**✅ M2 模組設計（產生 signal + 回測）**

**1️⃣ M2 功能說明（動態產生 signal + 支援策略驗證）**

|  |  |
| --- | --- |
| **功能** | **說明** |
| 🎯 根據 param\_id 自動重建 signal（即時計算） |  |
| 🧮 支援一次對多個策略參數進行回測 |  |
| 📁 可指定使用整個 param\_log，或使用 M3 精選清單 |  |
| 🧪 可切換為「訓練模式」或「驗證模式」，設定不同的回測期間 |  |
| 🧊 限流支援 | max\_workers + delay 控制 |
| 📤 輸出每組策略績效 + NAV |  |

**2️⃣ 使用者選單輸入（main\_controller）**

shell

複製編輯

【策略回測 M2】

請選擇要回測的策略來源：  
1. 全部 param\_log  
2. 精選策略清單（來自 M3）  
> 1

請選擇股票代碼（例如 AAPL）：  
> AAPL

請選擇策略類型（RSI、CROSS 等）：  
> RSI

請輸入回測時間範圍：  
起始日期：  
> 2023-01-01  
結束日期：  
> 2023-12-31

請輸入最大回測併發數（預設=3）：  
>

**3️⃣ 執行中與結束畫面**

**⏳ 執行中**

sql

複製編輯

➡️ 回測 RSI\_AAPL\_0001  
 ⮑ Sharpe=1.4, Return=12.8%, MaxDD=-11%  
 ⮑ 儲存至 NAV 與 performance 中...

**✅ 結束畫面**

css

複製編輯

📁 回測完成：  
✔️ NAV：nav/RSI\_AAPL\_0001.parquet  
✔️ 績效：performance/performance\_master.csv

**4️⃣ M2 讀取來源與欄位定義**

|  |  |
| --- | --- |
| **檔案來源** | **說明** |
| signals/param\_log\_\*.json | 所有參數清單 |
| signals/signal\_param\_map\_\*.json | param\_id 對應 |
| stock\_price.db | 提取對應區間股價 |
| M3 選出清單（可選） | selected\_strategies.json（含 param\_id + params） |

signal 是動態產生，不再讀取 signal.csv 檔案。

**5️⃣ M2 輸出定義**

|  |  |  |
| --- | --- | --- |
| **類型** | **路徑** | **說明** |
| NAV 資料 | nav/{strategy}\_{symbol}\_{param\_id}.parquet | 每筆對應淨值變化 |
| 績效彙總 | performance/{symbol}\_{strategy}\_YYYY-MM-DD\_to\_DD.csv | 每次跑結果獨立存檔 |
| 可選：統整報表 | performance/performance\_master.csv | 匯整全部股票／策略報告（可覆蓋或 append） |

**6️⃣ 程式碼模組結構建議**

bash

複製編輯

modules/  
├── m2\_backtester.py # 主流程（回測執行器）  
utils/  
├── backtest\_engine.py # 回測邏輯與 NAV 計算  
├── indicator\_utils.py # RSI、CROSS 訊號產生邏輯  
├── db\_loader.py # 從 DB 載入股價  
├── performance\_writer.py # 儲存績效/NAV  
├── param\_loader.py # 解析 param\_log, param\_map, selected\_strategies.json  
nav/  
├── RSI\_AAPL\_0001.parquet  
performance/  
├── RSI\_AAPL\_回測結果.csv