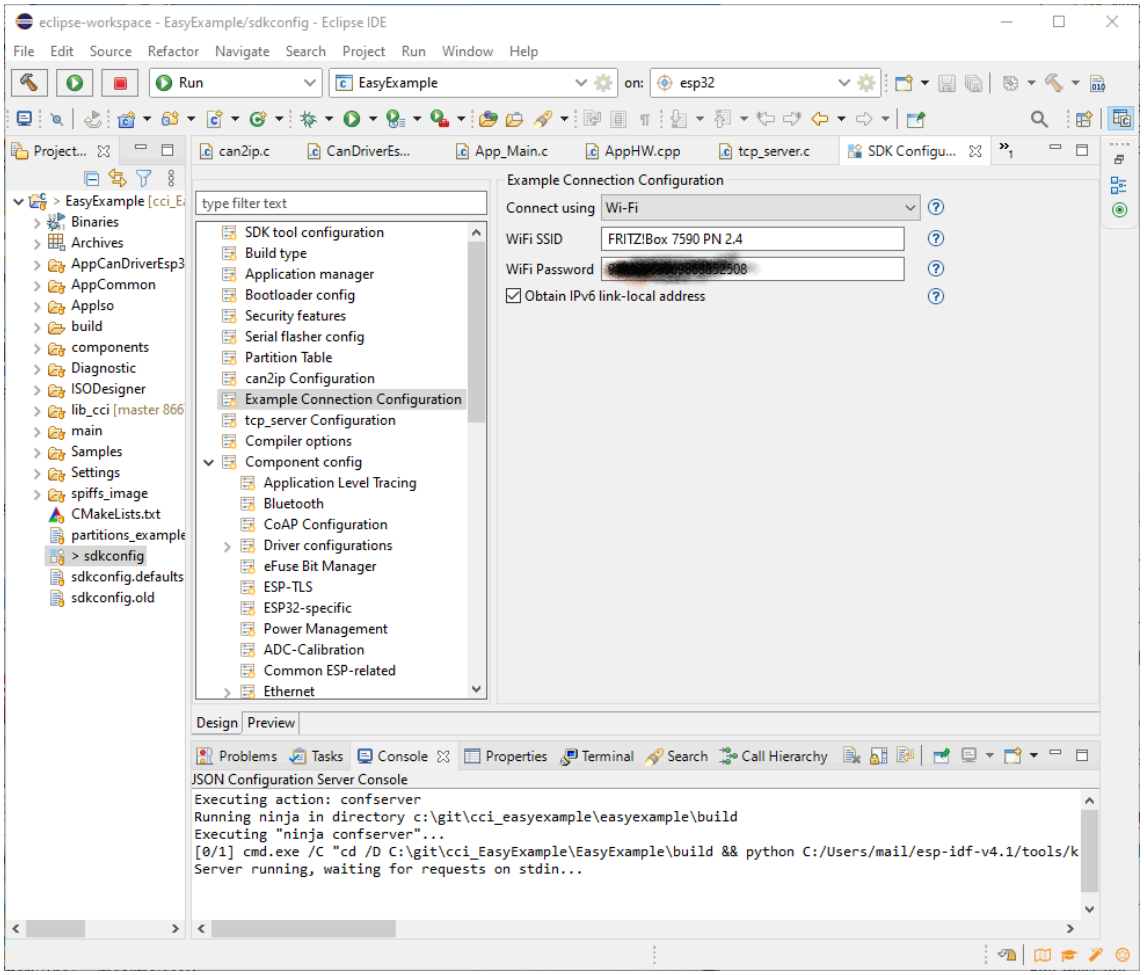
28. cci_EasyExample_CAN2IP

Special circumstances require special measures:

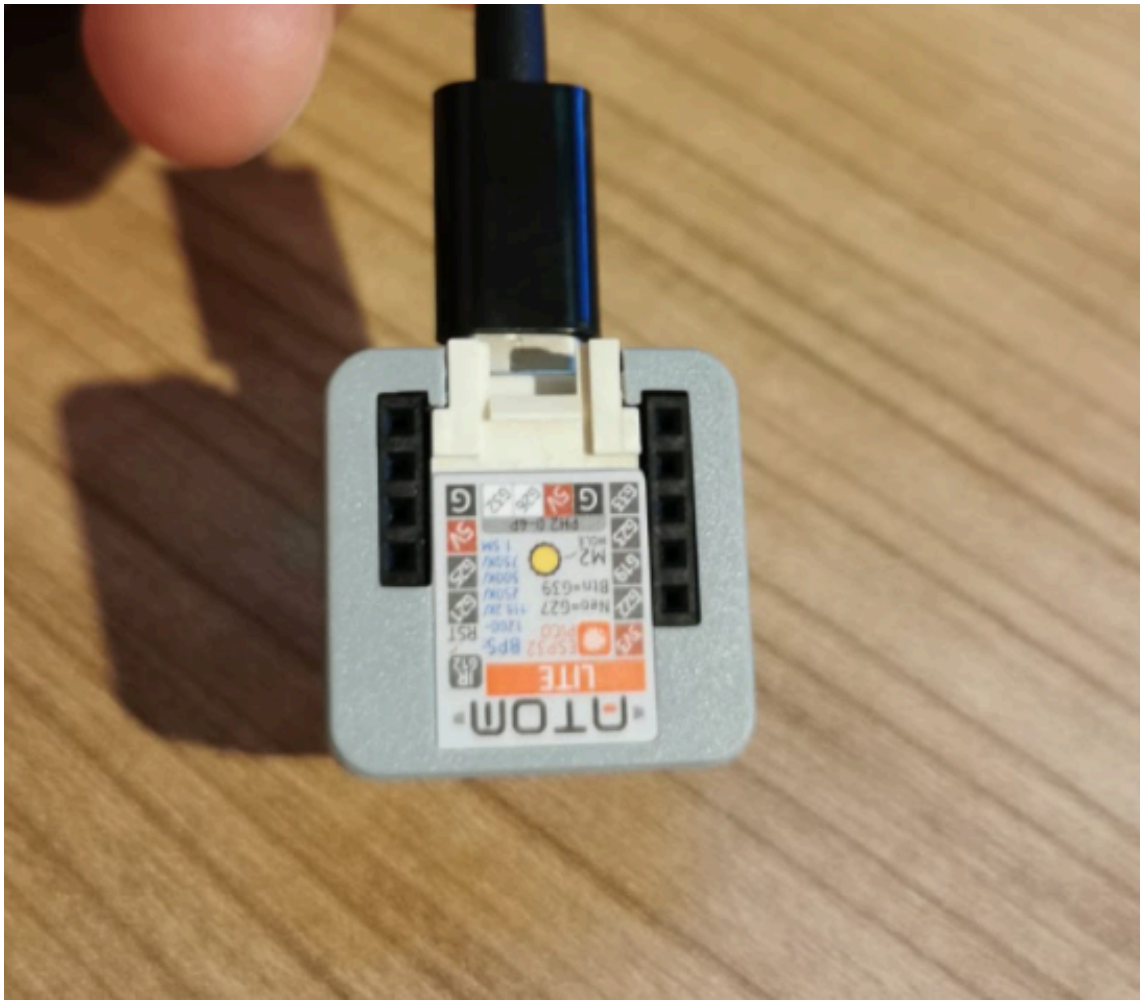
- Execute a PULL command on cci_EasyExample
- “CAN2IP” should be selected in sdkconfig
- Screenshot:



- Explanation: We now have two ways to connect to the ISOBUS ECU:
- The “virtual” CAN, which is actually WLAN: CAN2IP
- The “familiar” physical CAN, which is also present in the tractor: CAN
- You also need to enter your SSID and password in sdkconfig
- Screenshot:



- Now load the program onto the M5 Atom
- You don't necessarily need the CAN hardware for this.



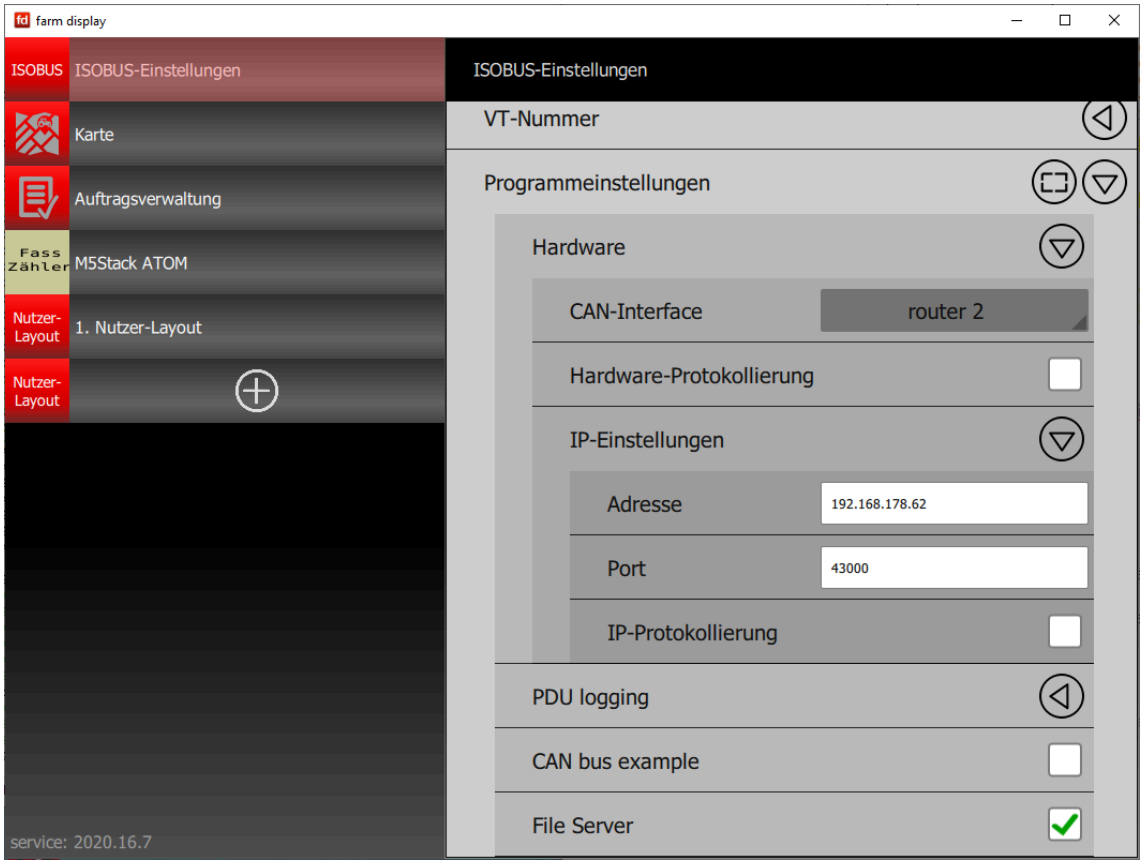
- An IP address now appears in the console.

```
I (809) wifi:Init static rx buffer size: 1600
I (814) wifi:Init static rx buffer num: 10
I (819) wifi:Init dynamic rx buffer num: 32
I (824) example_connect: Connecting to FRITZ!Box 7590 PN 2.4...
I (919) phy: phy_version: 4180, cb3948e, Sep 12 2019, 16:39:13, 0, 0
I (924) wifi:mode : sta (50:02:91:9f:51:1c)
I (924) example_connect: Waiting for IP
I (1044) wifi:new:<1,0>, old:<1,0>, ap:<255,255>, sta:<1,0>, prof:1
I (1824) wifi:state: init -> auth (b0)
I (1829) wifi:state: auth -> assoc (0)
I (1834) wifi:state: assoc -> run (10)
I (1869) wifi:connected with FRITZ!Box 7590 PN 2.4, aid = 5, channel 1, BW20, bssid = dc:39:6f:65:9a:d7
I (1869) wifi:security: WPA2-PSK, phy: bgn, rssi: -63
I (1874) wifi:pm start, type: 1

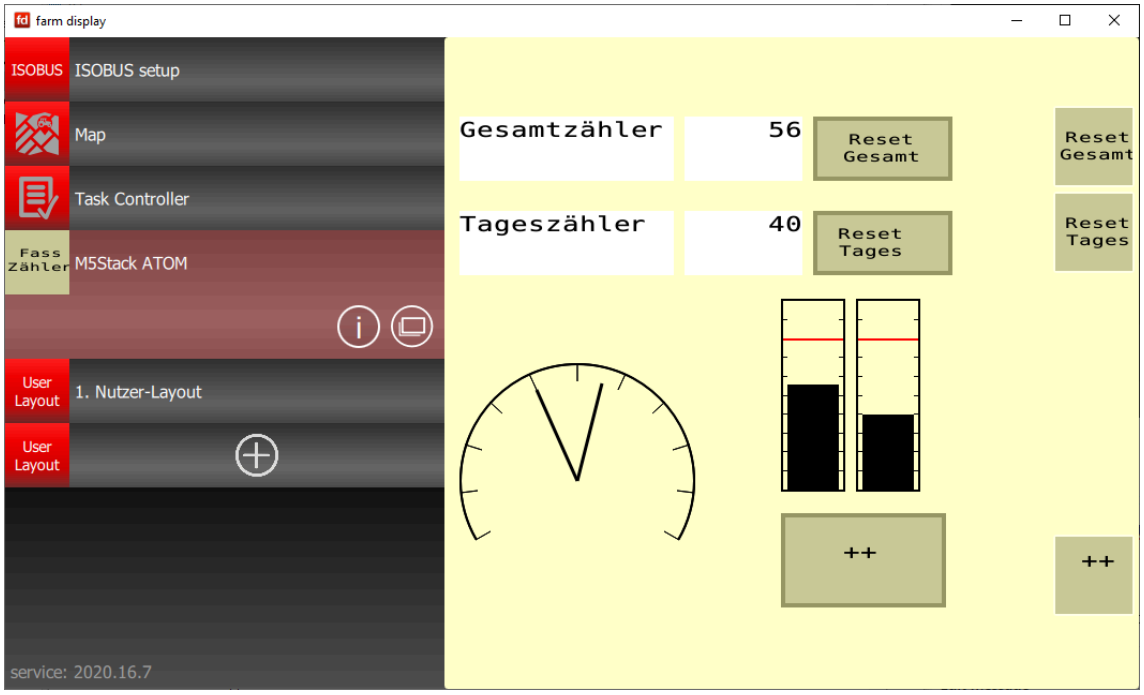
I (1919) wifi:AP's beacon interval = 102400 us, DTIM period = 1
I (2744) esp_netif_handlers: sta ip: 192.168.178.62, mask: 255.255.255.0, gw: 192.168.178.1
I (2749) example_connect: Got IP event!
I (3744) example_connect: Got IPv6 event!
I (3744) example_connect: Connected to FRITZ!Box 7590 PN 2.4
I (3744) example_connect: IPv4 address: 192.168.178.62
I (3749) example_connect: IPv6 address: fe80:0000:0000:0000:5202:91ff:fe9f:511c
app_main_tcp_server DONEapp_main_tcp DONEI (3759) tcp_server: Socket created
I (3764) tcp_server: Socket bound, port 43000
I (3769) tcp_server: Socket listening
I (4299) settings: getU8, section = CF-A, key = sourceAddress, value = 140
NE CH 0: This   Active   CF 8c (1400) Time:     3647
```

- Enter this IP address in the farm display.
- For installation, see: nx_farm_display
- Select “router 2” as the CAN interface.

- Screenshot:



- Click “Restart”.
- The object pool should now appear in the farm display.



You now have a completely virtual environment around the ISOBUS.
Advantage: You don’t have to build any hardware!

29. 📋 This is on the curriculum: C programming in ISOBUS

📌 **Source & Legal Basis: Source:** *Framework Curriculum for Preparation for the Master Craftsman Examination in Agricultural and Construction Machinery Mechatronics* **Publisher:** German Association of Agricultural and Construction Machinery Technicians V., Alfredstraße 102, 45131 Essen (Status: February 25, 2025) **Regulation:** *Master Craftsman Examination Regulation (LandBauMechMstrV)* of September 9, 2024 (Federal Law Gazette 2024 I No. 277, in force from August 1, 2025)

🌐 **Central Portal of the Master Craftsman Schools Munich:**

Visit the main platform: **www.ms-muc-docs.de** | 📋 Interactive Framework Curriculum 2025 | 📚 Wiki Overview | 📄 PDF Downloads

---

29.1 🎯 Relevance of C Programming & ISO 11783 C Libraries in the 2025 Curriculum

C/C++ programming for embedded systems is the foundation for implementing standards-compliant communication on agricultural and construction machinery according to ISO 11783.

29.2 📚 Curriculum Modules & Focus Areas

29.2.1 Part 1 – Learning Unit 1.4: Automated Functions (80 hours)

- **Implementation of ISOBUS Protocols:** Integration of C libraries for CAN communication and IsoAgLib.

29.2.2 Part 2 – Learning Unit 1.1: Systems Engineering (120 hours)

- **Software Architecture:** Memory management, bit operations, pointers, and system drivers in C/C++ for microcontrollers.

29.2.3 🌐 Relevant subpages on ms-muc-docs.de

- 🌐 Eclipse 4diac IDE & Color Reference on ms-muc-docs.de
- 🌐 MCU vs. MPU Comparison Guide on ms-muc-docs.de

29.3 🔗 Overview, Main Portal & Direct Links to all 6 Wikis

🌐 **Main Portal:** **www.ms-muc-docs.de** (Central Information Platform of the Master Schools at Munich East Station)

Wiki	Topic / Description	Link to ReadTheDocs
Wiki 0	**Master School Wiki** General information, Curriculum 2025 & Main Wiki.	📖 Open
Wiki 1	**C Programming in ISOBUS** Fundamentals of C programming and the ISOBUS environment.	🚜 Open
Wiki 2	**ISOBUS Virtual Terminal (VT)** User interfaces, ISO Designer, VT object pools & HMI.	🖥️ Open
Wiki 3	**ISOBUS Technology & Machine Systems** Hydraulics, pneumatics, motors, high-voltage technology, gearboxes & machine safety.	📡 Open
Wiki 4	**Visual Programming Languages** Low-code development, IEC 61499 & 4diac IDE (main curriculum topic).	🧩 Open
Wiki 4.1	**4diac Library Reference** Function block & adapter reference for the 4diac libraries.	🧩 Open
Wiki 4.2	**4diac Exercises: Training Solutions** Exercises and sample solutions for training with 4diac.	🧩 Open
Wiki 5	**Electronic Tools & Measuring Instruments** Handbook for crimping technology, soldering stations, oscilloscopes & microscopes.	🛠️ Open

29.3.1 📄 PDF Downloads for All Wikis

Wiki	PDF (DE)	PDF (EN)
Wiki 0 – Main Wiki	📄 DE	📄 EN
Wiki 1 – C Programming in ISOBUS	📄 DE	📄 EN
Wiki 2 – ISOBUS Virtual Terminal	📄 DE	📄 EN
Wiki 3 – ISOBUS Technology & Machine Systems	📄 DE	📄 EN
Wiki 4 – Visual Programming Languages	📄 DE	📄 EN
Wiki 4.1 – 4diac Library Reference	📄 DE	📄 EN
Wiki 4.2 – 4diac Exercises	📄 DE	📄 EN
Wiki 5 – Electronic Tools	📄 DE	📄 EN