

建構於 TMS320DSC25 之多媒體資訊家電系統平台

陳雅萍 陳敬 劉建宏 許勝杰

國立成功大學電機工程研究所

n2690164@cc.ncku.edu.tw

jchen@mail.ncku.edu.tw

{ liuken, Jason }@rtpc06.ee.ncku.edu.tw

摘要

本論文研究嵌入式系統平台之建立，並以配置 TMS320DSC25 (簡稱 DSC25) 系統晶片之硬體平台實作一多媒體資訊家電系統平台。DSC25 是美商德州儀器 (Texas Instrument, 簡稱 TI) 公司於數位相機產品中之一套解決方案，具有 ARM7TDMI 處理器 (簡稱 ARM) 與 C5409 數位信號處理器 (簡稱 DSP) 組成之主從式 (Master-Slave) 多處理器架構—以 C5409 處理多媒體影音資料，以 ARM7TDMI 處理器控制整體系統周邊與 DSP 之運作。我們根據其參考設計電路板之硬體平台及其配備之周邊裝置，建立 OSD (On-Screen Display)、AIC23、Key switch、UART 等輸出與輸入裝置之驅動程式，並移植一即時系統核心 $\mu\text{C}/\text{OS-II}$ 以建立系統軟體之基礎架構。此一平台具備基本之任務管理、任務排程、時間管理、記憶體管理等功能，並使用系統之信號功能 (signal) 管理周邊設備等硬體資源。本研究並以所建立的系統平台設計一 MP3 格式音樂檔案的播放器及所需之使用者介面，實作並藉以測試任務溝通、資源管理、周邊輸出與輸入等功能。此 MP3 應用程式可配合歌曲之播放於 OSD 顯示歌詞，產生與坊間 KTV 設備類似之效果。

1. 簡介

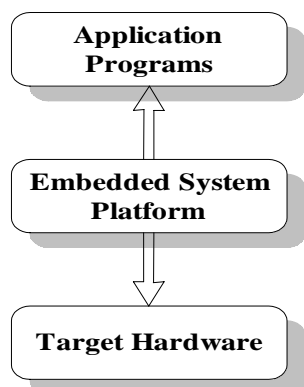
隨著個人電腦技術之快速進步，消費電子 (Consumer Electronics) 數位化，以及無線與有線通訊 (Communication) 技術的發展，加上積

體電路製程持續進展之推波助瀾，促進資訊及硬體之多元化技術整合，結合資訊、通訊、消費電子之 3C (Computer、Communication 與 Consumer Electronics) 產品與應用正快速竄起，與網路連線之功能也成為家電產品發展趨勢之一。未來的資訊家電產品將使消費者在居家之外可以很方便地經由上網或者是以電話控制遠在家裡的電器。目前消費市場已有獨立於個人電腦之外的 IP Phone、Net Meeting、上網機 (Network Appliance) 等產品，使現代人的生活更加便利。可隨身攜帶、輕薄短小的嵌入式系統產品更是蓬勃發展，例如目前市面上銷售的隨身碟是使用 USB 介面的電腦周邊產品，具有大量記憶儲存裝置，現在的產品技術已將其功能延伸為可以播放 MP3 格式音樂檔案的隨身聽裝置。在一般多媒體產品方面，隨著無線網路技術的進步，消費者除了有線網路之外也可以透過無線網路即時下載遠端多媒體資料，播放歌曲或影片，提供消費者舒適的生活與更多元之選擇。

當消費性電子產品功能單純時，其嵌入式系統使用單一程式控制硬體功能即可滿足功能性之需求。然而現今消費電子市場對電子產品的功能要求不斷增加，嵌入式系統無論是硬體或應用程式都愈趨複雜，對作業系統服務功能之需求甚為殷切。使用一嵌入式作業系統可將硬體平台隱藏，並管理硬體資源，以高階的抽象形式表示與硬體相關的低階控制；這種將硬體抽象化的方法，可協助應用程式之開發並解

決軟硬體之間資源使用與分配的難題，使得應用程式開發人員，不需要深入瞭解硬體平台，仍可開發所需之軟體。如圖一所示，一般嵌入式系統架構可以分成三個層次：上層為應用程式，下層為硬體平台，中間夾著一層軟體執行平台，也就是系統核心。系統核心將硬體平台包裝起來，並且提供一個與硬體無關的介面給上層的應用程式。由於電子科技產品進步的速度一日千里，目前嵌入式系統平台之發展特別強調『量身定做』之原則，具有由研發者基於特定用途或功能之需求進行設計客制化 (Customization) 的特質 [3, 4, 5, 6]。

本篇論文研究並設計一適合開發多媒體資訊家電之系統平台，以配置美商德州儀器公司 (Texas Instruments, 簡稱 TI) 為數位相機與相關產品所研發之多媒體系統晶片 (即 SoC, System-on-a-Chip)—TMS320DSC25 之參考設計評估模組 (Evaluation Module, 簡稱 DSC25 EVM) 為硬體平台[2]，移植 μ C/OS-II 即時系統核心[1]，建構一適合 TMS320DSC25 多媒體系統晶片特性之軟體執行平台。為提供應用軟體較為完備之執行環境，並根據 DSC25 EVM 所配置之電視訊號輸出、音效晶片等輸出與輸入周邊裝置設計其驅動程式，使應用程式開發者可研發處理或播放諸如 MP3、MPEG-1、MPEG-4、JPEG、H.263 等規格多媒體影音資料之應用程式。



圖一，嵌入式系統基本架構

本論文中，第二節介紹目前學術界有關嵌入式系統平台之研究。第三節描述 DSC25 EVM 系統平台的軟硬體架構與本研究所建構之系統核心。第四節闡述系統軟體實作與資源管理機制之建立。第五節說明以實作一 MP3 格式音樂播放之應用軟體為例，測試此多媒體資訊家電系統平台之功能。第六節為結論與對後續工作之規劃。

2. 相關研究

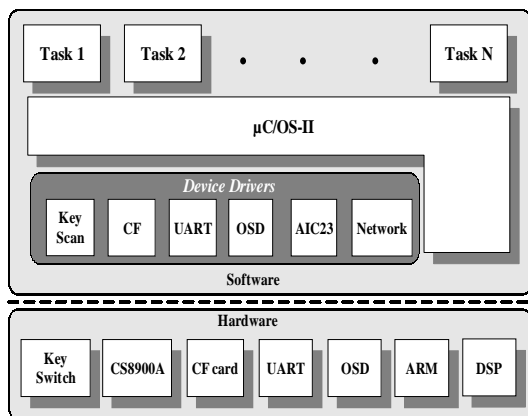
隨著資訊家電產品功能不斷進步，學術機構也致力於研發可因應不同功能需求之系統平台。與本論文在開發系統平台方向比較接近者包括 Maou 以 Motorola 公司研發之 Dragon Ball EZ 平台 所開發之 STARTRH 系統[7]，其使用 Motorola PPSM 即時系統核心[13]，此系統核心透過驅動程式控制硬體而不是直接取用硬體。其軟體發展環境是以 IBM 相容個人電腦，使用微軟視窗作業系統環境之編譯器與除錯器，建構為一適用於 PDA (Personal Digital Assistant) 應用程式開發環境的系統平台。STARTRH 目前已經進入商品化的階段，支援紅外線控制、筆端輸入觸控面版、按鍵、LCD 顯示面版等周邊裝置並具有圖形介面；LCD 顯示之文字可選擇 16 × 16 像素 BG 中文碼、16 × 16 像素 Big 5 中文碼與 8 × 8 像素英文的 bitmap 字型。

由於網路頻寬愈來愈大，造成網路設備之路由器 (Router) 必須維持愈來愈大的查詢表 (Look-up table)，以致封包 (Packet) 無法快速傳遞。學者 Pierre G. Paulin 為了有效解決此一問題，使用 NPU (Network Processor Unit) 的構想，利用管線化 (Pipelining)、平行處理 (Parallel processing) 與多重引線 (Multi-thread) 等方法實現以硬體加速封包標頭分析 (Parsing) 與檢核碼計算，提昇處理封包傳輸效率[9]。Paulin 根據這些構想設計了 STMicroelectronics 整合工具 [8]，研發者可以根據使用者對路由器處理功能之需求進行作開發工作。



3. 系統架構

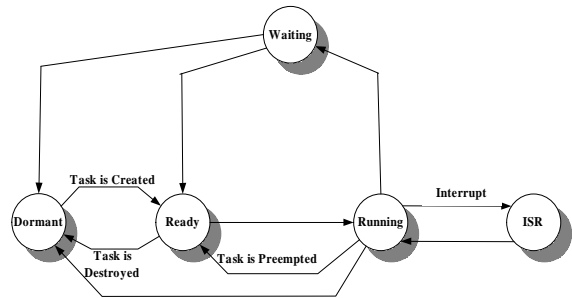
本系統平台以TI之DSC25 EVM為硬體平台，其具有由ARM7TDMI與C5409處理器組合而成的異質(Heterogeneous)雙CPU架構。ARM子系統可支援作業系統執行，負責控制包括中斷控制器(interrupt controller)、計時器(timer)、螢幕顯示(On-Screen Display, 簡稱OSD)介面、影像介面(video interface)、CCD控制器、USB(Universal Serial Bus)、序列埠(serial port)與使用者介面並且執行系統初始化、處理器內容切換(context switch)、多工(multi-tasking)與一般目的之任務控制。DSP子系統處理影像資料、資料壓縮與解壓縮的工作。ARM可直接讀寫DSP的記憶體，方便對DSP下指令與搬移資料供DSP處理。藉由HPI(Host Port Interface)之中斷機制和DSP溝通。



圖二，系統架構圖

建構本系統之軟體基礎架構是以移植Jean J. Labrosse設計之即時系統核心µC/OS-II[1]所實現。µC/OS-II[1]是一硬性即時系統核心[14]，具有體積小、任務執行時間明確等優點，並支援以優先權(priority)為基礎之搶佔式(preemptive)多工(multitasking)作業，足以滿足多媒體應用軟體播放影音資料時所需之軟性即時需求[14]。其組成可細分為任務處理、任務溝通機制、時間管理、記憶體管理等模組。µC/OS-II與DSC25 EVM組成之系統架構如圖二所示，最上層為應用程式，可以劃分為多個任務；硬體周邊裝置有Key switch、CS8900A網路晶片、Compact Flash card元件、UART介面、

OSD訊號輸出與AIC23音效晶片。µC/OS-II系統核心負責任務間的排程處理，並且將硬體周邊裝置集中管理，避免任務間因使用周邊資源時造成衝突。



圖三，µC/OS-II任務之狀態轉移圖

在µC/OS-II的系統環境下，一任務由使用者建立與刪除，並以一段反覆不斷執行之程式碼建構而成。系統核心共可處理64件任務，除了作業系統專用的優先權最高的4個與最低的4個之外，一般使用者最多可以建立56件任務。如圖三所示，應用程式之任務可能處於五種狀態其中之一。對系統核心而言，當一件任務存在程式區(RAM或ROM)，尚未開始執行時是處於Dormant狀態；當此任務備妥可被執行時，則進入Ready狀態。所有處於Ready狀態的任務當中，優先權最高者被系統置於Running狀態並執行。處於Ready狀態的任務若是它的優先權並不比目前處於Running狀態的任務高，它進入Running狀態的條件為下述二者之一：

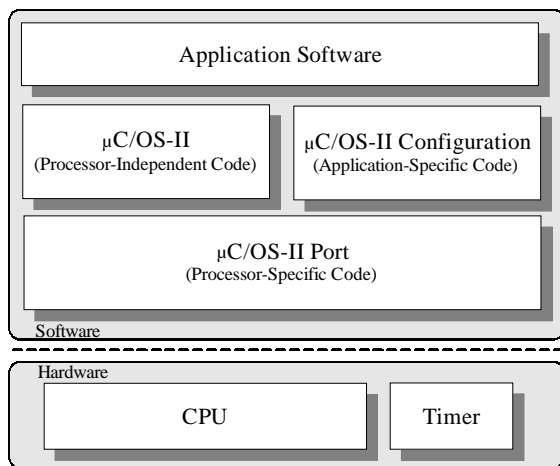
- (1) 原本可執行且處於Running狀態的任務結束執行或被刪除；
- (2) 原本處於Running狀態的任務進入Waiting狀態。

任何處於Running狀態的任務都可以被中斷，使這個任務進入ISR(Interrupt Service Routine)狀態。中斷發生時，正在執行(處於Running狀態)的任務被系統懸置(suspend)，由ISR取得CPU控制權。一個任務被刪除時，µC/OS-II將強制任務進入Dormant狀態，使這個任務不會再被排程(schedule)。

µC/OS-II之原始程式碼大部分是用ANSI C語言所撰寫，具有易於移植到任何硬體平台的特性。



如圖四所示， $\mu\text{C}/\text{OS-II}$ 的原始程式碼依其可移植性細分為與處理器硬體無關之程式碼(Processor Independent Code)，包括系統核心、記憶體管理、任務處理與溝通、時間管理等模組，這些部份與記錄系統組態(configuration)的 Application-Specific Code皆不需要大幅度修改；處理器硬體特定之程式碼(Processor-Specific Code)，包括中斷處理、與處理器頻率有密切關係的時間管理、tick之單位處理等模組，通常須使用處理器專用之組合語言撰寫。此外，因移植的目標平台之不同特性，部分C語言程式碼必須在移植時檢查或修改設定值，例如：設定記憶體是Little Endian或Big Endian、資料型態對於資料佔用記憶體寬度定義的標頭檔(Header files)等部份亦屬於移植時必須注意是否必須修改之Processor-Specific Code。

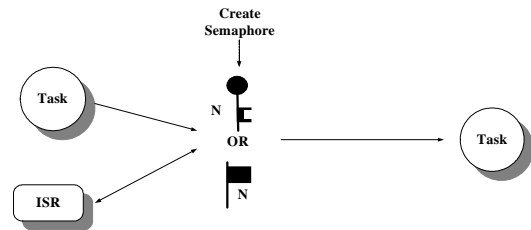


圖四， $\mu\text{C}/\text{OS-II}$ 程式碼模組與系統架構

4. 系統實作

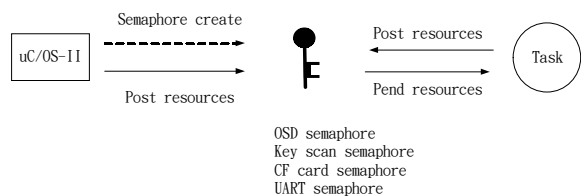
如前所述，建構於DSC25之多媒體資訊家電系統平台之硬體部份使用DSC25 EVM，而軟體開發工具則使用TI針對此硬體開發環境設計之CCS(Code Composer Studio)整合式開發環境，包括編譯器、除錯器、模擬器等。由於 $\mu\text{C}/\text{OS-II}$ 系統核心並不具備驅動程式，實作時，我們將所移植之系統核心、硬體平台的驅動程式與應用程式共同編譯成一個二元檔(binary)，經由JTAG[12]介面將此二

元檔下載到DSC25 EVM，再啟動系統並執行應用程式。



圖五，任務、ISR與號誌之關係

$\mu\text{C}/\text{OS-II}$ 提供號誌(semaphore)功能支援任務溝通與同步機制，本系統平台以號誌(semaphore)功能實作系統資源管理。 $\mu\text{C}/\text{OS-II}$ 的號誌包含一個16-bit整數(unsigned)記錄號誌數量與一組任務等待記錄，號誌數量可以是0到65535。任務、ISR與一個號誌三者間的關係如圖五所示，號誌所代表的是使用系統資源(鑰匙)的數量或者是觸發某些事件(或者旗幟)的數量。任務與ISR都可以釋放(Post)號誌，但只有任務可以等待(Pend)號誌的產生。



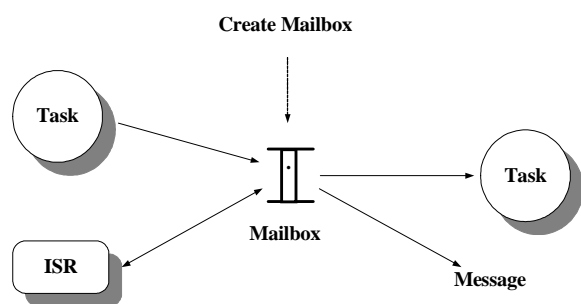
圖六，使用號誌管理系統資源

系統資源管理可以避免多個任務或應用程式因共用資源而造成資料混亂。本系統建立時於驅動程式內部設計並實作等待號誌之機制，系統核心於初始化階段針對畫面輸出、按鍵、UART、CF卡這些具備單一任務使用特性之驅動程式各產生一號誌，應用程式之任務可直接呼叫驅動程式使用硬體資源，如圖六所示。因本系統平台目前並不對死結(deadlock)問題提出解決辦法，應用程式設計者必須注意避免發生死結，例如設定一特定時間作為等待號誌之期限。

本系統平台使用 $\mu\text{C}/\text{OS-II}$ 的訊息信箱(mailbox)功能實現任務間溝通之機制。訊息信箱是一個



µC/OS-II物件，任務或ISR可使用指標送出一個應用程式設定的資料結構給其他任務，但只有任務可以等待訊息信箱內訊息的到來。任務、ISR與一個信箱三者間的關係如圖七所示，信箱在使用之前必須先被建立，信箱內只能有一個指向訊息之指標，初始化的指標值是空指標(Null pointer)。



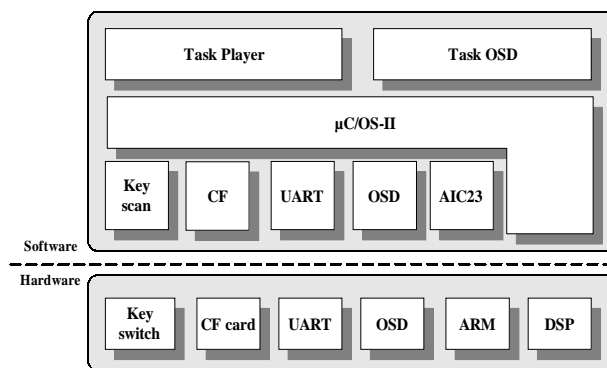
圖七，任務、ISR與一個訊息信箱的關係

5. 系統功能測試

本系統平台建置完成後，以實作一類似坊間KTV裝置之嵌入式系統，進行系統平台之I/O驅動、任務溝通與處理器之間溝通等功能測試。此嵌入式系統為一可播放MP3格式音樂(歌曲)並具備配合歌詞同步顯示字幕功能之應用程式。

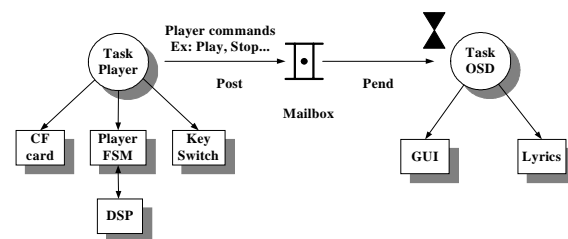
圖八顯示MP3格式音樂播放應用系統之架構。此應用系統之軟體規劃為Player播放器與OSD控制模組等二任務，系統核心µC/OS-II提供服務功能並管理硬體驅動程式；硬體部份之規劃為：ARM與DSP處理器分別執行周邊控制與影音資料處理工作，Key switch接收使用者之按鈕輸入、OSD輸出使用者介面、CF card用以儲存MP3格式之音樂或歌曲檔案、UART則用作研發階段之偵錯輔助工具。如圖九所示，Player任務處理CF卡內MP3檔案讀取、按鍵輸入、播放器與DSP之間的溝通，OSD任務分為控制使用者介面與歌詞顯示之功能。Player與OSD兩個任務之間經由µC/OS-II所提供之訊息信箱進行溝通，溝通之方向為Player任務接收使用者按下按鈕所輸入之命令，傳送命令至訊息信箱，OSD任務根據信箱內容改變輸出之內容。例如：使用者按下播放按鈕後，OSD任務將收到

Player任務所傳送之播放命令並取出歌詞與時間資料，於預設之時間顯示下一句歌詞，Player任務則檢視訊息信箱的內容，判斷狀態是否改變。



圖八，MP3應用程式架構圖

Player任務與OSD任務皆須向系統要求配置資源：Player任務要求使用按鍵、OSD任務要求使用電視畫面(OSD)輸出資源。若任何一方無法順利取得資源，MP3應用程式之流程設計是持續等待資源齊全為止。



圖九，Player任務與OSD任務間之關係圖

OSD任務之畫面顯示可以分成兩部份：字型與圖形介面。二者皆須設定畫面顯示像素(pixel)資料之表達格式與顏色。顯示之字型擷取自國喬中文系統之字型庫[10]，字體大小有兩種：16 × 15像素與24 × 24像素。中文與英文字型分別存放在兩個字型檔，預先儲存於CF卡內；歌詞則以ID3[11]標籤(tag)格式預先存入MP3歌曲檔案。中文字型顯示採用的方法為：按照文字編碼順序搜尋一中文或英文字符在字型檔中之位置，以取得字型模式(pattern)資料，根據字型模式(pattern)配合螢幕上每一列水平顯示像素資料之數量，將像素資料填入顯示介面之記憶體內。



6. 總結與後續研究

嵌入式系統產品之開發要求在有限資源下，能夠設計出一個滿足使用者需求的系統平台。我們根據TI之DSC25 EVM作為多媒體資訊家電硬體平台，移植 μ C/OS-II系統核心，加入控制硬體平台周邊UART、按鍵、畫面顯示與音效晶片控制的驅動程式，並且使用系統的號誌處理功能管理系統資源。任務之間的溝通採用系統核心所提供的訊息信箱功能，使任務之間處理工作達到同步。本系統平台目前具有ARM與DSP兩個處理器間之間工作委派的溝通功能、CF卡檔案處理基本功能。並且實作一個類似KTV之MP3格式之音樂(歌曲)播放器，使用MP3應用程式配合歌詞字幕顯示效果，測試系統平台處理多媒體資料播放功能。

本論文在移植 μ C/OS-II的過程部份，尚未完成bootloader之實作，後續工作包含將bootloader之實作與bootloader燒錄至Flash記憶體，使系統可直接從Flash記憶體開機執行，使移植核心的工作更趨完備。

7. 參考資料

- [1] Jean J. Labrosse, "Micro C/OS-II, The Real-Time Kernel", R & D Books, 1999.
- [2] Texas Instruments, "TMS320DSC25 DSP CPU and Peripherals", Technical Reference Manual Version 1.1.
- [3] Grant Martin and Frank Schirrmeister, "A Design Chain for Embedded Systems", Computer, IEEE, pp. 100-103, Mar 2002.
- [4] Alberto Sangiovanni-Vincentelli and Grant Martin, "Platform-Based Design and Software Design Methodology for Embedded Systems", Design & Test of Computers, IEEE, pp. 23-33, Nov/Dec 2001.
- [5] Gert Goossens, John Van Praet, Dirk Lanneer, Werner Geurts, Augusli Kifli, Clifford Liem and Pierre G. Paulin, "Embedded Software in Real-Time Signal Processing Systems: Design Technologies", Proceedings of the IEEE, pp. 436-454, Mar 1997.
- [6] Pierre G. Paulin, Clifford Liem, Marco Cornero Francois Nacabal and Gert Goossens, "Embedded Software in Real-Time Signal Processing Systems: Application and Architecture Trends", Proceedings of the IEEE, pp. 419-435, Mar 1997.
- [7] Peter Xiangdong Mao, Zhizhong Tang, Michael Chen, Youming Zhang and Vern Zheng, "Architecture of the Simplified Chinese Embedded System STARTH", High Performance Computing in the Asia-Pacific Region, 2000, Proceedings, pp. 1167-1170.
- [8] Pierre G. Paulin, "Embedded Systems Technologies for Application-Specific Architecture Platforms", System Synthesis, 2001. Proceedings. The 14th International Symposium, pp. 195.
- [9] Pierre G. Paulin and Faraydon Karim, "Network Processors: A Perspective on Market Requirements, Processor Architectures and Embedded S/W Tools", <http://www.ece.cmu.edu/~ece767/papers/perspective1.pdf>
- [10] 施威銘，劉廣治，"組合語言程式設計實例"，旗標出版社，民國七十九年出版。
- [11] ID3v2, <http://www.id3.org/develop.html>
- [12] IEEE 1149.1, <http://www.altera.com/literature/an/an039.pdf>
- [13] <http://www.ifoundrysys.com/company/news/pr020528-motorola.htm>
- [14] Jane W. S. Liu, "Real-Time Systems", Prentice-Hall Inc., 2000.

