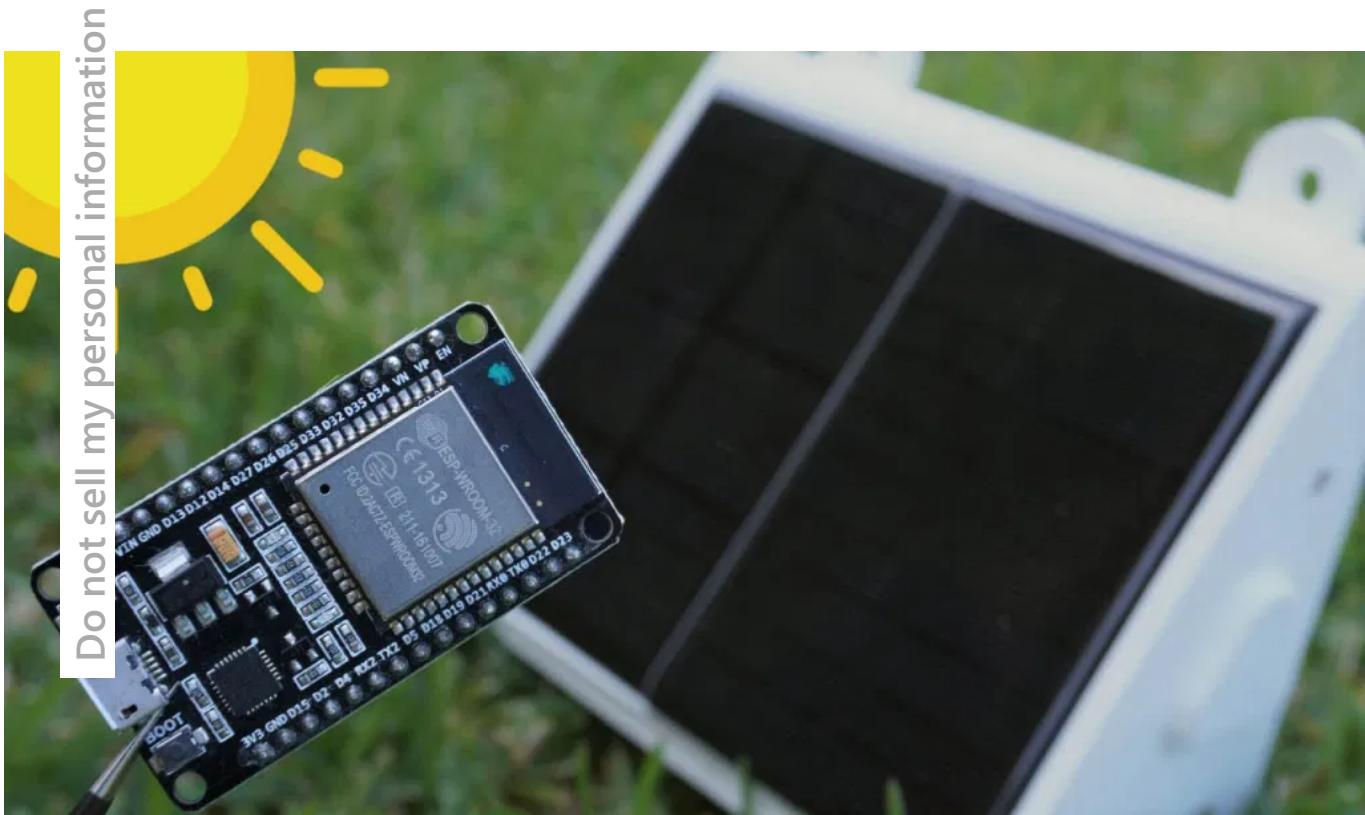


Power ESP32/ESP8266 with Solar Panels (includes battery level monitoring)

This tutorial shows step-by-step how to power the ESP32 development board with solar panels, a 18650 lithium battery and the TP4056 battery charger module. The circuit we'll build is also compatible with the ESP8266 or any microcontroller that is powered at 3.3V.



When you power your ESP32 with solar panels, it may be useful to use its deep sleep capabilities to save power. Learn everything you need to know about deep sleep with the ESP32 with our guide: [ESP32 Deep Sleep with Arduino IDE and Wake Up Sources](#).

Parts Required

To power the ESP32 or ESP8266 with solar panels, we'll use the following parts:

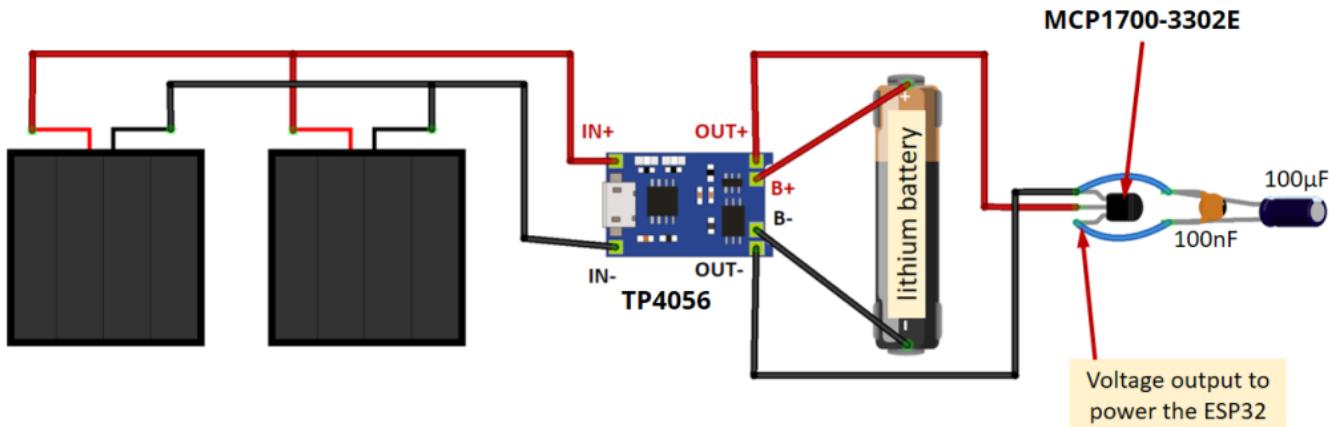
- Lithium Li-ion battery 18650
- Battery holder
- **Battery charger (optional)**
- **TP4056 Lithium Battery Charger Module**
- Voltage regulator:
 - Low-dropout or LDO regulator (MCP1700-3302E)
 - 100uF electrolytic capacitor
 - 100nF ceramic capacitor
- **Optional** – voltage divider for battery monitor:
 - 27k Ohm resistor
 - 100k Ohm resistor

You can use the preceding links or go directly to MakerAdvisor.com/tools to find all the parts for your projects at the best price!



ESP32 Solar Powered – Circuit Overview

The following diagram shows how the circuit to power the ESP32 with solar panels works.



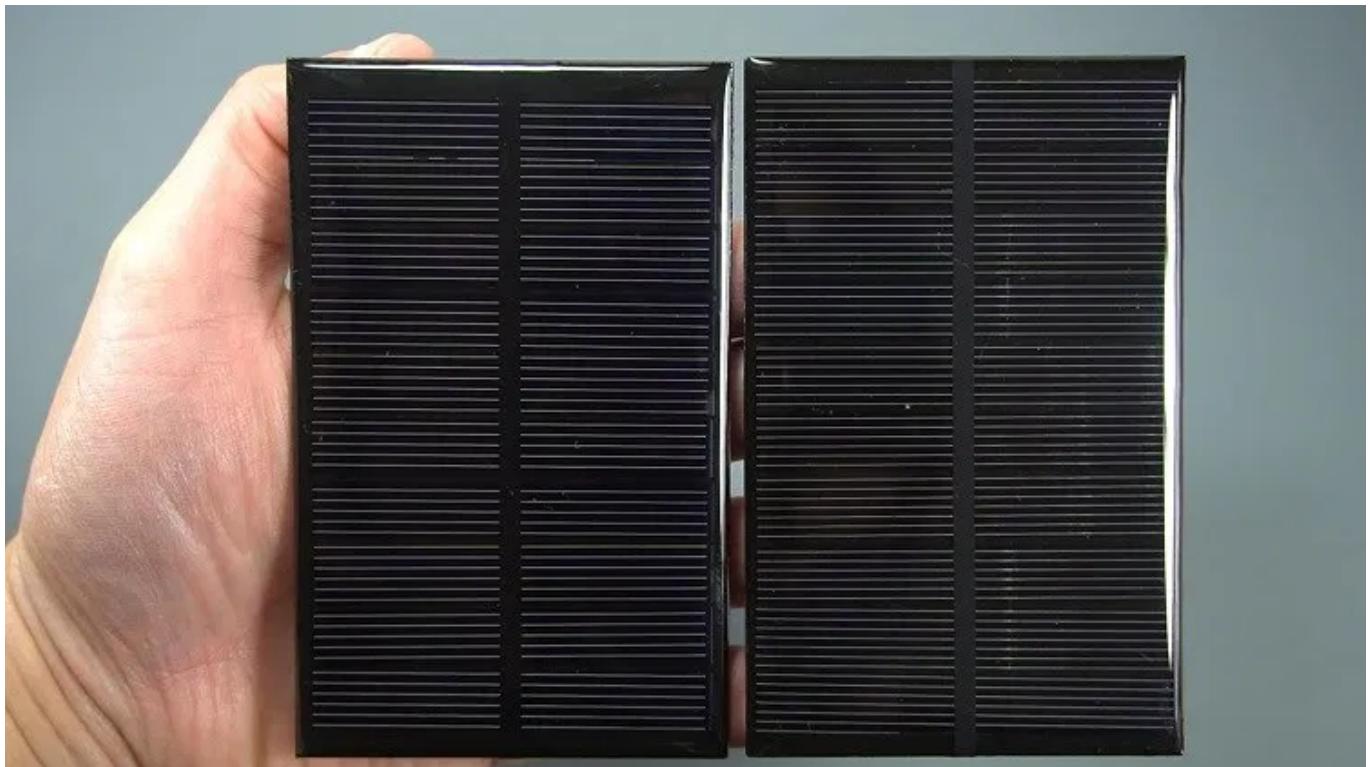
1. The solar panels output between 5V to 6V with direct sun.

prevent overcharging.

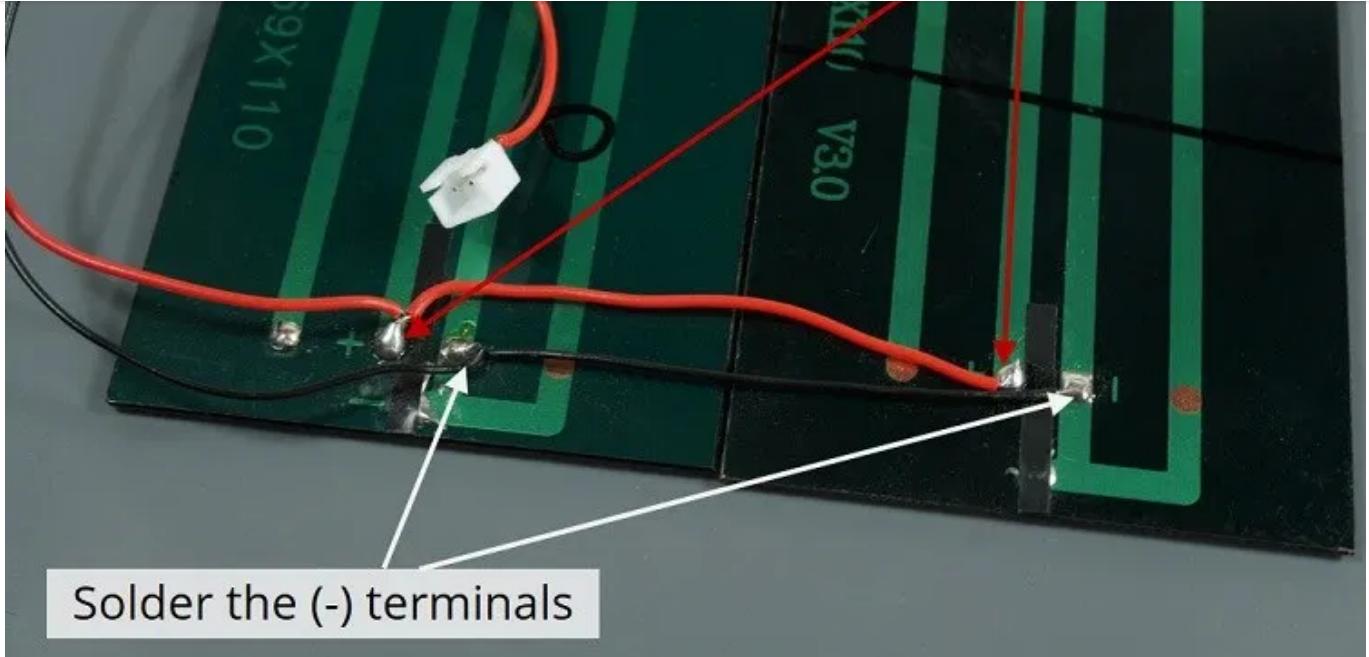
3. The lithium battery outputs 4.2V when fully charged.
4. You need to use a low dropout voltage regulator circuit ([MCP1700-3302E](#)) to get 3.3V from the battery output.
5. The output from the voltage regulator will power the ESP32 through the 3.3V pin.

Solar Panels

The solar panels we're using have an output voltage up to between 5V to 6V. If you want your battery to charge faster, you can use several solar panels in parallel. In this example we're using [two mini solar panels](#) as shown in the following figure.



To wire solar panels in parallel solder the (+) terminal of one solar panel to the (+) terminal of the other solar panel. Do the same for the (-) terminals. It may help taking a look at the following figure.

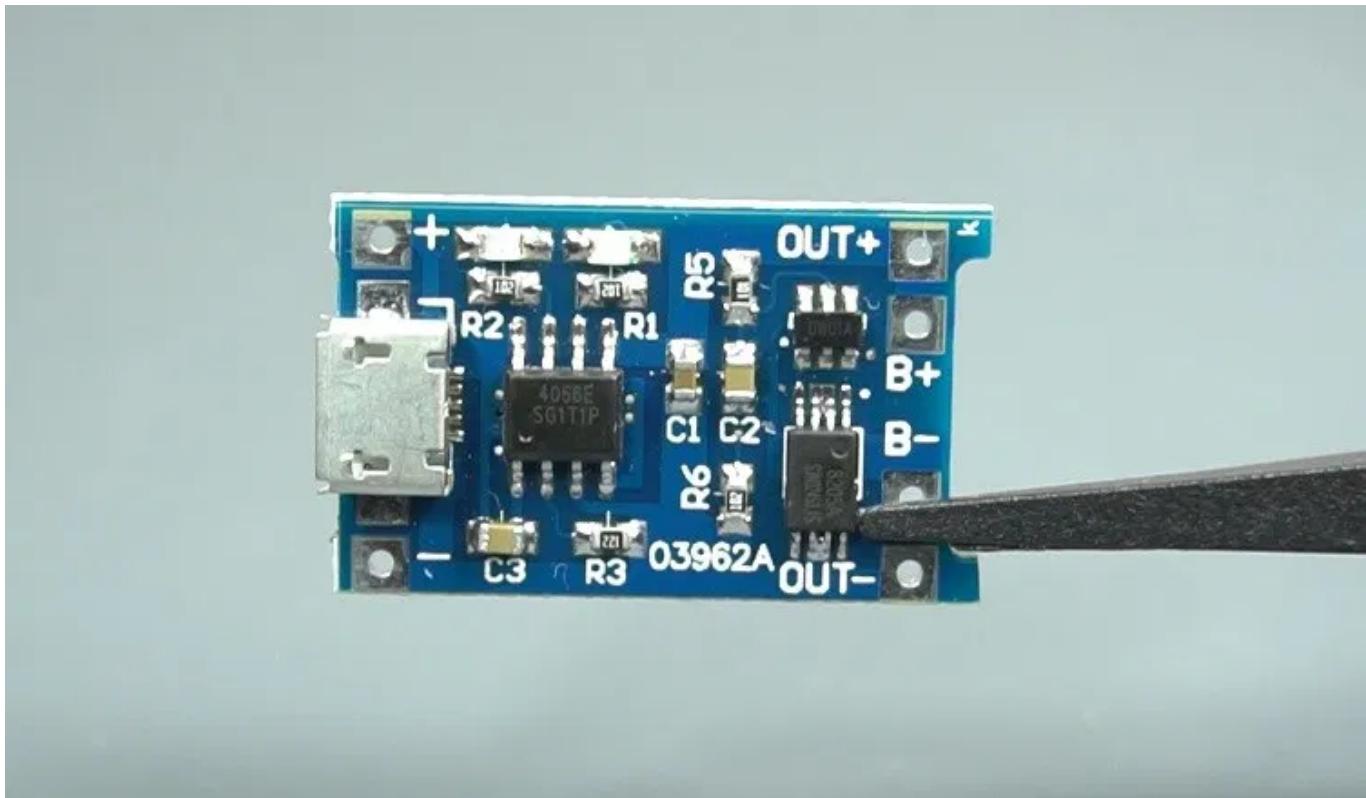


When wiring solar panels in parallel you'll get the same output voltage, and double the current (for identical solar panels). As you can see in the following figure, the solar panels output approximately 6V.



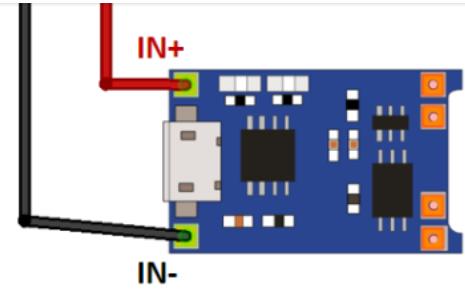
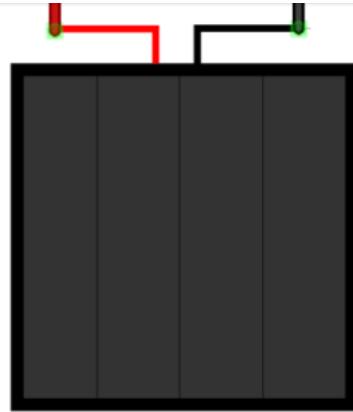
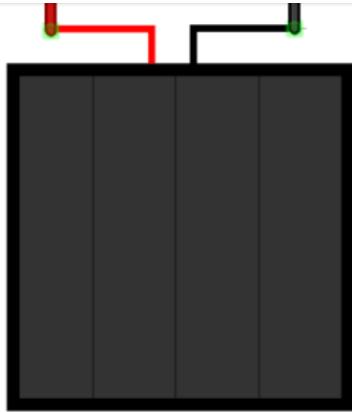
TP4056 Charger Module

The [TP4056 lithium battery charger module](#) comes with circuit protection and prevents battery over-voltage and reverse polarity connection.

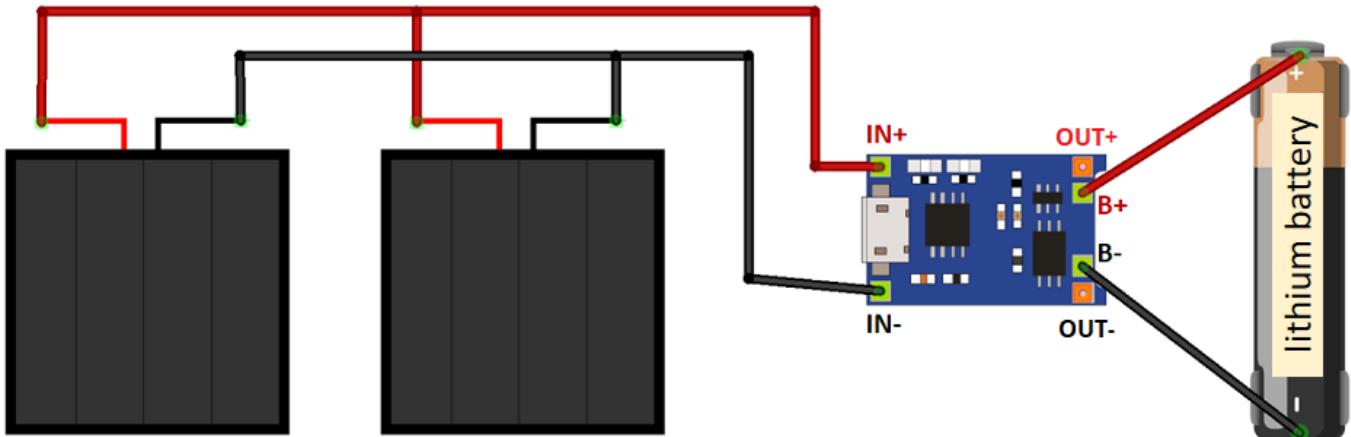


The TP4056 module lights up a red LED when it's charging the battery and lights up a blue LED when the battery is fully charged.

Wire the solar panels to the TP4056 lithium battery charger module as shown in the schematic diagram below. Connect the positive terminals to the pad marked with **IN+** and the negative terminals to the pad marked with **IN-**.



Then, connect the battery holder positive terminal to the **B+** pad, and the battery holder negative terminal to the **B-** pad.



The **OUT+** and **OUT-** are the battery outputs. These lithium batteries output up to 4.2V when fully charged (although they have 3.7V marked in the label).

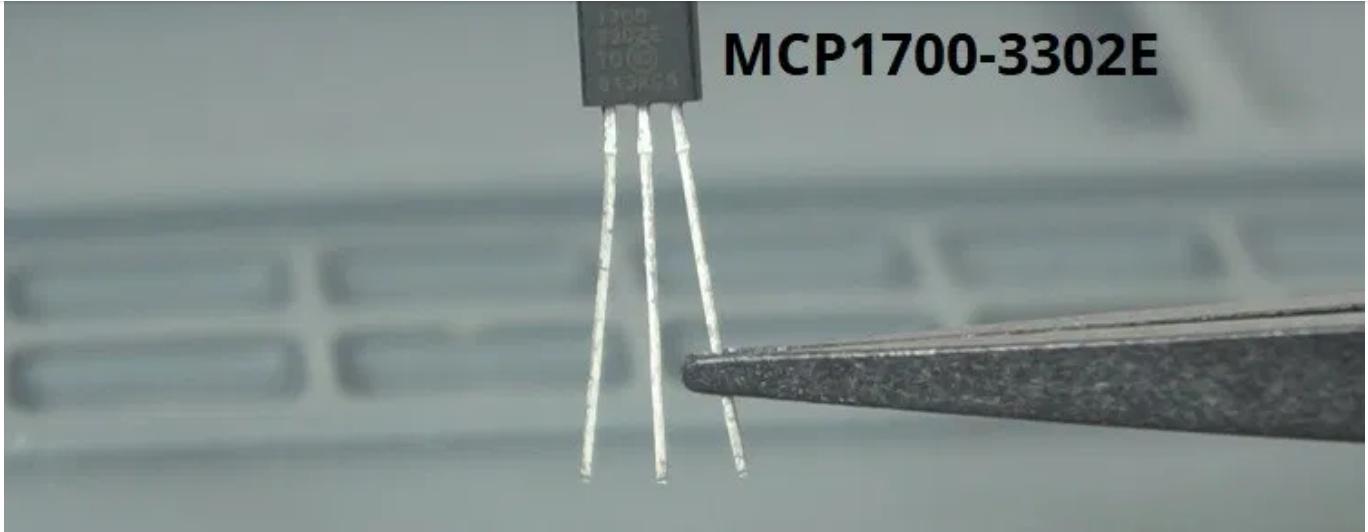


To power the ESP32 through its 3.3V pin, we need a voltage regulator circuit to get 3.3V from the battery output.

Voltage Regulator

Using a typical linear voltage regulator to drop the voltage from 4.2V to 3.3V isn't a good idea, because as the battery discharges to, for example 3.7V, your voltage regulator would stop working, because it has a high cutoff voltage.

To drop the voltage efficiently with batteries, you need to use a low-dropout regulator, or LDO for short, that can regulate the output voltage.



After researching LDOs, the [MCP1700-3302E](#) is the best for what we want to do. There is also a good alternative like the HT7333-A.



Any LDO that has similar specifications to these two are also good alternatives. Your LDO should have similar specs when it comes to output voltage, quiescent current, output current and a low dropout voltage. Take a look at the datasheet below.

Low Quiescent Current LDO

Features:

- 1.6 μ A Typical Quiescent Current
- Input Operating Voltage Range: 2.3V to 6.0V
- Output Voltage Range: 1.2V to 5.0V
- 250 mA Output Current for Output Voltages \geq 2.5V
- 200 mA Output Current for Output Voltages $<$ 2.5V
- Low Dropout (LDO) Voltage
 - 178 mV Typical @ 250 mA for $V_{OUT} = 2.8V$
- 0.4% Typical Output Voltage Tolerance
- Standard Output Voltage Options:
 - 1.2V, 1.8V, 2.5V, 2.8V, 3.0V, 3.3V, 5.0V
- Stable with 1.0 μ F Ceramic Output Capacitor
- Short Circuit Protection
- Overtemperature Protection

General Description:

The MCP1700 is a family of CMOS low dropout (LDO) voltage regulators that can deliver up to 250 mA of current while consuming only 1.6 μ A of quiescent current (typical). The input operating range is specified from 2.3V to 6.0V, making it an ideal choice for two and three primary cell battery-powered applications, as well as single cell Li-Ion-powered applications.

The MCP1700 is capable of delivering 250 mA with only 178 mV of input to output voltage differential ($V_{OUT} = 2.8V$). The output voltage tolerance of the MCP1700 is typically $\pm 0.4\%$ at $+25^\circ C$ and $\pm 3\%$ maximum over the operating junction temperature range of $-40^\circ C$ to $+125^\circ C$.

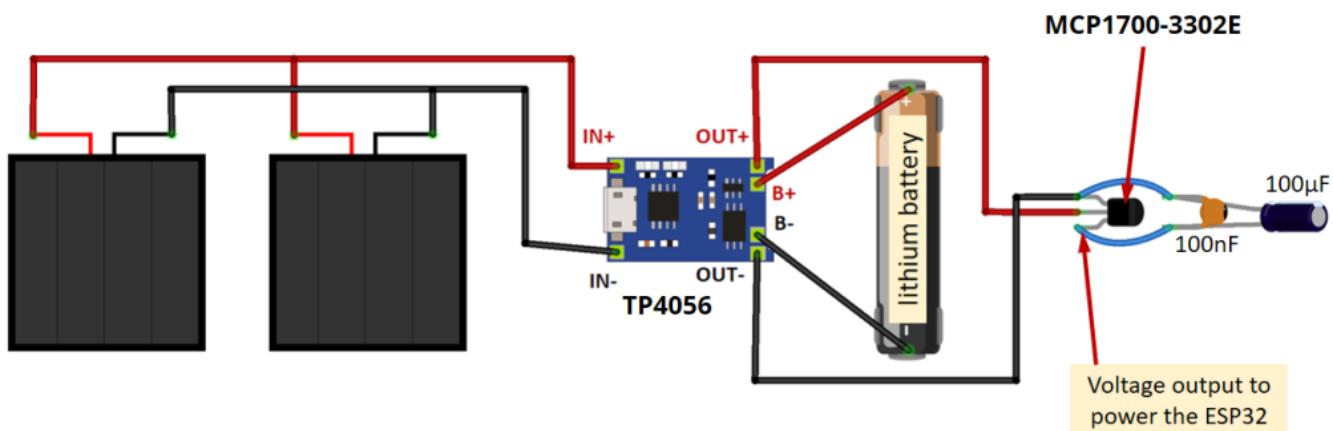
Output voltages available for the MCP1700 range from 1.2V to 5.0V. The LDO output is stable when using only 1 μ F output capacitance. Ceramic, tantalum or aluminum electrolytic capacitors can all be used for

Here's the MCP1700-3302E pinout: GND, VIN and VOUT pins.

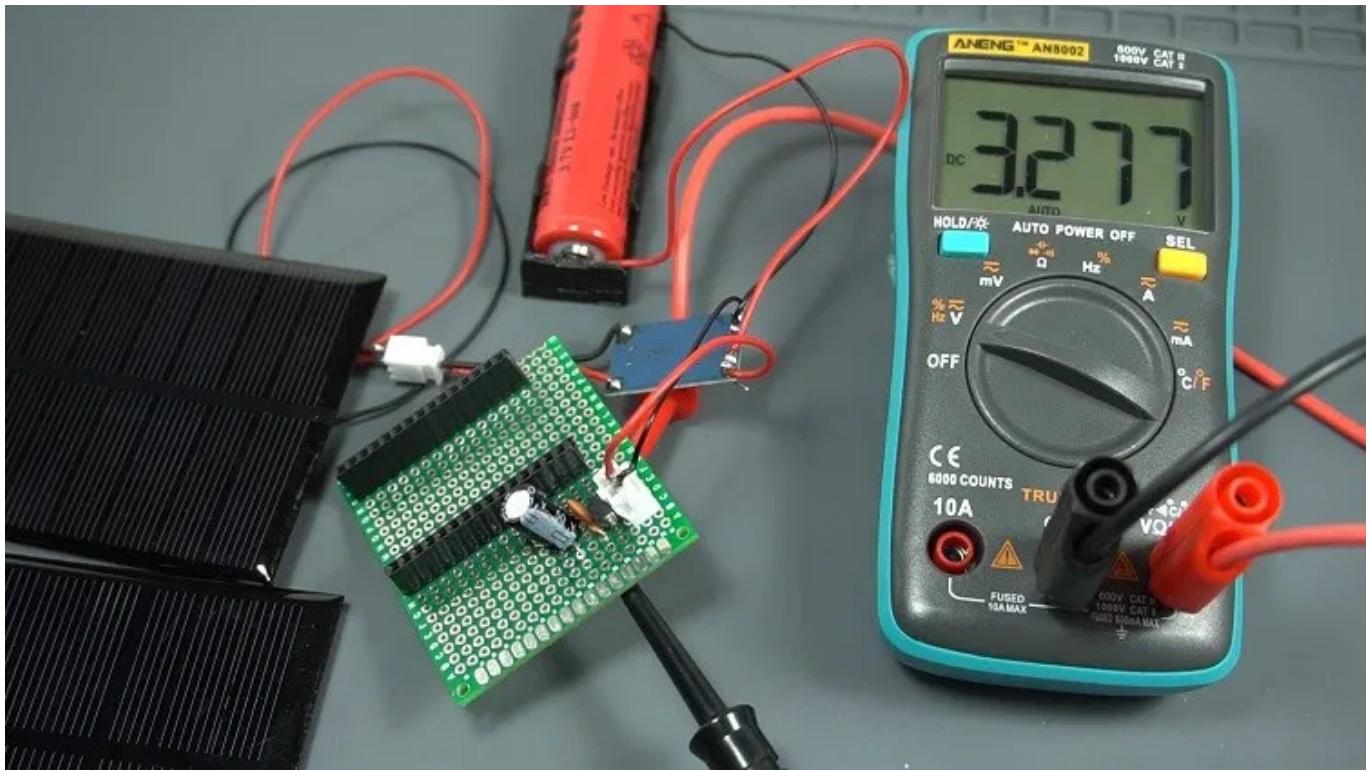


The LDOs should have a ceramic capacitor and an electrolytic capacitor connected in parallel to GND and Vout to smooth the voltage peaks. Here we're using a [100uF electrolytic capacitor](#), and a [100nF ceramic capacitor](#).

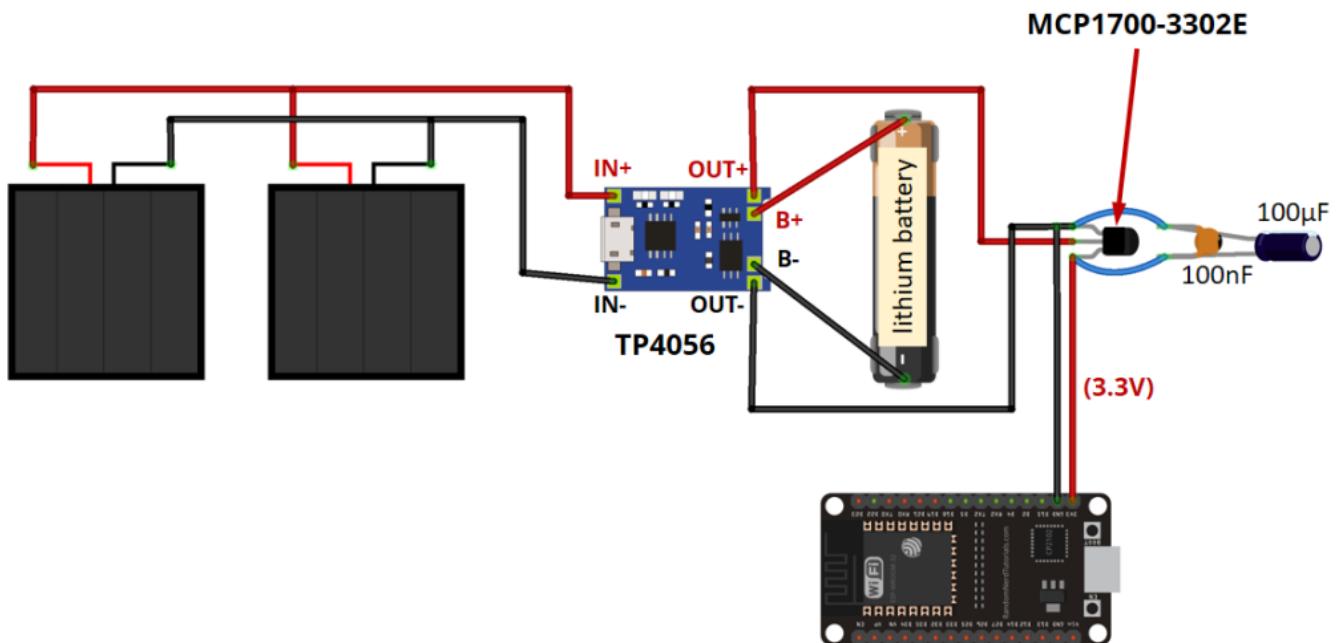
Follow the next schematic diagram to add the voltage regulator circuit to the previous setup.

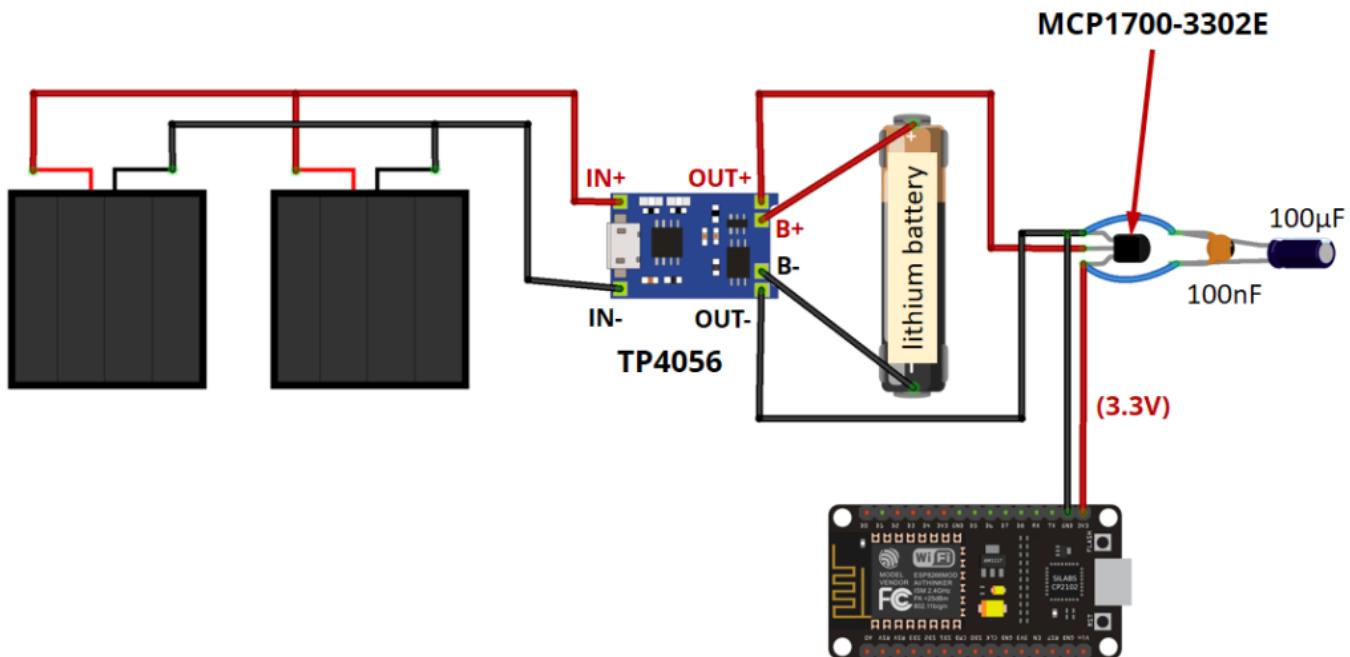


Warning: electrolytic capacitors have polarity! The lead with the white/gray strip should be connected to GND.



Finally, after making sure that you're getting the right voltage on the Vout pin of the voltage regulator, you can power the ESP32. Connect the Vout pin to the 3.3V pin of the ESP32 and GND to GND.

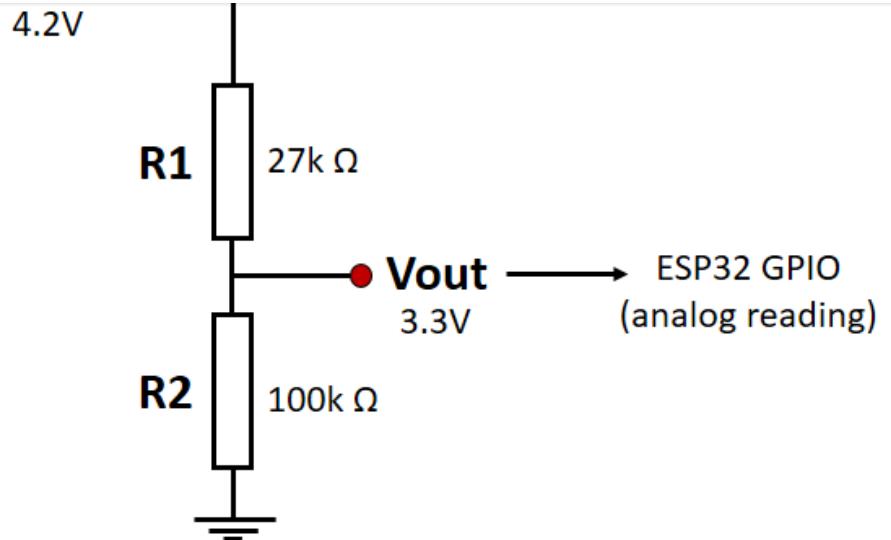




Battery Voltage Level Monitoring Circuit

When you have your ESP32 powered with batteries or solar powered as in this case, it can be very useful to monitor the battery level. One way to do that is reading the output voltage of the battery using an analog pin of the ESP32.

However, the battery we're using here outputs a maximum of 4.2V when fully charged, but the ESP32 GPIOs work at 3.3V. So, we need to add a voltage divider so that we're able to read the voltage from the battery.



The voltage divider formula is as follows:

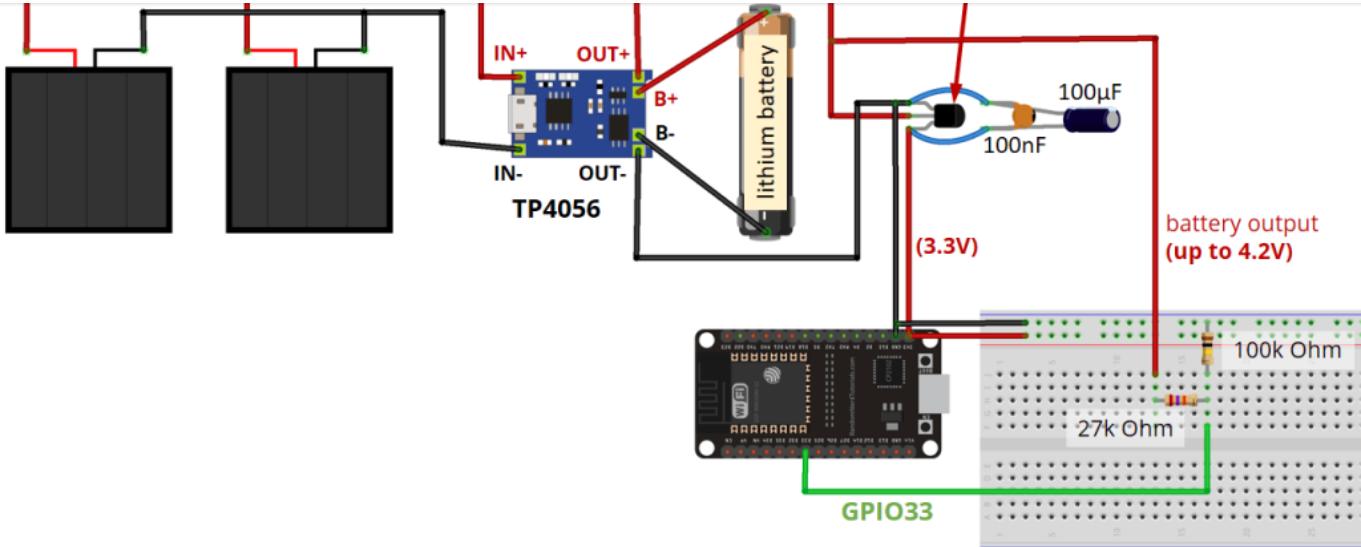
$$V_{out} = (V_{in} \cdot R_2) / (R_1 + R_2)$$

So, if we use R1=27k Ohm, and R2=100k Ohm, we get:

$$V_{out} = (4.2 \cdot 100k) / (27k + 100k) = 3.3V$$

So, when the battery is fully charged, the Vout outputs 3.3V that we can read with an ESP32 GPIO.

Add two resistors to your circuit as shown in the following schematic diagram.



In this case, we're monitoring the battery level through `GPIO33`, but you can use any other suitable GPIO. Read our [ESP32 GPIO guide](#) to learn which GPIOs are the best to use.

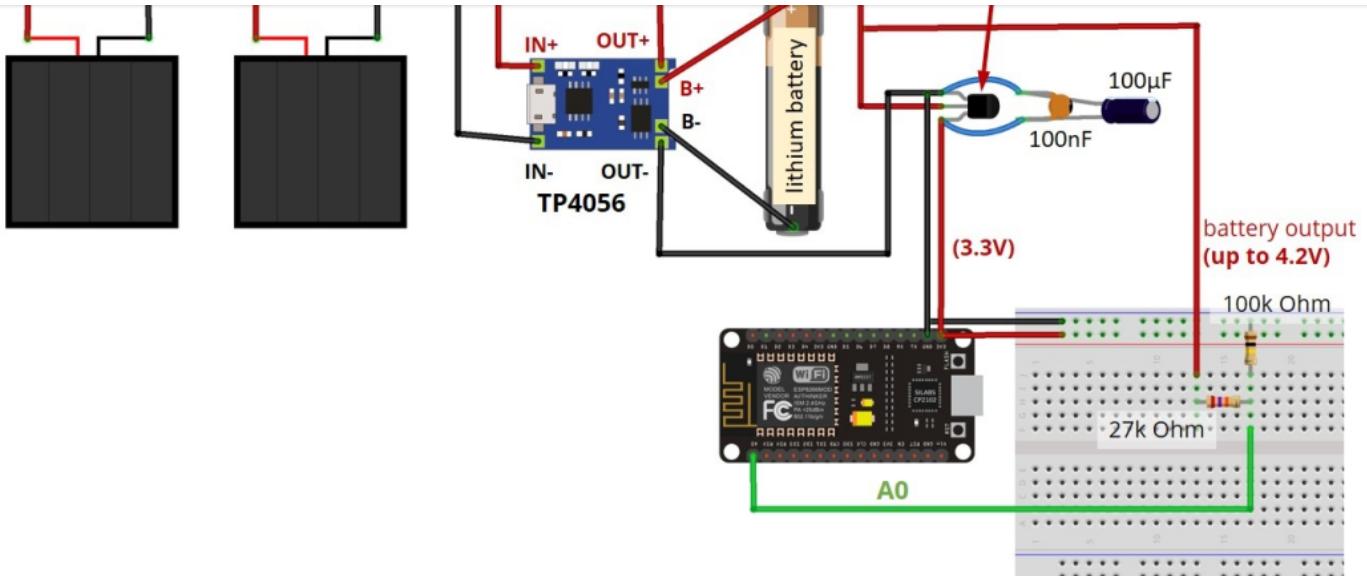
Finally, to get the battery level, you can simply read the voltage on `GPIO33` using the `analogRead()` function in your code (if you're using Arduino IDE).

```
analogRead(33);
```

You can also use the `map()` function, to convert the analog values to a percentage:

```
float batteryLevel = map(analogRead(33), 0.0f, 4095.0f, 0, 100);
```

If you're using ESP8266, it just supports analog reading on the `A0` pin. So, you need to wire the circuit as follows:



With the ESP8266 to read the analog value use:

```
analogRead(0);
```

Wrapping Up

In this article we've shown you how to power the ESP32 or the ESP8266 with solar panels, a lithium battery and a TP4056 battery charger module. The circuit we've shown you can also be used to power other microcontrollers that require 3.3V to operate.

When powering the ESP32 using solar panels or batteries, it is important to save power. For that, you can use the [ESP32 deep sleep capabilities](#).

Now, you can use this circuit to make your projects solar powered. For example, it would be interesting to modify the following projects to use solar panels:

- [ESP32 Data Logging Temperature to MicroSD Card](#)
- [ESP32 Web Server with BME280 – Mini Weather Station](#)
- [Low Power Weather Station Datalogger](#)

Thanks for reading.

PCBWay HIGH-QUALITY PCB
ONLY \$5 FOR 10 PIECES

- Rogers, HDI, aluminum and rigid-flex PCB are available now
- Production time 24 hours

PCB ASSEMBLY
Free shipping + Free stencil
ONLY \$30

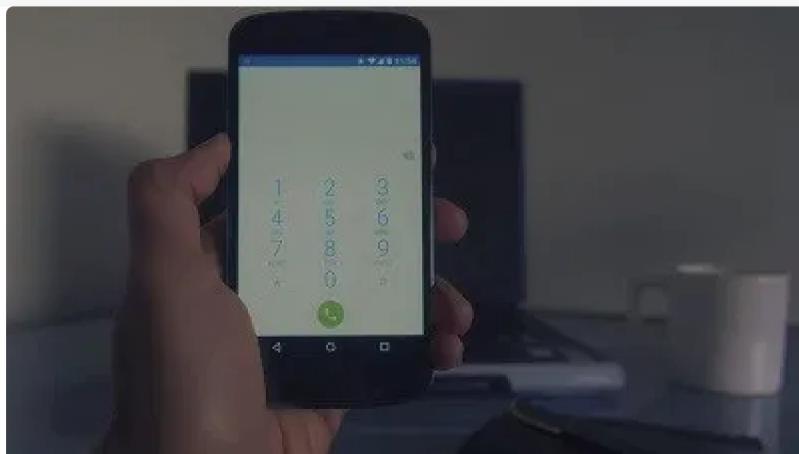
- Component sourcing
- Quality assurance

[eBook] Build Web Servers with ESP32 and ESP8266 (2nd Edition)

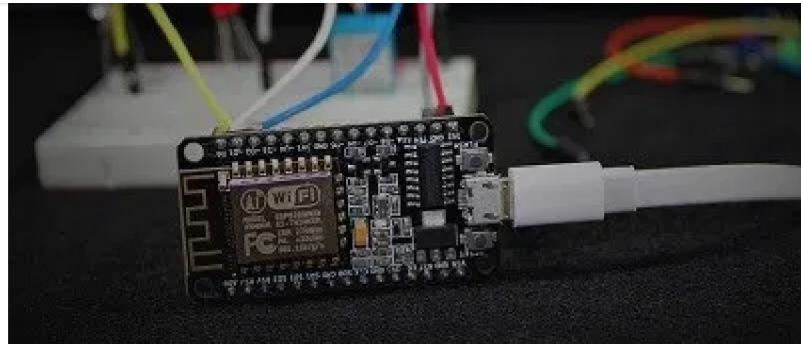


Build Web Server projects with the ESP32 and ESP8266 boards to control outputs and monitor sensors remotely. Learn HTML, CSS, JavaScript and client-server communication protocols [DOWNLOAD »](#)

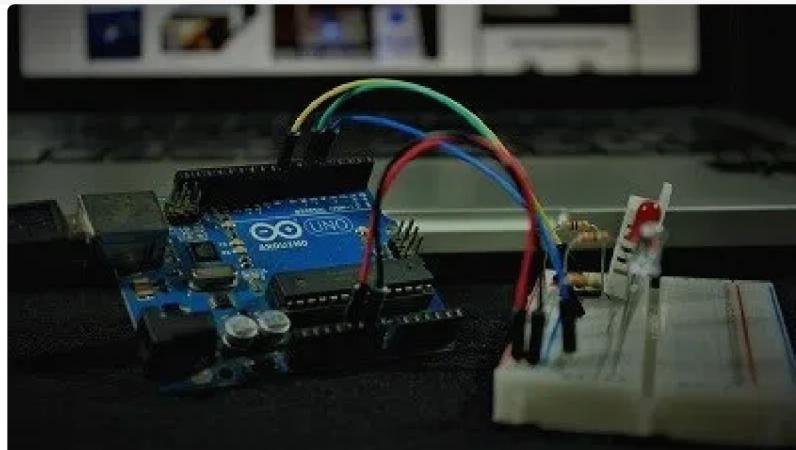
Recommended Resources



[Build a Home Automation System from Scratch »](#) With Raspberry Pi, ESP8266, Arduino, and Node-RED.



[Home Automation using ESP8266 eBook and video course »](#) Build IoT and home automation projects.



[Arduino Step-by-Step Projects »](#) Build 25 Arduino projects with our course, even with no prior experience!

What to Read Next...

[ESP32/ESP8266 Digital Inputs and Digital Outputs with MicroPython](#)

[Install ESP32 Filesystem Uploader in Arduino IDE](#)

[ESP8266 NodeMCU with Stepper Motor \(28BYJ-48 and ULN2003 Motor Driver\)](#)

newsletter!

Your Email Address

SUBSCRIBE

138 thoughts on “Power ESP32/ESP8266 with Solar Panels (includes battery level monitoring)”

Nick

May 2, 2019 at 2:56 pm

Please please please remove that link to 18650s and post one from a reputable source, most batteries won't be over 3200mAh and those are only from name brand sources!!

[Reply](#)

Ed

May 4, 2019 at 3:34 pm

Afaik the max input on the A0 of the esp8266 is 1 Volt. Need to adapt the resistors for that

[Reply](#)

May 5, 2019 at 9:17 am

Hi Ed.

If you're using an ESP8266 NodeMCU kit (as we're using in the example) it has a built-in voltage divider. So, the input range is 0 to 3.3V.

If you're using an ESP8266 bare chip, the input range of the A0 is 0 to 1V.

Regards,

Sara

[Reply](#)

ed

September 28, 2019 at 4:29 pm

That is certainly true, but I was under the impression the battery voltage was measured and that can be 4.2Volt, leading to 1.3Volts on A0.

That will not immediately kill the ESP, but make it impossible to measure battery voltage above 3.3 Volt.

I apologize for nitpicking, but might be an option to add a 100k resistor so one has $100k/(100k+100k+220k) = 1/4.2$ as voltage divider. Anyway, that's just a suggestion

[Reply](#)

ABHAY S BHARADWAJ

May 2, 2019 at 3:40 pm

When working with battery, you are using map function to convert 0v to 3.3v of the voltage divider into 0 to 100%.

The issue with batteries is that anything less than 3.4v (or something like this) is considered 0% (in case of LiPo/Lilon batteries). So 0V is a state where the battery is dead and cannot be brought back to life.

The map function should hence map between values 3.4v and 4.2v between 0 and 100% and not 0v to 4.2v.

[Reply](#)

Rui Santos

May 5, 2019 at 10:07 am

Thanks for pointing that out. I know it and that's definitely true. It's also not linear. At around 70% the battery is fully discharged, but it was just something that you can use to have an idea of the current level.

[Reply](#)

Arturo

November 19, 2021 at 2:30 pm

Hi Rui, would you consider to use a MOSFET to power up the divider, in order to avoid the energy drop during deep sleep time?
Is it worth?
Thanks

Amaral

December 22, 2021 at 11:33 am

Hi Arturo! I had the same problem, I got around it with a very simple and cheap solution. I used a Pc817 photocoupler. It switches efficiently.

[Reply](#)**ong thi**

January 1, 2021 at 12:56 pm

Can you show me how to place the map function?

[Reply](#)**v**

March 17, 2021 at 3:11 am

$\text{Vout} = (4.2100k)/(27k + 100k) = 3.3V$, When the battery voltage is 3V $\text{Vout}=2.3622V$, $2.36224095/3.3=2931$
batteryPower = map(analogRead(BATTERY_PIN), 2931, 4095, 3, 4.2);

[Reply](#)

May 2, 2019 at 4:05 pm

it should work on 4.2v without regulator, did you test it?

[Reply](#)

Rui Santos

May 5, 2019 at 10:05 am

If you power the ESP directly, it needs 3.3V. Otherwise, you'll fry your board...

[Reply](#)

zygfryd

September 30, 2021 at 9:41 pm

actually no

I am using bare ESP32 and S2 and all are powered directly from LiPo up to 4.2V – none died yet 😊

[Reply](#)

CARLOS D ALEXANDRIA BRUNI

May 2, 2019 at 4:21 pm

Carlos Bruni

[Reply](#)

DooMMasteR

May 2, 2019 at 6:48 pm

Removing the red and blue LED would be a good idea.

Also, can the ESP even sample voltages so close to Vcc? Shouldn't you voltage devide first?

And as already said, get a good cell, best would be a LiFePo4, which limited to 3.6 V can directly power the ESP32 and will be safer to handle and live longer.

A LiFePo4 protection circuit and a 3.6V schotky diode are enough to safely handle such a system.

[Reply](#)

Rui Santos

May 5, 2019 at 10:08 am

Thanks for the suggestions!

[Reply](#)

May 2, 2019 at 10:51 pm

Articolo interessante, come sempre.
Grazie, continuate così.

[Reply](#)

Rui Santos

May 5, 2019 at 10:08 am

Thank you for reading!

[Reply](#)

David McLaughlin

May 3, 2019 at 1:13 am

This is not a good board to be used to power anything at the same time as charging. You will always be drawing power from the battery at the same time as trying to charge it so the incorrect charging will be applied to the battery depending on what current your circuit draws. The proper type has power path control so that the incoming 5V is fed direct to the output and the battery disconnected from the output. Once the external 5V is removed, the battery is put back into the circuit and everything works fine. These boards are not good designs for anything other than standalone battery charging.

Chris Parsons

May 21, 2019 at 3:06 pm

Interesting – are you able to suggest a better charging circuit?

That is, one with power path control as you suggest?

Regards

Chris

[Reply](#)

Daijoubu

May 21, 2019 at 3:48 pm

Actually, the MCP73871 solar charge IC do load sharing, not as cheap as the TP4056 modules but cheap enough at \$3-4

Linear and handle 6V PV panel max

Chinese close of Adafruit's older design

[Reply](#)

Wolfgang

June 20, 2020 at 10:06 pm

1,69 US\$ Ali. Thx for Advise

[Reply](#)

Nikolas

May 21, 2019 at 5:23 pm

Hi Chris,

look at boards based based on the CN3065 or the BQ24210 that adjust the charge rate based at the available input current, that will work a lot better than the TP4056. At least boards with the CN3065 are available and almost as cheap.

Kind regards

[Reply](#)

Chris Parsons

May 22, 2019 at 11:21 am

Thanks very much, that's helpful 😊

[Reply](#)

May 22, 2019 at 9:17 pm

If you look at Andrea Spiess video, the CN3065 is not anymore efficient than the TP4056 with a small PV panel, might be better under shaded condition?

youtube.com/watch?v=ttyKZnVzic4

I'll save the money towards larger/more PV panel or battery instead.

[Reply](#)

Nikolas

May 22, 2019 at 9:48 pm

Hi,

the TP performs perfectly, on a sunny midsummer noon, no doubt. At least if Rprog is chosen correctly, so that the charge current is less than what the panel provides. In almost all other cases it will not. Any time the panel does not provide enough energy for the TP4056 to run at its programmed current rate defined by Rprog, it will work out of specs.

Somehow there may leak some energy into the Lipo, one cannot tell. Its out of specs.

On solar chargers, the current (e.g. Rprog on the TP) is actually defined by a micro controller that is part of the charger and the charge current is reduced if the panel provides limited power, so that voltage is maintained for the Charger to work within specs. It would be a funny

But why if there are cheap chargers that do.

Kind regards

Daijoubu

May 21, 2019 at 3:38 pm

Sadly, I don't think there's a cheap chinese clone out there that has load sharing, this guy designed an add-on board (+low power 3.3V step down):
hackaday.io/project/18205-tp4056-3v3-load-share-addon-board

It's pretty simple really, only need a mosfet to do the load sharing

The TP4056 would be OK if the load is less than C/10 for proper termination.

(So <100mA at the default 1A charging rate)

[Reply](#)

Barbudor

May 21, 2019 at 3:50 pm

Search for MCP73871 that offers current balancing between battery and circuit. Adafruit has a board for it and you can find a clone on Chinese sites. Waiting for mine and a handful of panels to try it out.

[Reply](#)

Nanie

September 17, 2020 at 7:37 pm

You are totally right. Greatscott on youtube designed a specific board for this purpose, and this is rather complicated.

[Reply](#)**James T Cullins**

September 18, 2020 at 4:09 am

Which board would you recommend?

[Reply](#)**Guido**

November 18, 2021 at 7:42 am

I guess it would be possible to disable the TP4056 by pulling CE low while the microcontroller is operating.

That pin is usually hardwired in to positive voltage on the common boards but a bit of careful tinkering might resolve that

Daijoubu

May 3, 2019 at 4:05 am

One can use a CN3791 and a 12V solar panel for more power.

[Reply](#)

Rui Santos

May 5, 2019 at 10:10 am

I haven't tried that module...

[Reply](#)

Barbudor

May 5, 2019 at 1:32 pm

Looks a great chip. Just ordered 2 modules for testing. Thanks

[Reply](#)

June 1, 2021 at 9:02 am

Hi I would like to use this charger module on my project.

Here are the components that I will be using:

1. 12V 2W solar panel
2. ESP32 CAM
3. 6V DC 4.5AH Sealed Lead Acid Battery

Can I combine these components for my project? Thank you!

[Reply](#)

Mark Woodrow

July 8, 2021 at 7:58 pm

You can certainly use a lead-acid battery which is charged by a solar panel, and then use a 3.3V voltage regulator to power your ESP32. Lead-acid batteries are better able to stand being trickle charged, and are generally much more robust in that way than Lithium ones. You might do better with a battery and panel that are rated at the same voltage, though, so I'd recommend a 12V battery, or 6V panels. Note that you need about 14V to actually charge a 12V battery (search for battery university for an explanation and details)

[Reply](#)

May 3, 2019 at 5:25 am

sorry if i mention it wrong, is it necessary to have a Voltage Regulator and the voltage divider circuit to made 3.3V for the ESP32? Why not just directly supply the Battery voltage to ESP32 (Vin) pin and (GND) pin, since on the broad itself already have a build in voltage regulator.

[Reply](#)

Rui Santos

May 5, 2019 at 10:09 am

That's because the input voltage would be 5V and the battery only outputs 4.2V when fully charged...

[Reply](#)

James G

October 18, 2019 at 4:10 pm

Hi Rui, excellent website and projects. I am really enjoying getting back into Electronics using your tutorials.

Just so that I can be clear on this point. The input voltage (via vin or by the micro usb) has a range of 6-12 V and because you're using a voltage lower than this range it wouldn't work (i.e. the AMS1117 would drop below

[Reply](#)**Sara Santos**

October 18, 2019 at 6:27 pm

Hi James.

That's right. That's why we don't use the VIN pin in this particular case. It wouldn't work with 4.2V from the lithium battery.

Regards,

Sara

[Reply](#)**James T Cullins**

September 18, 2020 at 4:15 am

Sara, I have mine wired to the VIN and it works down to 3.8v, I just checked it. I monitor it using Home Assistant. Having said that, I would not recommend feeding it less than the manufacturer's minimum.

If you have a recommendation for a circuit that will work on the VIN I had rather use it.

Mark Haller

May 3, 2021 at 5:53 pm

within the spec of the battery and the ESP32 voltage. There is a 1n5189 blocking diode on the USB which will protect that end but there will be a voltage across the battery as a result which will be at the higher end of the charge curve which is not ideal but not that important if the USB is not in regular use. I think the charger board will just see it as a fully charged battery and shut down. A second diode could be added to reduce the impact of this but it may be fiddly as they would have to be where the 1n5189 is so possibly better to not do this. I will have a play later. I could be totally wrong on all of this of course, but think i have it right. on the other comments, there can be high current draws on the 32 but the averages are not so bad and a fraction of what the battery can be charged at, from a purist point of view it is probably wrong but practically this should work pretty much indefinitely unless there are other things involved in the design people have added. I need the hold up to be a day or two and it will go to sleep if on battery so this is probably a great approach for that Rui. Thanks for posting, you do a great job for us amateurs and it is truly appreciated!

[Reply](#)**Niall Parker**

May 3, 2019 at 5:28 am

Not sure about the ESP32 but the ESP8266 will already have a passive voltage divider to take 3v3 down to 1v range. Putting another divider in front of that won't work unless you add a voltage follower to reduce the output impedance. Easier is just to use a series resistor added to the one built in, or replace the one on the board.

[Reply](#)

Rui Santos

May 5, 2019 at 10:11 am

Hi Niall.

If you're using an ESP8266 NodeMCU kit (as we're using in the example) it has a built-in voltage divider. So, the input range is 0 to 3.3V.

If you're using an ESP8266 bare chip, the input range of the A0 is 0 to 1V.

Regards,

Rui

[Reply](#)

Niall Parker

May 6, 2019 at 4:36 am

Understood about the voltage ranges, not sure if you understood my point about cascading passive voltage dividers. Try simulating and you see it will read low.

[Reply](#)

Barbudor

May 6, 2019 at 8:11 am

Indeed.

As there is already a divider on the board, we don't need a full divider but only an additional serie resistor.

a full range of 6.2V. Probably too much 😊

[Reply](#)

barbudor

May 3, 2019 at 10:12 am

Hi

Most battery charger modules come with a resistor to set the charging current to either 500mA or 1A. This is much more than what a typical small solar panel can provide.

If you get a small solar panel with 5V 1.5W, you will have at most 300mA.

The resistor should be changed to adapt the charging current.

See TP4056 datasheet for more details.

Very nice website by the way. I love it 😊

Best regards

[Reply](#)

Noob Hobbyist

May 28, 2020 at 11:49 pm

Thank you for that. I was wondering how a changing wattage can affect the circuit!

[Reply](#)

May 4, 2019 at 3:23 pm

Hi,

I have built a similar circuit and I suggest to put a diode (1N4004 or equivalent) from the solar panel to the TP4056 board, to avoid reverse current.

Vittorio

[Reply](#)

Rui Santos

May 5, 2019 at 10:09 am

Thanks for your input!

[Reply](#)

Zeni

August 8, 2019 at 1:52 pm

Thanks a lot for such a useful tutorial.

I request to give an exclusive tutorial for powering ESP32 board using 3V3 and Vin (V5) and ESP32 WROOM module using battery.

Thanks again.

Rui Santos

August 8, 2019 at 2:00 pm

Instead of using the solar panel, you can use the exact same circuit with the battery

[Reply](#)

Don Johnson

October 19, 2019 at 8:38 pm

Thanks a lot for the write-up! I built exactly this circuit, but the MCP1700-3302E isn't cutting it for me. When waking from deep sleep on an ESP-WROOM-32 and turning the WiFi on, it seems the 250mA limit on the regulator is insufficient. It works some times, but for me it does not reliably wake up and connect successfully.

While I don't know enough to know what's actually happening, it seems like maybe there's a current spike up to ~350-400mA needed to keep things running smoothly. If anyone is trying this circuit out and things aren't working, I'd recommend replacing the MCP1700-3302E with something with slightly higher current output (I think 500mA would be ample).

[Reply](#)

November 14, 2019 at 7:57 pm

Hi, in this article you use a 100uF capacitor with the voltage regulator while in your book Home Automation ESP8266 (4th edition) you use a 1000uF capacitor. Any comments?

[Reply](#)

ed

January 19, 2020 at 9:58 am

The MCP1700 is with its 250mA at the brink of what the ESP8266 can demand from time to time.

Same goes for the popular HT7333.

May I suggest the 500mA HT7833

[Reply](#)

GUERRA

January 31, 2020 at 9:41 am

Merci pour vos tutoriels, très très biens!

J'ai un problème avec cette alimentation. A vide j'ai bien les 3.3 V en sortie, mais dès que je branche un TTGO-T1 LoRa, elle s'écroule à 1.35 V (la pile est à 4.10 V)).

[Reply](#)

barbudor

February 10, 2020 at 10:19 pm

Bonjour Guerra. Je passais par là... Peu probable que les auteurs te répondent en français...

Comment as tu branché le TTGO ?

Il faut bien brancher sur le 3V3, pas sur le 5V

Cf le brochage :

<https://ae01.alicdn.com/kf/HTB1FjoewY1YBuNjSszeq6ybIFXa0.jpg>

[Reply](#)

Malcolm

March 1, 2020 at 4:28 pm

will these panels work?

ALLPOWERS Solar Panel 2.5W 5V/500mAh Mini Encapsulated Solar Cell

[Reply](#)

Sara Santos

March 2, 2020 at 10:55 am

Yes, I think those should work.

Regards,

Sara

[Reply](#)

Matt

April 30, 2020 at 8:26 am

Two 5V 1W Solar Panels seem to be not enough to charge my setup with the esp32.

My location is southern germany.

Thinking about adding a third solar panel.

Any advice?

[Reply](#)

Barbudor

April 30, 2020 at 8:45 am

Hi Matt

I'm using a single panel 6V 6W which can provide a charge current of 90mA in full sun (Paris FR suburbs) to charge a 1200mA LiPo.

My ESP using DeepSleep takes a measure every 5 minutes. When disconnecting the panel, battery loose around 2.5mV per hour so I have around 15 days of full autonomy.

I am now waiting for the battery to reach 3.1V before reconnecting the

Happy to give more details on request.
Best regards

[Reply](#)

Matt

April 30, 2020 at 8:56 am

Thanks for the info.
I will look into my setup then.
Two 5v 1.2W panels don't manage to charge my battery properly.
Only with direct sunlight it seems to charge.
Had a cloudy sky yesterday and my sensor which takes hourly readings
and uses deepsleep in between failed tonight at 11 pm...
Might be something wrong with the panels

[Reply](#)

Barbudor

April 30, 2020 at 11:12 am

Sorry I wrote 6W but meant 0.6W => 100mA.
You may want to monitor the status of the charging and charged
indicators from the Tp4056. I removed the leds and resistors and
connected the pins to GPIO with internal pull-up so I can include in my
reports battery voltage and charging status.
The sensor doesn't receive full sun before 2pm but on a bright day
charging can start 1 hour before. I can have generally 2 to 3 hours of
charging a day so probably equivalent to 200 (15-20% of battery

Have you check that your system is actually using very low power in deepsleep? Mine is around 40uA

[Reply](#)

Matt

May 1, 2020 at 3:36 pm

My TP4056 seems to be charging continuously once it is on solar power. So even with the cloudy sky it is charging according to the LED. I have not checked the sleep state, right now it is not possible anymore without reworking a few solder joints ^^. Might do that if adding a third solar panel will not work. Might actually need a multimeter which is digital. With my regular analog one i won't be able to measure the deepsleep power i am afraid

Matt

May 8, 2020 at 6:28 am

So i just replaced the two 5V 1W chinese panels with three german 6V 1W panels and everything works like a charm. Charges the battery to the maximum during the day, loosing roughly 6-7% over night, charging to the max again during the day.

The reason why i want for a third panel is that i mounted it in a 15 degrees angle, that should provide better conversion rates during winter.

Barbudor

May 8, 2020 at 6:56 am

Ah “Deutsche Qualität” is the thing! 😂

6V is better because the TP4056 need at least 4.6V at input which is available only in full sun with the 5V panels. So you have probably more hours of charging by this change.

Organising the panels to cover larger angle is a great idea too 👍 I should do something like this too.

Matt

May 11, 2020 at 8:11 am

^^ Now my sensor seems to stop working for a few hours at a time for no apparent reason. Last reading yesterday 11pm, next 8 am today, then skipped the next reading again.

Not sure what is going on.

I am suspecting that it might not find the bme from time to time and is then stuck in the

```
if (!status) {
```

```
Serial.println("Could not find a valid BME280 sensor, check wiring or  
change I2C address!");
```

```
while (1);
```

```
}
```

check. But that is just a wild guess....

Would not explain why it suddenly starts working again after sometimes hours of not posting anything.

Matt

July 5, 2020 at 10:03 pm

Quick update:

After making my own pcb for the solar /battery power supply (including battery charge monitoring) in accordance with the tutorial and modifying the weather station pcb from randomnerds (changed the ldr gpio so it wont have issues with the wifi library and added some more gpio breakouts) it works flawlessly. I assume that i had a bad solder joint on one of the cables causing the station to sometimes get a reading and sometimes not.

Battery charge is now at a constant 100%. Waking up once per hour to get a reading during night times does not even drain 1% till the sun goes up again. Happy with the result and using the data with grafana.

David

May 31, 2020 at 10:34 pm

Thank you for this project! Here are some things that I have learned:

The NodeMCU 32-S boards (as well as ESP32 chips) ADC pins do not have a built-in divider, and inputs can be up to 3.3V. The NodeMCU 8266 does have a built-in divider on A0 only (the only ADC).

The ESP32 ADC is not anywhere close to linear at the top of its range. So a

<content/uploads/2019/05/ADC-non-linear-ESP32.png> A voltage divider which puts the ADC value near 2 volts is better since the ADC is more or less linear in this range. This also makes the divider simpler, since you want to half the input voltage of 4.2 to 2.1 R1 and R2 can be equal.

Larger R1 and R2 values mean less current flow, so the battery will last longer. I chose 220kOhm resistors. The constant current flow through that divider is 10 only microamps. I am not sure what the downside is to this choice, perhaps a slower response time since the divider will be much slower to change, but with a solar/battery ESP32 in sleep mode for most of its life, I don't expect the voltage to change very quickly.

[Reply](#)

Sara Santos

June 1, 2020 at 2:23 pm

Thanks for sharing this 😊

[Reply](#)

Dieko

June 10, 2020 at 4:06 pm

Dear Sara and friends,

First: A big “thank you” for your effort to share your projects and findings! I hope all is well in these strange “Corona times”...

camere module. I already go the basics working, I get a stream in my browser etc.

Now I want to extend this so it works autonomous/remotely on battery + solar charging. This tutorial seems to have all I need BUT...

When I have the ESP32 Cam system running on 3.3V I see stripes in the picture/stream. Somewhere I read that this is related to the 3.3V operation and so I tested the setup powered with 5V. I can confirm that I don't have problems or stripes when running on 5V.

Or is that issue related to the camera (ID= 0x26)

The question is: Which hardware components do I need (or change) to run the setup on 5V ?

Some more questions...

Suppose I want a really simple setp which only starts if there is enough irradiation. How much power (mA) does the ESP32+Cam need for running. Can I connect it directly to a 5V solar panel (or maybe it needs 2 panels in parallel to get enough power). How much overvoltage can it take, can I connect it to a 6V panel or is that too much already? Or any higher voltage panel with voltage stepdown?

The market price of "regular" solar panels (200-300W, 34V, 7-9A) is only about 50 cent per watt, the "small" panels cost a lot more per watt.

It seems the Wifi uses some power, so is there something like "wifi sleep" which wakes up when wifi remote access is detected? (not wake-on-lan but wake-on-wifi?)

Bonus: Since bees only fly at daytime I want the system to go deepsleep over night

Should this be handled via the internal clock or with a light sensor? Or bee-motion-detection ??

another machine?

(I have a raspberry running which checks some IP cameras with ‘motion’)

Anyway, thank you for sharing and thank you for your support!

BTW: I cannot find a paypal donate button, where is it 😊

Or how can we support your site?

Best regards,

Dieko

[Reply](#)

Sara Santos

June 16, 2020 at 8:57 pm

Hi Dieko.

Thanks for sharing your project.

In this example, the rechargeable batteries output 4.2V when fully charged.

You can try powering the ESP32-CAM directly from that battery and see if it works as expected. I haven't experimented it, so I'm not sure.

Don't connect the solar panels directly to the ESP32.

If you want to power the ESP32-CAM using 5V, you can search how to power an Arduino (that works with 5V) using solar panels.

To save battery, it is better to put the ESP32-CAM in deep sleep at night.

It is also a good idea to integrate it with your other IP cameras using node-red.

There are many options, you have to experiment and see which scenario works better in your case.

You can also see our about page for another ways to support our work:

<https://randomnerdtutorials.com/about/>

Regards,
Sara

[Reply](#)

Wolfgang

June 20, 2020 at 9:22 pm

Hi,

i found the the following Information for the TP4056 on a Product Page.

” It is better that the charging current is 37% of the battery capacity. For example, the charging current should be 400 for the 1000mAH battery.”

Adjust the output current from 100 to 1000ma by simply changing the fixed resistor on the circuit board.

R1 (KOhm) I(mA)

30 50

20 70

10 130

5 250

4 300

3 400

2 580

1.66 690

1.5 780

1.33 900

1.2 1000

I've ordered all the Parts and see what happens in Reality 😊

[Reply](#)

Nicolas

July 6, 2020 at 11:11 pm

Hi, I'm following the tutorial a while ago. And I have had some complications, but I have already solved them. Some of these complications were:

In my region where I live, the regulator is not found, so I replaced it with an LD1086 regulator and I get the 3.3V. But here comes my doubt, because I connect it to the Vin pin but my ESP8266 MCU does not turn on. So my question is, should I connect it 3.3V to the Vin pin or the 3.3V pin.

Regards,

Nicolas

[Reply](#)

Sara Santos

July 13, 2020 at 10:14 am

Hi.

If you get 3.3V, you should connect it to the 3.3V pin.

Regards,

Sara

[Reply](#)

Carl R

June 25, 2021 at 1:13 pm

Sara, Rui, could you please clarify the 3.3v pin on the ESP32 for me. I was under the impression that it was an OUTPUT pin from the built in 5+V to 3.3V regulator. But 3.3v can also be applied there to power the board ? Also one other question, why can't a simple zener diode (3.3v) be used instead of the MCP1700 to regulate the output voltage ? Very good tutorial !

[Reply](#)

Hairyloon

July 8, 2020 at 6:23 pm

What is to stop this from discharging the battery too far?
Li-Ion batteries really don't like to go too low.

[Reply](#)

Daijoubu

July 8, 2020 at 6:38 pm

The TP4056 module depicted in this tutorial have a DW01 protection IC, voltages are rather low but will disconnect the battery when it drops below 2.4 ± 0.1 V, the FS312F-G is better suited and pin compatible at $2.9V \pm 0.08V$

Matt

July 8, 2020 at 8:08 pm

Nothing will stop a deep discharge. At one point there won't be enough energy to power your esp. That is well before the battery is anywhere near a critical low level. If you size and place your solar cells correctly you should always get a charge, even with cloudy winter skies.

Using deep sleep you could go days without charging.

[Reply](#)**Douglas Mauro**

December 23, 2020 at 9:05 pm

I'm working this problem right now. I'm trying to work in a voltage supervisor so when volt gets too low, I can drop all power and let the solar and batteries come back to charge.

I was hoping that grounding the EN pin of the esp32 would stop 3.3v from coming out other side. It looks like even when a chip is disabled it'll still pass through the volt regs, etc.

[Reply](#)

August 18, 2020 at 6:23 am

Does anyone have a solution to my problem? When I follow the instructions and try to use wifi on my esp32 boards it reboots, I was using the HT7333 LDO but I've read that the 250mA could be a bit to low so I switched to a LM1117T (800mA) now but I still have the same issues.

It really seems to be something about the wifi + current issue? Also tried different capacitors without look.

[Reply](#)

Pascal

September 4, 2020 at 6:34 am

Hello

Interesting article, very interesting And inspiring site.

Maybe off topic but found discussion around battery powered esp32 in the below article:

radioshuttle.de/en/media-en/tech-infos-en/battery-powered-esp32/

[Reply](#)

Sara Santos

September 4, 2020 at 9:50 am

Regards,
Sara

[Reply](#)

Robert Bock

September 17, 2020 at 5:38 pm

Great article. I have completed the original ESP32 Camera book. I have started the ESP32 book now.

How do I interface this with the ESP32 Cam AI Thinker. I want to make an autonomous surveillance camera.

[Reply](#)

Sara Santos

September 19, 2020 at 11:32 am

Hi.

This can be applied to the ESP32.

Power it through the 3.3V pin and it should work.

Regards,
Sara

[Reply](#)

September 20, 2020 at 2:41 pm

Esp32Cam has AMS1117 regulator at 5V input. It has maximum input voltage limit 15V.

With high voltages power dissipation is the key. You cannot make too much heat to regulator.

If you take Chinese 6V 100x100mm solar panel it has appr. 7,8V free running voltage and appr. 200mA short circuit current.

It can be connected with only serial diode (1N4007) directly to small 6V 4Ah closed lead acid battery. It will not overcharge that without regulator.

Then you can connect Esp32Cam directly to 6V battery (or use also serial diode 1N4007 as polarity safe device).

This is what I am going to try next with my time lapse camera ver.2

[Reply](#)

dierickx

September 23, 2020 at 7:56 am

Hi

This is a great tutorial.

I noticed that it is possible to use 2 solar panels (in serie) with only one TP4056.

My question, is it possible to use as well 2 solar panels , one TP4056 and 2 x batteries 18650?

thx

Matt

September 23, 2020 at 9:42 am

Two batteries or more could be used but you would have to balance them first (make sure both have the same charge)

I am running on three solar panels (one at an angle which works better during winter or overcast), one tp module and one battery. With deep sleep my power never drops below 99%.

I take measurements every 30 minutes. So one battery should give you plenty of power

[Reply](#)**Bob**

October 1, 2020 at 8:43 pm

Hi Matt,

you mentioned using german 6V/1W solar cells. I have trouble finding those, Would you be so kind and share your supplier?

Thx!

[Reply](#)**Matt**

November 3, 2020 at 10:28 am

[Reply](#)

Manuel Seixas

November 19, 2020 at 1:36 pm

What about the same type to control for a 12,8v battery?

[Reply](#)

Marcel

February 6, 2021 at 10:05 am

Hello all,

at first, thanks for this really great tutorial!

Currently, I'm working on the circuit to measure the battery level over A0 with my NodeMCU v3. But there is one point i don't understand.

Why is the red line (3.3V) connected from the ESP8266 to the breadboard? It seems to be a dead line ending up in the breadboard because there is no other connection. One thing, I could imagine: The red line coming from battery (4.2V) to the breadboard is not only connected to the resistor (27k), but also connected to the 2nd row of the breadboard. Hope, you got my issue. I'm fairly new to all the circuit stuff so please be polite. 😊

[Reply](#)**Hairyloon**

February 6, 2021 at 12:00 pm

Good question.

My guess is that it is just best practice when wiring up to a breadboard to connect the power to the power lines.

People using breadboards are likely in the habit of adding on extra circuits and expect there to be power on the power lines and if you then forget to check the obvious when it doesn't work, it can get confusing.

[Reply](#)**Marcel**

February 6, 2021 at 12:11 pm

Thanks for response.

So it is not necessary to connect the power line from ESP8266 to the breadboard.

I researched a lot by understanding voltage dividers and this power line made me crazy. 😊

[Reply](#)

February 6, 2021 at 10:32 pm

It doesn't appear necessary to have a breadboard at all to be honest.
It seems a little excessive just to hold two resistors and a couple of wires.

[Reply](#)

Marcel

February 14, 2021 at 8:54 am

Sure, but I use the breadboard for initial tests whether the circuit is working. If it's all fine I replace all that.

For your information: Everything is working and I'm doing some tests with all sensors and stuff. Output voltage of TP4056 in indirect sunlight is at 4,4V currently. Hope, it compensates all the consumers even in this bad spot without direct sunlight (real spot is in sunlight).

Marcel

February 14, 2021 at 9:04 am

Edit: *Output voltage of SOLAR PANELS is 4,4V in indirect sunlight.

February 11, 2021 at 10:35 pm

What about TP4056, I heard that this module cant charge and discharge (load) at same time, because its not safe for battery 18650, how to fix this problem?

[Reply](#)

Patto

February 19, 2021 at 9:23 pm

Hi

Awesome site and great tutorials.

I have a NodeMCU-32S and am not sure if I need to do anything code wise with the 3.3pin. Isn't it a V out pin? Or can I just connect to it and it reads voltage in?

Cheers

[Reply](#)

dentalan

March 21, 2021 at 7:12 pm

Hello, very nice project. I have almost completed my version and I intend to put ii my yard.

hot. So now I use 3 TP4056.

[Reply](#)

Ruben Perez

April 12, 2021 at 3:57 pm

Hello to all, I would love to share my experience with this project (Thanks for sharing it!).

I built the project using a NodeMCU8266 with two temperature sensors (DS18B20), using EspHome to program and deep sleep to save energy, awaking every 5 minutes for the temperature reading and sending results using MQTT to OpenHAB. This was working for weeks without problem, but coinciding with several days of very cloudy days (I am not sure if there was related, but was this days), the device stop working.

I have disassembled all, and Esp8366 was fried, it doesn't responds using USB cable to computer or anything... I am trying to find a possible cause and I am trying to switch NodeMCU8266 by a NodeMCU Esp32. All seems to be ok, battery, charger, regulator... At output of regulator I am now reading 3.2 V... but at NodeMCU I am reading 3.1 (3.08...), and the device doesn't power up, led blinks all time without proper boot.

I am not sure about what could have happened and I don't know also what can I do to make Esp32 working with this (on some coment before, I have read that regulator could not provide enough power to power wifi on Esp32 (plus temperature sensors...)).

[Reply](#)**ivan**

April 15, 2021 at 7:56 am

Hi Ruben sorry for out of topic i also do deep sleep mode project i used DHT 22 and esp and i used thingsboard as to see the result ...its working but my problem is after the deep sleep for 1-2 hrs its just keep reconnecting to wifi i already left to for 1 whole day still same so would dont mind to share ?

[Reply](#)**Ruben Perez**

April 15, 2021 at 8:14 am

Hi Ivan, do you want my configuration of the device?
I am using EspHome to program it, is ok? For me is not problem, it works perfectly awaking from deep sleep every 5 minutes (also test 10 minutes).

Ruben

[Reply](#)**Ruben Perez**

April 15, 2021 at 8:21 am

https://esphome.io/components/deep_sleep.html

[Reply](#)

[ivan](#)

April 16, 2021 at 3:52 am

may i know how u used the esphome code in arduinio ide please..
cause i used <https://randomnerdtutorials.com/esp32-deep-sleep-arduino-ide-wake-up-sources/> <https://randomnerdtutorials.com/esp32-deep-sleep-arduino-ide-wake-up-sources/> this method

Ruben Perez

April 16, 2021 at 7:10 am

Hi Ivan,

EspHome is not based on code, is based on configuration files.

As I only have to add to my configuration file:

Example configuration entry

deep_sleep:

run_duration: 30s

sleep_duration: 10min

for a 30 seconds run after 10 minutes of deepsleep. No more is needed, is explained on the link that I put before. You can add more items for example, if you want to update via OTA (as in deepsleep can't receive

Ruben

[Reply](#)

ivan

April 14, 2021 at 2:46 am

hi sara im using this board (banggood.com/LILYGO-TTGO-T-Energy-ESP32-8MByte-PSRAM-WiFi-bluetooth-Module-18650-Battery-ESP32-WROVER-IB-Development-Board-p-1427125.html?cur_warehouse=CN) i want to monitor the battery like until how many it can supply ... i just try the adc i put in 5v pin and connect to monitor and i used 2 10k as voltage divider and it shown once i put in 3.3Vpin the board burn and no port shown in the arduino ide can i know how to monitor the battery

[Reply](#)

Ruben Perez

April 15, 2021 at 8:18 am

Ivan, are you using this device with the sara board? are you charging their battery via solar panel? I am interested on this, as I have tested that my NodeMCU Esp32 doesn't works powered by 3.3V, probably, as I read, due a start up power requeriments (the device never boots properly, led flash quickly all time), no only with sara project, also powering by a 3.3V little power source. NodeMCU Esp8266 works without problem at 3.3V.

[Reply](#)**ivan**

April 16, 2021 at 3:55 am

i guess some esp 32 board dont support the 3.3V cause when i test the adc in 5V pin the reading shown but once i used 3.3V the board loss the port number and it still got light but cant find the port on the arduino ide ...

[Reply](#)**Rogério**

April 14, 2021 at 11:31 pm

Hello congratulations on the article.

I'm assembling a project and I don't need the solar charger, so I took it out of my project, but I'm using the 4.2v battery and created the voltage reducing circuit and the circuit to read the battery, when I turn on the battery I do the measurement and it is coming out of the 3.25v reducing circuit but when I turn on esp32 it turns on the esp led light but does not transmit anything.
Does anyone know what may be happening?

[Reply](#)**Rogério**

April 19, 2021 at 11:42 am

[Reply](#)**Ruben Perez**

April 19, 2021 at 12:22 pm

Probably have the same problem as me (and other people). Not all NodeMCU circuits can boot at 3.3V or, maybe not all sources could provide enough power to start up. It seems that WiFi needs 400mA to power up, and not all circuits and batteries can provide this. With this project, I can boot NodeMCU Esp8266, but not Esp32.

Ruben

[Reply](#)**Rogério**

April 20, 2021 at 12:02 am

Thanks Rubens,

The article demonstrates how it works on both Esp32 and Esp8266, I found it very strange because many said it worked. But thank you for your return.

[Reply](#)

June 7, 2021 at 3:09 am

Hi I would like to use CN3791 Charger Module on my project.

Here are the components that I would like to use with CN3791:

1. 12V 2W solar panel
2. ESP32 CAM
3. 6V DC 4.5AH Sealed Lead Acid Battery

Can I combine these components for my project? Thank you!

[Reply](#)

Neil Joshua Limiac Layson

June 24, 2021 at 8:51 am

Help

[Reply](#)

David

August 15, 2021 at 6:02 pm

No, the CN3791 is designed for charging Lithium Ion batteries with a charging voltage of 4.2V. A 6V lead acid battery will need at least 6.5V in order to charge.

Mauro

June 25, 2021 at 12:52 pm

Aren't you using TP4056 for 'load-sharing' in this configuration, for which it is not suitable? The TP4056 may register the battery as 'not-full' when you are drawing current through it, which can cause the battery to over-charge (and we all know what that may result in). Why not putting a bypass-diode from 5v solar panel to V=Out (or a diode from Vin to Vout to prevent load-sharing) so that if there is enough power from the solar, that it used instead of drawing through the TP4056?

[Reply](#)**Willy Ray**

July 6, 2021 at 12:26 am

Question: If I have a pair of panels @6v and sufficient amperage and I *only* care about my nodes running when the sun is on them, can't I just run the panels unregulated straight to the VIN pin?

[Reply](#)**Matthias Wurst**

July 6, 2021 at 11:59 am

0.0v. So you can power it directly via USB for a fraction of a second before you burn it out. 😊

You will need to transform the power coming from the solar panels down to 3.3 volt.

Without a battery you probably will have issues on cloudy days.

[Reply](#)

Willy Ray

July 6, 2021 at 3:44 pm

Ok, but don't most of these development boards have power regulation on them? I'm looking at a board with a V5 pin on it. USB is 5 volts, and *that* doesn't smoke my device.

What about this?

techexplorations.com/guides/esp32/begin/power/

[Reply](#)

Matthias Wurst

July 6, 2021 at 8:03 pm

The pins usually don't go via the power regulator. So a usb port can be 5v and it is transformed to the needed 3.3v, the pins usually are not. In the end it depends on your board.

Even with a 5V pin I would not risk connecting 6V to it.

It might work but why risk damaging the board when a converter is

expensive

[Reply](#)

Schalk

August 14, 2021 at 11:41 am

Hi.

Thank you for your excellent articles and other materials. I own several of your books and highly recommend them.

I question: All over the internet, including in this article, people use TWO solar panels in series. I understand about the capacity. My question: why not just use a higher capacity panel?

Using my currency (so ignore the value, look at the ratios):

1.2W panel: R42, so two = 2.4W = R84

2.5W panel: R76

Maybe on other sites / shops the money works out differently?

Most important question: is there some electronics reason to have two in parallel rather than one of the same capacity?

Kind regards

Schalk

[Reply](#)

Hi.

I don't think there is any problem with using one solar panel with double the capacity instead of two.

I think that at the time we wrote the tutorial, we used two because those models were more widely available.

Regards,
Sara

[Reply](#)

Peter

September 30, 2021 at 12:13 pm

Hi, thanks for the project. i'm trying to use a solar pannel (5V 14W) and a tp4056 to alimentate a esp01. I bought one battery "103454"3.7V. i want to use this battery when the sun goes down. when i connect the battery to the tp4056, i have only 1.8V to alimentate the esp01.

[Reply](#)

Sara Santos

September 30, 2021 at 1:52 pm

Hi.

1.8V is not enough.

Sara

[Reply](#)

Jo

October 20, 2021 at 2:09 pm

Can someone teach or guide me how to write the coding for monitoring the battery percentage and status? Thank you.

[Reply](#)

zygfryd

October 21, 2021 at 10:34 am

monitoring the battery is usually done by measuring the voltage on it. however, it is not linear function meaning: if you consider LiPo to go from 4.2V(fully charged) to 3.0V (cut off) 3.6V is NOT 50% of the health of the battery.

So it will be always estimation, it depends on battery type but even on the actual battery you have.

But once you are ok with measuring and reporting voltage, then you simply add resistor voltage divider and connect the output of that to ADC on your ESP32

[Reply](#)

October 27, 2021 at 10:40 pm

When I connect a ESP8266, D1 mini and nodeMCU dev, to this circuit the solar panels short out. The ESPs work on battery power, as soon as I plug the panels in everything goes dead. Even using 1 panel, so it is not in that wiring.

I am boosting the TP4056 circuit to 5.1V into Vin, but plugging the solar panel directly to the board does the same thing.

A D1 battery shield seems to have the same problem.

Any ideas what is going on?

[Reply](#)

zygfryd

October 28, 2021 at 10:42 am

my solar panels are small, so to avoid “shorting” them I replaced R3 on TP4056 to 4k7 and it works this way: when ESP works and it needs $\leq 300\text{mA}$ the power comes fully from battery. When ESP goes to sleep, total current from solar panel goes to charging – yes, slow charging, but I don’t mind. My ESP sleeps for 5min then it works for 8s – no harm on charging and usually my battery is full by midday.

[Reply](#)

DON

November 1, 2021 at 12:41 am

works fine for a D1 mini that did not work with the smaller, 130 mA panels in parallel.

[Reply](#)

DON

October 27, 2021 at 11:13 pm

I know from experience (not “I read”) that a 100K resistor is all that is needed for a 4.2 volt Liion battery on a nodeMCU dev board or a D1 mini.

That will bring the internal voltage to 1 volt = 1024, then you do the math using 4.2 v not 3.7 volts.

I.E.: $(\text{analogRead(A0)/1024}) * 4.2 = \text{battery voltage.}$

$(1024/1024) * 4.2 = 4.2.$

As previously stated NON-linear, but close enough; add a fudge factor if you need to.

Other voltage batteries will be different, the formulae are out there.

[Reply](#)

Daniel

October 29, 2021 at 3:00 pm

This project doesn't work, I bought a Mcp1700t-3302 and when I connect the energy on my esp32, my esp only blinking. While when I connect my esp

[Reply](#)**zygfryd**

October 29, 2021 at 4:35 pm

try adding 220uF

[Reply](#)**Luke**

February 3, 2022 at 11:00 am

Sorry zygfryd, I'm having the same problem of Daniel, the 220uF you suggested must be placed instead of the 100uF?

[Reply](#)**Greg**

December 28, 2021 at 11:52 am

hello I am a beginner I would like to use your advice to power a 5v esp32cam
is it enough to put a boost?
Thanks.

Wesley

February 18, 2022 at 5:29 pm

My battery is outputting more than 3.7V, and the MPC1700 output is 3.3v however when I attach to the 3.3v esp32 pin its voltage drops to around 2.8V and my program is not running.

[Reply](#)**Wesley**

March 4, 2022 at 8:26 pm

After putting a 10kuf capacitor and adding this to the code

```
#include "soc/soc.h"  
#include "soc/rtc_cntl_reg.h"  
  
void setup() {  
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); // disable detector  
}
```

It resolved the issue.

[Reply](#)

Hi Sara,

I'm using ESP32 Node MCU for this project. Everything works fine until the sun stops shining, the power voltage drops and the ESP32 Node MCU Red Power LED flickers OFF. When the sun does eventually return the voltage slowly increases until the ESP32 Node Red power LED flickers ON but the ESP fails to boot. Even when the battery is charged beyond 3.3v the unit remains in a comatose state until a manual restart is initiated using the EN button.

This is not great for units out in remote locations. Do you know how this system can recover successfully from a flat battery once the sun starts shining?

[Reply](#)

Leave a Comment

Name *

Email *

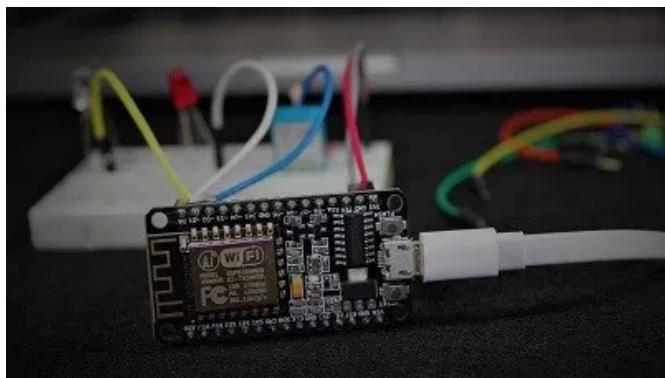
Notify me of follow-up comments by email.

Notify me of new posts by email.

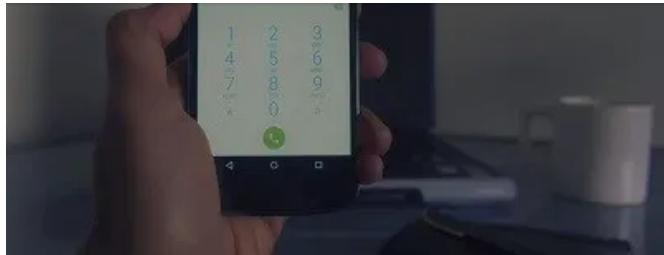
Post Comment



[Visit Maker Advisor – Tools and Gear for makers, hobbyists and DIYers »](#)



[Home Automation using ESP8266 eBook and video course »](#) Build IoT and home automation projects.



**Build a Home Automation System from
Scratch » With Raspberry Pi, ESP8266,
Arduino, and Node-RED.**

≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



≡ Menu



[About](#) [Support](#) [Terms and Conditions](#) [Privacy Policy](#) [Refunds](#) [Complaints' Book](#)

[MakerAdvisor.com](#) [Join the Lab](#)

Copyright © 2013-2022 · RandomNerdTutorials.com · All Rights Reserved

Exclusive Member of Mediavine Home