


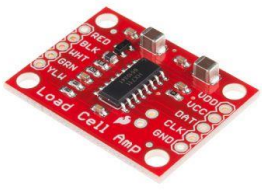



















Προϊόντα : <https://grobotronics.com>

	<p>Breadboard Jumper Wires 140Pcs (Bulk) Κωδικός: 19-00014101</p> <p>Καλώδια σύνδεσης €2.90</p>
	<p>Μετατροπέας DC-DC Step-Down 5V 3A - USB Κωδικός: 19-00015320</p> <p>για τροφοδοσία το PICO W με USB €5.90 Αν η τάση εισόδου είναι μικρότερη του 6v σταματάει να δουλεύει</p>
	<p>Header Kit for Raspberry Pico - 20Pin x2 & 3Pin x1 Κωδικός: 11-00012618</p> <p>Για το PICO €1.20</p>
	<p>Ενισχυτής Αισθητήρα Φορτίου SparkFun - HX711 Κωδικός: 03-00013879</p> <p>Για σύνδεση αισθητήρων €13.80 SparkFun γιατί είναι ο ποιο σταθερός</p>
	<p>Raspberry Pi Pico W - Wireless Κωδικός: 06-00012122</p> <p>€9.00</p>
	<p>Καλώδιο USB 2.0 A σε USB B micro 1.8m Μαύρο Κωδικός: 05-00093181</p> <p>Για σύνδεση με Η/Υ</p> <p>€2.50</p>
	<p>Πλακέτα Δοκιμών 400 Οπές - Άσπρη Κωδικός: 11-00012002</p> <p>Για σύνδεση όλων €3.20</p> <p><u>Τεμ: 2</u></p>

	<p>Αισθητήρας Φορτίου - 50kg Κωδικός: 03-00010245</p> <p>Τους αισθητήρες φορτίου 50kg μπορούμε να τους πάρουμε και από μια ζυγαριά μπάνιου</p> <p>€6.80</p> <p>Τεμ: 4</p>
	<p>Jumper Wires 15cm Female to Female - Pack of 10 Κωδικός: 11-00015551</p> <p>€1.45</p>
	<p>Jumper Wires 15cm Male to Male - Pack of 10 Κωδικός: 11-00015552</p> <p>€1.45</p>
	<p>Ribbon 10wire 15cm - Female to Male Κωδικός: 11-00015585</p> <p>€0.80</p>
	<p>JST XH Jumper 3 Wire Assembly - 20cm Κωδικός: 19-00080192 Για τον αισθητήρα DHT22</p> <p>€0.4838 Τεμ: 2</p>
	<p>JST XH Jumper 4 Wire Assembly - 20cm Κωδικός: 19-00080193 Για τον Ενισχυτή Αισθητήρα Φορτίου - HX711</p> <p>Τεμ: 2 €0.4</p>
	<p>Molex Connector Male 3-Pin 2.54mm (Angled) Κωδικός: 05-00250205 Για τον αισθητήρα DHT22</p> <p>€0.08 Τεμ: 2</p>

	<p>Molex Connector Male 4-Pin 2.54mm (Angled) Κωδικός: 05-00250208</p> <p>Για τον Ενισχυτή Αισθητήρα Φορτίου - HX711 Τεμ: 2 €0.08</p>
	<p>Αισθητήρας Θερμοκρασίας-Υγρασίας DHT22 Κωδικός: 19-00012287 €4.80</p>
	<p>Αντίσταση Carbon 1/4W 5% 10Kohm Κωδικός: 05-00145013 Για Αισθητήρας Θερμοκρασίας-Υγρασίας DHT22 Η συσκευασία έχει Τεμ: 10 εμείς χρειαζόμαστε 1 €0.2</p>
	<p>Κουτί Κατασκευών 176x126x57.4mm - Grey (Kradex Z74JPH ABS) Κωδικός: 05-00071415 Για να τα βάλουμε όλα μέσα €9.30</p>
	<p>JST SM 3 Wire Assembly Set (Male/Female) Κωδικός: 19-00020818 Για τον αισθητήρα DHT22 €0.60</p>
	<p>ST SM 4 Wire Assembly Female - 10cm Κωδικός: 19-00023001 Για τον Ενισχυτή Αισθητήρα Φορτίου - HX711 €0.3</p>
	<p>Καλώδιο Micro USB σε Jack 5.5x2.1mm Αρσενικό Κωδικός: 19-00011511 Για τροφοδοσία του pico w Συνδέεται στο Καλώδιο Τροφοδοσίας - Θηλυκό Jack και Pico W €1.20</p>



Καλώδιο Τροφοδοσίας - Θηλυκό Jack 5.5x2.1mm σε Γυμνό Καλώδιο

Κωδικός: 19-00552130

Για τροφοδοσία του pico w

Συνδέεται στον Μετατροπέα DC-DC Step Down και Micro USB

€0.50



Μετατροπέας DC-DC Step-Down 1.3-30V 3A

Κωδικός: 19-00015318

Είσοδος ρεύμα από το DFROBOT Gravity I2C Ψηφιακός Μετρητής
Έξοδος ρεύμα στο Καλώδιο Τροφοδοσίας - Θηλυκό Jack -> Pico W
Ρύθμιση τάσης εξόδου 4V

€2.90



Pin Header 1x40 Male 2.54mm Black (Long Centered)

Κωδικός: 19-00011945

Θα χρειαστούν για διάφορες συνδέσεις

€0.40



DFROBOT Gravity I2C Ψηφιακός Μετρητής Watt

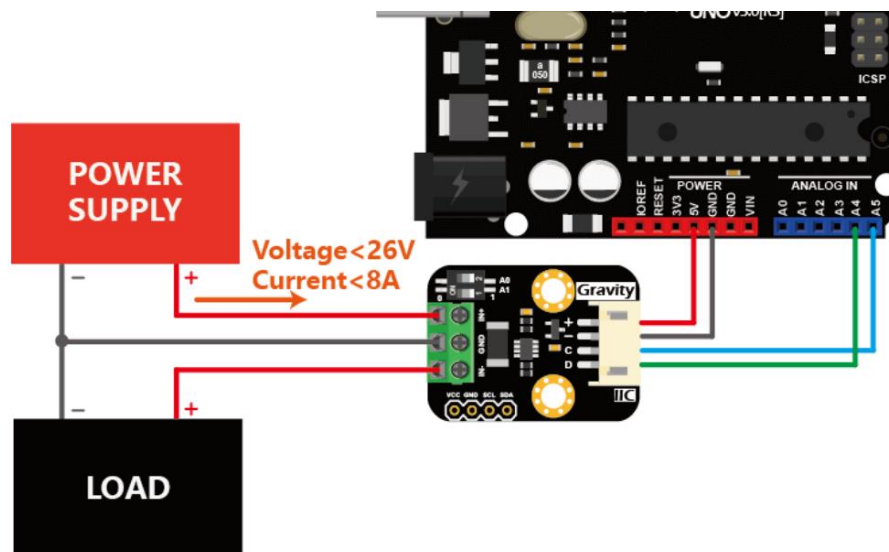
Κωδικός: 25-00016131

Για μέτρηση τάσης μπαταρίας

Είσοδος από μπαταρία 12V

Έξοδος στο Μετατροπέας DC-DC 12V (εικόνα →LOAD)

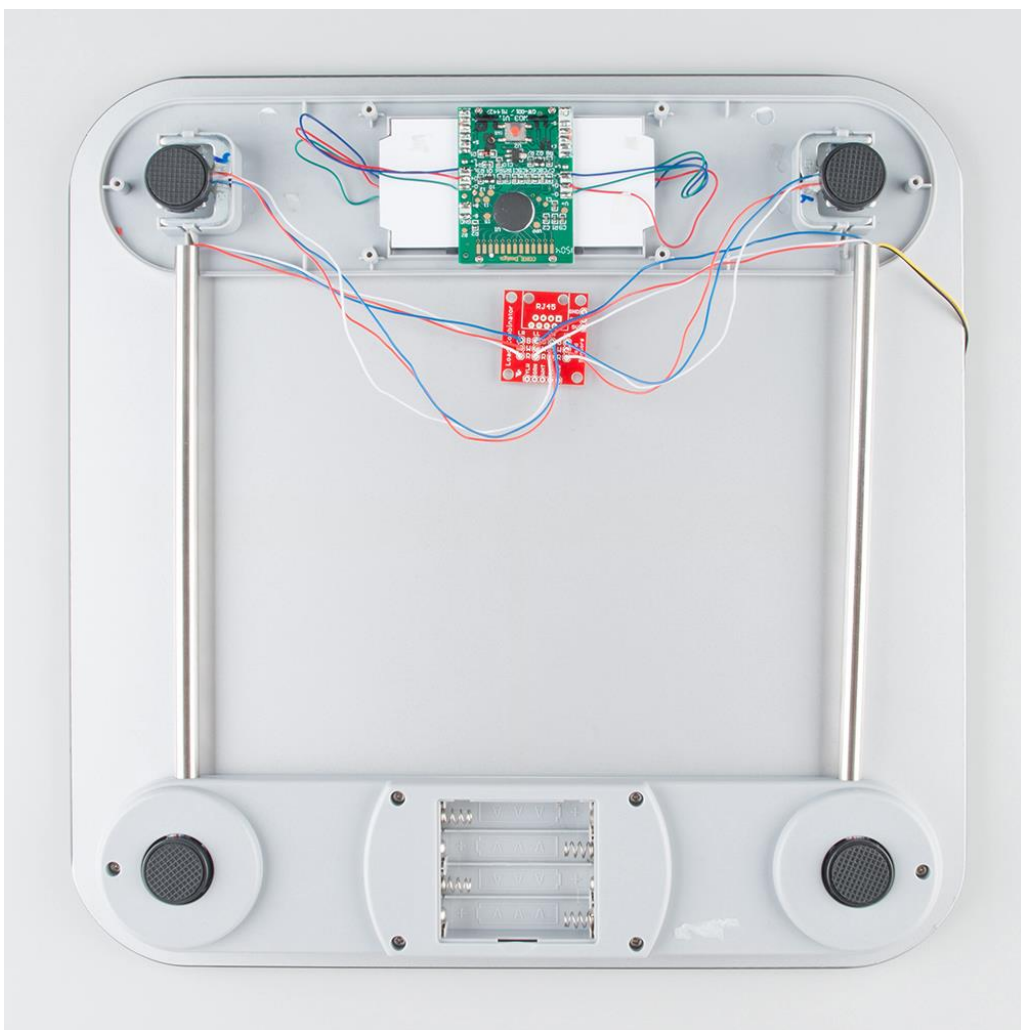
€9.40



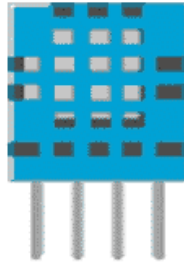
***Τους αισθητήρες φορτίου 50kg Μπορούμε να τους πάρουμε και από μια ζυγαριά μπάνιου μαζί με τις βάσεις τους, διαφορετικά πρέπει να κάνουμε εμείς βάσεις για να έχει κενό να ταλαντεύεται.**

<https://circuitjournal.com/50kg-load-cells-with-HX711>

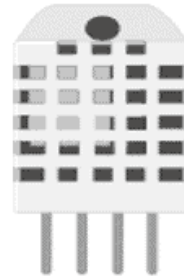
<https://www.thingiverse.com/thing:2624188>



**Αισθητήρας για τον χώρο της ζυγαριάς,
δεν μπορεί να είναι έξω γιατί δεν είναι αδιάβροχος**



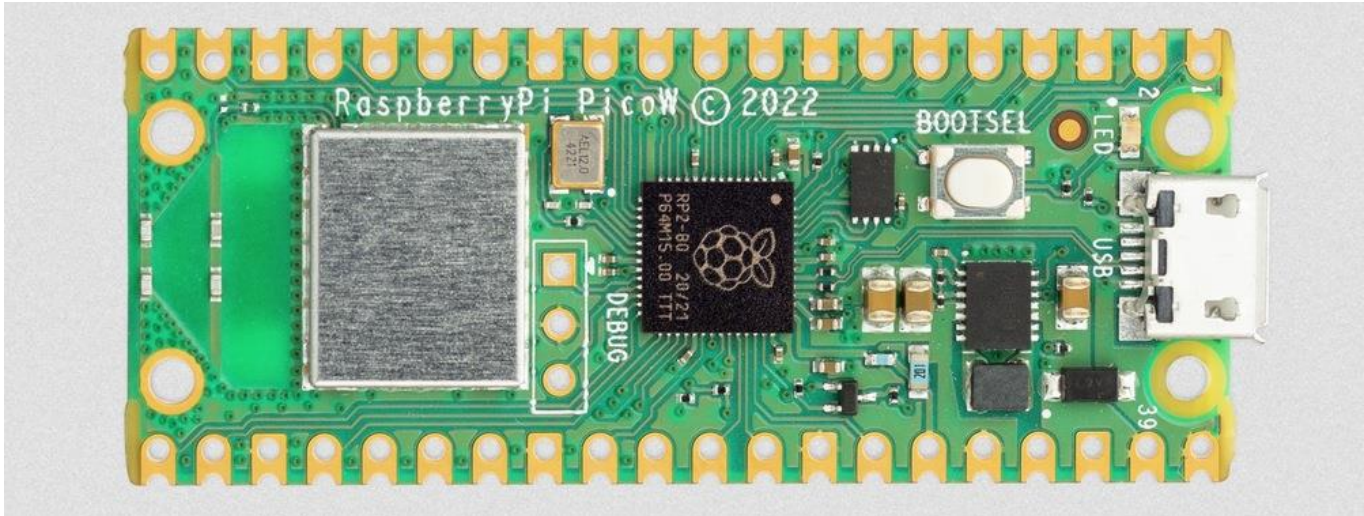
DHT11



DHT22

	DHT11	DHT22
Operating Voltage	3 to 5V	3 to 5V
Max Operating Current	2.5mA max	2.5mA max
Temperature Range	0-50°C / $\pm 2^{\circ}\text{C}$	-40 to 80°C / $\pm 0.5^{\circ}\text{C}$
Humidity Range	20-80% / 5%	0-100% / 2-5%
Sampling Rate	1 Hz (reading every second)	0.5 Hz (reading every 2 seconds)
Advantage	low cost	More Accurate

Pico W link → <https://micropython.org/download/rp2-pico-w/>



Vendor: Raspberry Pi

Features: BLE, Dual-core, External Flash, USB, WiFi

Source on GitHub: [rp2/PICO W](https://github.com/micropython/rp2-pico-w)

More info: [Website https://projects.raspberrypi.org/en/projects/get-started-pico-w/1](https://projects.raspberrypi.org/en/projects/get-started-pico-w/1)

Installation instructions

Flashing via UF2 bootloader

To get the board in bootloader mode ready for the firmware update, execute `machine.bootloader()` at the MicroPython REPL. Alternatively, hold down the BOOTSEL button while plugging the board into USB. The uf2 file below should then be copied to the USB mass storage device that appears. Once programming of the new firmware is complete the device will automatically reset and be ready for use.

Firmware https://micropython.org/download/RPI_PICO_W/

Releases

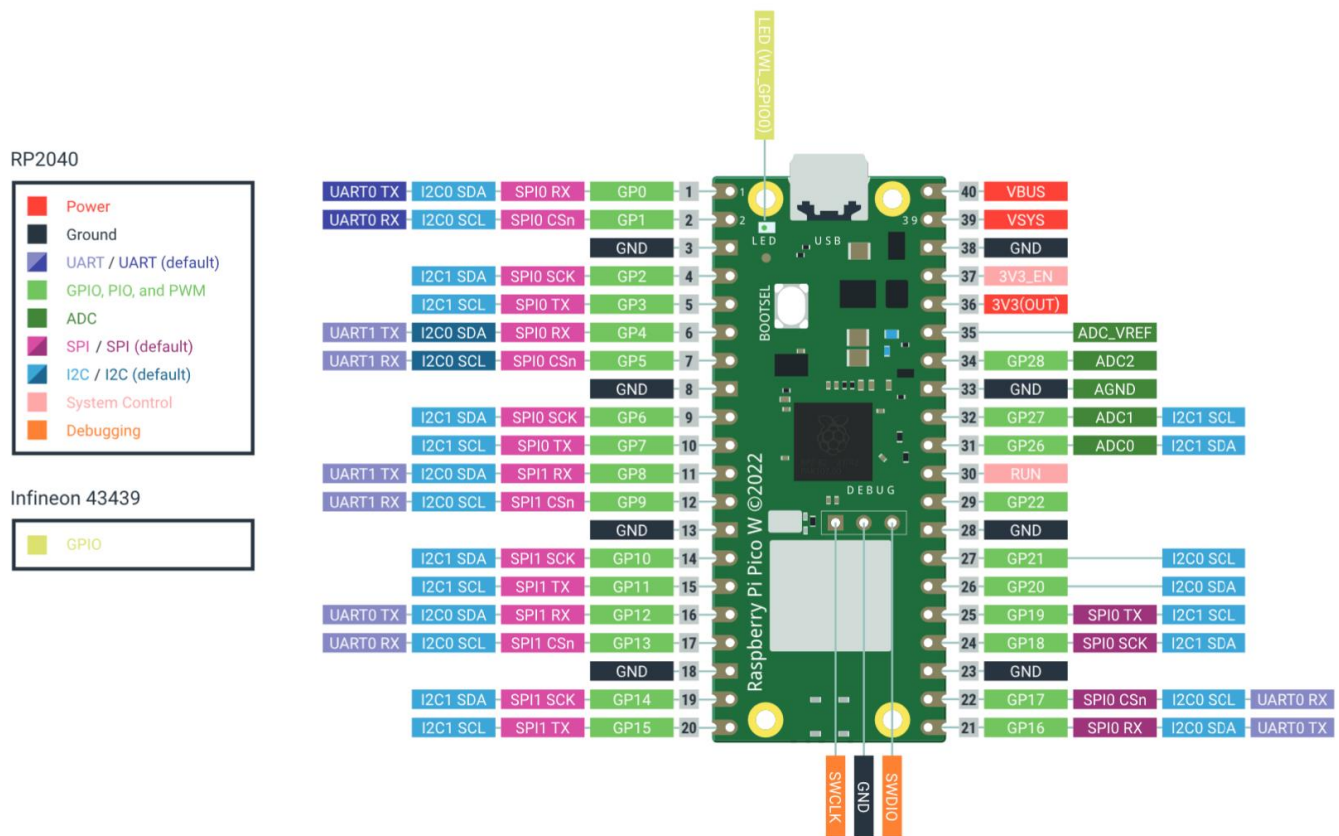
[v1.20.0 \(2023-04-26\) .uf2](#) [\[Release notes\]](#) (latest)

RP2040 Datasheet

Hardware design with RP2040

Raspberry Pi Pico W Datasheet

Raspberry Pi Pico W Pinout



Download the [Pinout Diagram \(PDF\)](#)

4.5. Powering Pico

The simplest way to power Pico is to plug in the micro-USB, which will power VSYS (and therefore the system) from the 5V USB VBUS voltage, via D1 (so VSYS becomes VBUS minus the Schottky diode drop).

If the USB port is the **only** power source, VSYS and VBUS can be safely shorted together to eliminate the Schottky diode drop (which improves efficiency and reduces ripple on VSYS).

If the USB port is **not** going to be used, it is safe to power Pico by connecting VSYS to your preferred power source (in the range ~1.8V to 5.5V).

ΠΡΟΣΟΧΉ

**Μια πηγή εισόδου ρεύματος θα πρέπει να
χρησιμοποιούμε**

Resetting Flash memory

Pico's BOOTSEL mode lives in read-only memory inside the RP2040 chip, and can't be overwritten accidentally. No matter what, if you hold down the BOOTSEL button when you plug in your Pico, it will appear as a drive onto which you can drag a new UF2 file. There is no way to brick the board through software. However, there are some circumstances where you might want to make sure your **Flash memory is empty**. You can do this by dragging and dropping a special UF2 binary onto your Pico when it is in mass storage mode.

- Download the flash_nuke.uf2 [UF2 file](#)

[Post](#) by [Roberthh](#) » Fri Jul 16, 2021 3:56 pm

There is a special .uf2 image which just **wipes out the whole flash**. I could not find the link again, but IO have a copy of that file. After running it, you must install the firmware .uf file again.

[flash_nuke.zip](#)

(8.87 KiB)

[Post](#) by [fdufnews](#) » Fri Jul 16, 2021 4:32 pm

flash_nuke will erase all the flash. if the problem is only an "uncontrolled" main.py there is a less damaging solution.

[https://www.raspberrypi.org/forums/view... 6&t=305432](https://www.raspberrypi.org/forums/view...6&t=305432)

MicroPython_RenameMainDotPy only zap main.py and preserve the filesystem so if you already have **any file saved on the file system they will be kept**.

Λογισμικό

IDE <https://thonny.org/>

Δικό μου repo στο Github με όλα τα αρχεία

https://github.com/dannyswolf/Bee_Scale_Pico_W

Θα πρέπει να βάλετε και ένα αρχείο με όνομα:
credentials.py

```
secrets = {  
    'url' : linkToAddDataToDjango', # Βαλέ τα δικάσου στοιχεία  
    'ssid' : NameOfSsid, # Βαλέ τα δικάσου στοιχεία  
    'password' : WifiPassword, # Βαλέ τα δικάσου στοιχεία  
    'timezone' : 'Europe/Athens',  
    'mqtt_username' : hivemqUsername', # Βαλέ τα δικάσου στοιχεία  
    'mqtt_key' : hivemq_Key, # Βαλέ τα δικάσου στοιχεία  
    'broker' : 'URL.s1.eu.hivemq.cloud', # Βαλέ τα δικάσου στοιχεία  
    'port' : 8883,  
    'DEBUG' : False, # True για εκτυπώσεις στην κονσόλα του Thony  
}
```

Pico W into lightsleep

<https://github.com/ghubcoder/micropython-pico-deepsleep/issues/8#issuecomment-1430355998>

HiveMQ

<https://github.com/FDelporte/HiveMQ-examples/tree/main/picow-to-hivemq>

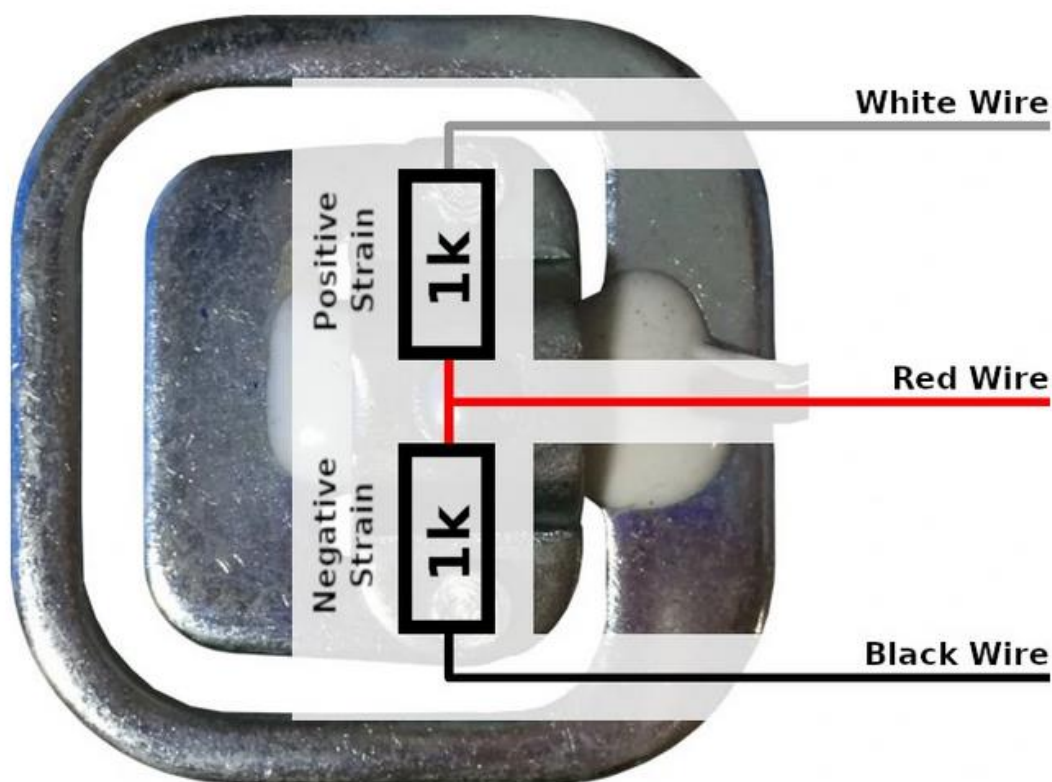
lib <https://github.com/FDelporte/HiveMQ-examples/tree/main/picow-to-hivemq/lib/umqtt>

Συνδεσμολογία αισθητήρων

Θα πρέπει να βρούμε ποιο είναι το μεσαίο καλώδιο. Για να το βρούμε μετράμε την αντίσταση μεταξύ των καλωδίων. Το ένα άκρο με το άλλο άκρο έχουν 2kΩ αντίσταση ενώ το μεσαίο με τα άκρα έχουν 1kΩ. Το μεσαίο καλώδιο συνδέουμε στο HX711. Για να δούμε αν τα έχουμε συνδέσει καλά μεταξύ τους

Θα πρέπει το E+ με το E- να έχουν 2Ω | E+ με A- 1.5Ω | E+ με A+ 1.5Ω | A+ με το A- να έχουν 2Ω

I have white, red, and black:



Να ελέγξω πως έχω βάλει τα καλώδια γιατί στέλνει αρνητικό νούμερο αν είναι ανάποδα

Raspberry Pico W

HX711

HX711

LOAD CELL

PIN 6	GP 4	→	DAT
PIN 7	GP 5	→	CLK
PIN 36	3V3 OUT	→	VCC
PIN 36	3V3 OUT	→	VDD
PIN 13	GND	→	GND

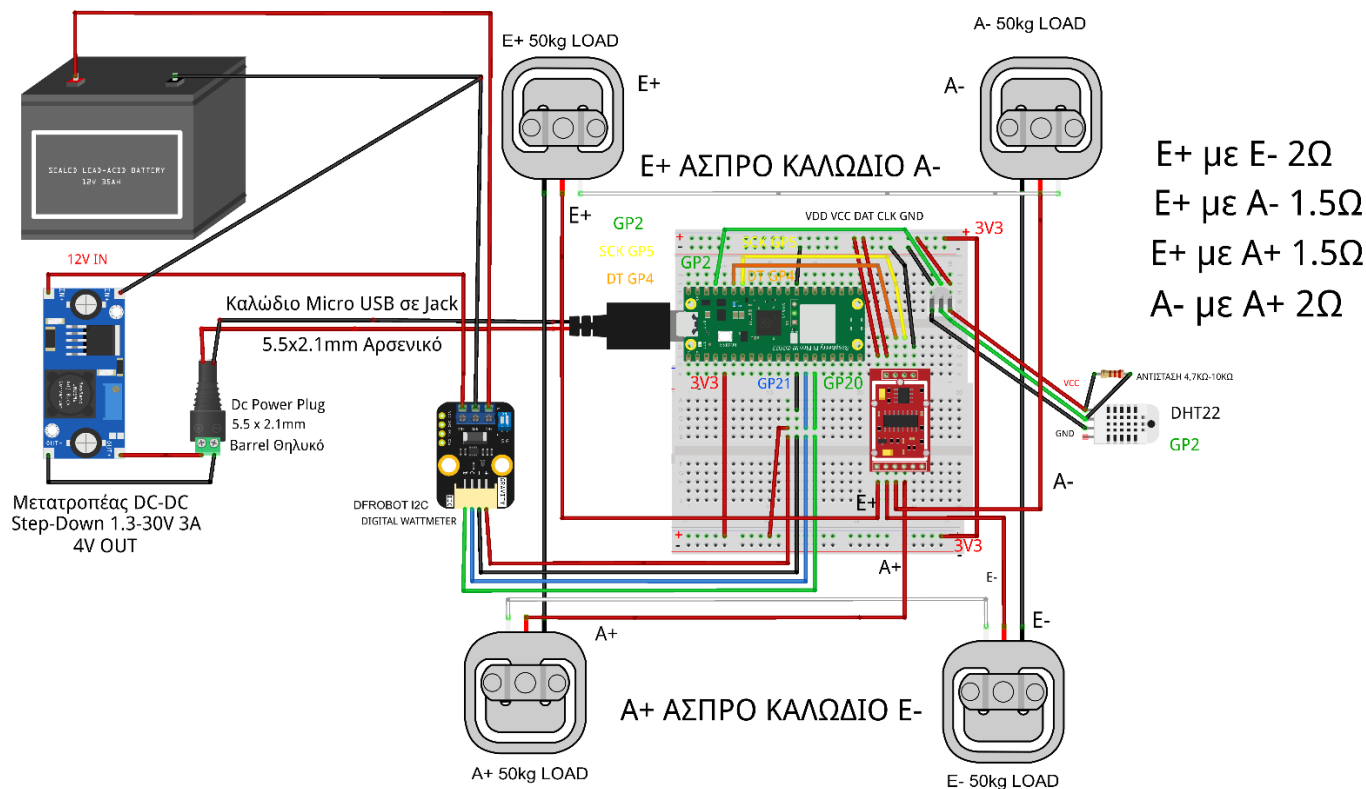
Pin RED	→	E+
Pin BLACK	→	E-
Pin WHITE	→	A-
Pin GREEN	→	A+

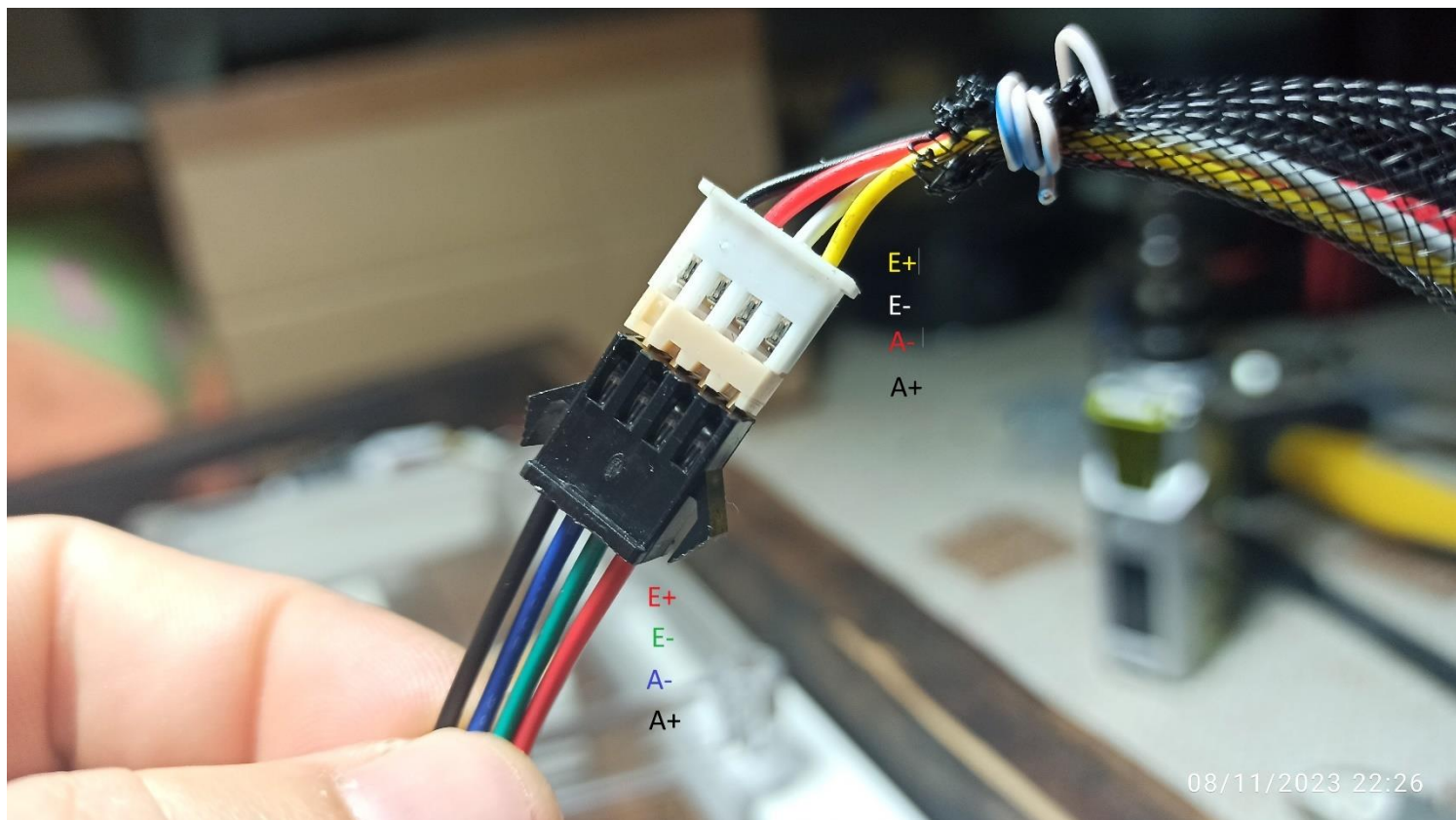
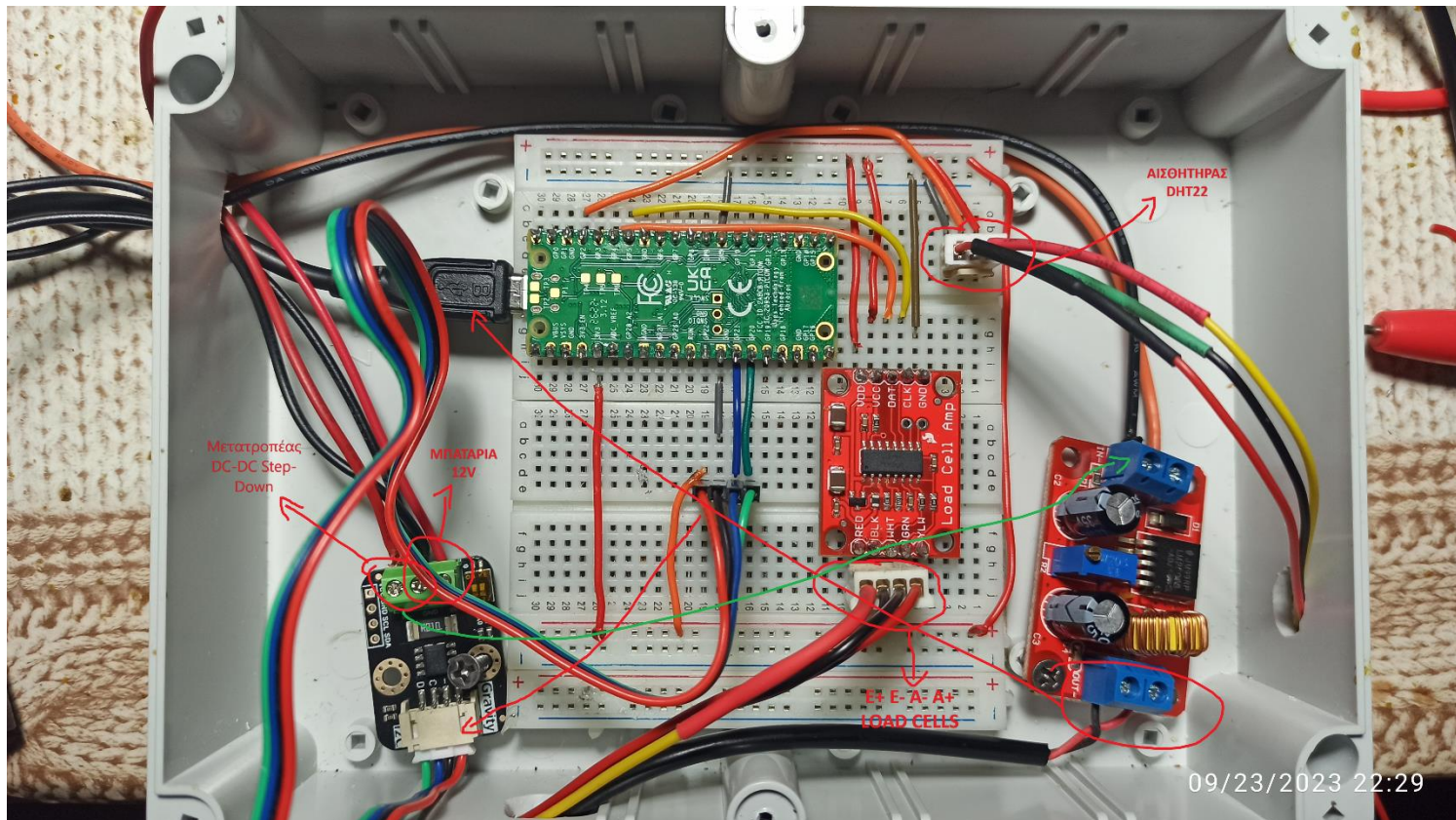
Διάγραμμα

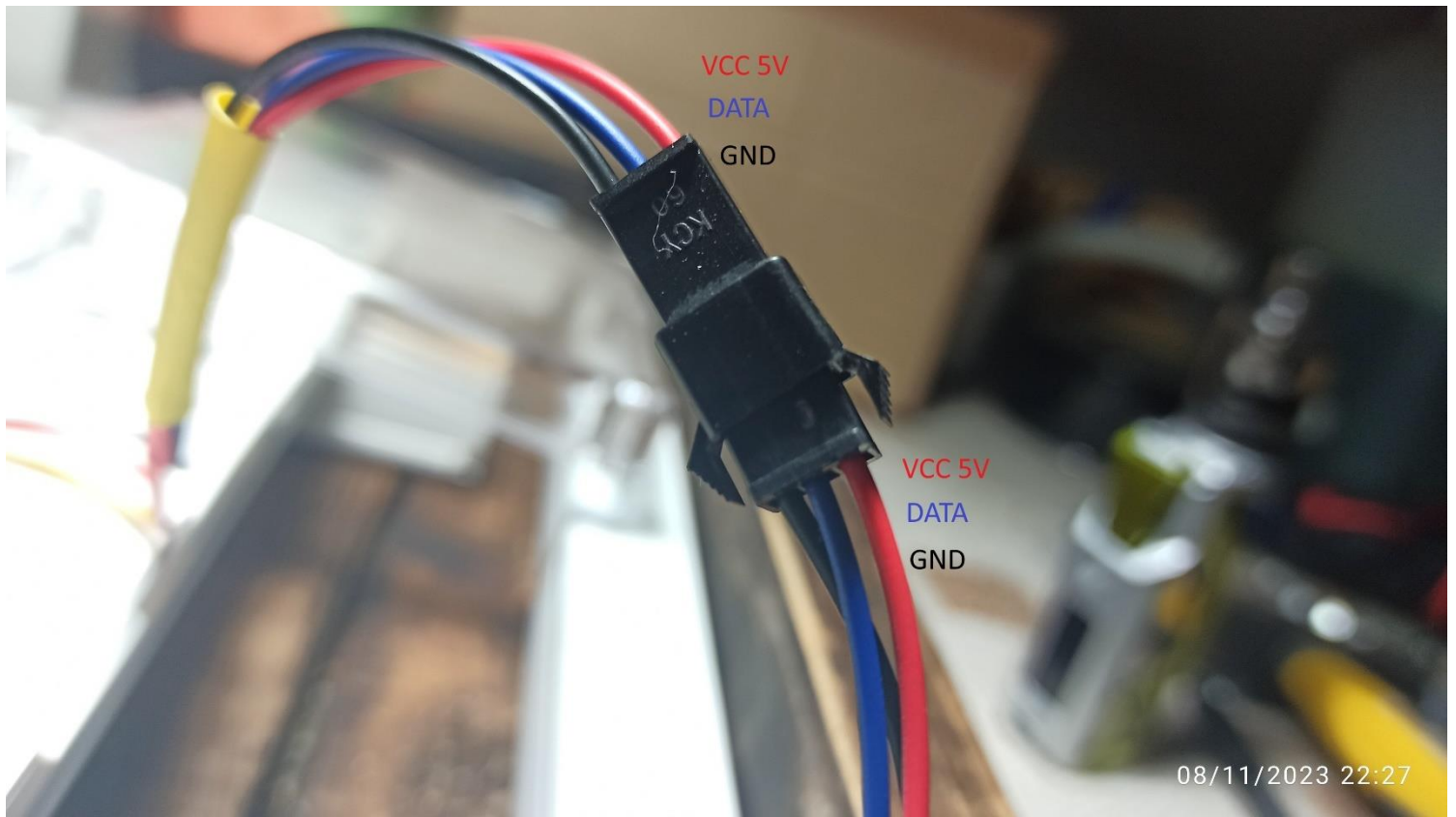
Το PICO W το έχω βάλει ανάποδα (στην εικόνα κάτω είναι κανονικά) με το wifi chip προς τα κάτω για να βλέπω τα gr pins

Οι αντιστοιχίες είναι όπως δείχνει η εικόνα αν το wifi chip είναι προς τα κάτω και όχι όπως φαίνεται στην εικόνα.

ΘΑ ΠΡΕΠΕΙ ΤΟ PICO W ΝΑ ΕΙΝΑΙ ΜΕ ΤΟ WIFI ΚΑΤΩ
ΓΙΑ ΝΑ ΤΑΙΡΙΞΑΖΟΥΝ ΤΑ GP ΟΠΩΣ ΣΤΗΝ ΕΙΚΟΝΑ







VCC 5V

DATA

GND

VCC 5V

DATA

GND

08/11/2023 22:27

Αποστολή δεδομένων

Υπάρχουν δύο τρόποι να στέλνει τα δεδομένα, και στου δύο χρησιμοποιώ Django σε Debian lxd container με apache2 (Qnap Nas)

Πρώτα Εγκατάσταση Django μέσα στο Container με venv python και μετά μπορείς να βάλεις τα αρχεία από το github repo

1. Requests
2. Mqtt

Και οι δύο τρόποι υπάρχουν στα αρχεία στο github με διαφορετικούς μεθόδους python

1. Ο πρώτος στέλνει urequests

```
response = urequests.get(url=secrets['url'], headers=headers, data=payload)
```

το url είναι που έχουμε ορίσει εμείς στο django στο urls.py:

```
path('add/', add_weight, name='add_weight'),
```

και στο views.py:

```
def add_weight(request, *args, **kwargs):  
    και εδώ τα περνάμε στην βάση που παίρνουμε από το urequests ( data=payload)
```

2. Ο Δεύτερος στέλνει τα δεδομένα στο hivemq.cloud.

Κάνεις έναν δωρεάν λογαριασμό στο hivemq.cloud

Δημιουργείς χρήστες με τα ανάλογα δικαιώματα, ένα για publish που χρειάζεται το pico για να στέλνει τα δεδομένα, ένα για subscribe για το Django για να παίρνει τα δεδομένα από hivemq

Στο mqtt.py αρχείο στο pico ο server είναι το Cluster URL που θα το βρεις στο Cluster Details στην σελίδα του hivemqt (είναι διαφορετικό για κάθε λογαριασμό)

Έτσι το pico στέλνει τα δεδομένα στο hivemq με την μέθοδο `sent_to_HiveMQ_Cloud(data_to_send)` που είναι στο αρχείο mqtt.py μέσα στο pico w.

Κάνεις install το paho-mqtt <https://pypi.org/project/paho-mqtt/> στο venv του debian

Μέσα στο φάκελου του app στο Django κάνεις ένα νέο αρχείο mqtt_client.py

Και εκεί ορίζεις το broker_url που είναι πάλι το Cluster URL, username – pass τον χρήστη που έκανες μόνο για subscribe και το ορίζεις στο αρχείο credentials.py

```
client.username_pw_set(username, password)
```

Μετά κάνεις connect και subscribe

```
client.connect(broker_url, 8883)
```

```
client.subscribe("picow/#", qos=2) # " picow/#" → είναι το topic
```

Τα δεδομένα τα παίρνουμε με την μέθοδο `on_message(client, userdata, msg)`: στο αρχείο mqtt_client.py που είναι στον server μέσα στο φάκελο του app του Django, και τα αποθηκεύουμε στην βάση

Λαμπάκι Pico W

Όταν ξεκινάει το pico αναβοσβήνει το λαμπάκι ανάλογα σε ποιο μέρος της διαδικασίας είναι

- 2 φορές get_temp_humidity() -----πριν πάρει την θερμοκρασία και υγρασία
- 3 φορές get_weight() -----πριν πάρει το βάρος
- 4 φορές get_pico_temp() -----πριν πάρει θερμοκρασία pico
- 5 φορές measure_vsys() -----πριν πάρει την τάση
- 5 δευτερόλεπτά ανοιχτό wifi connected -----όταν συνδεθεί στο Wifi
- 6 φορές get_date_time() -----πριν πάρει την ημερομηνία
- 7 φορές sent_data() -----πριν στείλει τα δεδομένα
- 8 φορές response.status_code == 200 -----επιτυχημένη αποστολή δεδομένων
- 9 φορές disconnect() -----πριν την αποσύνδεση