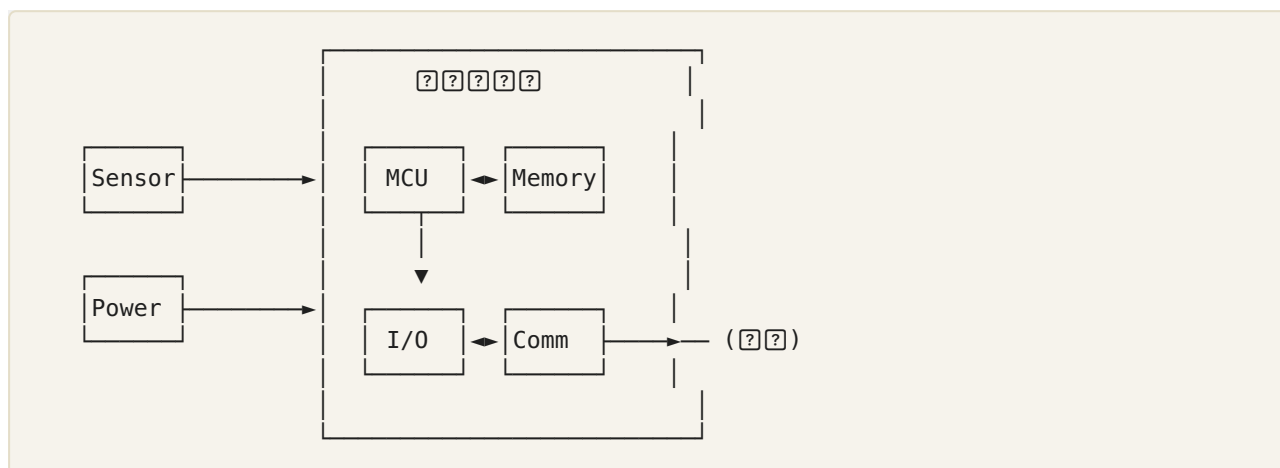


第 4 章 7：嵌入式系統

嵌入式系統是「硬體 + 韌體」的整體設計，現代電子產品 99% 都是嵌入式。

一、嵌入式系統組成



核心元件

1. MCU/MPU/SoC：運算核心
2. 記憶體：Flash + SRAM/DRAM
3. 電源：穩壓 + 管理
4. I/O：GPIO、ADC、PWM 等
5. 通訊：UART、SPI、I²C、USB、Ethernet
6. 時鐘：晶體 + PLL
7. 感測器：依應用

二、處理器分類

MCU (Microcontroller)

- 整合 CPU + Flash + RAM + 外設
- 低功耗、即時
- 應用：感測器、控制、IoT

主流 MCU 家族

家族	廠商	特色
AVR (8/16-bit)	Microchip (Atmel)	Arduino 用
PIC	Microchip	早期主流
8051	多家	經典 8-bit
MSP430	TI	超低功耗
ARM Cortex-M0/M3/M4/M7/M33/M55	ARM IP	業界主流
ARM Cortex-A	ARM IP	高階（手機應用）
RISC-V	開源	新興
ESP32	Espressif	Wi-Fi/BT 整合

經典 MCU IC

- STM32 (ST)：Cortex-M，廣泛應用
- NXP LPC / i.MX：工業 / 汽車
- TI MSP430 / TM4C
- Microchip PIC32 / SAM
- Renesas RA / RX / RL：日系
- Nordic nRF52/53：BLE 強項
- ESP32-S3 / C3：Wi-Fi + BT

MPU (Microprocessor)

- 沒有內建 Flash，需外接 DDR + Flash
- 高效能
- 跑 Linux / RTOS

範例：i.MX 6/8、Allwinner A33、Rockchip RK3399、樹莓派的 BCM2711

SoC (System on Chip)

- 整合 CPU + GPU + DSP + 各種週邊
- 行動 / IoT 主流
- Apple A 系列、高通 Snapdragon、聯發科

DSP (Digital Signal Processor)

- 專門處理訊號
- 應用：音訊、影像、通訊
- TI C6000、ADI SHARC

NPU / TPU (神經處理單元)

- AI 運算加速
- Google Edge TPU、寒武紀、地平線

三、ARM Cortex-M 系列

主流嵌入式 MCU 核心。

主要型號

核心	性能	用途
Cortex-M0	0.9 DMIPS/MHz	低成本、簡單
Cortex-M0+	1.0 DMIPS/MHz	低功耗
Cortex-M3	1.25 DMIPS/MHz	主流
Cortex-M4	1.25 DMIPS/MHz + FPU + DSP	訊號處理
Cortex-M7	2.14 DMIPS/MHz + cache	高性能
Cortex-M33	1.5 DMIPS/MHz + TrustZone	安全 IoT
Cortex-M55	4.5 DMIPS/MHz + Helium	AI

核心特性

- 32-bit RISC
- Thumb-2 指令集
- NVIC 中斷控制器
- SysTick
- DAP debug 介面

四、外設介面

GPIO（通用 I/O）

- 輸入：浮接、上拉、下拉、interrupt
- 輸出：推挽、開汲極
- 中斷觸發：上升、下降、雙邊、電平

UART / USART

- 異步串列通訊
- 點對點
- 速度：1200 bps – 數 Mbps
- 用於：偵錯、Modem、GPS

SPI（Serial Peripheral Interface）

- 同步、4 線（SCK、MOSI、MISO、CS）
- 主從架構
- 速度：MHz–100 MHz
- 用於：Flash、ADC、感測器

I²C（Inter-Integrated Circuit）

- 同步、2 線（SCL、SDA）
- 多主多從、地址尋址
- 速度：100k / 400k / 1M / 3.4M
- 用於：感測器、EEPROM、即時時鐘

CAN

- 工業/汽車匯流排
- 差動、抗噪
- 1 Mbps（Classic） / 8 Mbps（FD）

USB

- USB 1.1：12 Mbps
- USB 2.0：480 Mbps
- USB 3.x：5/10/20 Gbps

- USB-C/PD：100W+ 充電

Ethernet

- MAC + PHY
- 10/100/1000 Mbps
- TSN (Time-Sensitive Networking)

Wi-Fi / BLE / Zigbee / Thread

- 整合在 SoC 或外接模組

ADC / DAC

(已在 03-中級篇/10 介紹)

PWM / Timer

- 計時、馬達控制、PWM 輸出
- Capture / Compare

五、即時作業系統 (RTOS)

為什麼需要 RTOS？

裸機 (bare metal) 大型專案困難維護：- 多任務難協調 - 即時性難保證 - 程式碼複雜

常見 RTOS

RTOS	特色
FreeRTOS	開源、廣泛
Zephyr	Linux 基金會、模組化
RT-Thread	中國開源
ThreadX / Azure RTOS	微軟
μC/OS	商業，認證好
VxWorks	高可靠 (航太、軍用)
QNX	商業，汽車

核心概念

- Task / Thread
- 排程器 (preemptive priority)
- 同步：semaphore、mutex、queue
- 中斷管理
- 記憶體管理
- 計時器

範例：FreeRTOS

```
void task1(void *param) {
    while (1) {
        // 延遲
        vTaskDelay(pdMS_TO_TICKS(100));
    }
}

int main() {
    xTaskCreate(task1, "T1", 256, NULL, 1, NULL);
    vTaskStartScheduler();
    while (1);
}
```

六、嵌入式 Linux

為什麼用 Linux？

- 大量現成驅動與函式庫
- 網路功能完整
- 適合複雜應用

適合 Linux 的情境

- 計算量大 (影像、AI)
- 多通訊 (Wi-Fi、ETH、USB host)
- GUI (顯示器)
- 時間要求不嚴苛

不適合 Linux 的情境

- 真實即時 (μs 級)

- 超低功耗
- 開機極快 (< 100 ms)
- 簡單應用 (成本高)

Linux 嵌入式發行版

- Yocto Project / OpenEmbedded
- Buildroot
- Ubuntu Core
- 廠商專用 (如 i.MX BSP)

七、嵌入式硬體設計考慮

1. MCU 選用

評估： - 性能 (MHz、MIPS) - 記憶體 (Flash、RAM 大小) - 外設 (哪些介面、多少通道) - 功耗
- 成本 - 工具鏈成熟度 - 社群支援

2. 開發板原型

- Arduino (簡易)
- ESP32 / Raspberry Pi (中高)
- Nucleo / Discovery / EvalBoard (廠商)

3. PCB 設計

- 振盪器靠近 MCU
- 去耦電容每腳一顆
- 偵錯介面 (SWD、JTAG)
- USB 連接器 (如有)
- 重置按鈕、開機選擇腳

4. 偵錯介面

- SWD (Serial Wire Debug)：2 線，現代主流
- JTAG：4-5 線，早期
- UART console：簡單但有限

偵錯器： - ST-Link (ST 系列) - J-Link (廣泛支援，Segger) - Keil ULINK - Black Magic Probe (開源)

5. 韌體更新

- ISP (In-System Programming)：開發
- Bootloader 更新：在系統運行時更新
- OTA (Over-the-Air)：遠端更新

八、感測器整合

常見感測器

類型	範例 IC
溫度	LM35 (類比)、DS18B20 (1-wire)、TMP102 (I ² C)
濕度	DHT22、SHT3x、BME280
壓力	BMP180、BME280
加速度	ADXL345、MPU6050
陀螺儀	L3GD20、ICM-20948 (9 軸)
磁力	HMC5883L
光	TSL2591、APDS-9960
距離	VL53L0X、HC-SR04 (超聲)
心率	MAX30100 / 30102
氣體	MQ 系列、SGP30
GPS	u-blox NEO-6M / 8M / 9M
影像	OV2640、HM01B0
聲音	MAX9814、INMP441 (I ² S)

介面選擇

- 簡單：類比 (ADC)
- 標準：I²C、SPI

- 高速資料：SPI、I²S、Parallel、MIPI CSI

九、低功耗設計

MCU 功耗模式

模式	範例 (STM32L4)	喚醒源
Run	12 MHz, 100 μ A/MHz	—
Sleep	CPU off, 外設 on	IRQ
Stop	RAM 保留, 外設 off	RTC, EXTI
Standby	大部分斷電	RTC, Pin
Shutdown	全斷電	Pin

設計策略

1. 大部分時間 deep sleep
2. 用 RTC 或外部 IRQ 喚醒
3. 縮短 active 時間
4. 用 DMA 減少 CPU 時間
5. 外設 clock gating
6. 降低時鐘頻率

量測

- 微安級電流量測： μ Current Gold、Joulescope、PPK2

十、安全 (Security)

威脅

- 韌體竊取
- 資料竊取
- 篡改
- 重播攻擊

防護

- Secure Boot：驗證簽章
- Encrypted Firmware
- TrustZone (Cortex-M33+)
- Hardware Crypto Engine (AES、SHA、RSA)
- TRNG (真隨機亂數)
- Secure Element (如 ATECC608、SE050)

安全 MCU

- ST STM32H5 / U5 / WBA
- NXP LPC55S
- Microchip CryptoAuthentication
- Renesas RA6M5

十一、本章總結

✅ 您應該掌握：

- [x] 嵌入式系統組成
- [x] MCU/MPU/SoC 區別
- [x] ARM Cortex-M 系列
- [x] 主要外設介面 (UART、SPI、I²C、USB、CAN)
- [x] RTOS 基本概念
- [x] 嵌入式 Linux 場景
- [x] 偵錯介面與工具
- [x] 感測器整合
- [x] 低功耗設計
- [x] 安全設計

下一章

08-FPGA[?]?.md 一 FPGA 設計入門。