

Bezzera Duo Bridge – Bauanleitung

main-1eaf1f7 · github.com/larszu/bezzera-duo-protocol

Inhalt

1. Firmware auf den ESP32
2. Verkabelung
3. Pinbelegung beider Boards
4. Ersatz-Display zeigt keine Werte
5. Fernzugriff

Firmware auf den ESP32

Die Bridge läuft auf zwei Boards:

Board	Was geht
Waveshare ESP32-S3-ETH (ESP32-S3, 8 MB PSRAM, 16 MB Flash, Ethernet), empfohlen	alles: Matter, Netzkabel, Update über die Weboberfläche, 24 h Verlauf
klassischer ESP32 (ESP32-D0WD/WROOM-32, 4 MB Flash, USB über CP2102/CH340)	alles außer Matter, Netzkabel und Update über die Weboberfläche; Verlauf 5 min

Die Flash-Seite erkennt das Board selbst. Drei Wege, vom einfachsten zum flexibelsten.

1. Im Browser (empfohlen)

1. Am Rechner **Chrome oder Edge** öffnen (Safari, Firefox und Handys können kein Web Serial).
2. Die Flash-Seite öffnen: <https://larszu.github.io/bezzera-duo-protocol/>
3. ESP32 per USB-C anstecken, **Installieren** klicken, Anschluss wählen (meist „USB JTAG/serial debug unit“).
4. Nach dem Flashen fragt die Seite nach dem Heim-WLAN (Improv). Netz wählen, Passwort eingeben.
5. Die Bridge ist danach unter <http://duo.local> erreichbar.

Die Seite baut GitHub bei jeder Änderung an `bridge/` neu (`.github/workflows/firmware.yml`); zu jedem Versions-Tag gibt es die Datei zusätzlich als Release.

2. Mit Arduino CLI (aus dem Quelltext)

```
brew install arduino-cli # macOS; Linux/Windows:
arduino.github.io/arduino-cli
arduino-cli core install esp32:esp32@3.3.12
bash tools/firmware_bauen.sh # baut build/s3 und build/esp32
esptool --chip esp32s3 --port /dev/cu.usbmodem... write-flash 0x0
build/s3/duo_bridge.ino.merged.bin
esptool --chip esp32 --port /dev/cu.usbserial... write-flash 0x0
build/esp32/duo_bridge.ino.merged.bin
```

Der klassische ESP32 wird aus einer Kopie ohne `partitions.csv` gebaut (`PartitionScheme=huge_app`, 3 MB Programm, kein zweiter Programmplatz).

- `FlashSize=16M,PartitionScheme=custom`: die eigene Aufteilung `bridge/duo_bridge/partitions.csv` (zwei App-Bereiche à 6 MB, genug für Bluetooth und Matter). Der Einstellungsspeicher liegt an derselben Stelle wie vorher, gespeicherte Einstellungen bleiben beim Wechsel erhalten.
- `PSRAM=opi`: der ESP32-S3R8 dieses Boards. Ohne PSRAM läuft alles, nur der Verlauf reicht dann 5 Minuten statt 24 Stunden.
- `CDC0nBoot=cdc`: Ausgabe und Befehle über den USB-C-Anschluss.

3. Mit esptool (fertige Datei aus einem Release)

```
pip install esptool
esptool --chip esp32s3 --port /dev/cu.usbmodem... write-flash 0x0 duo_bridge-esp32s3-
vX.Y.bin
esptool --chip esp32 --port /dev/cu.usbserial... write-flash 0x0 duo_bridge-esp32-
vX.Y.bin
```

Update eines laufenden ESP32-S3 ohne Kabel: Weboberfläche → Diagnose → Firmware, Datei `duo_bridge-esp32s3-update-vX.Y.bin`.

Wenn der ESP32 nicht erkannt wird

- Anderes USB-Kabel: Viele Kabel laden nur.
- Download-Modus erzwingen: **BOOT** gedrückt halten, **RESET** kurz drücken, BOOT loslassen, dann flashen. Danach einmal RESET.
- macOS fragt beim ersten Anstecken „Zubehör verbinden?“ – erlauben.

Ersteinrichtung ohne Rechner

1. Handy mit dem WLAN **duo-bridge** verbinden, Passwort **espresso1**.
2. Die Seite öffnet sich als Anmeldeportal von selbst (sonst irgendeine Adresse aufrufen, z. B. `http://espresso.maschine`).
3. **Einstellungen** → **Heim-WLAN** → **Netze suchen**, Netz antippen, Passwort.
4. Unter **Zugang** das WLAN-Passwort der Bridge ändern und ein Passwort für die Seite setzen (Benutzer `duo`). Das Standardpasswort steht öffentlich in diesem Repo.

Per USB geht es auch: `n <ssid> <passwort>` (siehe README, Befehle über USB).

Danach

- Verkabelung zur Maschine: README, Abschnitt ESP32-Bridge → Verkabelung.
- Waage: **Einstellungen** → **Waage** → **Bluetooth-Waage** → **Waage suchen**, Waage antippen.
- Von unterwegs: [fernzugriff.md](#).

Verkabelung

Steckplan mit Lochpositionen: [docs/breadboard_hybrid.png](#) . Welcher Stift das auf dem ESP32-S3-ETH bzw. einem klassischen ESP32-DevKit ist und wie man von einem aufs andere umbaut:

[docs/pinbelegung.md](#) .

Verbindung	wie
Mainboard schwarz (Ersatzkabel braun) + Display braun	ESP32 GND
Mainboard rot (grün) ↔ Display grün	direkt, nicht an den ESP32
Mainboard rosa (weiß)	10 kΩ → GPIO17 , GPIO17 20 kΩ → GND
ESP32 GPIO18	→ Display weiß
Display gelb	→ ESP32 GPIO15
ESP32 GPIO16	→ Mainboard grün (gelb)
ESP32 5V	frei

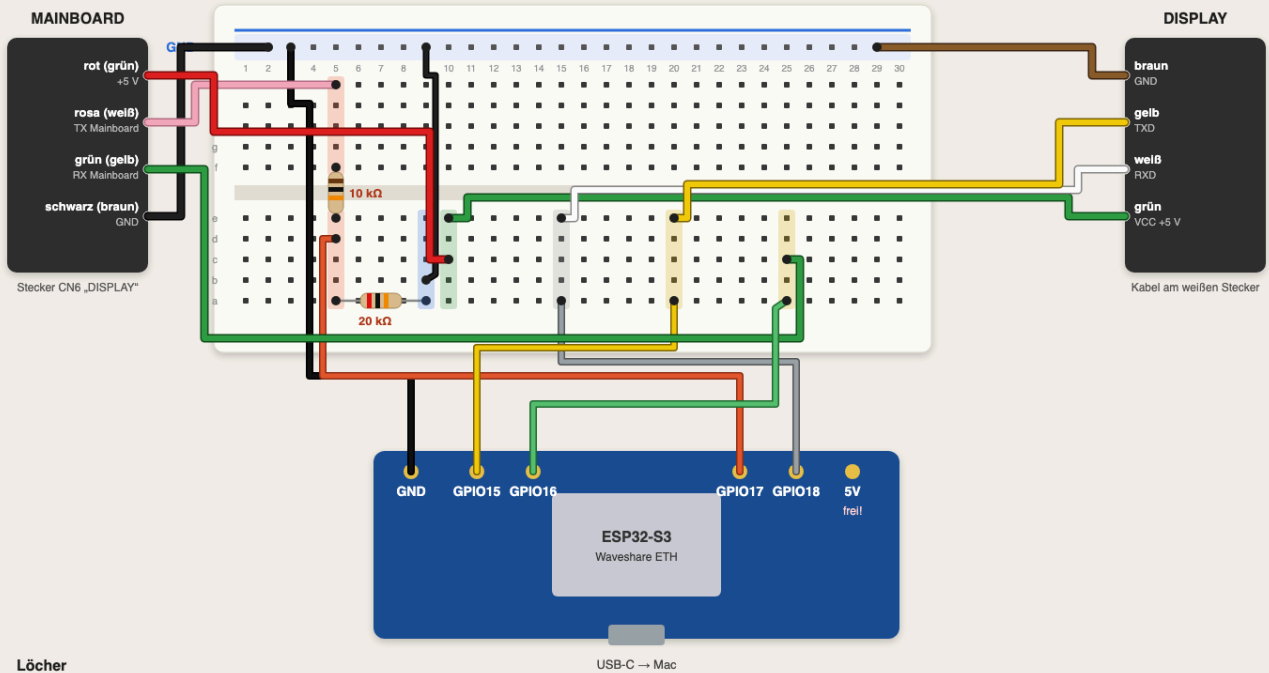
Nur Display, ohne Maschine (Seiten durchschalten, Flash lesen): Display grün an ESP32 **5V**, braun an GND, gelb an GPIO15, weiß an GPIO18.

Fallen beim Aufbau

- **Display-braun und Display-grün vertauscht verpolt das Display.** Seine Schutzdiode schließt dann die 5 V der Maschine kurz (~0 V an der Versorgung). Das Display hat es überstanden.
- **Ohne gemeinsame Masse** kommen nur Störbytes an: Maschinen-GND direkt an einen GND-Pin des ESP32.
- **10 kΩ und 20 kΩ vertauscht** ergibt ~1,6 V an GPIO17, genau im Grenzbereich — Datensalat. Richtig: 10 kΩ zwischen Leitung und GPIO17, 20 kΩ nach GND.
- **Nie das Display zusätzlich direkt an die Maschine stecken**, solange der ESP32 in der Leitung ist: zwei Sender auf einer Leitung, 5 V aus zwei Quellen.

ESP32-Bridge zwischen Mainboard und Display (Hybridmodus)

Netzstecker gezogen · ESP32 per USB am Mac · 5V-Pin des ESP32 bleibt frei · Farben: Originalkabel am Mainboard (neues Kabel in Klammern)



Löcher

Spalte 5 oben (j): Mainboard rosa (neu: weiß) = TX Mainboard

5f → 5e: 10 kΩ über die Mittelrinne (braun-schwarz-orange)

5d: Kabel zu ESP32 GPIO17

5a → 9a: 20 kΩ (rot-schwarz-orange), 9b Draht zur GND-Schiene

Spalte 10: Mainboard rot (neu: grün) in 10c + Display grün in 10e → +5 V, NICHT zum ESP32

Spalte 15: Display weiß in 15e, 15a → ESP32 GPIO18

Spalte 20: Display gelb in 20e, 20a → ESP32 GPIO15

Spalte 25: Mainboard grün (neu: gelb) in 25c, 25a → ESP32 GPIO16

GND-Schiene: Mainboard schwarz (neu: braun), Display braun, ESP32 GND

USB-C → Mac

Pinbelegung beider Boards

Die Firmware nutzt auf beiden Boards **dieselben GPIO-Nummern**

(`bridge/duo_bridge/duo_bridge.ino`, `PIN_...`). Beim Umbau von einem Board aufs andere bleiben die Kabel also an denselben Signalen, nur die Lage und die Beschriftung der Stifte sind anders.

Signal	GPIO	Waveshare ESP32-S3-ETH	klassischer ESP32 (DevKit, 30 Pins)	Kabel
RX vom Display	15	Stift „15“	D15 (auch IO15/G15), rechte Reihe, 13. von oben	Display gelb
TX zum Mainboard	16	Stift „16“	RX2 (GPIO16), rechte Reihe, 10. von oben	Mainboard grün (Ersatzkabel gelb)
RX vom Mainboard	17	Stift „17“	TX2 (GPIO17), rechte Reihe, 9. von oben	Mainboard rosa (weiß) über 10 kΩ, 20 kΩ nach GND
TX zum Display	18	Stift „18“	D18 (IO18), rechte Reihe, 7. von oben	Display weiß
Masse	GND	„GND“	GND , rechte Reihe, 14. von oben (oder links unten)	Mainboard schwarz + Display braun
5 V (nur ohne Maschine)	–	„5V“	VIN , linke Reihe unten	Display grün, nur im Betrieb ohne Maschine

Lage beim klassischen DevKit: Antenne oben, USB unten. Rechte Reihe von oben: D23, D22, TX0, RX0, D21, D19, **D18**, D5, **TX2**, **RX2**, D4, D2, **D15**, **GND**, 3V3. Andere Board-Varianten (38 Pins, „ESP32-WROOM-32 DevKitC“) tragen dieselben GPIO-Namen an anderer Stelle: nach der Beschriftung gehen, nicht nach der Position.

Achtung beim klassischen ESP32

- „**RX2**“/„**TX2**“ **sind nur Namen**: Die Firmware nutzt GPIO16 als Sender zum Mainboard und GPIO17 als Empfänger. Kabel nach der Tabelle oben stecken, nicht nach RX/TX auf dem Aufdruck.
- **Boards mit PSRAM (WROVER)** belegen GPIO16/17 intern. Dort funktioniert die Bridge nicht; ein WROOM-Board ohne PSRAM nehmen (`esptool chip-id` zeigt „Embedded PSRAM“ nicht an).
- **Spannungsteiler** am Eingang vom Mainboard (10 kΩ zur Leitung, 20 kΩ nach GND) wie beim S3: die Maschine sendet 5 V, der ESP32 verträgt 3,3 V.
- Kein Ethernet, kein Matter, Update nur über die Flash-Seite ([flashen.md](#)).
- Wenig RAM: ntfs als `http://ntfy.sh/...` eintragen (ohne Verschlüsselung, sonst fehlt Speicher). Kalender und Telegram brauchen TLS und können scheitern.

Umbau S3 → klassisch (oder zurück)

1. Maschine aus.
2. Kabel am alten Board abziehen und nach der Tabelle am neuen anstecken: gleiche GPIO-Nummer, gleiche Farbe.
3. Neues Board über die [Flash-Seite](#) flashen; sie erkennt das Board selbst.
4. Einstellungen (WLAN, Passwort, Profile) liegen im Board und müssen einmal neu gesetzt werden.

Ersatz-Display zeigt keine Werte

Neues Display (Ersatzteil 2.2) eingebaut, Seiten erscheinen, Touch piept, aber **keine Temperaturen, keine Uhr**? Dann spricht das Display einen anderen Rahmenkopf (5A A5) als das Mainboard (C6 A5). Eine SD-Karte stellt es um, ohne Bridge, ohne Programm:

1. [display_kopf_c6.zip](#) laden, den Ordner `DWIN_SET` auf eine leere microSD kopieren (FAT32, 4-KB-Cluster, höchstens 8 GB).
2. Maschine einschalten, Karte ins Display stecken, nach einigen Sekunden ziehen.
3. Maschine am Hauptschalter aus und wieder ein. Jetzt erscheinen die Werte.

Die Karte ändert nur eine Einstellung des Displays (Register R3 von 5A auf C6), keine Seiten und Bilder. **Nur anwenden, wenn das Display keine Werte zeigt:** An einem Mainboard, das selbst 5A A5 spricht, würde sie das Display stumm machen (rückgängig mit R3=5A). Erprobt mit Display 2.2 an Mainboard 2.1, Hintergrund und Sicherung vorher in [docs/versionen.md](#). Ohne Gewähr.

Fernzugriff

Die Bridge ist ein kleiner Webserver im Heimnetz. Von außen kommt man nur über einen sicheren Tunnel hinein; **nie** einen Port im Router freigeben – die Bridge kann die Maschine einschalten.

Vorher: Passwort setzen

Einstellungen → Zugang → Passwort für diese Seite. Ohne Passwort darf jeder im Netz die Maschine bedienen.

Weg 1: Tailscale über ein Gerät, das ohnehin läuft (empfohlen)

Tailscale direkt auf dem ESP32 gibt es nicht ausgereift (zu wenig Speicher für WireGuard plus Tailscale-Steuerung). Stattdessen reicht ein Rechner im Heimnetz als **Subnet-Router**, z. B. der Heimserver, ein Raspberry Pi oder ein NAS:

Am einfachsten mit dem Skript aus diesem Repo, es erkennt das Heimnetz selbst:

```
sudo tools/tailscale_heimnetz.sh
```

Oder von Hand:

```
## auf dem Gerät im Heimnetz (Linux)
curl -fsSL https://tailscale.com/install.sh | sh
echo 'net.ipv4.ip_forward = 1' | sudo tee /etc/sysctl.d/99-tailscale.conf && sudo
sysctl -p /etc/sysctl.d/99-tailscale.conf
sudo tailscale up --advertise-routes=192.168.0.0/24    # eigenes Heimnetz eintragen
```

1. In der Tailscale-Verwaltung (login.tailscale.com → Machines) beim Gerät „Edit route settings“ die Route freigeben.
2. Auf dem Handy die Tailscale-App installieren, anmelden.
3. Im Browser die **IP-Adresse** der Bridge öffnen (steht unter Einstellungen → Heim-WLAN).
`duo.local` geht über Tailscale nicht, weil mDNS nur im eigenen Netz funktioniert.

Tipp: Im Router der Bridge eine feste IP geben, dann bleibt das Lesezeichen gültig.

Weg 1b: Maschine als eigenes Gerät im Tailscale-Netz (Docker)

Statt das ganze Heimnetz freizugeben, steht die Bridge als eigenes Gerät „bezzera“ in der Tailscale-App, mit fester Adresse für ein Symbol auf dem Home-Bildschirm. Auf einem Rechner im Heimnetz mit Docker:

```
sudo docker run -d --name bezzera-tailscale --restart unless-stopped \
-e TS_HOSTNAME=bezzera -e TS_STATE_DIR=/var/lib/tailscale -e TS_USERSPACE=true \
-v /opt/bezzera-tailscale/state:/var/lib/tailscale tailscale/tailscale:latest
sudo docker logs bezzera-tailscale          # Anmelde-link öffnen und bestätigen
sudo docker exec bezzera-tailscale tailscale serve --bg --tcp 80 tcp://<IP der
Bridge>:80
sudo docker exec bezzera-tailscale tailscale ip -4    # Adresse für das Handy
```

--tcp statt --http , damit die Seite unter dem Namen und unter der 100.x-Adresse antwortet. Auf dem iPhone: Tailscale verbinden, Adresse in Safari öffnen, Teilen → Zum Home-Bildschirm. Der Bridge im Router eine feste IP geben.

Weg 2: Home Assistant

Läuft Home Assistant mit Fernzugriff (Nabu Casa oder eigener Tunnel), ist die Bridge dort als Gerät „Bezzera Duo“ mit Schalter „Maschine“ eingebunden (Einstellungen → Home Assistant). Ein- und Ausschalten geht dann über die Home-Assistant-App von überall, ganz ohne zweiten Tunnel.

Weg 3: Kalender

Wer nur zu festen Zeiten Kaffee will: **Maschine** → **Ein- und Ausschalten** → **Kalender**. Termine mit dem Stichwort im Titel schalten die Maschine vorher ein. Der Kalender wird über das Internet gelesen, das geht auch, wenn man selbst unterwegs ist.