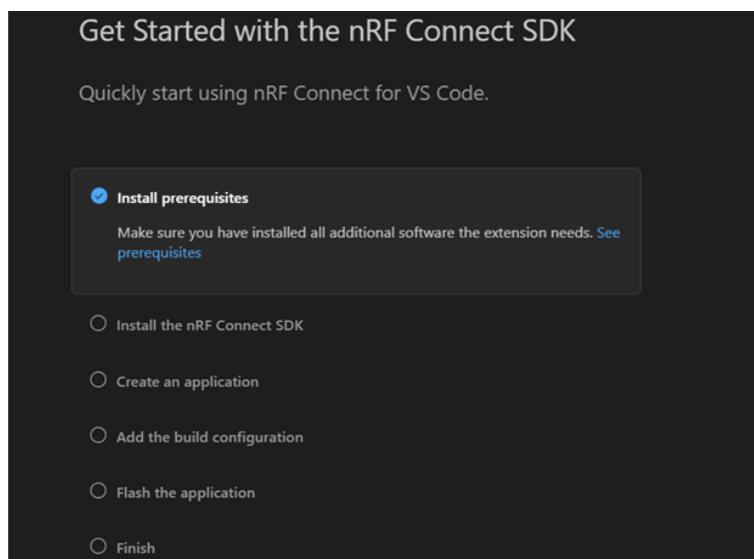


Zephyr OS開發工作事項: 安裝環境、範例專案導入、編譯與燒錄的詳細流程，適合從零開始學習使用nRF54L15與Zephyr OS.

整個全部流程結合了nRF Connect SDK與Zephyr RTOS，nRF54L15系列的SDK版本需要使用NCS 2.7.0或以上，以支援新的硬體架構與功能。(選nRF Connect SDK - 3.1.0))

安裝nRF Connect SDK開發環境

1. 先安裝nRF命令行工具（nRF Command Line Tools），包括SEGGER J-Link驅動，方便燒錄和調試。
2. 安裝Visual Studio Code（VS Code），作為開發IDE。
3. 在VS Code擴展商店安裝「nRF Connect for VS Code」擴展套件，包含nRF Connect SDK安裝、管理和開發工具。



2. 在「nRF Connect for VS Code」擴展中，點擊「Manage SDKs」選項，選擇安裝最新版本的nRF Connect SDK（NCS）。安裝時建議安裝近根目錄的路徑且路徑不宜太長，以避免路徑錯誤。

3. 下載並安裝適合的Toolchain（編譯工具），通常nRF Connect SDK會自動建議匹配版本。

Work Log:

C:\nRF\nsc3.1.0

C:\nRF\toolchain

C:\ncs\v3.1.0

C:\ncs\toolchains

選擇並配置nRF54L15開發板專案

(BTW, 建議你們先拿 nRF54L15 EVK 來 porting)

1. 在VS Code開啟nRF Connect SDK資料夾，定位到範例工程，如 `nrf/samples/bluetooth/peripheral_uart` 等。
2. 點擊「Add Build Configuration」，在板子列表中選擇nRF54L15開發板目標，如 `nrf54l15dk_nrf54l15_cpuapp`，並確認SDK版本與Toolchain版本。
3. 可選擇預設的 `prj.conf` 設定檔或自訂配置。
4. 設定完成後，執行Build開始編譯專案。

(範例工程，如 `C:/ncs/v3.1.0/nrf/samples/bluetooth/peripheral_uart`)

編譯與下載至開發板運行

1. 編譯成功後，使用USB Type-C線連接開發板的Debugger USB口與PC，不需其他連線。
2. 在VS Code「ACTIONS」面板中點擊「Flash」按鈕，將編譯出的韌體燒錄至nRF54L15開發板。
3. 透過串口監控（UART）可查看韌體運行日誌，進行Debug與驗證。

這整個流程結合了nRF Connect SDK與Zephyr RTOS (並選用nRF54L15系列的SDK版本.)

[以上步驟已包含安裝環境、範例專案導入、編譯與燒錄的詳細流程，適合從零開始學習使用nRF54L15與Zephyr OS]

(用指令重啟nRF裝置: to be done)

[west flash --skip-rebuild -r jlink]