

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA ĐIỆN - ĐIỆN TỬ



## THIẾT KẾ HỆ THỐNG NHÚNG

---

# BÁO CÁO BÀI TẬP LỚN SMART WATER BOTTLE

---

**GVHD:** Bùi Quốc Bảo

**Lớp:** TN01 - **Nhóm:** 11

SV thực hiện	MSSV
Huỳnh Nhật Huy	2311149
Bùi Nguyễn Quang Thương	2313401

TP Hồ Chí Minh, Tháng 9 Năm 2025

## Mục lục

<b>1</b>	<b>Requirement</b>	<b>3</b>
<b>2</b>	<b>Architecture</b>	<b>5</b>
<b>3</b>	<b>Usecase</b>	<b>6</b>
3.1	Use Case UC_0001 – Sensor (Measurement) (Cảm biến – Đo lường)	6
3.1.1	Scope (Phạm vi)	6
3.1.2	Actors	6
3.1.3	Stakeholders & Interests	6
3.1.4	Preconditions	6
3.1.5	Postconditions	6
3.1.6	Main Flow of Events	6
3.1.7	Alternative Flows	7
3.1.8	Exception Flows	7
3.1.9	Special Requirements	7
3.1.10	Assumptions	7
3.1.11	Notes	7
3.2	Use Case UC_0002 – Real-time Display (Hiển thị mức nước theo thời gian thực)	7
3.2.1	Scope	7
3.2.2	Actors	7
3.2.3	Stakeholders & Interests	7
3.2.4	Preconditions	8
3.2.5	Postconditions	8
3.2.6	Main Flow	8
3.2.7	Alternative Flows	8
3.2.8	Exception Flows	8
3.2.9	Special Requirements	8
3.2.10	Assumptions	8
3.2.11	Notes	9
3.3	Use Case UC_0004 – Daily Goal Management (Mục tiêu uống nước hằng ngày)	9
3.3.1	Scope	9
3.3.2	Actors	9
3.3.3	Stakeholders & Interests	9
3.3.4	Preconditions	9
3.3.5	Postconditions	9
3.3.6	Main Flow	9
3.3.7	Alternative Flows	9
3.3.8	Exception Flows	9
3.3.9	Special Requirements	10
3.3.10	Assumptions	10
3.3.11	Notes	10
3.4	Use Case UC_0001 – Power Supply (Cấp nguồn tổng)	10
3.5	Use Case UC_0002 – Sensor (Measurement) (Cảm biến – Đo lường)	11
3.6	Use Case UC_0003 – Time (Thời gian)	13
3.7	Use Case UC_0004 – Display (Hiển thị)	15
<b>4</b>	<b>Hardware Requirement</b>	<b>17</b>



## 5 Schematic

18

## 1 Requirement

### REQ-1: Đo và hiển thị mức nước theo thời gian thực

- REQ-1.1: Bình phải có cảm biến mức nước để phát hiện chính xác lượng nước trong bình
- REQ-1.2: Sai số đo của cảm biến không được vượt quá  $\pm 5$  ml trong dải 0 ml đến dung tích tối đa.
- REQ-1.3: Hệ thống phải cập nhật và hiển thị giá trị đo trên LCD trong vòng 5 giây sau khi mức nước thay đổi.
- REQ-1.4: Khi lượng nước trong bình bằng 0 ml, màn hình hiển thị cảnh báo “Hết nước”.
- REQ-1.5: Khi lượng nước đạt mức tối đa (đầy bình), màn hình hiển thị cảnh báo “Đầy bình”.

### REQ-2: Theo dõi lượng nước uống hằng ngày và hiển thị mục tiêu

- REQ-2.1: Hệ thống phải có bộ nhớ để lưu trữ tổng lượng nước người dùng uống trong ngày
- REQ-2.2: Bình phải so sánh lượng nước uống được với mục tiêu hằng ngày. Có thể thay đổi theo cài đặt của người dùng (mặc định 2000ml)
- REQ-2.3: Màn hình LCD phải hiển thị tiến độ uống nước trong ngày dưới dạng số ml đã uống
- REQ-2.4: Hệ thống phải tự động đặt lại bộ đếm lượng nước về 0 ml lúc 0 giờ mỗi ngày.

### REQ-3: Giao diện hiển thị qua màn hình LCD đạt chuẩn

- REQ-3.1: Màn hình phải rõ ràng, dễ đọc dưới điều kiện ánh sáng trong nhà và ánh sáng yếu.
- REQ-3.2: Màn hình LCD phải tiêu thụ công suất nhỏ hơn 0.2W trong chế độ hoạt động bình thường.
- REQ-3.3: Khi dung lượng pin yếu (dưới 10%), màn hình phải hiển thị cảnh báo pin yếu.
- REQ-3.4: Màn hình phải hiển thị các thông tin sau:
  - + Lượng nước hiện tại trong bình
  - + Lượng nước đã uống trong ngày
  - + Phần trăm dung lượng pin hiện tại

### REQ-4: Yêu cầu về nguồn và an toàn

- REQ-4.1: Hệ thống phải được cấp nguồn bằng pin hoặc nguồn trực tiếp 9V
- REQ-4.2: Hệ thống phải hoạt động tối thiểu 1 tháng liên tục sau mỗi lần thay pin.
- REQ-4.3: Toàn bộ linh kiện điện tử phải được bọc cách điện và chống nước đạt chuẩn IPX4 trở lên.

- REQ-4.4: Khi phát hiện rò rỉ điện hoặc chập mạch, hệ thống phải tự động ngắt nguồn để đảm bảo an toàn.
- REQ-4.5: Vật liệu chế tạo vỏ bình phải an toàn thực phẩm, ví dụ:
  - + BPA-free
  - + Inox 304
  - + Nhựa Tritan

### **REQ-5: Yêu cầu về giá thành và khả năng tiếp cận**

- REQ-5.1: Giá bán lẻ của bình nước thông minh phải dưới 500.000 VNĐ để phù hợp với sinh viên và cá nhân.
- REQ-5.2: Các linh kiện sử dụng trong bình phải phổ biến, dễ tìm, chi phí thấp để giảm giá thành sản phẩm.
- REQ-5.3: Chi phí bảo trì và thay thế linh kiện (pin, màn hình, cảm biến) phải thấp, không vượt quá 20% giá trị sản phẩm.
- REQ-5.4: Bình phải được sản xuất theo hướng quy mô nhỏ – trung bình, đảm bảo tính thương mại hóa và giá cả cạnh tranh.

### **REQ-6: Yêu cầu về độ bền và tuổi thọ sử dụng**

- REQ-6.1: Vỏ bình phải được làm từ vật liệu an toàn thực phẩm (Inox 304, Tritan, hoặc nhựa BPA-free).
- REQ-6.2: Toàn bộ hệ thống điện tử phải hoạt động ổn định trong ít nhất 2 năm với điều kiện sử dụng bình thường.
- REQ-6.3: Bình phải chịu được va đập thông thường và chịu rơi từ độ cao 1 mét xuống nền cứng mà không hỏng hóc nghiêm trọng.
- REQ-6.4: Cảm biến mức nước và màn hình LCD phải có độ tin cậy cao, hoạt động ổn định trong ít nhất 10.000 giờ sử dụng.
- REQ-6.5: Bình phải chịu được nhiệt độ môi trường từ 0°C đến 50°C (điều kiện sinh hoạt bình thường) mà không ảnh hưởng đến chức năng đo và hiển thị.

### **REQ-7: Yêu cầu về thiết kế ngoại hình tiện lợi và thẩm mỹ**

- REQ-7.1: Bình phải có dung tích tiêu chuẩn từ 700 ml – 1000 ml, phù hợp cho nhu cầu cá nhân thường ngày.
- REQ-7.2: Trọng lượng bình (chưa có nước) không được vượt quá 800 g để thuận tiện mang theo.
- REQ-7.3: Thiết kế vỏ ngoài phải dễ cầm nắm, chống trơn trượt, và có nắp chống tràn để sử dụng an toàn khi di chuyển.
- REQ-7.4: Màn hình LCD phải được bố trí ở vị trí dễ quan sát nhưng không gây cản trở thao tác uống nước.

- REQ-7.5: Bình phải có ít nhất hai lựa chọn màu sắc để phù hợp với nhiều đối tượng người dùng.
- REQ-7.6: Thiết kế tổng thể phải thanh lịch, hiện đại, phù hợp cho cả môi trường học tập, làm việc và thể thao.
- REQ-7.7: Bình phải có phụ kiện hỗ trợ như quai xách hoặc dây treo để tăng tính tiện lợi khi mang theo.

## 2 Architecture

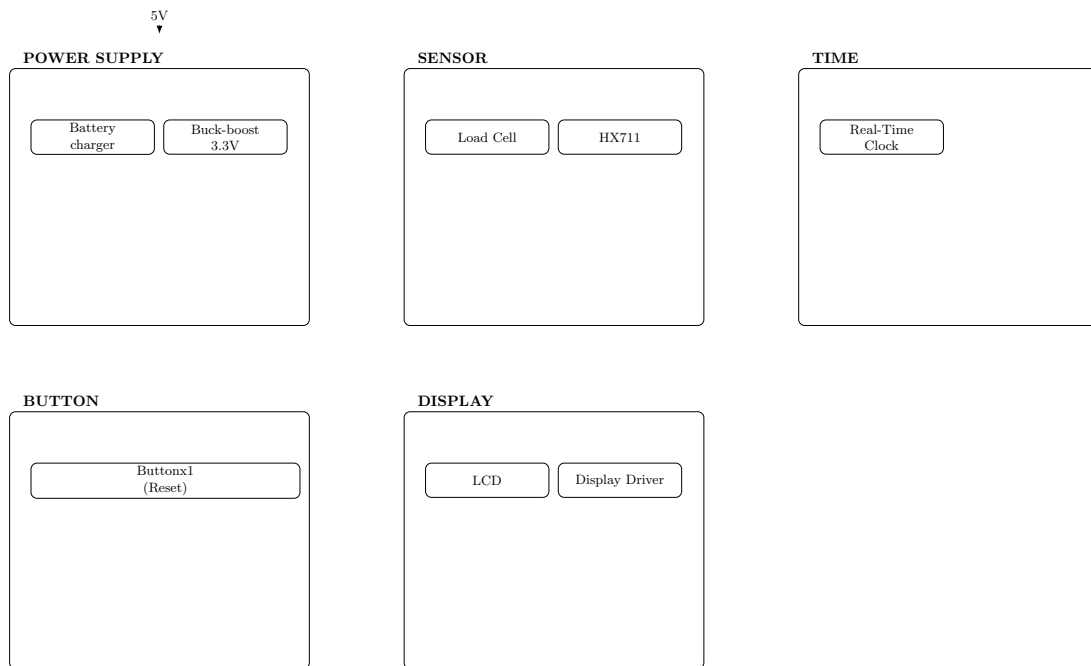


Figure 1: System Architecture

## 3 Usecase

### 3.1 Use Case UC\_0001 – Sensor (Measurement) (Cảm biến – Đo lường)

#### 3.1.1 Scope (Phạm vi)

Đo liên tục mực nước (mL) từ cảm biến mức/khối lượng; lọc nhiễu; hiệu chuẩn/Tare; xuất kết quả đo hợp lệ phục vụ hiển thị và tính tổng uống.

#### 3.1.2 Actors

**Primary:** MCU (vi điều khiển)

**Secondary:**

- Level/Weight Sensor
- Sensor Driver (ADC/HX711)
- Calibration/Tare
- Time (RTC/tick)
- Storage/Logger

#### 3.1.3 Stakeholders & Interests

- Người dùng: số đo chính xác, ổn định, nhanh.
- Hệ thống: điện năng thấp, dữ liệu có time stamp phục vụ tính “đã uống trong ngày”.

#### 3.1.4 Preconditions

- Cảm biến lắp đúng; driver khởi tạo; sample rate cấu hình.
- Có hệ số hiệu chuẩn/tare.
- RTC hoạt động.

#### 3.1.5 Postconditions

- Có volume\_mL hợp lệ (đã lọc và trừ bì).
- Gửi kết quả kèm time stamp cho Display/UI và Logger.

#### 3.1.6 Main Flow of Events

1. RTC phát tick  $T$  ms  $\rightarrow$  MCU kích hoạt đo.
2. MCU gọi Sensor Driver `read_raw()`.
3. Driver loại offset, phát hiện overflow/timeout, áp lọc (IIR/MA)  $\rightarrow$  `filtered_value`.
4. MCU áp Calibration/Tare  $\rightarrow$  `volume_mL`.
5. Kiểm tra validity: dải đo, slew limit, ổn định.
6. Đóng gói (time stamp)  $\rightarrow$  Display/UI + Logger.

### 3.1.7 Alternative Flows

- A1 Adaptive filtering: động (nhANH khi biến động / dài khi tĩnh).
- A2 Geometry lookup: level  $\rightarrow$  mL theo bảng hình học.

### 3.1.8 Exception Flows

- E1 Sensor/ADC fault: mất tín hiệu, CRC/parity fail, timeout  $\rightarrow$  `sensor_fault`, giữ giá trị cuối, cảnh báo nội bộ.
- E2 Out-of-range: kẹp biên, log sự kiện.
- E3 Time invalid: mất RTC  $\rightarrow$  gán cờ `time_unsynced`.

### 3.1.9 Special Requirements

- Sai số  $\leq \pm 5$  mL
- End-to-end đo  $\rightarrow$  sẵn sàng hiển thị  $\leq 5$  s khi mức thay đổi
- Công suất đo trung bình tối ưu để đáp ứng yêu cầu 1 tháng pin

### 3.1.10 Assumptions

- Dùng load cell + HX711/ADC hoặc level sensor có bảng tra.
- RTC khả dụng.

### 3.1.11 Notes

Trì hoãn ngắn khi có nhiễu (LCD/backlight bật) trước khi lấy mẫu.

## 3.2 Use Case UC\_0002 – Real-time Display (Hiển thị mức nước theo thời gian thực)

### 3.2.1 Scope

Cập nhật và hiển thị `volume_mL` trên LCD; sinh cảnh báo “Hết nước” và “Đầy bình”; hiển thị thông tin tổng hợp theo

### 3.2.2 Actors

**Primary:** Display/UI (LCD Controller)

**Secondary:** MCU, Measurement, Battery Monitor, Storage/Logger

### 3.2.3 Stakeholders & Interests

- Người dùng: số liệu rõ ràng, dễ đọc, phản hồi nhanh.
- Hệ thống: giao diện gọn, tiết kiệm năng lượng.



#### 3.2.4 Preconditions

- LCD hoạt động.
- Có dữ liệu `volume_mL` hợp lệ.
- Có định dạng hiển thị.

#### 3.2.5 Postconditions

- LCD hiển thị đúng: mực nước hiện tại, “đã uống”, % pin.
- Nếu rơi vào ngưỡng đặc biệt → thông điệp cảnh báo tương ứng.

#### 3.2.6 Main Flow

1. Nhận dữ liệu `volume_mL` mới (event từ UC\_0001).
2. So sánh với giá trị trước → quyết định cập nhật.
3. Cập nhật LCD trong  $\leq 5$  giây kể từ lúc thay đổi
4. Đồng thời hiển thị:
  - Lượng nước hiện tại (mL),
  - Lượng đã uống trong ngày (mL, từ UC\_0003),
  - % pin hiện tại
5. Nếu `volume_mL == 0` → hiện “Hết nước”
6. Nếu `volume_mL  $\geq V_{full}$`  → hiện “Đầy bình”

#### 3.2.7 Alternative Flows

- A1 Chế độ ánh sáng yếu: tăng độ tương phản/kí tự lớn
- A2 Tiết kiệm điện: giảm tần suất refresh khi không thay đổi.

#### 3.2.8 Exception Flows

- E1 LCD fault: không cập nhật được → log + thử lại theo backoff.
- E2 Dữ liệu đo không hợp lệ → giữ hiển thị gần nhất, gắn cờ lỗi nhỏ.

#### 3.2.9 Special Requirements

- Dễ đọc trong nhà/ánh sáng yếu
- Công suất LCD  $< 0.2$  W trong chế độ thường

#### 3.2.10 Assumptions

- Kích cỡ font và layout đã chuẩn hoá cho người dùng phổ thông.

### 3.2.11 Notes

Hạn chế flicker; chỉ refresh phần thay đổi.

## 3.3 Use Case UC\_0004 – Daily Goal Management (Mục tiêu uống nước hằng ngày)

### 3.3.1 Scope

Quản lý mục tiêu uống/ngày, mặc định 2000 mL; so sánh tiến độ và cung cấp thông tin cho UI.

### 3.3.2 Actors

**Primary:** User (qua nút/app)

**Secondary:** MCU, Storage, Display/UI

### 3.3.3 Stakeholders & Interests

- Người dùng: dễ đặt/chỉnh mục tiêu theo thể trạng.
- Hệ thống: lưu bền vững, đơn giản.

### 3.3.4 Preconditions

- Có giao diện đặt mục tiêu.
- Bộ nhớ cấu hình sẵn.

### 3.3.5 Postconditions

- Lưu `goal_mL` (mặc định 2000 mL).
- Cung cấp % tiến độ =  $\text{intake\_today\_mL} / \text{goal\_mL}$ .

### 3.3.6 Main Flow

1. Người dùng xem mục tiêu hiện tại.
2. Người dùng thay đổi mục tiêu (nếu muốn).
3. Hệ thống lưu NVM và tính lại % tiến độ.
4. UI hiển thị số mL đã uống và có thể kèm % hoặc thanh tiến độ.

### 3.3.7 Alternative Flows

A1 Presets: 1500 / 2000 / 2500 mL theo gợi ý.

A2 Giới hạn: kẹp mục tiêu trong dải an toàn.

### 3.3.8 Exception Flows

E1 Lỗi ghi NVM: retry; nếu thất bại giữ giá trị cũ và cảnh báo.

### 3.3.9 Special Requirements

- Cho phép chỉnh mục tiêu; mặc định 2000 mL.

### 3.3.10 Assumptions

- Có ít nhất 1 phương thức nhập (nút, hoặc app cấu hình).

### 3.3.11 Notes

Có thể bổ sung mẹo nhắc uống theo tiến độ.

## 3.4 Use Case UC\_0001 – Power Supply (Cấp nguồn tổng)

### Scope

Quản lý cấp nguồn cho toàn hệ thống: chọn nguồn USB/battery, phân phối qua regulator đến MCU, LCD, cảm biến.

### Actors

**Primary Actor:** Hệ thống (firmware Power Manager)

**Secondary Actors:** Người dùng, USB nguồn ngoài

### Stakeholders & Interests

- Người dùng: thiết bị luôn bật ổn định, an toàn
- Hệ thống: chuyển đổi mượt giữa USB và pin, không reset

### Preconditions

- Pin còn dung lượng hoặc đang cắm USB 5V
- Mạch regulator sẵn sàng

### Postconditions

- Tất cả rail (3.3V/5V) đạt mức danh định
- MCU, LCD, cảm biến hoạt động bình thường

### Main Flow of Events

1. Hệ thống kiểm tra nguồn vào: USB present? Battery voltage?
2. Nếu có USB và cấu hình ưu tiên USB → chọn đường USB làm nguồn chính
3. Nếu không có USB → chọn pin
4. Bật các regulator theo thứ tự (sequencing: 3.3V trước, 5V sau nếu cần)
5. Chạy tự kiểm (Power-Good, brown-out flags)

6. Thông báo trạng thái cấp nguồn cho MCU (GPIO/ADC/PMIC IRQ)
7. Vào chế độ hoạt động bình thường

### Alternative Flows

**A1 – Chỉ dùng pin:** Bỏ qua kiểm USB, cấp nguồn từ pin → quay lại bước 4

**A2 – USB yếu (sag):** nếu VBUS < 4.5V → rơi về pin → quay lại bước 4

### Exception Flows

**E1 – Brown-out khi chuyển nguồn:** nếu rail tụt < ngưỡng → reset mềm, retry sequencing tối đa N lần

**E2 – Quá nhiệt PMIC:** cắt tải không thiết yếu (LCD backlight), ghi log, cảnh báo người dùng

### Special Requirements

- Thời gian chuyển nguồn USB ↔ Battery < 50 ms, không được reset MCU
- Có tín hiệu Power-Good và bảo vệ ngược cực / quá dòng
- EMC/ESD theo chuẩn IEC 61000-4-2 ở cổng USB

### Assumptions

- Board đã có mạch ORing/PMIC để chọn nguồn
- MCU đọc được cờ trạng thái nguồn

### Notes

- Ưu tiên USB để sạc và cấp tải; pin làm dự phòng
- Có thể thêm “Low-Power Mode” khi pin yếu

## 3.5 Use Case UC\_0002 – Sensor (Measurement) (Cảm biến – Đo lường)

### Scope (Phạm vi)

Đo level/weight (mức/khối lượng) của bình bằng Level/Weight Sensor; đọc dữ liệu thô qua Sensor Driver; áp Calibration/Tare để suy ra thể tích (mL); đồng bộ chu kỳ đo và gán Time (thời gian) cho bản ghi.

## Actors

**Primary Actor(s):** MCU (vi điều khiển)

**Secondary Actors:** :

- Level/Weight Sensor – phần cứng tạo tín hiệu thô theo mức nước/khối lượng
- Sensor Driver – giao tiếp ADC/HX711, lọc nhiễu, trả dữ liệu xử lý
- Calibration/Tare – phần mềm hiệu chuẩn/trừ bì
- Display/UI – hiển thị kết quả đo cho người dùng
- Storage – ghi log kết quả đo để tính tổng/ngày

## Stakeholders & Interests

- Người dùng: số đo mL chính xác, ổn định, phản hồi nhanh
- Hệ thống: tiêu thụ điện thấp, chống nhiễu, dữ liệu có time stamp để tính daily intake

## Preconditions (Điều kiện trước)

- Cảm biến lắp đúng, dây tín hiệu tốt
- Sensor Driver đã khởi tạo, sample rate được cấu hình
- Calibration/Tare có offset & gain
- Time (RTC/tick) hoạt động

## Postconditions (Điều kiện sau)

- Có volume\_mL hợp lệ (đã lọc và trừ bì)
- Dữ liệu kèm time stamp được gửi cho Display/UI và Logger

## Main Flow of Events

1. Time phát tick theo chu kỳ T ms → MCU kích hoạt đo
2. MCU gọi Sensor Driver để read\_raw() từ cảm biến qua ADC/HX711
3. Sensor Driver loại offset, phát hiện overflow/timeout, áp lọc (IIR/MA) → filtered\_value
4. MCU áp Calibration/Tare: tare\_offset & gain → volume\_mL
5. MCU kiểm tra validity: dải đo, slew limit, stable detection
6. MCU đóng gói kết quả với time stamp → Display/UI + Logger
7. Nếu phát hiện pour/drink event → tăng sample rate tạm thời, sau đó khôi phục

## Alternative Flows

**A1 – Adaptive filtering:** Khi biến động lớn dùng IIR nhanh; khi tĩnh dùng MA dài

**A2 – Geometry lookup:** Nếu dùng level sensor thì tra bảng level → mL

**A3 – Auto-cal reminder:** Nếu sai lệch tích lũy vượt ngưỡng, nhắc Calibration/Tare

## Exception Flows

**E1 – Sensor/ADC fault:** Mất kết nối, CRC/parity fail, timeout → đặt sensor\_fault, giữ giá trị cuối, cảnh báo

**E2 – Out-of-range:** Giá trị vượt dải → kẹp biên, log sự kiện

**E3 – Time invalid:** Mất RTC → vẫn đo nhưng gắn cờ time\_unsynced

**E4 – Wrong tare:** tare\_offset sai (bình không trống khi tare) → cảnh báo, bỏ tare khung hiện tại

## Special Requirements

- Accuracy  $\leq \pm 2\%$  trong dải 0–1000 mL khi đã Calibration/Tare
- Noise sau lọc khi tĩnh:  $< \pm 1-2$  mL
- End-to-end latency từ đo đến hiển thị:  $< 300$  ms khi đang uống
- Sample rate: 10–20 Hz (burst 50–80 Hz khi rót/uống)
- Power budget trung bình  $< 5$  mA (ADC/Amplifier bật/tắt theo chu kỳ)
- Lưu calibration data vào NVM kèm checksum/version

## Assumptions

- Phần cứng load cell + HX711/ADC hoặc level sensor có bảng tra
- Bình có hình học đã biết; RTC/tick hệ thống khả dụng

## Notes

- Khi LCD backlight bật hoặc có nhiễu, trì hoãn vài trăm ms trước khi đo
- Có thể bật temperature compensation nếu có NTC

## 3.6 Use Case UC\_0003 – Time (Thời gian)

### Scope (Phạm vi)

Quản lý Time Reading (đọc thời gian thực), Time Setting (chỉnh ngày/giờ từ nút), Day Rollover (chốt dữ liệu khi sang ngày mới) dựa trên Realtime Clock (RTC/system tick) để:

- Đồng dấu thời gian cho đo lường
- Reset bộ đếm theo ngày
- Hiển thị đúng trên LCD

## Actors

**Primary Actor:** MCU (vi điều khiển)

**Secondary Actors:** :

- Người dùng (nút chỉnh giờ)
- Realtime Clock (RTC)
- Display/UI
- Logger/Storage

## Stakeholders & Interests

- Người dùng: thời gian hiển thị đúng, chỉnh giờ nhanh và dễ
- Hệ thống: có time stamp tin cậy, rollover ngày ổn định để tính daily intake và days-met-goal

## Preconditions (Điều kiện trước)

- RTC đã chạy ổn định (có nguồn nuôi) hoặc system tick dự phòng
- Nút TIME/HOUR/MIN hoạt động và đã debounce
- Timezone mặc định đã thiết lập (hoặc local time)

## Postconditions (Điều kiện sau)

- Hệ thống có current time hợp lệ để đóng dấu log
- Khi rollover ngày, dữ liệu được chốt và bộ đếm reset đúng

## Main Flow of Events

1. MCU đọc thời gian từ RTC/system tick, cập nhật time cache mỗi T ms/s
2. Cập nhật Display/UI (hiển thị HH:MM), gán time stamp vào mẫu đo/nhật ký
3. Người dùng nhấn TIME → vào chế độ cài đặt; HOUR/MIN để tăng; nhấn TIME lần nữa để xác nhận
4. MCU ghi giá trị mới xuống RTC, đồng bộ time cache, log sự kiện *time\_changed*
5. Day Rollover: tại 00:00:00 → MCU sinh sự kiện *midnight*, chốt daily intake, đánh giá goal, reset bộ đếm ngày
6. Tiếp tục vòng lặp: đọc, hiển thị, ghi log, rollover hàng ngày

## Alternative Flows

**A1 – Auto Sync:** Khi kết nối USB/App, hệ thống nhận time sync → cập nhật RTC, log *time\_changed*

**A2 – 24h/12h Display:** Người dùng chọn định dạng hiển thị, internal time vẫn 24h

**A3 – DST/Timezone:** Khi đổi DST/múi giờ, áp offset cho hiển thị/time stamp tương lai; log quá khứ giữ nguyên

## Exception Flows

- E1 – RTC lost/invalid:** Nếu RTC reset (mất pin nuôi) → dùng system tick tạm, gán cờ *time\_unsynced*, yêu cầu Time Setting/Auto Sync
- E2 – Button bounce/long-press error:** Nếu rung/kẹt → debounce hoặc bỏ qua, giới hạn repeat rate
- E3 – Missed rollover:** Nếu thiết bị ngủ qua 00:00 → khi dậy, so sánh ngày, chạy rollover bù
- E4 – Back-in-time change:** Nếu chỉnh giờ lùi nhiều → đánh dấu *time\_warp* để không cộng sai daily intake; tách log trước/sau thay đổi

## Special Requirements

- Accuracy RTC: sai số  $\leq \pm 20$  ppm ( $\approx \pm 1.7$  s/ngày) hoặc có periodic sync
- Rollover latency: xử lý trong  $\leq 1$  s sau midnight
- UI responsiveness: vào/thoát Time Setting  $\leq 100$  ms; mỗi lần tăng HOUR/MIN cập nhật hiển thị  $\leq 100$  ms
- Debounce nút: 20–50 ms; long-press  $\geq 1$  s; auto-repeat 5–10 Hz
- Power: RTC backup cell  $< 1 \mu\text{A}$ ; giữ thời gian khi ngủ sâu

## Assumptions

- Có RTC crystal 32.768 kHz hoặc internal RC đã hiệu chỉnh, có backup supply
- Các nút TIME/HOUR/MIN nối GPIO hỗ trợ interrupt
- Không yêu cầu xử lý leap seconds

## Notes

- Khi Auto Sync từ USB/App, hiển thị thông báo “Time synced” ngắn
- Day Rollover nên theo local time; khi đổi múi giờ, rollover ngày tiếp theo theo múi giờ mới

### 3.7 Use Case UC\_0004 – Display (Hiển thị)

**Scope (Phạm vi):** Quản lý hiển thị Status View (Today, Goal, Streak, Last sip), điều hướng Menu/Setting UI, giao tiếp Display Driver (LCD/OLED – I<sup>2</sup>C), tối ưu Brightness Control để tiết kiệm pin mà vẫn rõ ràng.

**Primary Actor(s) (Tác nhân chính):** MCU (vi điều khiển)

**Secondary Actors (Tác nhân phụ):**

- Người dùng (nút bấm)
- Sensor (Measurement)
- Storage
- Time



#### Preconditions (Điều kiện trước):

- Display Driver đã khởi tạo, bus I<sup>2</sup>C ổn định
- Có dữ liệu từ Sensor (mL), Storage (goal, đơn vị), Time (giờ/phút) để hiển thị
- Nút điều hướng (UP/DOWN/OK/BACK) hoạt động, đã debounce

#### Postconditions (Điều kiện sau):

- Status View (Today/Goal/Streak/Last sip) hiển thị cập nhật, dễ đọc
- Người dùng vào Menu/Setting UI, thay đổi tham số và thấy phản hồi ngay
- Brightness Control áp dụng theo môi trường/cấu hình, tiết kiệm pin

#### Main Flow of Events (Luồng sự kiện chính):

1. MCU nhận dữ liệu từ Sensor/Storage/Time, dựng Status View.
2. MCU gọi Display Driver (I<sup>2</sup>C) để vẽ văn bản/biểu tượng theo layout (chống nhấp nháy).
3. Định kỳ (250–500 ms) hoặc theo sự kiện, Status View được refresh một phần để giảm nhấp nháy và tiết kiệm năng lượng.
4. Người dùng nhấn nút → chuyển sang Menu/Setting UI:
  - Điều hướng mục (Goal, Unit, Time, Brightness...).
  - Chỉnh giá trị, preview, OK để lưu.
5. Khi lưu, MCU cập nhật Storage, phản chiếu thay đổi lên Status View.
6. Áp Brightness Control: giảm sáng khi idle/ánh sáng thấp; tăng sáng khi có thao tác.
7. Sau idle lâu, chuyển về Status View hoặc dim/off.

#### Alternative Flows (Luồng thay thế):

- A1 Quick Toggle: Nhấn ngắn để luân phiên Status sub-views (Today ↔ Goal ↔ Streak ↔ Last sip).
- A2 Units/Locale: Người dùng chọn mL/oz, 24h/12h; UI cập nhật tức thì.
- A3 Auto-Dim by Time: Sau N giây không bấm → giảm sáng; có bấm → tăng sáng lại.
- A4 Low-Power Screen: Khi pin yếu, UI chuyển layout tối giản.

#### Exception Flows (Luồng ngoại lệ):

- E1 I<sup>2</sup>C error: Giao tiếp lỗi → retry; nếu quá số lần, đặt cờ `display_fault`, chuyển low-update mode.
- E2 Font/Asset missing: Dùng fallback font/biểu tượng.
- E3 Oversized text: Tràn khung → truncate/scroll, log để chỉnh UI sau.
- E4 Brightness sensor fail: Không đọc được → chuyển manual brightness.

#### Special Requirements (Yêu cầu đặc biệt):

- Readability: chữ  $\geq 12\text{--}14\text{ px}$ , tương phản cao
- UI latency: phản hồi nút  $\leq 100\text{ ms}$ ; cập nhật khung hình  $\leq 200\text{ ms}$
- Refresh policy: partial update để giảm nhấp nháy, tiết kiệm điện
- Power: với Brightness Control bật, tiết kiệm  $\geq 30\%$  năng lượng
- Layout: Status View gồm 4 mục: Today, Goal, Streak, Last sip
- Accessibility: hỗ trợ High-contrast, Large font

#### Assumptions (Giả định):

- Màn LCD/OLED nhỏ ( $0.96\text{--}1.3''$ ), điều khiển qua I<sup>2</sup>C
- Nút tactile có interrupt để UI nhanh nhạy
- Không yêu cầu đồ họa phức tạp (bitmap lớn)

#### Notes (Ghi chú):

- Khi Status View hiển thị Last sip, nên kèm time stamp ngắn.
- Nếu dùng OLED: ưu tiên nền tối + auto-dim để tăng tuổi thọ màn hình.

## 4 Hardware Requirement

- **Nguồn:** Sử dụng **pin Li-ion 1 cell** ( $3.7\text{ V}$ , dung lượng  $\geq 500\text{ mAh}$ ), sạc qua **USB-C 5 V**. Mạch quản lý nguồn bao gồm IC sạc, bảo vệ quá dòng/quá áp và **bộ ổn áp 3.3 V** (và  $5\text{ V}$  nếu cần cho màn hình hoặc cảm biến). Hiệu suất chuyển đổi yêu cầu  $\eta \geq 85\%$ ; điện áp ổn định trong khoảng  $\pm 5\%$  để đảm bảo hoạt động của vi điều khiển và cảm biến.
- **MCU:** Vi điều khiển họ **STM32F1**, hỗ trợ **ADC 12-bit**, **I<sup>2</sup>C**, **GPIO** và chế độ **Sleep-/Low Power**. Tốc độ xung nhịp từ  $16\text{--}72\text{ MHz}$ ; tiêu thụ dòng thấp ( $< 10\text{ }\mu\text{A}$  ở chế độ ngủ). Sai số dao động nội  $\pm 1\%$ ; có khả năng lưu trữ dữ liệu trong bộ nhớ **Flash/NVM** để ghi lại thông tin hiệu chuẩn và nhật ký sử dụng.
- **Cảm biến:**
  - **Load cell** loại 4 dây, dải đo  $0\text{--}1\text{ kg}$ , độ phân giải  $1\text{--}2\text{ g}$ .
  - **Bộ ADC khuếch đại HX711**, độ phân giải 24-bit, tốc độ lấy mẫu  $10\text{--}80\text{ Hz}$ , sai số tuyến tính  $\pm 0.02\%$  FS.
  - Dữ liệu được lọc trung bình và gửi định kỳ về MCU qua giao tiếp số (DOUT, SCK). Độ chính xác khối lượng tổng thể đạt  $\pm 5\text{ ml}$  sau hiệu chuẩn.
- **Thời gian (RTC):** Sử dụng **DS3231** (I<sup>2</sup>C). Độ chính xác  $\pm 2\text{ ppm}$  (tương đương sai số khoảng  $\pm 1\text{ phút/năm}$ ); có pin dự phòng giữ thời gian khi mất nguồn chính. RTC cung cấp thời điểm cho các bản ghi và tự động **reset dữ liệu theo ngày** vào 00:00.
- **Hiển thị:** Màn hình **LCD/OLED 0.96 inch** giao tiếp I<sup>2</sup>C, điện áp hoạt động  $3.3\text{ V}$ , dòng tiêu thụ  $< 20\text{ mA}$ . Hiển thị các thông số: lượng nước trong ngày, mục tiêu (Goal), số ngày liên tiếp (Streak), và thời điểm uống cuối (Last sip). Sai số hiển thị không đáng kể (do tính toán nội MCU).

- **Nút nhấn:** Một nút nhấn đa chức năng RESET Nối GPIO có ngắt, xử lý **debounce** bằng phần mềm hoặc RC, thời gian phản hồi  $< 50$  ms.
- **Cảnh báo :** Đèn LED nhỏ để thông báo tình trạng pin yếu, lỗi cảm biến hoặc nhắc hiệu chuẩn.
- **Cơ khí/Độ bền:** Vỏ chống trượt, bảo vệ cơ khí cho load cell, chịu rơi ở độ cao khoảng 1 m; hoạt động trong dải nhiệt độ  $0-50^{\circ}\text{C}$ ; cấp bảo vệ nước tương đương **IPX4**.

## 5 Schematic

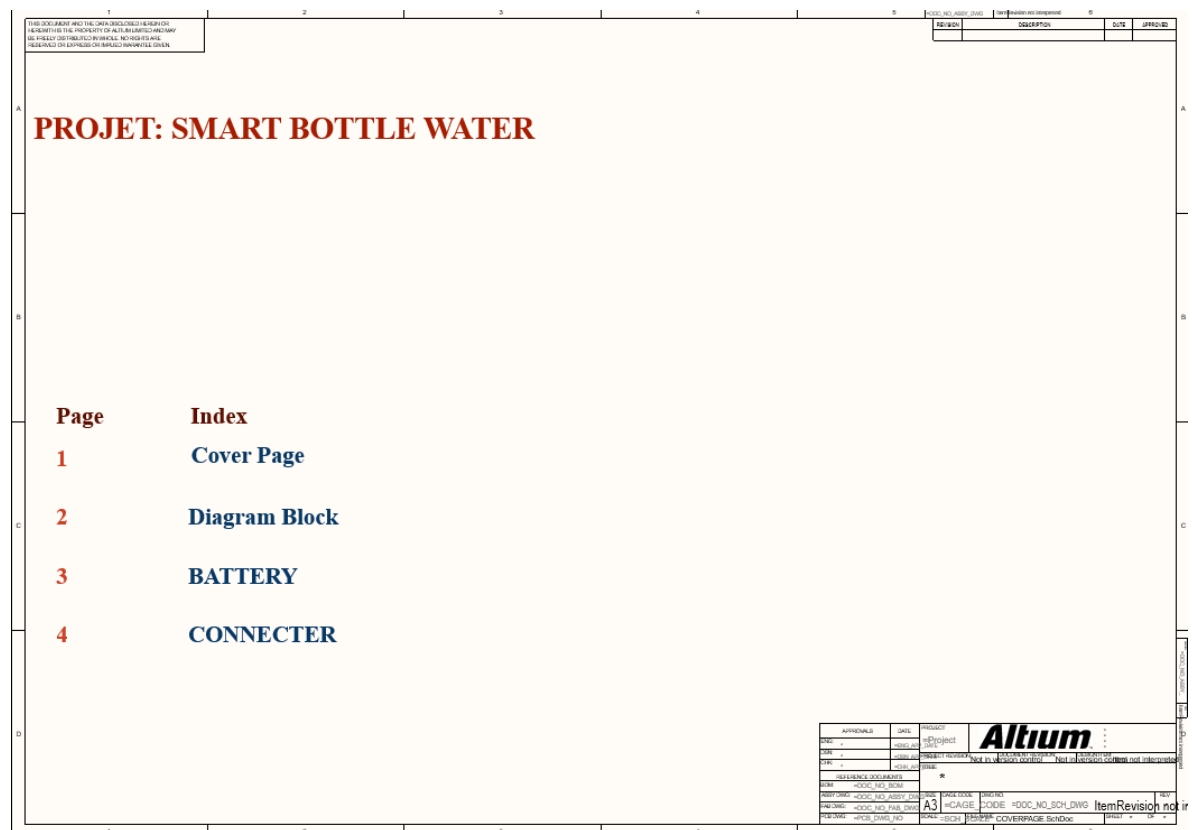


Figure 2: COVER PAGE

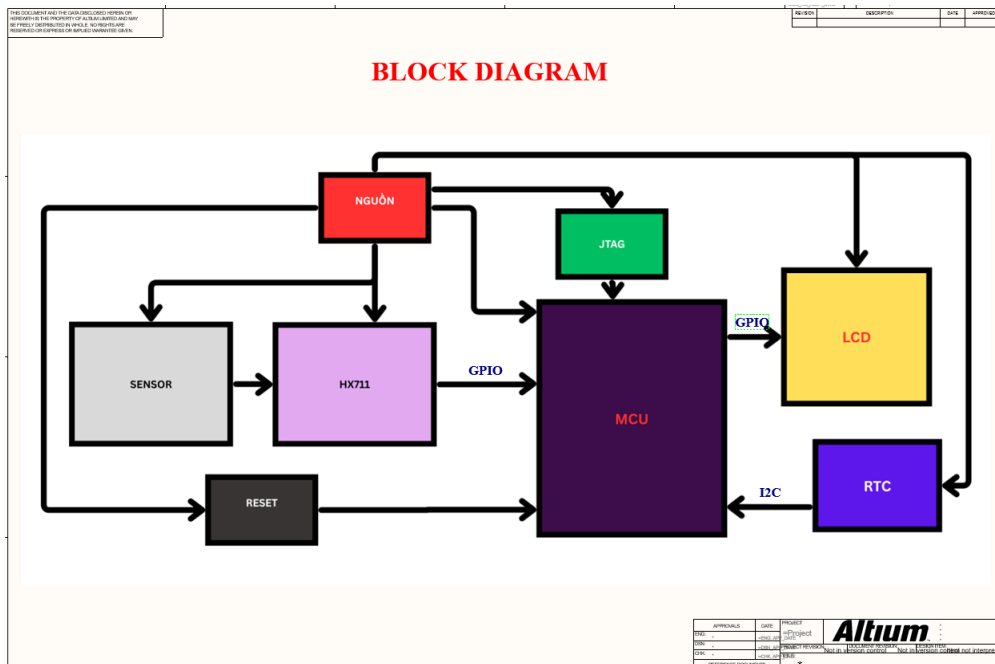


Figure 3: BLOCK DIAGRAM

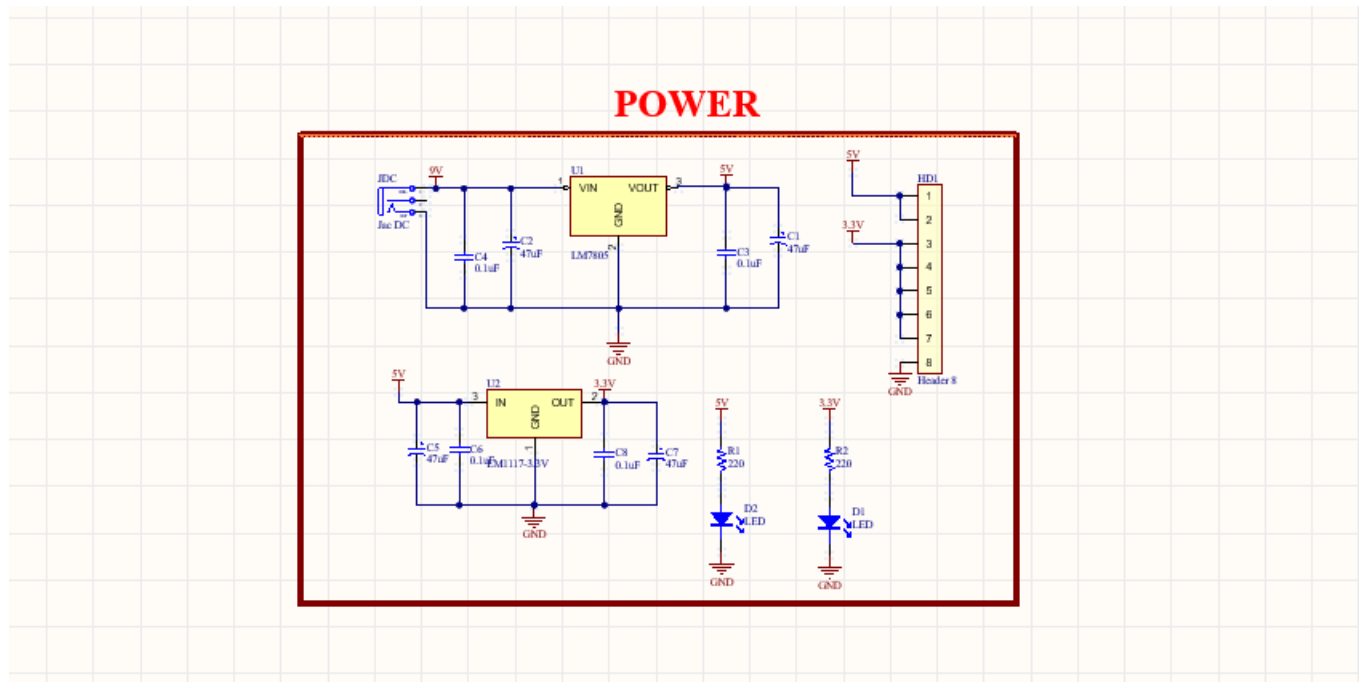


Figure 4: POWER

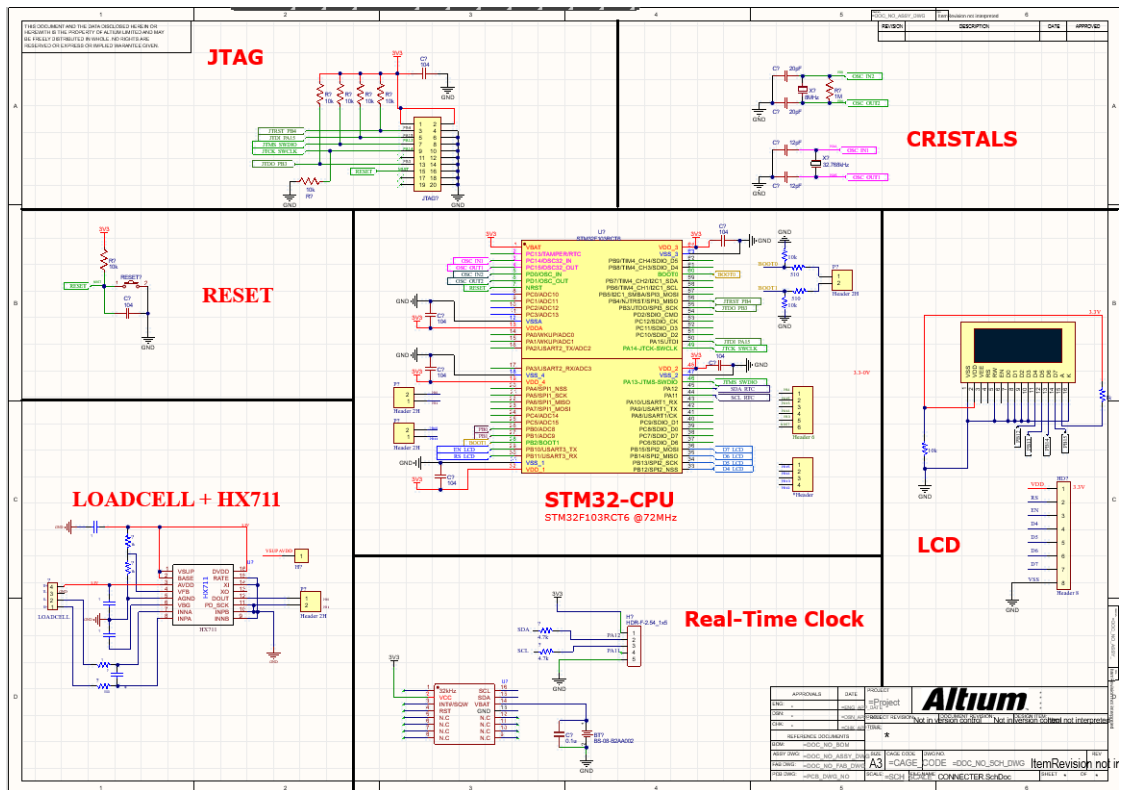


Figure 5: CONNECTER