

# 第 9 章 嵌入式作業系統

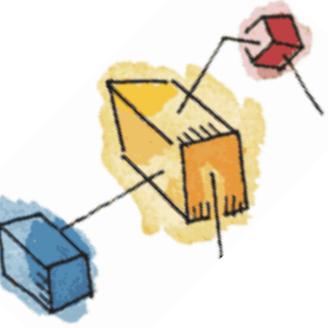


# 本章提要

- 9.1 嵌入式系統
- 9.2 嵌入式作業系統的特色
- 9.3 eCos
- 9.4 TinyOS

# 嵌入式系統

- 結合某些硬體或元件，設計用來共同執行某項專門的功能的系統
- 嵌入式系統的數量遠超過通用用途的電腦系統，而且涵蓋的應用程式很廣泛
- 嵌入式系統與使用環境是緊密耦合的



# 嵌入式系統實例

表 9.1 嵌入式系統實例與其市場 [NOER05]

市場別	嵌入式裝置
汽車工業	啟動系統 引擎控制 煞車系統
消費性電子產品	數位與類比電視 機上盒 (DVD、VCR、有線電視盒) PDA (個人數位助理, Personal digital assistants) 廚房家電 (電冰箱、烤麵包機、微波爐、烤箱) 汽車 玩具 / 電動遊戲 電話 / 手機 / 呼叫器 相機 全球定位系統 (GPS)
工業控制	製造業的機器人和控制系統 感測器
醫療業	注入唧筒 洗腎機 義肢裝置 心臟監視系統
辦公室自動化	傳真機 影印機 印表機 螢幕 掃描器

# 嵌入式系統架構

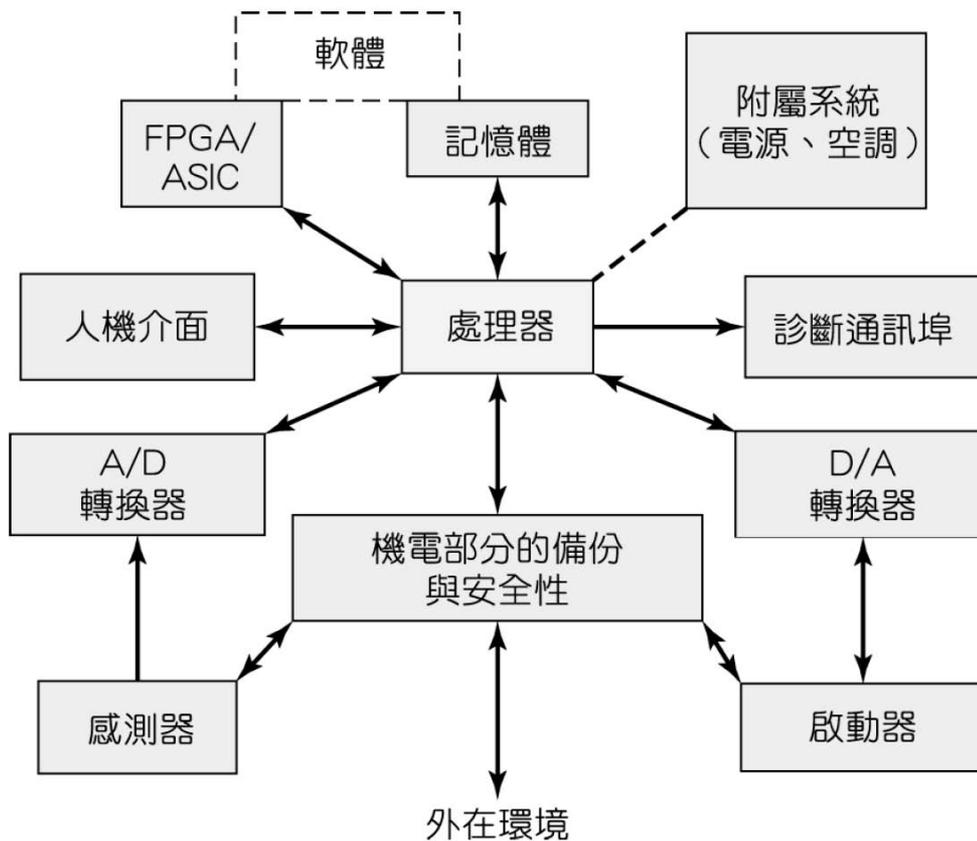
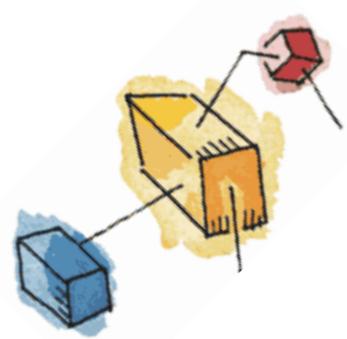


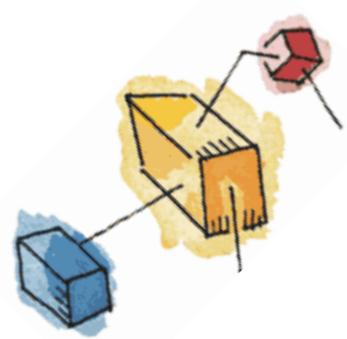
圖 9.1 嵌入式系統可能的架構範例



# 嵌入式作業系統的特色

- 即時運作
- 反應快速
  - 對外部事件做出回應
- 可設定組態
  - 由於嵌入式系統種類眾多，因此嵌入式**OS**需要有彈性的組態

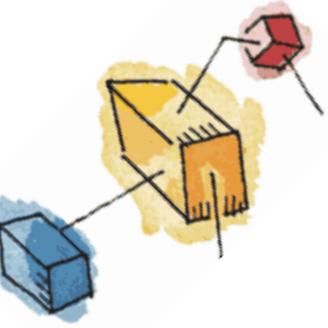




# 嵌入式作業系統的特色

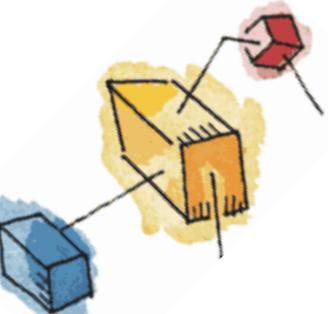
- I/O裝置的彈性
- 串流式保護機制
- 直接使用中斷





# 改造現有的OS

- 增加即時能力
- 將運算串流化
- 加入必要的功能



# eCos

- EEmbedded Configurable Operating Systems的縮寫
- 開放原始碼
- 免授權費用
- 即時OS
- 最多人使用的嵌入式OS



# eCos的組態設定工具

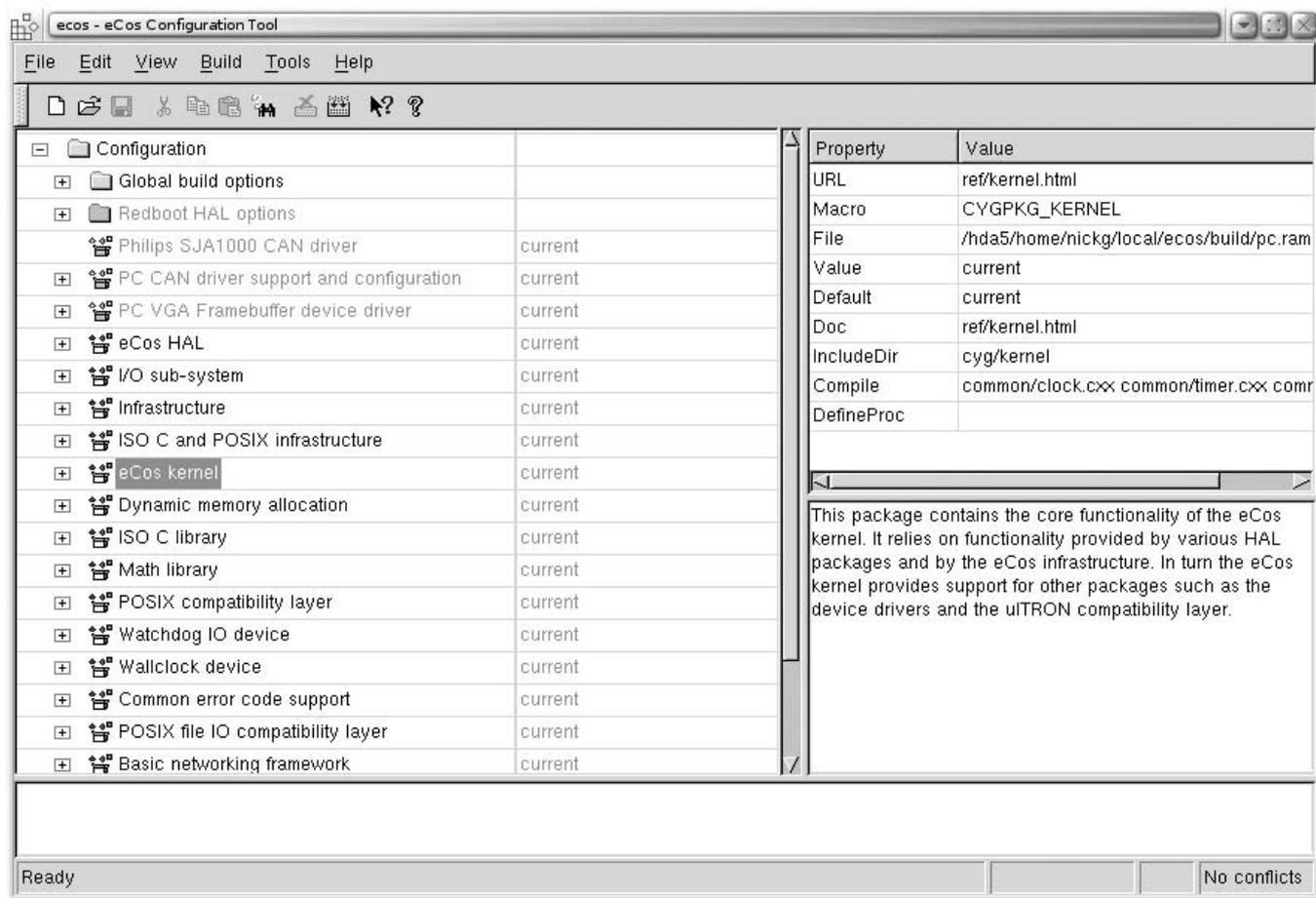


圖 9.2 eCos 的組態設定工具 ( Configuration Tool ) — 最上層

# eCos的組態設定工具

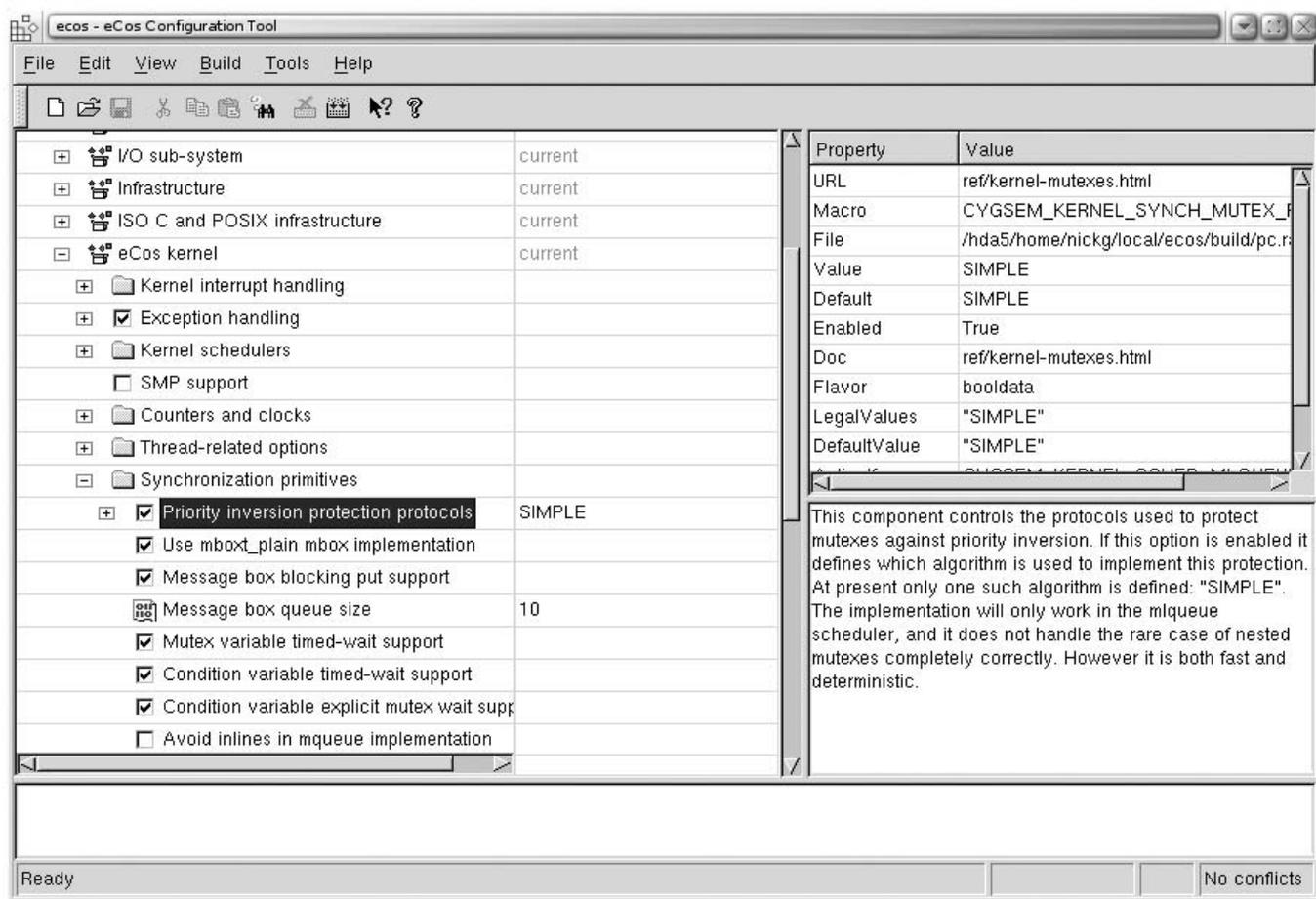


圖 9.3 eCos 的組態設定工具 — Kernel 的詳細資訊

# 載入一組eCos組態設定值

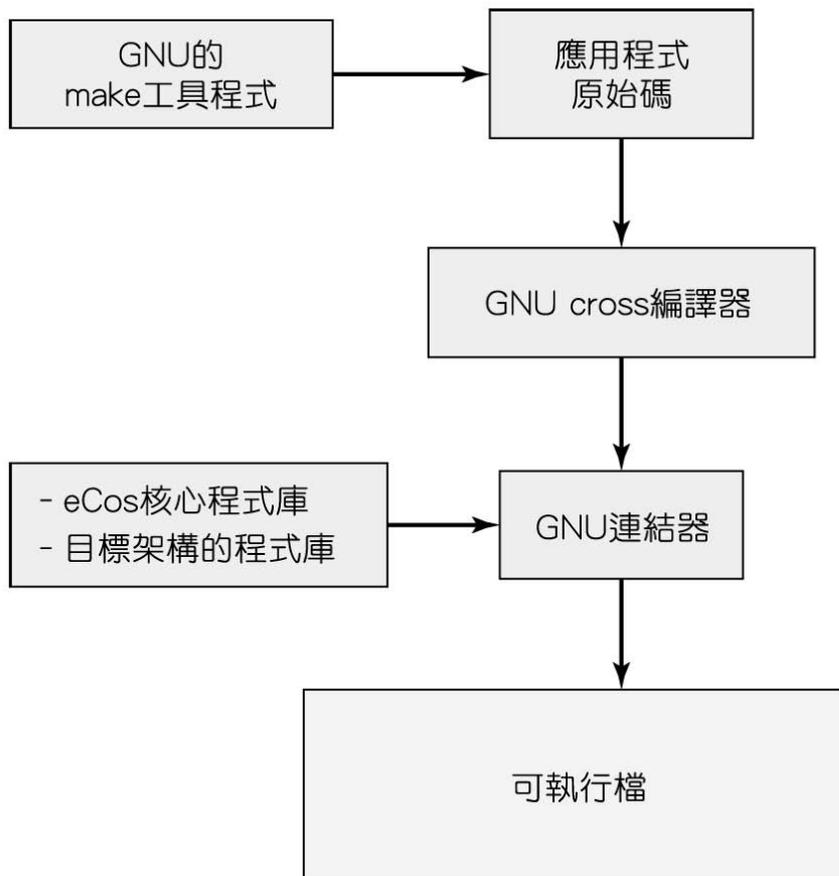
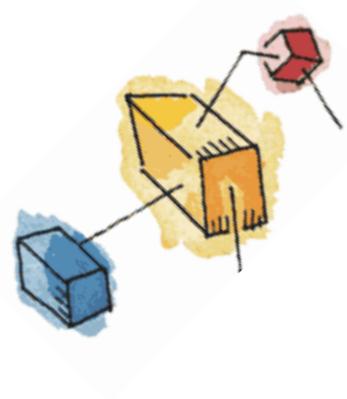


圖 9.4 載入一組 eCos 組態設定值



# eCos的層級架構

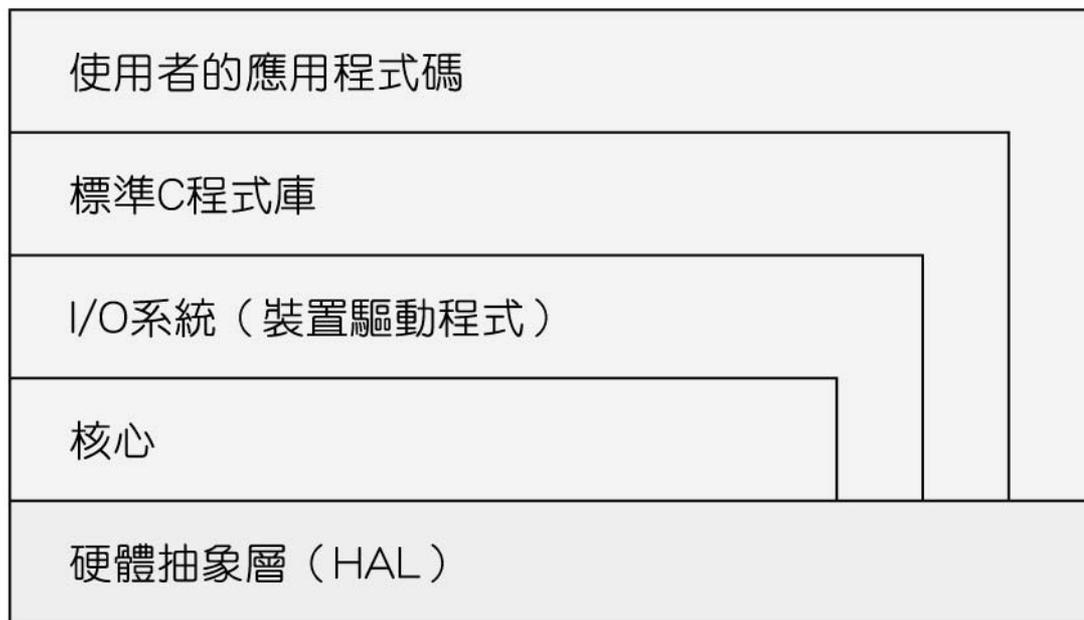
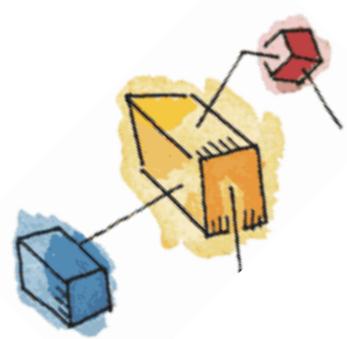


圖 9.5 eCos 的層級架構





# 硬體抽象層 (HAL)

- 對上層顯示出一致的API
- 每種硬體平台都不同



# HAL

```
1 #define HAL_ENABLE_INTERRUPTS() \
2   asm volatile ( \
3     "mrs r3, cpsr;" \
4     "bic r3, r3, #0xC0;" \
5     "mrs cpsr, r3;" \
6     : \
7     : \
8     : "r3" \
9   );
```

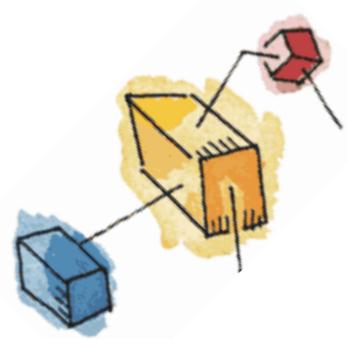
(a) ARM架構

```
1 #define HAL_ENABLE_INTERRUPTS() \
2   CYG_MACRO_START \
3   cyg_uint32 tmp1, tmp2 \
4   asm volatile ( \
5     "mfmsr %0;" \
6     "ori %1,%1,0x800;" \
7     "rlwimi %0,%1,0,16,16;" \
8     "mtmsr %0;" \
9     : "=r" (tmp1), "=r" (tmp2)); \
10  CYG_MACRO_END
```

(b) PowerPC架構

圖 9.6 Hal\_Enable\_Interrupts() 巨集的兩種實作版本

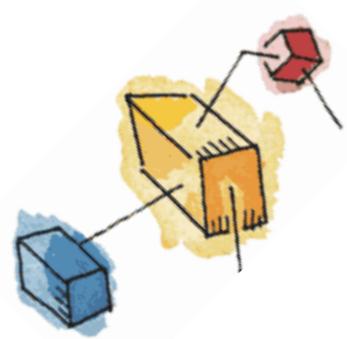




# HAL模組

- **Architecture (架構)**
  - 負責定義處理器系列的種類
- **Variant (變化版本)**
  - 支援系列中特定處理器的功能
- **Platform (平台)**
  - 擴充HAL支援緊密耦合的周邊裝置

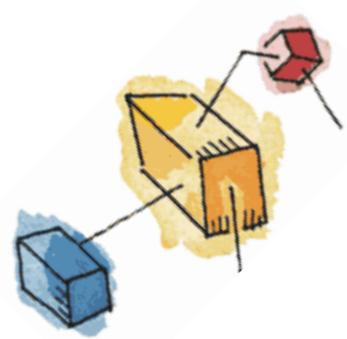




# eCos核心

- 中斷延遲要短
- 工作切換延遲要快
- 所需記憶體要少
- 行為確定性高





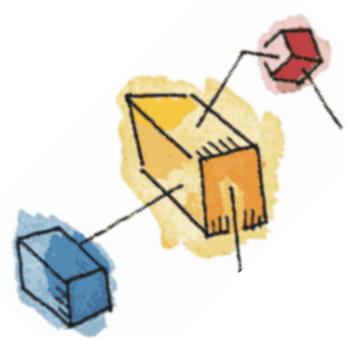
# 不在eCos核心的功能

- 記憶體配置
- 裝置驅動程式



# TinyOS

- 總共只需要**400**個位元組的記憶體
- 不是即時的**OS**
- 沒有核心
- 沒有記憶體保護功能



# Tiny OS

- 是一種以元件為主的OS
- 沒有行程
- 沒有記憶體配置系統
- 中斷和例外處理與周邊裝置是獨立的
- 完全非懸置的系統



# 無線感測器網路拓樸

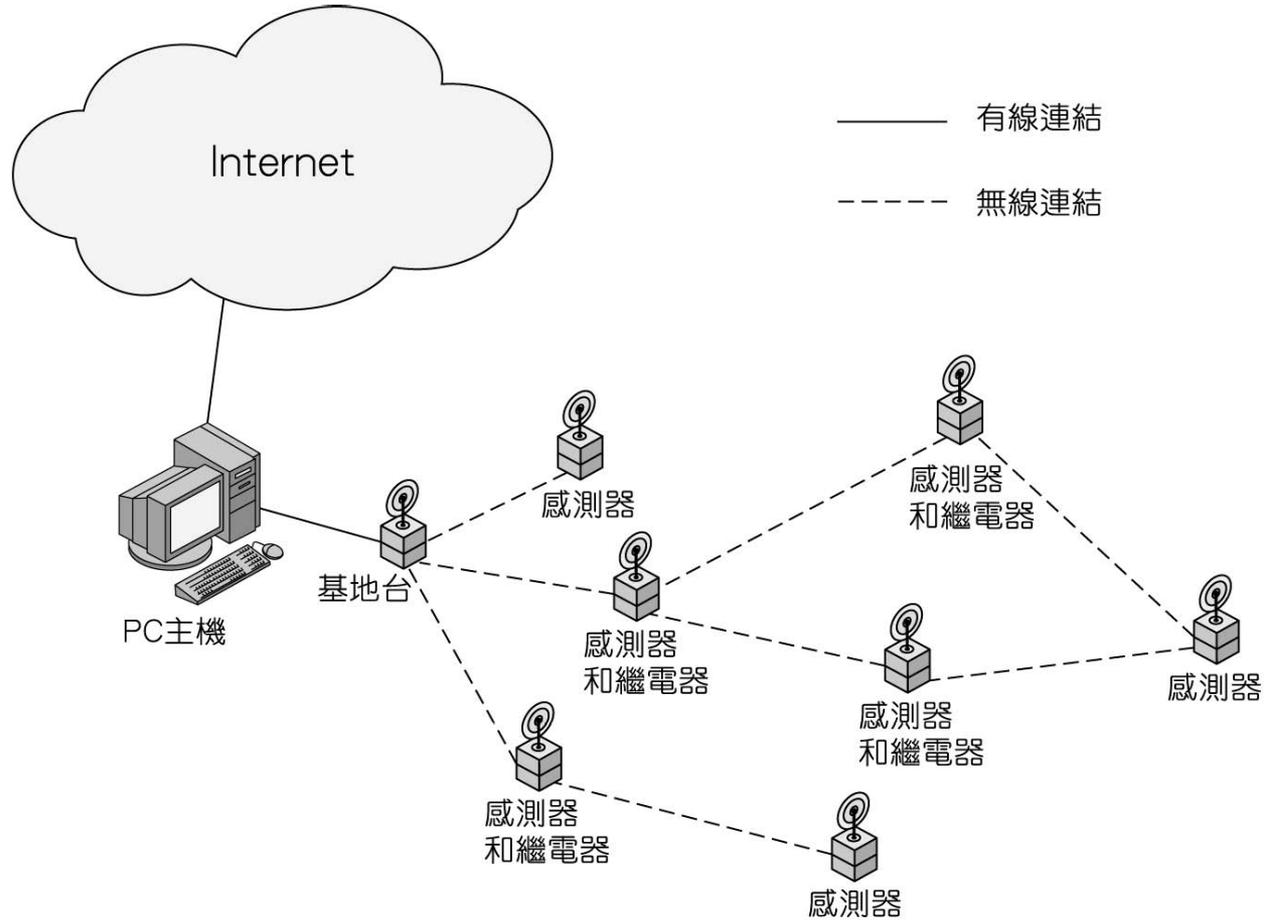
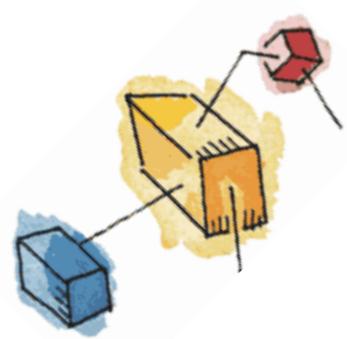


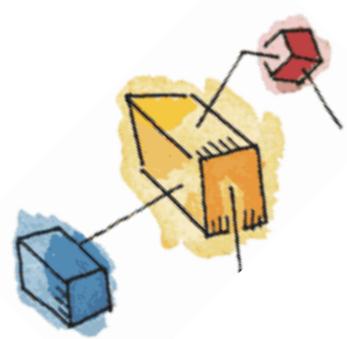
圖 9.10 典型無線感測器網路的拓樸



# TinyOS的目標

- 具有高並行性
- 能在有限資源下運作
- 能適應硬體的演進
- 支援廣泛種類的應用程式
- 支援各種不同的平台
- 要穩定





# TinyOS的元件

- Single-hop網路功能
- Ad-hoc路由功能
- 電源管理
- 計時器
- 非揮發性儲存體的控制

