

第四章 系統軟體設計—系統初始化

本系統的軟體設計包含了三部份—系統初始化、軟體設計概念及嵌入式電腦語音溝通系統和替代性輸入裝置之軟體功能。本章將介紹嵌入式軟體發展套件，如何對本系統之作業系統、周邊設備之驅動程式等進行初始化動作。

首先我們採用了微軟 Microsoft[®] 所提供的 Windows CE.NET 5.0 軟體發展套件，其中包含了 PB (Platform Builder) 5.0 和 EVC++ 4.0 軟體。為了完成作業系統映像檔(Image:nk.bin)的製作，我們選用了 PB 5.0，而所需要之步驟有以下三大部份—產生映像檔平台(Creating a Platform Image)、初始化並建立映像檔(Customize and build the Platform)及映像檔傳輸至機器(Download Image to the Platform)[19]。系統軟體架構圖如圖 4-1 所示。

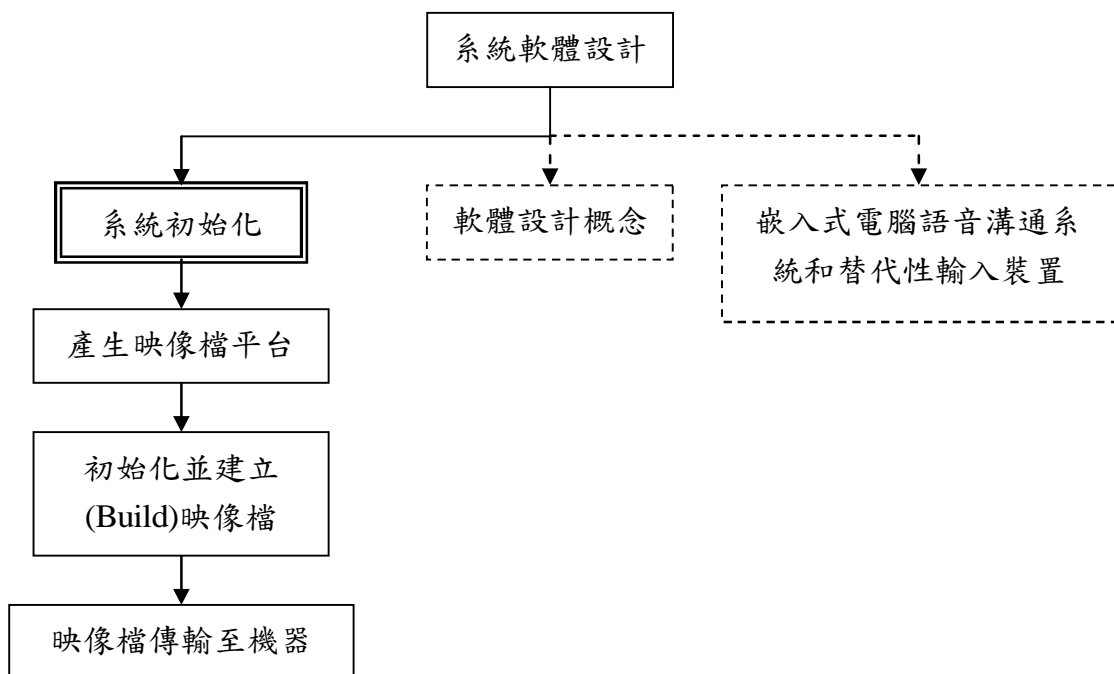


圖 4-1 系統軟體設計之系統初始化架構圖

第一節 系統初始化—產生映像檔平台

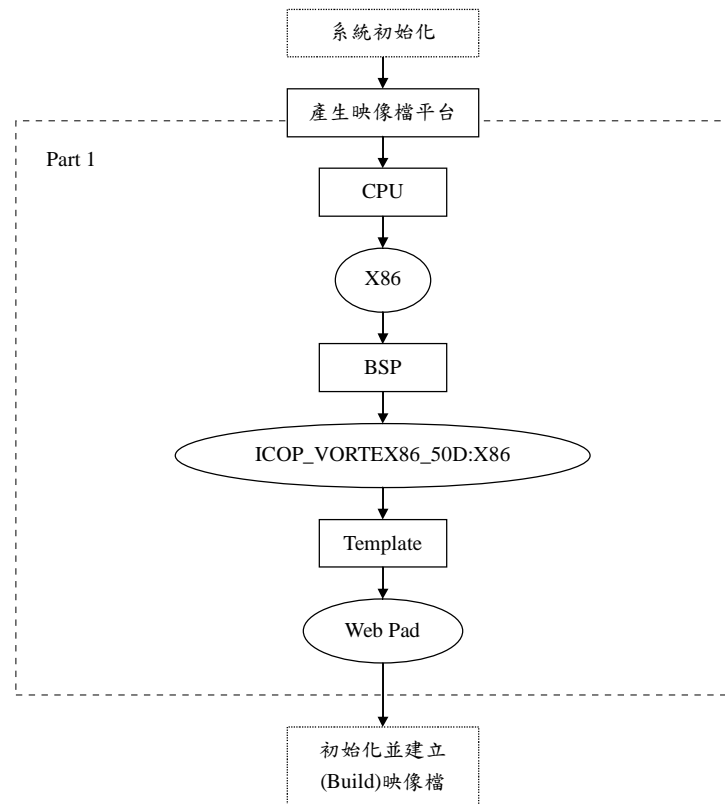


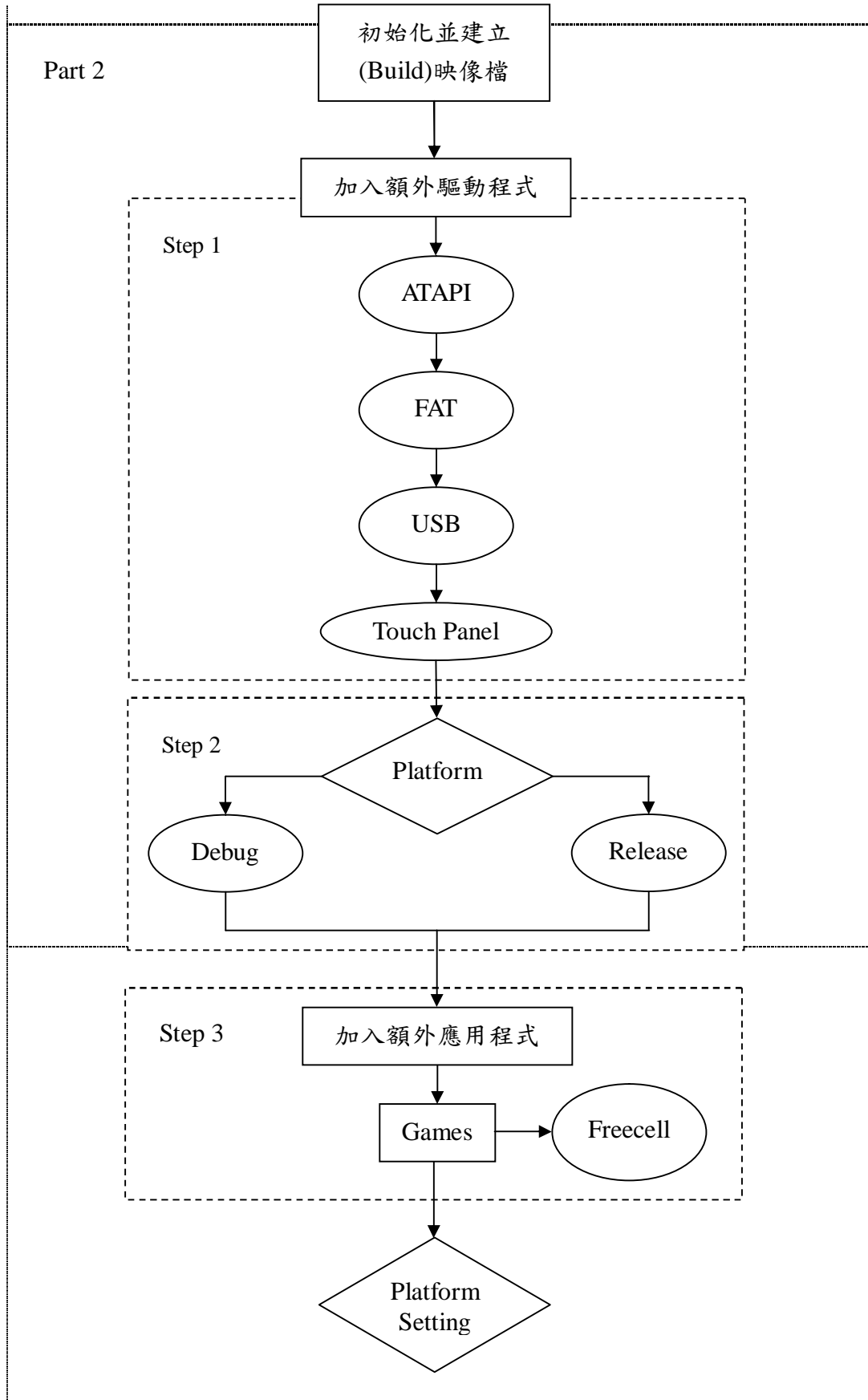
圖 4-2 產生映像檔平台流程圖

為了要建構我們所希望的掌上型作業系統，因此利用系統發展軟體 PB 5.0 所提供之精靈功能進行研發。圖 4-2 展現了如何產生映像檔平台的流程。首先將 PB5.0 系統發展軟體灌入電腦，需要約 2G 的硬碟空間，當完成後，即可開始進入產生映像檔平台的步驟了。我們需選擇合適的 CPU，不同之嵌入式系統支援不同版本的 CPU，例如 ARMV4、SH4、MIPS16、x86 等等。由於本系統之 CPU 是以 x86 系列為主，因此我們選擇了模擬器(Emulator)版本，以及 x86 版本的 CPU。選擇模擬器主要原因是在發展系統過程中，透過模擬器我們可以利用桌上

型電腦對嵌入式系統之作業系統或其軟體進行直接的測試和除錯，而不需每次都必須將映像檔或軟體傳送到機器上運行才知道它們出錯的部份。當測試完成後，再透過選擇 x86 系列的 CPU，依據 Emulator 所選擇正確的選項重新將作業系統映像檔建構完成並傳輸到機器上。

當選擇了正確的 CPU 後，則進入到選擇 BSP(Board Support Package)的階段。市面上常見的嵌入式 BSP(以 X86 系列為主)有 AMD GEODE:X86 系列、CEPC:X86 系列、ICOP_VORTEX86_50D:X86 等。BSP 主要包含了不同的硬體驅動程式，而我們選擇了 ICOP_VORTEX86_50D BSP 作為我們系統的主要硬體驅動程式來源。接著應該選擇合適的系統模組(Template)。PB 5.0 提供了多種不同的模組，例如 Custom Device、Digital Media Receiver、Web Pad、Internet Appliance 等。不同的系統模組提供不同的功能，例如 Web Pad 提供了觸控式顯示和無線網路的功能，也提供類似 IE(Internet Explorer)、Windows Messenger、Windows Media Audio/Video/MP3 等軟體，而若選擇 Custom Device，則可自主的選擇所需之軟體，但對於不熟悉的新手而言，不建議選擇此選項。選擇了 Web Pad 模組後，就會進入到初始化並建立映像檔平台的階段。

第二節 系統初始化—初始化並建立(Build)映像檔



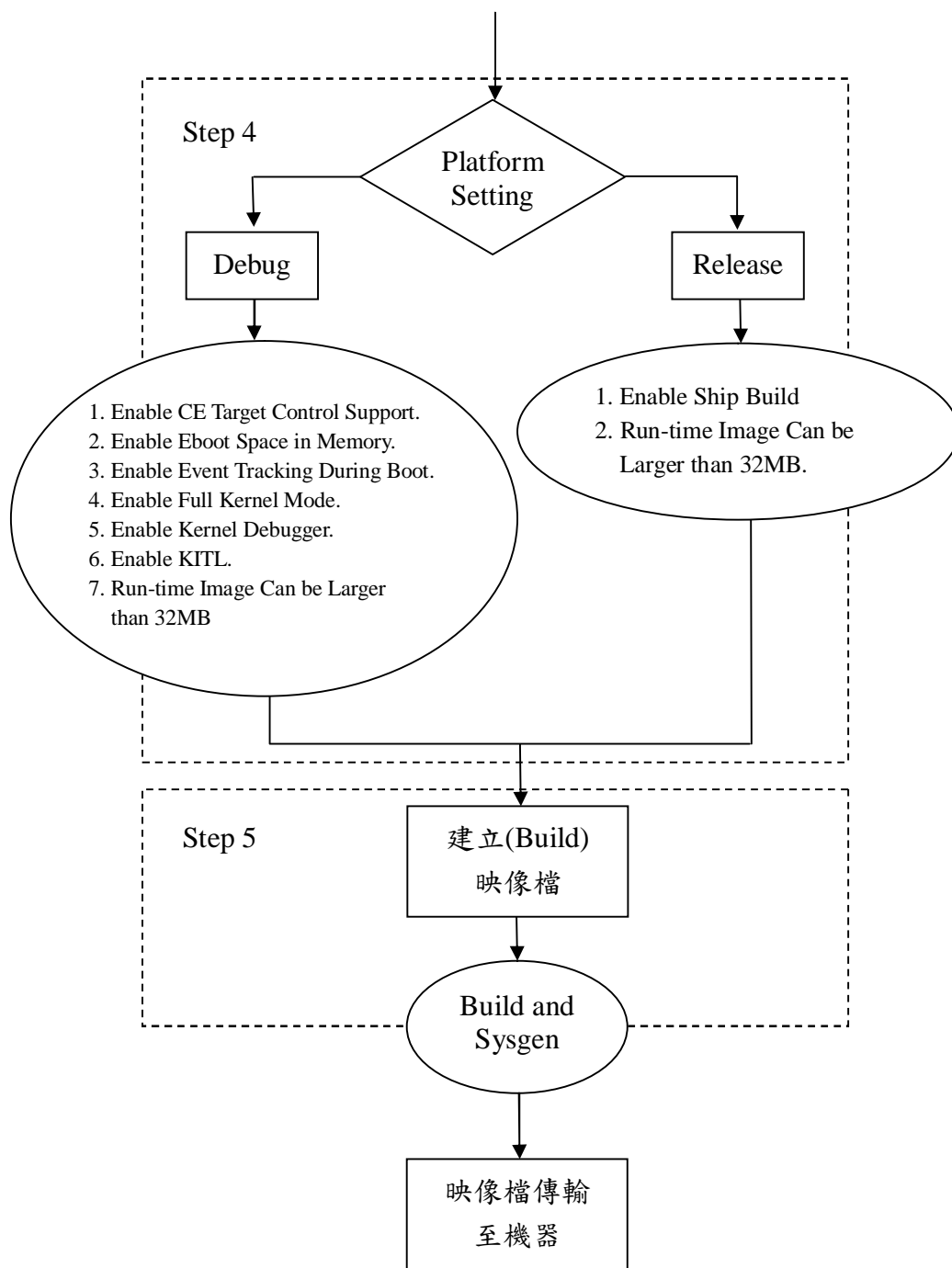


圖 4-3 初始化並建立(Build)映像檔流程圖

在第二部份的初始化並建立映像檔中，共需以下五個步驟來完成：

- 一、加入額外驅動程式(Add Additional Device Driver to Platform)、
- 二、平台選擇(Select Platform-Debug/Release)、
- 三、加入額外應用程式(Adding Applications to the Platform)、

四、平台設定(Additional Platform Settings)、

五、建立映像檔(Build the Platform)。

一、加入額外驅動程式(Add Additional Device Driver to Platform)

當我們在第一部份選擇 BSP 時雖然提供了多種硬體的驅動程式，但對於有些硬體仍然需要手動加入驅動程式才能完成。以嵌入式小硬碟(DOM)為例，我們需要加入 ATAPI PCI/IDE Storage Block Driver，接著我們就需要有 FAT File System 來讀取檔案系統。由於 USB 規格已經成為現時主要外接式物件規格的潮流，因此我們也加入了 USB Storage Class Driver。有了這個驅動程式，我們則可透過機器(嵌入式系統)輕易的來對 USB 隨身碟等物件進行存取的动作，增加了其便捷性。除此之外，對於驅動觸控面板，我們則需要額外在新建的平台(workspace)固定檔案裡加上合適的程式。其中在 workspace 的 project.reg 中需加入如表 4-1 的程式片段[18]：

表 4-1 project.reg 中加入觸控面板驅動程式之方式

```
[HKEY_LOCAL_MACHINE\Drivers\TouchKit]
    "RBSIZE"=dword:800
    "RBTIME"=dword:989680
    "FLAGS"=dword:1
    "BeepFrequency"=dword:1F4
    "BeepTime"=dword:64
    "ZFilterThreshold"=dword:800
    "ZFilterBound"=dword:00100001
    "SoundType"=dword:0
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\14371_1\0_0_0\255_255_255\USB_TOUCH_Driver]
    "DLL"="USBTouch.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\14371_2\0_0_0\255_255_255\USB_TOUCH_Driver]
    "DLL"="USBTouch.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\4660_1\0_0_0\255_255_255\USB_TOUCH_Driver]
    "DLL"="USBTouch.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\3823_1\0_0_0\255_255_255\USB_TOUCH_Driver]
    "DLL"="USBTouch.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\3823_2\0_0_0\255_255_255\USB_TOUCH_Driver]
    "DLL"="USBTouch.dll"
```

Project.reg 檔案主要是定義硬體註冊設定(hardware registry settings)，其變數

定義如表 4-2：

表 4-2 project.reg 檔案變數定義

[KEY1]
“ValueName1”={Value Type}:{data}

因此，[HKEY_LOCAL_MACHINE\Drivers\TouchKit]為 key，等號左邊雙掛號內("RBSIZE")為此 key 多個變數中的其中一變數名稱，等號右邊冒號左邊則為此變數的資料結構，右邊則為此變數之值。Reg 檔案裡的變數資料結構可分為 5 種，有字串(string)、字組(DWORD)、多字串(multi_string)、二進位(binary)以及十六進位(hex)[24]。

除了 project.reg 檔案需要加入程式片段以外，project.bib(Binary Image Builder) 檔案也需加入程式片段。Project.bib 主要是定義作業系統映像檔的模組(modules) 和檔案進入點(file entries)，例如定義不同的.exe 檔案和音樂(.wave)檔案等。表 4-3 為額外加入之程式片段[19]，其檔案路徑(path)可自行變更。

表 4-3 project.bib 中加入觸控面板驅動程式之方式

; Name	Path	Memory	Type
=====	=====	=====	=====
TouchKit.exe	c:\CAL\TouchKit.exe	NK	S
DrawTest.exe	c:\CAL\drawtest.exe	NK	S
Calbration.exe	c:\CAL\calbration.exe	NK	S
UpdateEEPROM.exe	c:\CAL\UpdateEEPROM.exe	NK	S
USBTouch.dll	c:\CAL\UsbTouch.dll	NK	SH

以上兩個加入程式的步驟完成後，觸控面板就可以使用了。當觸控的驅動程式完成後，則進入第二步驟的平台選擇(Select Platform-Debug/Release)。

二、平台選擇(Select Platform-Debug/Release)

PB 5.0 提供兩種平台讓我們做選擇，其中一種為測試版(Debug)，另外一種為開放版(Release)。對於在發展系統、進行測試的人們而言，測試版是好的選擇，因為可以輕易的透過網路的傳輸而將建立好的映像檔傳送到機器上進行測試。至於測試成功後，需存入嵌入式小硬碟作為永久使用的映像檔而言，開放版則是合適的選擇。當我們確認測試版已經無誤後，就可選擇開放版而將映像檔放入至建好的嵌入式小硬碟中。

三、加入額外應用程式(Adding Applications to the Platform)

此階段主要是以個人喜好而加入不同的應用程式，例如不同的遊戲軟體或音樂軟體，其方法與第一步驟的加入驅動程式類似。

四、平台設定(Additional Platform Settings)

此步驟主要是在平台設定的組態(configuration)中之 Build Options 選擇合適的變數選項。若在第二步驟的平台選擇當中以測試版為主，在此階段則需多加額外的設定，以下則為應加入之變數。

1. Enable CE Target Control Support(SYSGEN_SHELL=1)
2. Enable Eboot Space in Memory(IMGEBOOT=1)
3. Enable Event Tracking During Boot(IMGCELOGENABLE=1)
4. Enable Full Kernel Mode(no IMGNOTALLKMODE=1)
5. Enable Kernel Debugger(no IMGNODEBUGGER=1)
6. Enable KITL(no IMGNOKITL=1)
7. Run-time Image Can be Larger than 32MB(IMGRAM64=1)

對於為何選擇以上之變數，可參考微軟 msdn 網站[25]。對平台選擇開放版而言，只需選擇加入兩種變數的設定：

1. Enable Ship Build(WINCESHIP=1)
2. Run-time Image Can be Larger than 32MB(IMGRAM64=1)

對於此兩種變數的解釋，也可參考微軟 msdn 網站[26]。其中 WINCESHIP=1 主要是取消 debug 的功能，進而讓映像檔的容量變小。

五、建立映像檔(Build the Platform)

在第二部份的最後步驟，則是將映像檔建立起來，選擇 Build and Sysgen 選項，是因為更改了原本的作業系統設定，至於其他 Build 選項也可參考微軟 msdn 網站[28]。

第三節 系統初始化—映像檔傳輸至機器

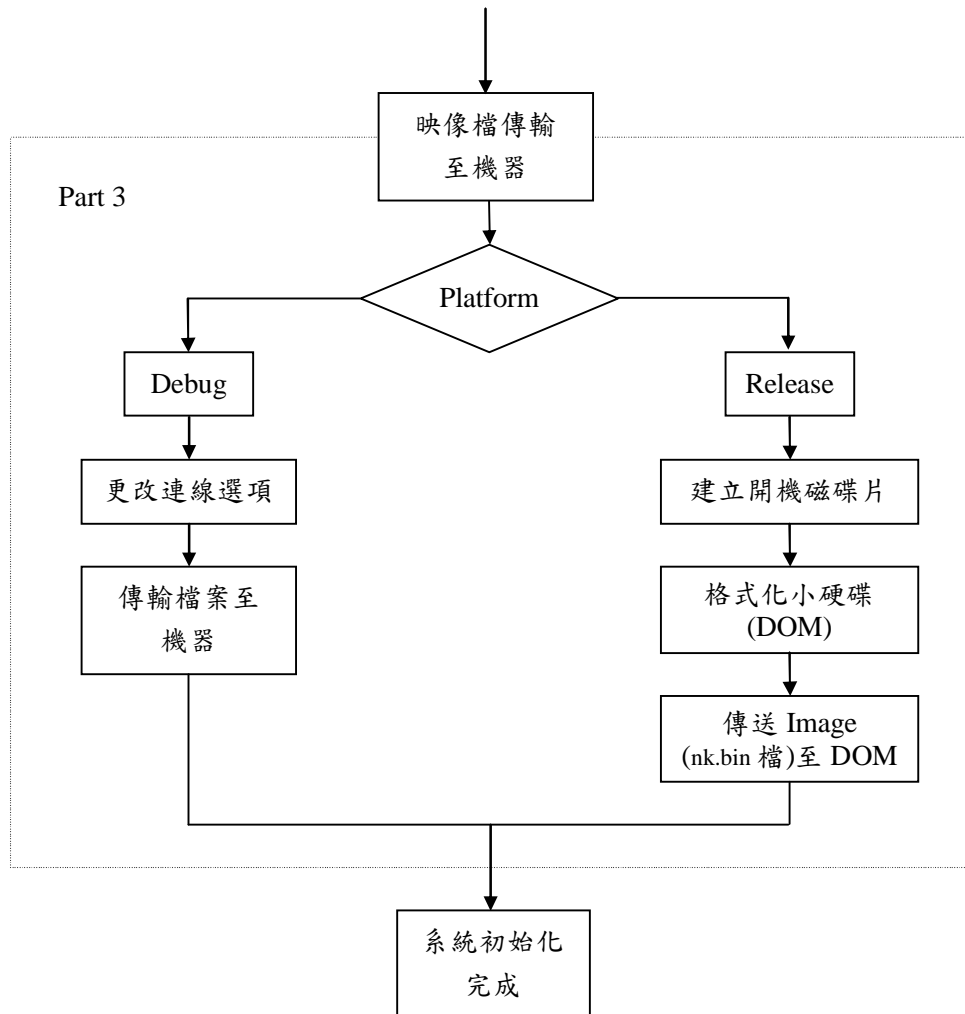


圖 4-4 映像檔傳輸至機器流程圖

在第三部份的映像檔傳輸至機器裡，選擇測試版和開放版的進行方式有很大的不同。以測試版(Debug)為例，我們必須先在遠端連線選項(Target Device Connectivity Options)中選擇正確的設定，下載(Download)的選項可透過不同方式將資料傳輸至機器，例如透過 RS-232(Serial Port:boot)、Ethernet(eboot)等。在本研究裡我們選擇了 Ethernet 選項，而傳輸(Transport)的選項也應以 Ethernet 作為選擇。此時我們將網路線接至機器上，並打開電源，Ethernet 選項的設定視窗則

會開始出現等待網路從機器傳回的 Device Boot Name，不過在此之前我們必須將電腦的防火牆關上，否則無法回傳此值。更改連線選項完成後，系統發展軟體 PB 5.0 將會自動將建立好之映像檔(nk.bin)傳輸至機器上。在機器螢幕上可以選擇不同的檔案傳輸方式。其中有直接使用嵌入式小硬碟內的 nk.bin 檔作為作業系統映像檔(local nk.bin)，也可透過網路(eboot)，或者透過 COM Port(sboot)等方式。由於是測試版，因此我們選擇了透過網路來傳輸。當傳輸完畢後，PB 5.0 端畫面將會有 Debug 的訊息，提供資訊以方便我們直接控制傳輸至機器上之映像檔，包括作業系統、驅動程式和軟體等。其中應該注意的是，當機器電源關閉後，此次傳輸的一切將不會儲存在機器上，因此在進行測試版時，無須將嵌入式小硬碟格式化，也不需將它裡面原有的映像檔刪除，所以也帶來了方便。

當確定無錯誤訊息後，則可進入將測試版的映像檔變成開放版的映像檔階段。對於不同版本的映像檔，我們都必須重新進行建立的動作。當開放版的映像檔建立完成後，需將嵌入式小硬碟格式化，因為我們希望將此映像檔保留在小硬碟內。格式化之前，則需先建立系統開機片。Windows CE.NET 5.0 軟體發展套件除了提供 PB 5.0 以完成作業系統映像檔的製作以外，也提供額外的發展工具，例如提供將嵌入式小硬碟製作成 FAT 檔案格式的執行檔、製作 DOS 開機片的執行檔等。

首先在電腦中執行 Windows CE.NET 5.0 軟體發展套件所提供的 WebSetup.exe 檔案，接著將 setupdisk.144(Disk Image)檔案放入磁碟片中，此資料

夾包含了 fdisk.exe、format.exe 等可將嵌入式小硬碟重新格式化的執行檔。另外我們也需要將 CEPCBoot.144(Disk Image)檔案放入不同的磁碟片中，而此資料夾則包含了利用 DOS 開機所需要的檔案，包括了 autoexec.bat、config.sys、loadcepe.exe 等。

準備了以上兩張磁碟片後，我們需要將軟碟機外接到機器的軟碟插槽上。機器需選擇以磁碟片開機，此時將讀到 setupdisk.144 的資料。首先在 A 槽下輸入 fdisk 指令，主要是將嵌入式小硬碟進行格式化。接著輸入 format c:指令，此指令是讓嵌入式小硬碟能以 DOS 開機(具有 FAT 檔案系統)。完成此階段後，可將 CEPCBoot.144 的磁碟片放入軟碟槽，執行 sys c:指令，主要是為了傳輸 DOS 啟動檔案(Boot files)。當執行結束後，copy *.* c:指令將會把軟碟機裡的全部檔案複製到 c 槽底下，到此大致上完成了格式化嵌入式小硬碟的步驟。

當重新啟動機器後，螢幕將出現 DOS 畫面，讓我們選擇以哪種方式啟動映像檔。在開放版與測試版不一樣的地方乃是我們在這個步驟選擇以 local nk.bin 作為我們的映像檔(原始設定檔會以網路選項作為預設的開機選擇)。因此為了讓機器啟動時直接選擇 local nk.bin 選項，我們必須更改 autoexec.bat 和 config.sys 兩個檔案。其中 autoexec.bat 更改的部分如表 4-4。

表 4-4 autoexec.bat 檔案保留部分

```
if "%CONFIG%" == "CEPC_LOCAL" goto CEPC_LOCAL
if "%CONFIG%" == "CLEAN" goto CLEAN

:CEPC_LOCAL
REM
#####
REM Launch LOADCEPC using a local NK.BIN image.

loadcepc nk.bin
goto END
```

只需將 autoexec.bat 檔案保留以上部分即可。程式第一行執行 CEPC_LOCAL 的部份，即是執行 loadcepc nk.bin。在 config.sys 檔案中也只需保留如表 4-5 之程式即可。在程式第四行 menudefault=CEPC_LOCAL,1 是指定預設執行的檔案為 local 之 nk.bin，而變數 1 表示延遲 1 秒鐘才進入系統。

表 4-5 config.sys 檔案保留部分

```
[menu]
menuitem=CEPC_LOCAL, Boot CE/PC (local nk.bin)
menuitem=CLEAN, Clean Boot (no commands)
menudefault=CEPC_LOCAL,1
menucolor=7,1

[CEPC_LOCAL]

[CLEAN]

[COMMON]
buffers=10,0
files=30
break=on
lastdrive=Z
dos=high,umb
device=himem.svs /testmem:OFF
```

設定了直接開機的作業系統映像檔位置後，我們還需預先將我們製作好的映像檔放入到嵌入式小硬碟中。至於放檔案的方式，可透過以下的幾個步驟完成。首先將已經格式化的嵌入式小硬碟插在機器提供的額外 IDE 連接介面上，並設其跳腳(jump)設定為附屬(slave)。由於機器原本即提供一現成的嵌入式小硬碟，此嵌入式小硬碟的跳腳則接在主要(master)的位子。利用現成的嵌入式小硬碟開機，接著Windows 畫面將出現第一和第二顆嵌入式小硬碟的目錄，分別為 Hard Disk 1 和 Hard Disk 2。將已經放入映像檔的usb 隨身碟插入usb 插槽，Windows 畫面也將測出第三顆嵌入式小硬碟的目錄(Hard Disk 3)，此時就可將usb 裡的映像檔儲存至第二顆嵌入式小硬碟裡。當檔案傳輸完畢後，處於附屬跳腳的這顆嵌入式小硬碟就可獨立的選擇以 local nk.bin 的方式直接開機了，當然其跳腳則需設為主要。

完成了以上主要三個部份後，本系統的軟體初始化就已經完成了。