Battery Life Estimate

Battery Life Estimate

https://docs.google.com/spreadsheets/d/1Q-Q_fvTAb0JCVSVsL16ZMg1RRjRiRT8HyTsrPTgaA3Y/edit?usp=sharing

Display & Stepper System

Component	Current (mA)	Voltage (V)	Duty Cycle (%)	Usage (mAh)	Power (mW)	Average Power (mW)
ESP32-C3	80	3.3	50	40	264	132
SSD1306 Display	15	3.3	80	12	49.5	39.6
WS2812B LED	60	5	20	12	300	60
X27 Stepper Motor	20	5	40	8	100	40

Soil Moisture System

Component	Current (mA)	Voltage (V)	Duty Cycle (%)	Usage (mAh)	Power (mW)	Average Power (mW)
ESP32-C3	80	3.3	30	24	264	79.2
Moisture Sensor	5	3.3	50	2.5	16.5	8.25

Power

Device	Daily Power Usage (mWh)	Estimated Battery Needed (mAh)	Chosen Battery (mAh)
Soil Moisture System	2098.8	567	1200
Display System	6518.4	1762	2600

Device	Battery Type	Capacity	Estimated Runtime
Soil Moisture System	Li-ion (LiPo)	1200mAh, 3.7V	**~2 days**

Display & Stepper Motor System	18650 Li-ion	2600mAh, 3.7V	**~1.5 days**

SEN(0193)

Specification

- Operating Voltage: 3.3 ~ 5.5 VDC
- Output Voltage: 0 ~ 3.0VDC
- Operating Current: 5mA
- Interface: PH2.0-3P
- Dimensions: 3.86 x 0.905 inches (L x W)
- Weight: 15g

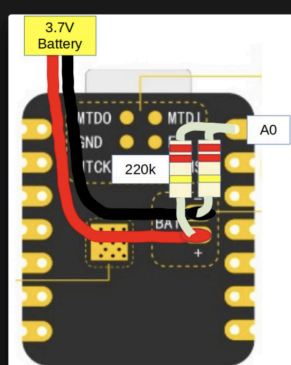
[ESP32_xiao-C3]

Check the battery voltage

Due to the limitation of the number of pins in the ESP32C3, engineers had no extra pins to allocate to the battery for voltage measurement in order to ensure that the XIAO ESP32C3 has the same number of GPIOs as the other XIAO series available.

But if you prefer to use a separate pin for battery voltage measurement, you can refer to the genius operation of msfujino. We would also like to give a special thanks to msfujino for all the experience and efforts shared for the XIAO ESP32C3.

The basic operating idea is: The battery voltage was divided by 1/2 with 200k and connected to the A0 port so that the voltage could be monitored.



Reflection on Battery Sizing and Optimization

1. **How did you determine your "days of use" metric?**

- **Calculating the total power consumption** of each device per hour in **mAh (milliamp-hours)**.
- **Multiplying by 24 hours** to get the total daily power consumption.

- **Dividing the battery capacity by the daily consumption** to estimate how many days the device can run before recharging.
- **Considering duty cycles** (since components like ESP32, stepper motor, and LEDs do not run continuously).
- **Adding a safety margin** to account for inefficiencies in real-world usage.

For example, for the **Soil Moisture System**:

- Daily consumption ≈ **622mAh/day**
- Using a **1200mAh LiPo battery** → Estimated runtime = **~2 days**.

For the **Display & Stepper Motor System**:

- Daily consumption ≈ **1762mAh/day**
- Using a **2600mAh 18650 battery** → Estimated runtime = **~1.5 days**.

2. **What do you think is the optimum size for the battery in your device?**

For the **Soil Moisture System**, a **1200mAh LiPo battery** is optimal because:

- It provides **~2 days of continuous operation**, balancing power needs and compact size.
- If solar charging is added, a **smaller battery (600mAh) could work**, reducing weight and cost.

For the **Display & Stepper Motor System**, a **2600mAh 18650 battery** is the best choice because:

- It **supports high-current loads** like the stepper motor and LED.
- It lasts **~1.5 days per charge**, which is reasonable for user convenience.
- A larger **5000mAh battery** would extend runtime but increase cost and size.

---

3. **What hardware/software/cost/effort trade-offs could improve the user experience?**

Hardware Optimizations:

- **Reduce LED brightness or duty cycle** to cut power consumption.
- **Use a step-down voltage regulator** for more efficient power conversion.
- **Switch to a low-power ESP32 mode** when the system is idle.
- **Integrate a solar charging circuit** for the Soil Moisture System to extend battery life.

Software Optimizations:

- **Use deep sleep mode on ESP32** to lower power consumption when the device is not actively transmitting data.
- **Implement event-based wake-up** (e.g., moisture sensor wakes up only when needed instead of running constantly).
- **Optimize stepper motor control** by running it at a lower speed or reducing activation time.

Cost vs. Performance Trade-offs:

- **LiPo batteries are more compact and rechargeable but cost more** than AA batteries.
- **Using an 18650 rechargeable battery for the display system** is cost-effective compared to disposable AA batteries.
- **Adding Bluetooth instead of Wi-Fi saves power but reduces remote access options**.

Effort vs. User Experience:

- **A simple plug-in USB charging circuit** improves ease of use.
- **Automating stepper motor movements** instead of requiring manual control can improve the product's usability.