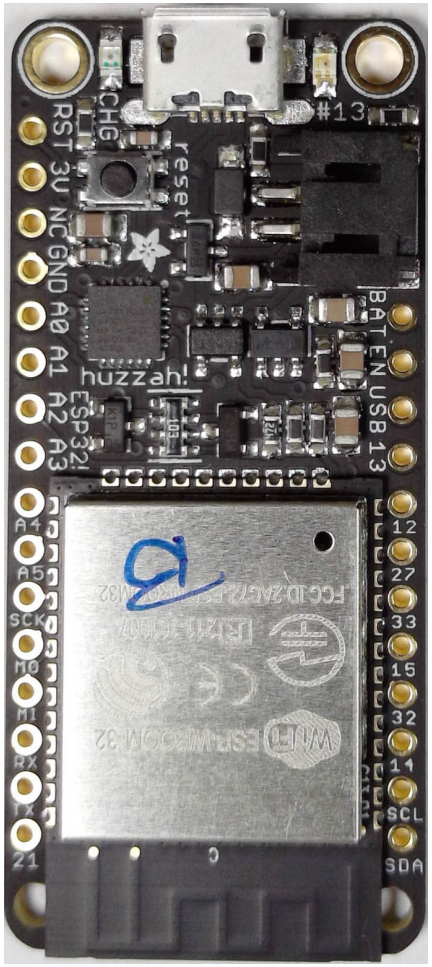
# Electronics for IoT

# Power Management

Bernhard E. Boser
University of California, Berkeley
boser@eecs.berkeley.edu

# Power ESP32 from Solar Cell



# Challenges

- Solar cell only deliver ~ 30mA
  - ~ 100mA in full sun
- ESP32 consumes up to 200mA
- No sun at night
- How connect solar cell to ESP32?
- ESP32 needs 3.3V
- Solar cell generates up to 5V

# Huzzah32 Power Supply

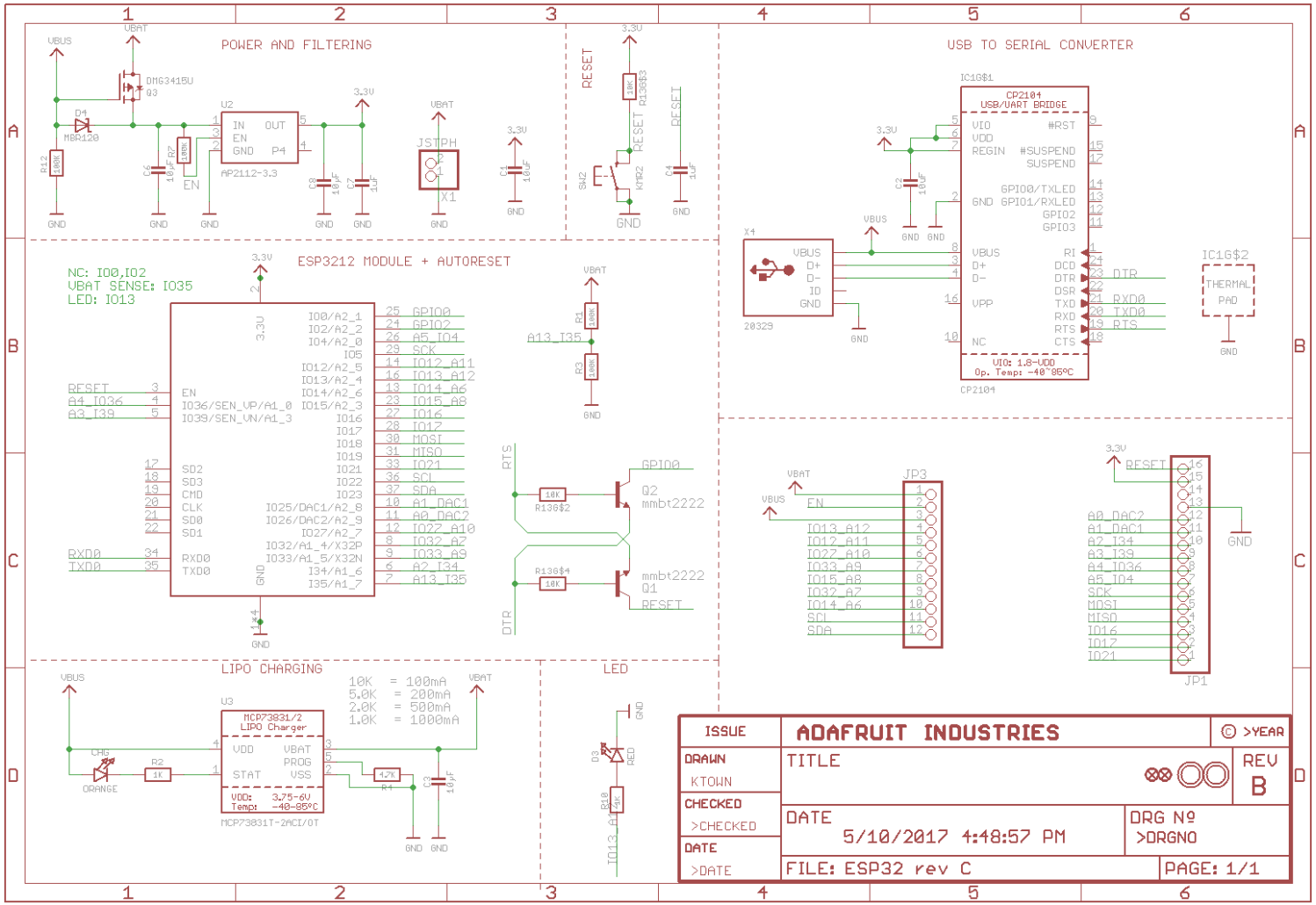
- USB is 5 Volts
- ESP32 needs 3.3 V
- How does the Huzzah32 generate 3.3 V from 5 V?

# Power Regulators

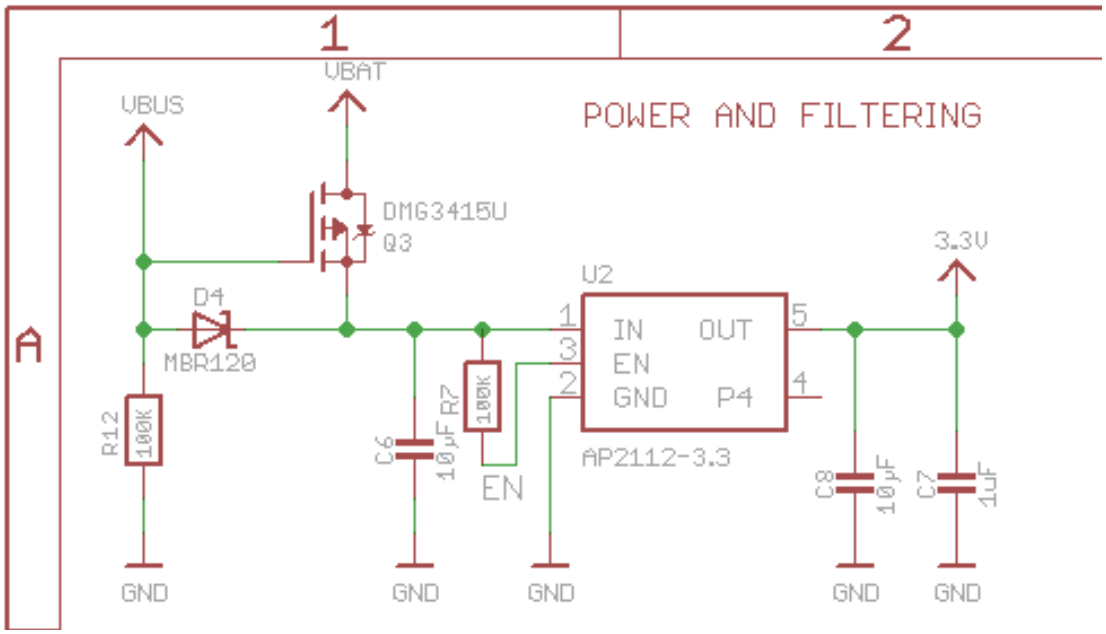
# Types of Regulators

# Power Regulator Efficiency

# Huzzah32 Power Regulator



# Huzzah32 Power Supply



# AP2112 600mA Regulator

DIODES INCORPORATED

Lead-free Green

**AP2112**

**600mA CMOS LDO REGULATOR WITH ENABLE**

## Description

The AP2112 is CMOS process low dropout linear regulator with enable function, the regulator delivers a guaranteed 600mA (min.) continuous load current.

The AP2112 is available with a fixed output voltage of 1.2V, 1.8V, 2.5V, 2.6V, or 3.3V. The LDO has an output accuracy of ±1.5% and a very fast loop response providing excellent performance for dealing with line and load transients. The AP2112 includes an auto discharge function which connects the output to ground via 60Ω of resistance when the device is disabled.

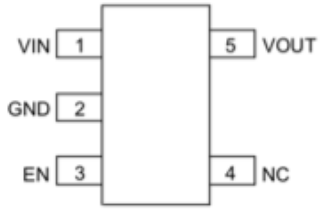
The regulator features low power consumption, and provides SOT25, SOT89-5, and SO-8 packages. Previously SOT-23-5, SOT-89-5 and SOIC-8 packages were respectively identified as SOT23-5, SOT89-5 and SO-8 but have been renamed to match the latest Diodes Incorporated's nomenclature.

## Features

- Output Voltage Accuracy: ±1.5%
- Output Current: 600mA (Min.)
- Foldback Short Current Protection: 50mA
- Enable Function to Turn ON/OFF $V_{OUT}$

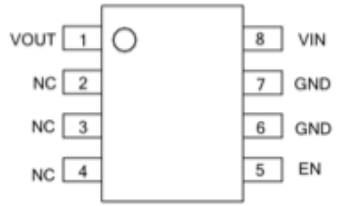
## Pin Assignments

(Top View)



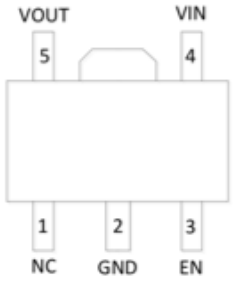

SOT25

(Top View)

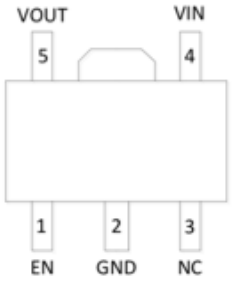



SO-8

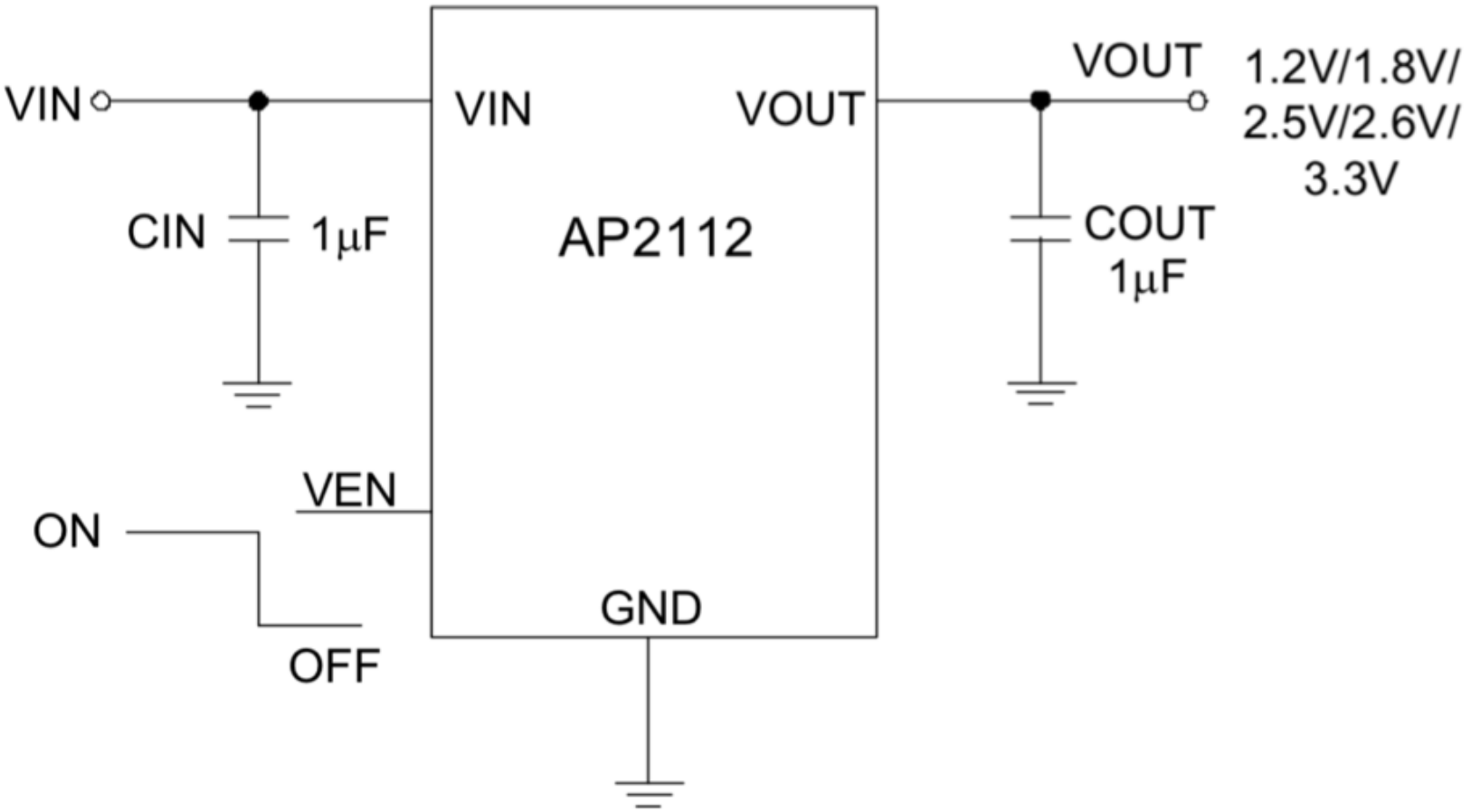
(Top View)




(Top View)



# AP2112


VIN
CIN
1μF
VIN
AP2112
VOUT
VOUT
1.2V/1.8V/
2.5V/2.6V/
3.3V
COUT
1μF
VEN
ON
OFF
GND

# AP2112

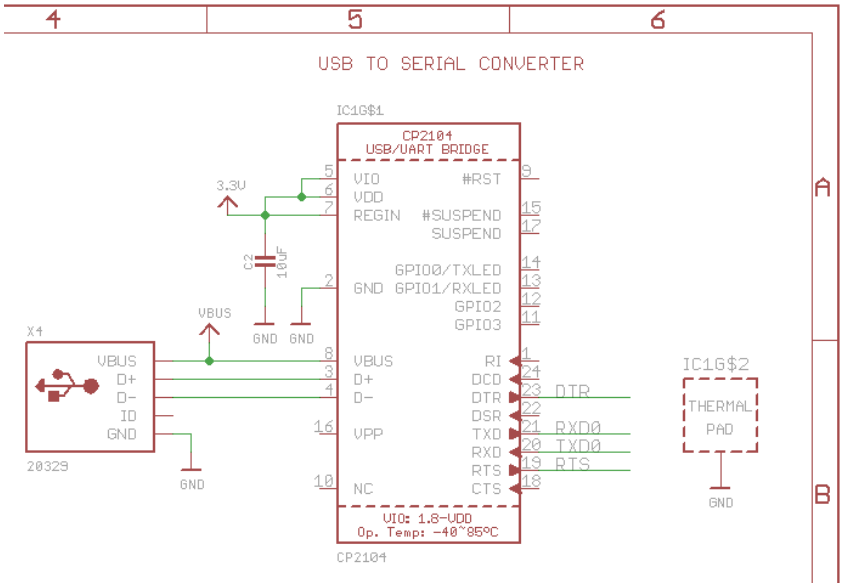
## Absolute Maximum Ratings (Note 5)

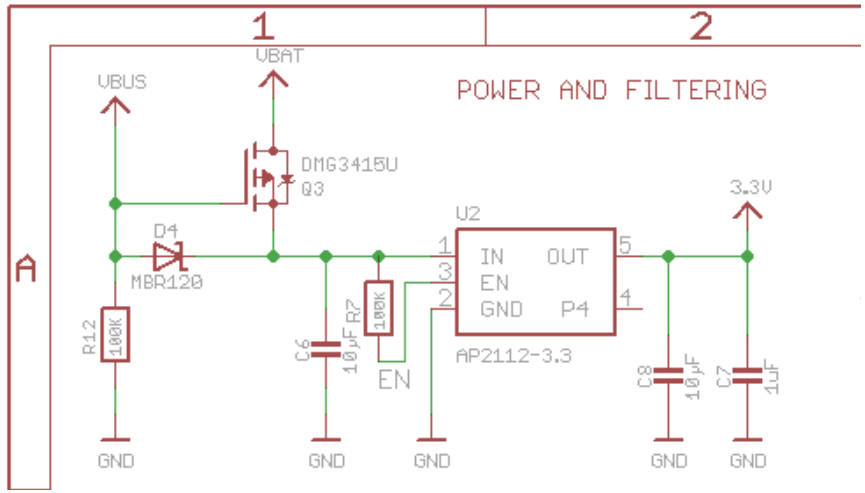
| Symbol | Parameter | Rating | Unit |
|---|---|---|---|
| $V_{CC}$ | Power Supply Voltage | 6.5 | V |
| $T_J$ | Operating Junction Temperature Range | +150 | °C |
| $T_{STG}$ | Storage Temperature Range | -65 to +150 | °C |
| $T_{LEAD}$ | Lead Temperature (Soldering, 10 Seconds) | +260 | °C |

## Recommended Operating Conditions

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{IN}$ | Supply Voltage | 2.5 | 6.0 | V |
| $T_A$ | Ambient Operation Temperature Range | -40 | +85 | °C |

# Power Huzzah32 from Solar Cell



# Huzzah32 Current

# Solar Power

# Running Huzzah32 from Solar Cell

# Solar Powered Weather Station

- Design plan:
  1) Reduce Huzzah32 power requirement
  2) Add battery for nights, cloudy days

- Let's address first (1), then (2) …

# Weather Station Requirements

# ESP32 Current Consumption

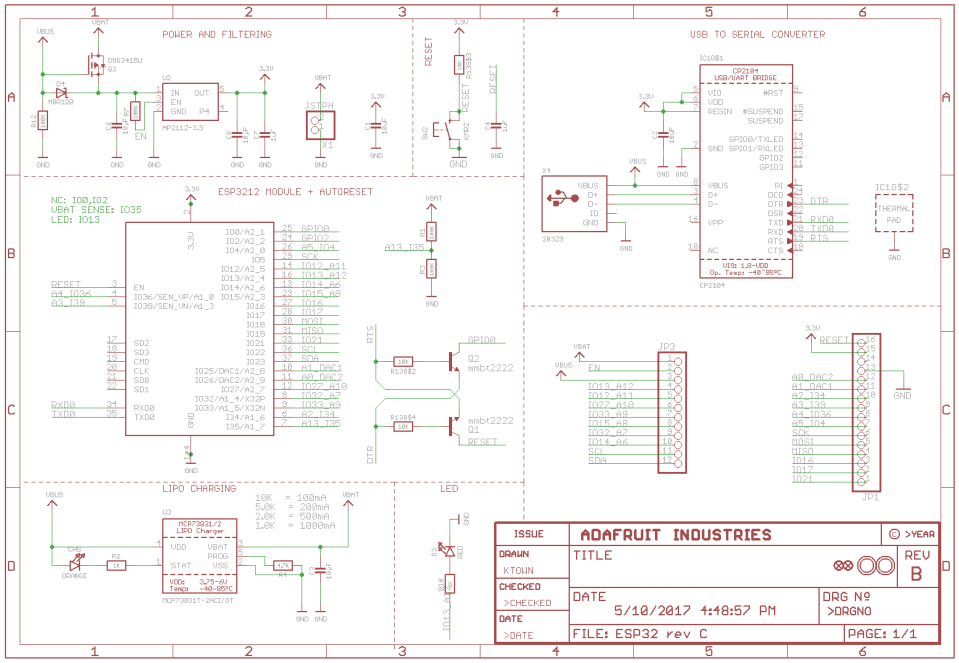
# ESP32 Deepsleep

# ESP32 Average Current - Example

# No so fast …

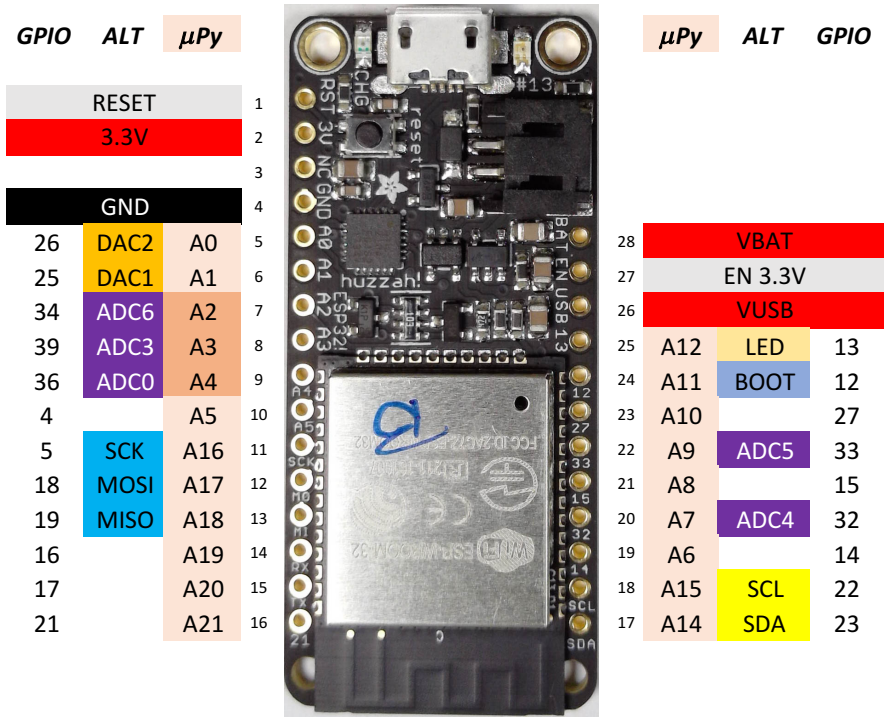
# Not so Fast

- Lots of stuff on Huzzah32 board
- Not just ESP32!
- How much current do the other circuits consume?
- How can we find out?
- Hmm, lots of datasheets to consult …



# Measure Huzzah32 Supply Current

- No need for USB
  - Do not power it!
- Connect supply to battery
- But how make the connection?



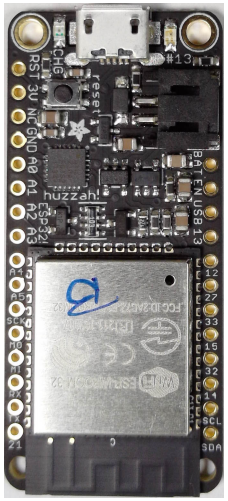
# Lab Supply


KEYSIGHT
E3631A
0-6V,5A/0-±25V,1A
Triple Output DC Power Supply
6.000V
0.016A
CV
+6V
ADJUST
FUNCTION
SELECT
+6V
+25V
−25V
Track
Display Limit
Recall
Store
Error
I/O Config
Output On/Off
Local
Voltage Current
Power
On
Off
6V
+
−
⊥
±25V
+
COM
−
± 240 VDC MAX TO ⊥

# Power from Lab Supply



| GPIO | ALT | µPy | | µPy | ALT | GPIO |
|---|---|---|---|---|---|---|
| | RESET | | 1 | | | |
| | 3.3V | | 2 | | | |
| | | | 3 | | | |
| | GND | | 4 | | | |
| 26 | DAC2 | A0 | 5 | | | |
| 25 | DAC1 | A1 | 6 | | | |
| 34 | ADC6 | A2 | 7 | | | |
| 39 | ADC3 | A3 | 8 | | | |
| 36 | ADC0 | A4 | 9 | | | |
| 4 | | A5 | 10 | | | |
| 5 | SCK | A16 | 11 | | | |
| 18 | MOSI | A17 | 12 | | | |
| 19 | MISO | A18 | 13 | | | |
| 16 | | A19 | 14 | | | |
| 17 | | A20 | 15 | | | |
| 21 | | A21 | 16 | | | |
| | | | | 28 | VBAT | |
| | | | | 27 | EN 3.3V | |
| | | | | 26 | VUSB | |
| | | | | 25 | A12 | LED 13 |
| | | | | 24 | A11 | BOOT 12 |
| | | | | 23 | A10 | 27 |
| | | | | 22 | A9 | ADC5 33 |
| | | | | 21 | A8 | 15 |
| | | | | 20 | A7 | ADC4 32 |
| | | | | 19 | A6 | 14 |
| | | | | 18 | A15 | SCL 22 |
| | | | | 17 | A14 | SDA 23 |



# Huzzah32 Current Consumption

- Just processor on 43 mA
- Processor & WiFi on 120 … 200 mA
- Deepsleep 420 μA

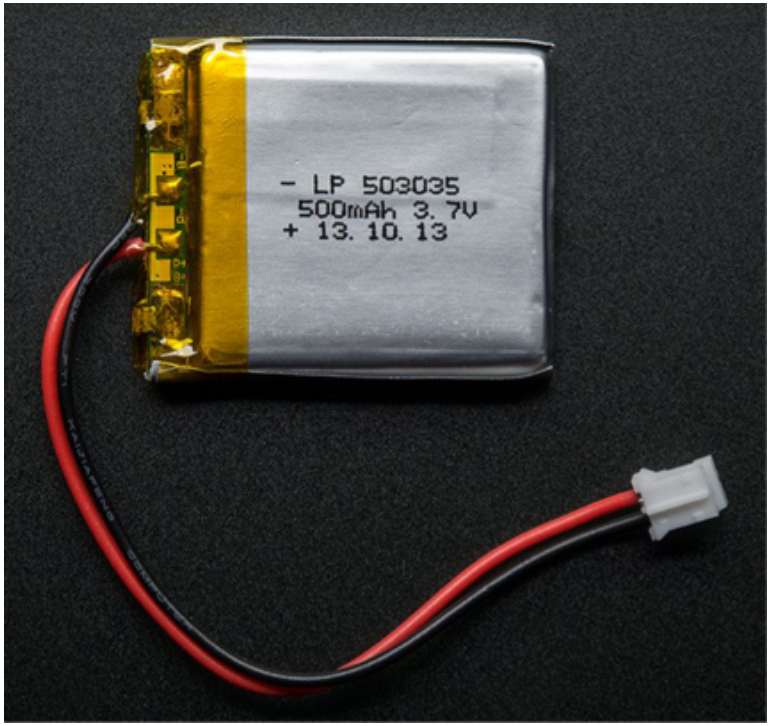
- Hmm, a bit more than 10 μA
- Are we still ok?
- What if not

# ESP32 Average Current - Example

**ESP32**

**Huzzah32**

# Battery

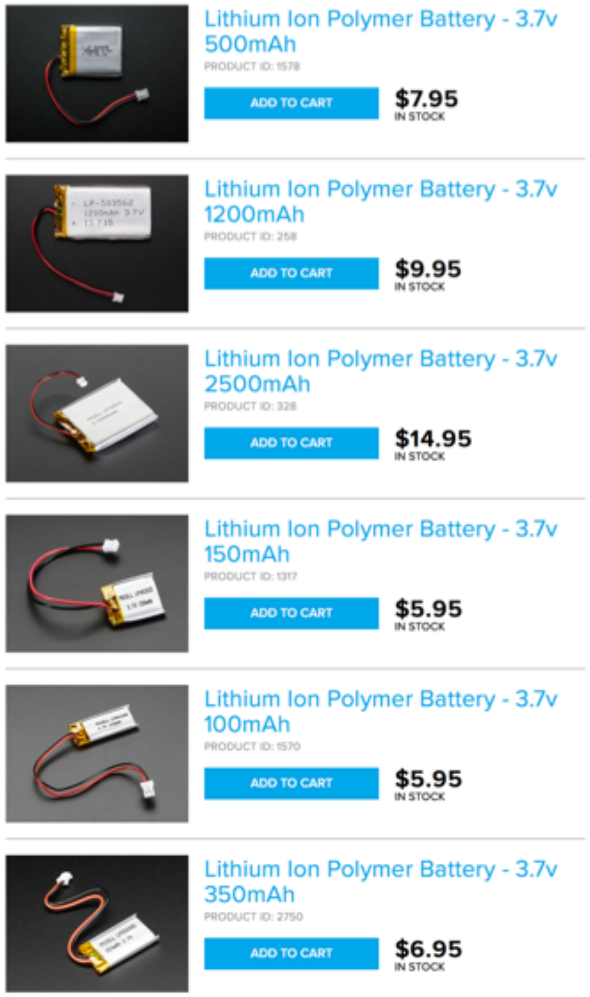


# Battery only Run-Time

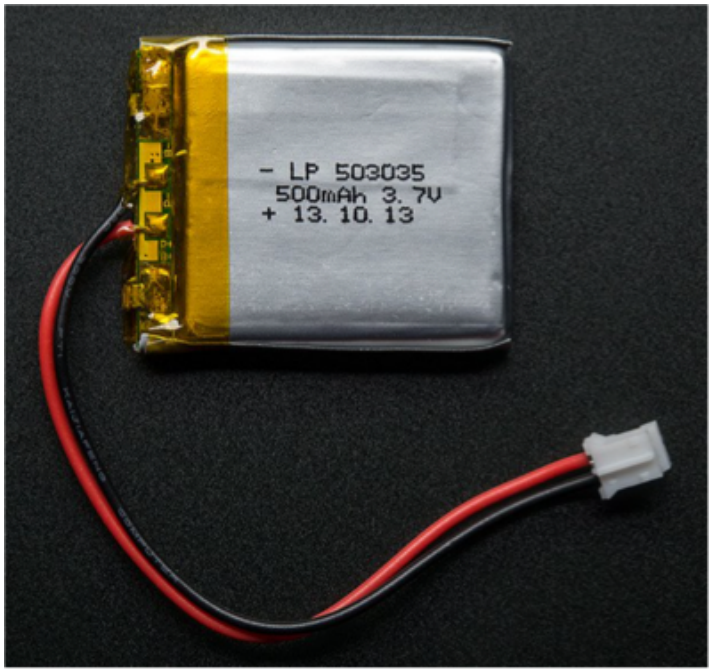
# Energy vs Ah

# Minimum Battery Capacity

# LiPo Batteries for Huzzah32



# Solar Power at Night

# Putting Everything Together …

- Huzzah32
  - Peak current
  - Average current
- Solar cell
  - Peak current
  - Average current
- Battery
  - Max run time
  - Average current
  - Capacity
- Are we good?

# What is Deepsleep?

- Processor off
- RTC still running
  - Can wake up the processor
- What is this good for?

# 10mA → 200mA

- Keep ESP32 in `deepsleep` most of the time
- Wake up every 10min or so to take measurements
  - Send results to cloud
- E.g.
  - 6 measurements per hour
    - 10 seconds @ 200mA
  - Deepsleep for rest of time
    - 3600 – 60 sec @ 10μA
  - Average current

# Putting Everything Together

- Solar cell
- Huzzah32 with ESP32
- INA219 (why?)
- Weather sensors

- Firmware

# Deepsleep

- Processor, WiFi OFF
- Only functions still power
  - RTC, including `deepsleep` wakeup timer
  - Small amount of memory
  - Few peripherals
- After deepsleep, processor restarts
  - Executes `boot.py, main.py`
  - WiFi disconnected (reconnect if needed)
  - RTC still has correct time

# Enter Deepsleep

```
from machine import deepsleep

# sleep for x milli seconds
milli_seconds = 20000
deepsleep(milli_seconds)
```

- Processor restarts after `milli_seconds` delay
- Optional: wake from external pin (later)

# Deepsleep Memory

- RTC memory
  - 64 integers (32 bits), pos 0 … 63
  - One string, up to 2048 characters
  - Retained during deepsleep
- Syntax:

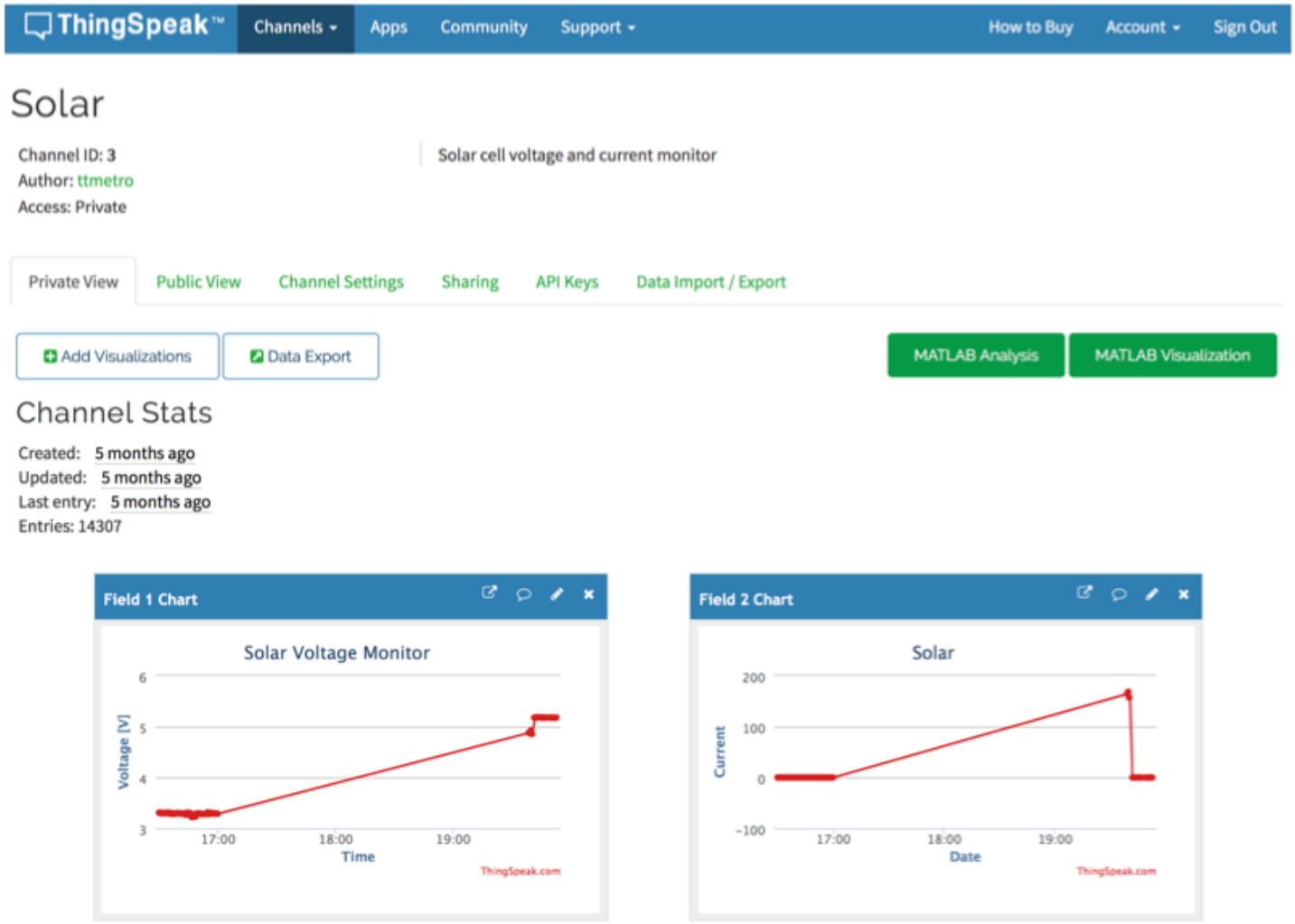
```
from machine import RTC

rtc = RTC()

# read and write RTC memory
rtc.write(0, 123)
print(rtc.read(0))
rtc.write_string("hello world")
print(rtc.read_string())
```

# Keeping Track of Measurements

# Thingspeak.com



# Skeleton `boot.py` for Solar Weather Station

# Summary

- Low power operation
  - Turn power off (most of the time)
  - Deepsleep
  - Average current << peak current
  - Duty cycle
- Beware of other circuits that consume power
  - Sensors?
  - Sleep/power down modes?
  - Check datasheets and/or measure
- Test!