**Chap02**

**1. 請回答以下以Keil C51開發實驗版軟體環境使用問題。請將Chap 2課程範例Led.c為例,建置專案.**

**(1)在Debug(偵錯)模式下，要看資料記憶體(RAM) 0x00位址要如何達成? (請放截圖)**

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片  AI 產生的內容可能不正確。 |
| Start/Stop Debug Session(Ctrl+F5) |

(2)在Debug(偵錯)模式下，要看程式記憶體(ROM) 0x0010位址要如何達成? (請放截圖)

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片  AI 產生的內容可能不正確。 |
| Start/Stop Debug Session(Ctrl+F5) |

(3)若要在程式記憶體中找到程式計數器(PC)指向的記憶體要如何達成?並如何對應反組譯(Disassembly)確認正確性?(請放截圖)

|  |
| --- |
| 一張含有 文字, 螢幕擷取畫面, 軟體, 行 的圖片  AI 產生的內容可能不正確。 |
| 在 Disassembly Window 中，程式計數器 (PC) 的當前位置會以黃色高亮顯示，您可以檢查該位置的反組譯碼是否與程式碼一致。Memory Window 會顯示指定地址的記憶體內容，您可以檢查程式碼是否正確載入到該地址。 |

**(4)在Debug(偵錯)模式下，請問單步偵錯中Step Out是什麼?**

* 當執行到一行包含函數呼叫的程式碼時，Step In 會進入該函數內，讓您可以逐步檢視函數的內部執行過程。這對於需要深入了解函數內部邏輯時非常有用。

**(5)在Debug(偵錯)模式下，請問單步偵錯中Step in，Step Over的不同?**

* 當執行到一行包含函數呼叫的程式碼時，Step Over 會直接執行整個函數，而不進入該函數內。執行完函數後，程式計數器會移到下一行程式碼。這對於已知函數內部邏輯且無需檢查其執行過程時很有幫助。

**(6) )在Debug(偵錯)模式下，請加入 P2, i至local watch. 並顯示P2於記憶體位置(請放截圖)**

|  |
| --- |
| 一張含有 文字, 軟體, 螢幕擷取畫面, 行 的圖片  AI 產生的內容可能不正確。 |
| * **開啟 Local Watch 窗口**： * 在 Debug 模式下，從工具列選擇 View -> Watch Windows -> Local。 * 或者直接打開 Watch 1 窗口也可用來檢視變數。 * **加入變數**： * 在 Local Watch 中，找到 P2 和 i，右鍵點擊，選擇 "Add to Watch". * 如果變數未自動顯示，您可以手動輸入變數名稱（如：P2 和 i） |

**(7)在Debug(偵錯)模式下，請將P2.0和P2.1的邏輯分析儀(logical analyzer)模擬顯示(請放截圖)**

|  |
| --- |
| 一張含有 文字, 行, 字型, 繪圖 的圖片  AI 產生的內容可能不正確。 |
| * **啟用 Logical Analyzer**： * 在 Debug 模式下，從工具列選擇 View -> Analysis Window -> Logic Analyzer。 * **新增信號**： * 在邏輯分析儀的窗口中，右鍵點擊以開啟選單，選擇 Setup. * 在 Setup 中手動輸入 P2.0 和 P2.1 作為要觀察的信號。 * 配置時間軸的參數，例如取樣速率和顯示範圍。 |