

# 行政院國家科學委員會專題研究計畫 成果報告

## 嵌入式網路通訊裝置評比技術與工具之研發--總計畫(2/2) 研究成果報告(完整版)

計畫類別：整合型  
計畫編號：NSC 99-2220-E-009-043-  
執行期間：99年08月01日至100年07月31日  
執行單位：國立交通大學資訊工程學系(所)

計畫主持人：林盈達

處理方式：本計畫可公開查詢

中華民國 100 年 10 月 30 日

## 目錄 (Content)

[1] 計畫總覽.....	1
<u>摘要與 EBL 實驗室</u> .....	1
<u>計畫說明</u> .....	1
[2] 緣由與目的.....	1
[3] 研究內容.....	2
<u>嵌入式系統共通測試平台(Embedded Common Test Platform, CTP)</u> .....	2
<u>SSLTC (SSL VPN Tunnel Capacity)</u> .....	3
<u>IVQT (Integrated Voice Quality Test)</u> .....	4
<u>嵌入式系統應用效能評比測試(Embedded System Performance, ESP)</u> .....	5
<u>嵌入式系統耗電分析評比測試(Embedded Power Consumption, EPC)</u> .....	6
<u>嵌入式核心與網路協定行為評比技術與工具 (Kernel Profiling Tool, KPT)</u> .....	7
<u>嵌入式儲存裝置測試系統 (Embedded Storage Benchmark System, ESBS)</u> .....	8
[4] 研究成果與討論.....	9
[5] 參考文獻.....	10

## [1] 計畫總覽

### ■ 摘要與 EBL 實驗室

近年來嵌入式系統被大量的應用在各種不同的產品上，使用者可以很輕易地透過各種網路進行資料存取，不論是有線網路或是無線網路 Wi-Fi, GSM, WiMAX 等網路介面技術，到目前為止並沒有一個夠好的方法可以針對各項嵌入式系統網路裝置及元件進行評比，因此如何挑選適當的元件並幫助產品開發者測試該裝置的穩定度及效能，將成為未來開發嵌入式系統網路通訊裝置的重要議題。

本計畫目的在於提供一系列完整的測試工具與測試方法，其中包括應用效能測試、耗電測試、分析與診斷機制、元件效能測試(例如：儲存裝置)，將其應用在目前主流的嵌入式網路通訊裝置，為其建造「嵌入式網路通訊裝置測試軟體發展中心 (Embedded Benchmarking Lab, EBL)」來協助廠商及使用者評比嵌入式網路通訊裝置，亦可做為日後學術相關研究時有實際的產品數據提供參考。

### ■ 計畫說明

為了達到嵌入式網路通訊裝置在測試上的「垂直整合」，整個計畫目標針對當前嵌入式網路通訊裝置所面臨的問題進行研究，並配合相關的技術和理論研發各項技術以及測試評比平台，內容共計四個子計畫進行嵌入式網路通訊裝置評比：

#### **嵌入式網路通訊裝置應用效能評比技術與工具之研發**

提供使用者應用效能評比工具進行使用者應用效能評比測試，在對待測的嵌入式網路通訊裝置進行效能評比測試時，可以利用總計畫所提供的測試設備與流量設計出各種不同的測試項目，藉以觀察並評估不同的環境條件下對系統效能的影響。

#### **嵌入式網路通訊裝置耗能評比基準與工具**

### 之研發

此工具主要整合嵌入式網路通訊裝置的耗電分析及測試設備，建立一套完整的自動化耗電量測環境。本計畫會從測試平台取得其產生的測試封包並蒐集回覆封包以分析其反應時間及流量等，並將測試結果回傳給嵌入式系統耗電分析與評比工具產生黑箱測試評估報告。

#### **嵌入式網路通訊裝置核心與通訊行為效能評比技術與工具之研發**

主要目的是對系統核心的訊息與通訊協定行為進行分析，並開發測試與評比的工具，其工作包含建立一套嵌入式網路通訊裝置的標準測試流程及分析機制，開發具高親和力之網路核心系統與通訊協定行為的整合式分析工具，最後結合總計畫提供的測試平台對網路通訊過程進行細部動作的分析，診斷其潛在的問題。

#### **嵌入式網路通訊裝置儲存裝置效能評比基準與工具之研發**

本工具主要針對以快閃記憶體為基礎之儲存裝置，設計與實作出完整之效能評比程序、工具以及技術。具體工作項目包括：作業系統核心與韌體之事件擷取機制，標準測試程序之建立，測試標竿之設計。

**關鍵詞：** 嵌入式系統、網路通訊裝置、耗電量、系統核心事件、儲存媒體、開放原始碼

## [2] 緣由與目的

近年來嵌入式系統被大量的應用在各種不同的產品上，使用者可以很輕易地透過各種網路進行資料存取，不論是有線網路或是無線網路 Wi-Fi, GSM, WiMAX 等網路介面技術，到目前為止並沒有一個夠好的方法可以針對各項嵌入式系統網路裝置及元件進行評比，因此如何挑選適當的元件並幫助產品開發者測試該裝置的穩定度及效能，將成為未來開發嵌入式系統網路通訊裝置的

重要議題。

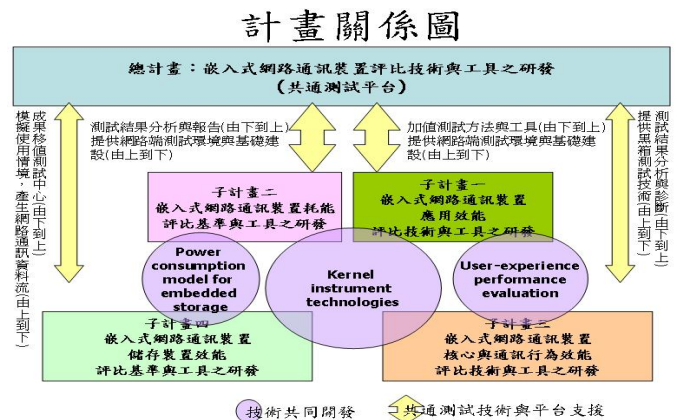
### [3] 研究內容

本計畫是三年期計畫的第二年，目的在於提供一系列完整的測試工具與測試方法，其中包括黑箱(Black Box)、灰箱(Gray Box)與白箱(White Box)測試、應用效能測試、耗電測試、分析與診斷機制、元件效能測試(例如：儲存裝置)，將其應用在目前主流的嵌入式網路通訊裝置，為其建造「嵌入式網路通訊裝置測試軟體發展中心(Embedded Benchmarking Lab, EBL)」來協助廠商及使用者評比嵌入式網路通訊裝置，亦可做為日後學術相關研究時有實際的產品數據提供參考。

有鑑於此，本計畫的目標即對於當前嵌入式網路通訊裝置所面臨到的問題，進行相關的研究及討論，配合相關的技術和理論，研發各項技術以及測試評比平台，為達到此目的，本計畫共將開發包含總計畫及四個子計畫共四項相關測試工具進行嵌入式系統評比，分別如下所述：

- ✓ 嵌入式系統共通測試平台(Embedded Common Test Platform, CTP 1.1.1)
- ✓ 嵌入式系統應用效能評比測試(Design and Implementation of Benchmarking Tools and Methods for Embedded System Performance, ESP 1.2.1)
- ✓ 嵌入式系統耗電分析評比測試(Design and Development of Power Consumption Analysis and Benchmarking Tools for Embedded Network and Communication Devices, EPC 1.3.1)
- ✓ 嵌入式核心與網路協定行為效能評比技術與工具(Design and Development for Kernels and Protocols Analysis and Benchmarking Tools for Embedded Networking and Communication Devices, KPT 1.4.1)
- ✓ 嵌入式儲存裝置評比測試(Design and Implementation of Benchmarking Tools and Methods for Storage Systems, ESBS 1.5.0)

為了達到嵌入式網路通訊裝置在測試上的「垂直整合」，本計畫針對當前嵌入式網路通訊裝置所面臨的問題進行研究，並配合相關技術和理論建置**嵌入式網路通訊裝置測試中心：方法與工具之實驗、開發與推廣 (Benchmarking Lab for embedded networking and communication devices: Experiment, Development, and Deployment of methodologies and tools, SYS)**，總計畫本身包含開發一個子系統為嵌入式系統共通測試平台[CTP 1.1.0]，另外尚有其它四個子系統分別為：**嵌入式系統應用效能評比測試[EUE 1.2.0]**、**嵌入式系統耗電分析評比測試[EPC 1.3.0]**、**嵌入式核心與通訊行為效能評比測試[KPT 1.4.0]**及**嵌入式儲存裝置評比測試[ESB 1.5.0]**，而各系統與總計畫間之介面及互動關係如圖一所示。

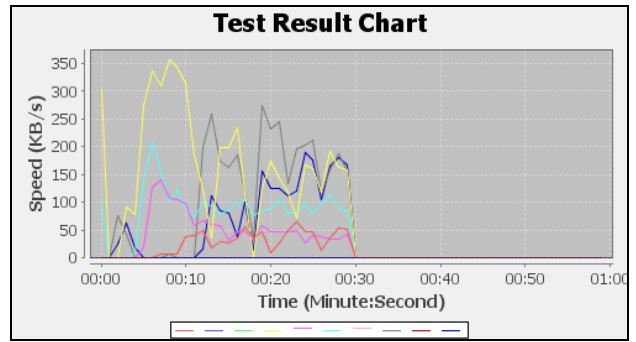


圖一 計畫關係圖

本計畫在整個軟體開發過程除了依照CMMI所規定之規格撰寫系統需求文件、系統設計文件與系統整合測試文件，並完成設計與開發各子系統，各子系統之研究內容及如下所示：

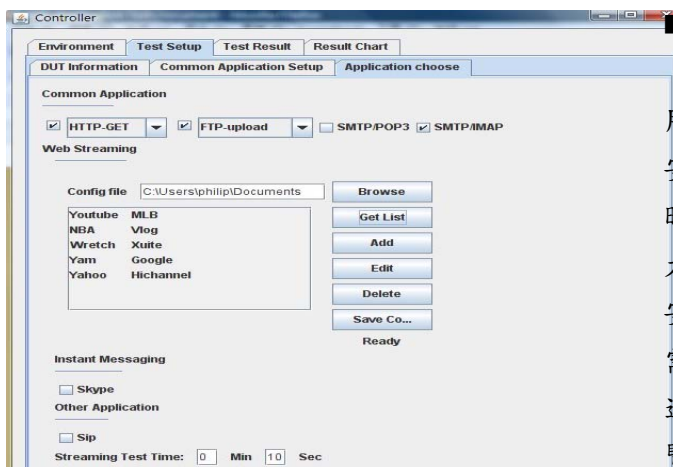
- 嵌入式系統共通測試平台(Embedded Common Test Platform, CTP 1.1.1)  
為了測試嵌入式網路通訊裝置所需要

的有線及無線網路的應用層流量，總計畫開發應用層測試工具(App-Test Tool)針對不同的網路服務，提供各種需要進行測試的應用層流量來對待測裝置以自動化的方式進行流量測試，目前可以設定的流量種類(如圖二所示)有一般常用協定的應用層流量：超文件傳輸協定(Hypertext transmission protocol, HTTP)、檔案傳輸協定(File Transfer Protocol, FTP)、簡單信件傳輸協定(Simple Mail Transfer Protocol, SMTP)、郵局通訊協定(Post Office Protocol 3)，串流(Streaming)流量或者是即時訊息(Instant Message, IM)流量，進行各種應用層流量測試。



圖三 測試過程的 throughput 變化及測試結果曲線圖

透過總計畫開發的應用層測試工具就可以提供一個共通測試平台做為四項子計畫的測試基底，觀察在該流量之下，對待測裝置的效能影響及評比。



圖二 設定需要產生的應用層流量及參數

應用層測試工具在執行的過程中也能夠得知所有應用層的吞吐量(throughput)，從圖三可以看出所有流量上下傳的最高速度及平均速度，讓測試工程師可以觀察待測裝置的吞吐量變化，最後應用層測試工具會根據這些吞吐量的變化將其繪製成測試結果曲線圖用來做進一步的分析。

Environment	Test Setup	Test Result	Result Chart			
Application	Instant Spe...	Max Speed	Average Speed	Total Receive	DUT Setting	Result
Web Strea...	14 KB/s	93 KB/s		1072 KB		
Youtube	177 KB/s	662 KB/s		7688 KB		
MLB	26 KB/s	63 KB/s		826 KB		
NBA	178 KB/s	230 KB/s		3596 KB		
Xuite	21 KB/s	97 KB/s		1799 KB		
Google	94 KB/s	289 KB/s		3884 KB		
Yahoo	124 KB/s	293 KB/s		5006 KB		
Vlog	102 KB/s	249 KB/s		3401 KB		
Wretch	23 KB/s	217 KB/s		3769 KB		
Yam						

### SSLTC (SSL VPN Tunnel Capacity)

SSLTC 為子計畫一所開發的測試工具，用戶端可以利用 VPN 方式連線，進行私密安全連線，以防止資料在網路上進行傳輸時，遭到駭客竊取，但是加密連線必須使用大量的運算，以及過濾，使得該類型的網路安全產品可容納的 VPN 容量無法達到實際需求，考量在大部份的情況下，利用 VPN 進行連線時，許多的連線都是在進行網頁瀏覽，因此並非需要在 Client 和 Server 端之間保持建立連線的狀況態，也可由此減輕 VPN 網通裝置的負擔，利用 SSL VPN 的方式進行資料的流通，以 Linux 為 SSL VPN 的 client，產生同時多條的 SSL VPN tunnels (Full Tunnel Mode)與 users 來連上 SSL VPN server，可用來自動化測試 tunnel capacity、user capacity、stability。

本系統的軟體架構主要是由幾項元件所構成: Traffic Generator– SSL VPN Tunnel, Traffic Generator – Background Traffic, Controller，以下說明各個元件所扮演的角色以及如何進行測試與蒐集數據：

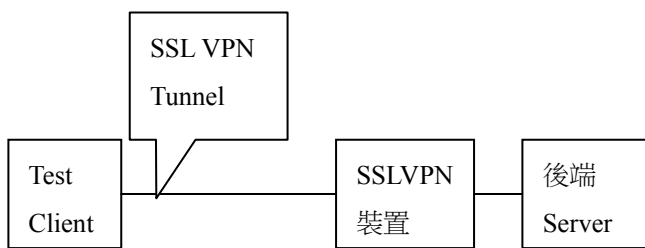
**Traffic Generator – SSL VPN Tunnel：** 使用一台 PC 製造出同時上千條的 SSL VPN tunnels。



**Traffic Generator – Background Traffic**：除了建立 SSL VPN tunnel 外，也可產生網路流量導入 SSL VPN tunnel 中，如 HTTP、FTP.. 等等。

**Controller**：會使用一台 PC 作為 controller，controller 的功能一是用來 trigger traffic generators，二是用來抓取 DUT 上的 information 來判斷該次測試的結果是 Pass or Fail。

SSLTC 之系統架構如下所示：



SSLTC 針對特定的 SSL (Secure Socket Layer) 虛擬私有網路 (Virtual Private Network, VPN) 的裝置，根據其連線的規範開發連線程式，透過個人電腦的方式模擬出多個 SSL VPN 的客戶端(Client)和 SSL VPN 裝置進行連線(如圖四所示)，測試不同的 SSL VPN 裝置最多可建立的通道(Tunnel)數量，並提供吞吐量測試來進行單一通道及合併所有通道之吞吐量測試，再透過總計畫的應用層測試工具將產生不同的應用層背景流量，將其導入至 SSL VPN 通道中，就可以評比不同的 SSL VPN 裝置在導入不同的應用層流量至 SSL 通道時，傳輸狀態的穩定度以及吞吐量效能。

#	User	Access	Login Address	Connected Time	Inbound (Bytes)	Outbound (Bytes)
1	vpn3	Network-Extension	192.168.1.235	00:00:27	160	250
2	vpn4	Network-Extension	192.168.1.235	00:00:21	160	250
3	vpn5	Network-Extension	192.168.1.235	00:00:15	160	250
4	vpn6	Network-Extension	192.168.1.235	00:00:09	160	250
5	vpn7	Network-Extension	192.168.1.235	00:00:03	160	250

圖四 模擬多個 client 來測試 SSL VPN 裝置

#### ■ IVQT (Integrated Voice Quality Test)

IVQT(Integrated Voice Quality Test)為子計畫一所開發的語音測試工具，由於市面上有許多語音功能的產品都整合至嵌入式網路裝置中，因此本工具主要目的是提供一個語音測試的環境，以虛擬定位(Virtual Position)的技術來模擬嵌入式網路通訊裝置在移動式的行為，再以自動化的方式進行語音撥接的測試，最後透過總計畫提供背景流量的測試平台就可以用來觀查及分析嵌入式裝置在各種不同的網路情境下的語音通訊品質。IVQT系統的軟硬體架構主要是由幾項元件所構成：

#### **NIST-Net Controller**：需能支援使用

NIST-Net 控制對於網路環境的影響，如在收話端與受話端之間 Traffic 受環境影響的控制，封包延遲或 Loss 等條件，用以測試在該環境下對於語音通話品質的影響。

#### **Background Traffic Generator**：需能支援

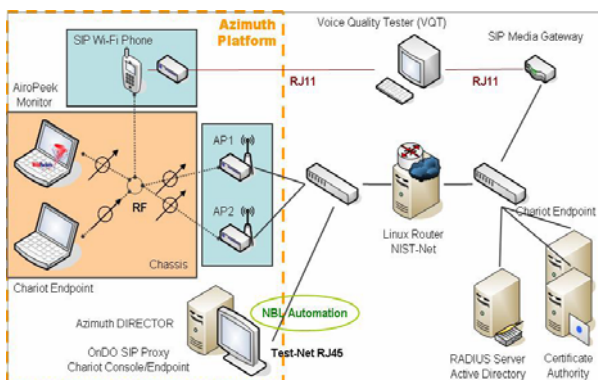
在不同型態的 Background Traffic 之下，擷取語音相關的封包進行語音品質測試，Background Traffic 可以為 HTTP, FTP, or P2P etc.，藉以觀察在不同的 traffic 以及不同的 loading 之下對 Voice Quality 的影響。

#### **Azimuth Platform(For Wireless Only)**：利

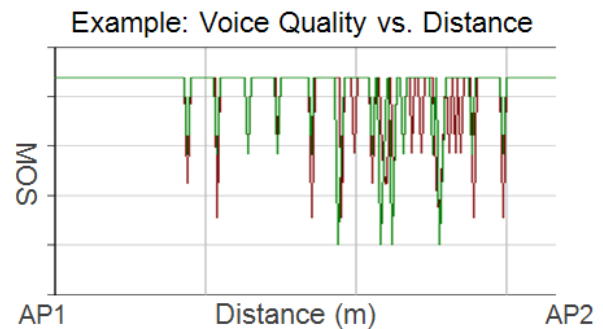
用 Azimuth 平台進行 RF Attenuator 控制，可以擬似對於 WiFi Phone 的移動模式，藉以

測試在移動狀態中的 WiFi Phone 對於 Voice Quality 的影響，藉由本測試可以分析關於 WiFi Phone 與 AP 在移動狀態下的適應性，由本測試可以進一步進行在移動中需進行 Roaming 時，WiFi Phone 對於 Roaming 機制的調適，選擇以何機制和時機進行 Roaming 的動作，以及在 Roaming 的同時對於語音品質的影響為何。

**Integrated Controller**：上面的測試都是只有針對單一的情況進行測試，但是這樣的測試無法進行多次的重複使用，每次進行測試時都要重新再設定不同的環境和不同的條件及劇本，做一次測試要花費許多的時間進行其它工具的設定，再者，在動態設定的情況下，會造成各工具之間 Timing 的非同步，造成測試不精確，因此本項目支援的目的是要做到在測試時可以利用單一視窗介面，進行對 WiFi Phone, NIST-Net 和 Azimuth 同步的控制以及批次處理。其系統架構如下所示：



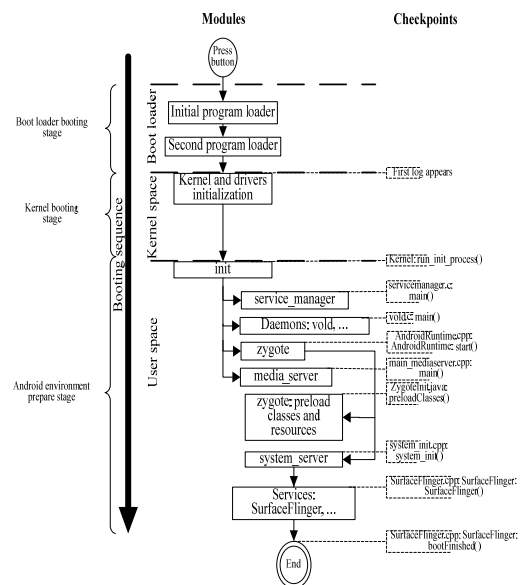
圖五為一嵌入式網路式裝置從無線網路基地台 AP1 行進至 AP2 時，在進行交遞 (Handover) 時的語音品質狀況，從圖中可以看出該裝置距離 AP1 越遠時語音品質會不斷下降，行進至 AP1 及 AP2 的中間位置時語音訊號會不斷飄移，其語音品質也會是最差的狀況，藉由虛擬定位的技術即可對不同支援語音功能的嵌入式網路通訊裝置進行交遞測試的評比。



圖五 Handover 的語音品質狀況

## ■ 嵌入式系統應用效能評比測試 (Embedded System Performance, ESP 1.2.1)

子計畫一提供測試嵌入式裝置效能評比工具。主要目的是希望透過運行各種分析工具模組，使之能應付嵌入式系統多層架構與多種程式語言特性(如 Android 的多種語言架構)，同時也解決嵌入系統上儲存空間有限造成分析結果 log 的儲存空間不足的問題。以下是於開機時間分析評測流程圖：



圖六 開機流程

按下電源開關後，Initial Program Loader (IPL) 會檢查硬體元件並啟動 Second Program Loader (SPL)，接著 SPL 會依序將壓縮的 kernel image 從 Flash 載到記憶體、將 image 解壓縮，然後執行 kernel 裡的第一個函式——start\_kernel。kernel 首先初始化各個資料結構(data structure)和任務(task)，並加載入驅動程式。接著，user space 的程式依序初始化並開始執行——service\_manager、daemon programs、zygote 及 media\_server。

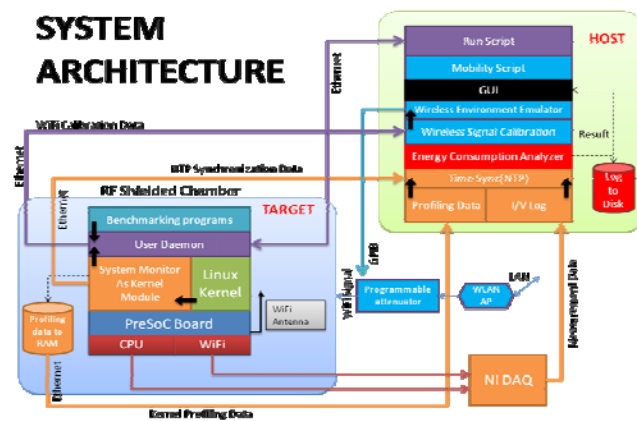
Zygote繼續預載一些Java class和resource，以加速Java應用程式，然後啟動system\_server。

system\_server 是一個管理所有的Android服務的程式，。它可以掃描flash，找出所有已安裝的應用程式，然後為每一程式創建thread以提供服務。其中一個重要的服務是負責螢幕拖曳的SurfaceFlinger，在啟動過程也結束於SurfaceFlinger裡的bootFinished()執行完成。

根據上述啟動程序，我們分為開機過程分為三個區塊：導引程式、kernel space和user space，最初的三個checkpoint即被設置於這三個區塊的邊界。然後，我們使用了階段性插入的方法的逐啟動效能分析，並確定了以下七個重要checkpoint：service\_manger、daemon、zygote、media\_server、JAVA class & resource、system\_server以及SurfaceFlinger()。

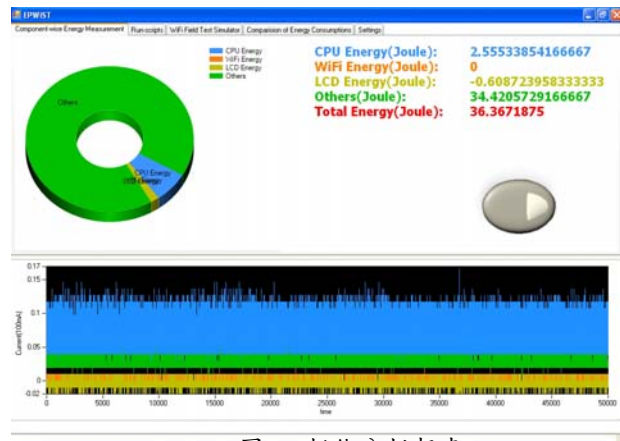
## ■ 耗電測試分析評比工具(Embedded Network and Communication Devices, EPC 1.3.1)

Power Memo 為子計畫二所開發用來量測嵌入式裝置為的耗電工具，主要的設計目的是提供一個系統負載度低且能夠有效精準剖析系統耗電資訊的工具，並且可同時搭配無線網路的測試環境，在動態測試中觀查移動中的裝置耗電量。下方是PowerMemo的系統架構圖：



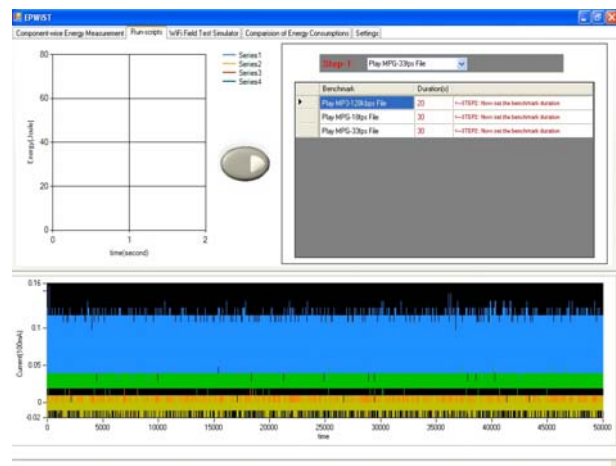
圖七 PowerMemo系統架構圖

主要區分成Host端及Target端兩個部分，在Host端有開發一個量測工具，主要目的同時記錄系統上各裝置的耗電狀況，並著重在耗電分析，希望以可讀性佳的圖表來呈現的耗電結果，不但能同時量測各components的耗能狀況，並對耗能分析產生報表與圓餅圖，如圖八所示。



圖八 耗能分析報表

在Target端的部分又可分為兩部分，第一部分是開發Benchmark程式進行自動化測試。主要是建立communication-intensive benchmark，像是streaming MP3, streaming video (可在Host端自行定義要在target上執行的benchmark腳本，如圖九所示)。



圖九 建立Benchmark腳本

第二部分是Control Daemon的開發，主要是提供host控制target的管道，可以由host任意執行或停止benchmark，由host掌控在target所有的benchmarks藉此達

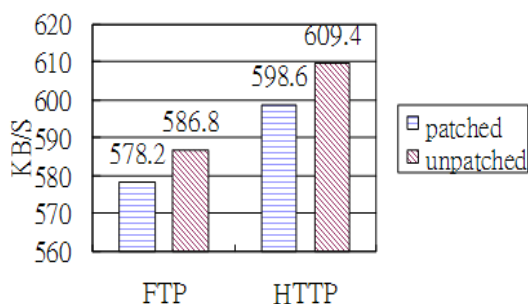


到大量自動化量測的目標。

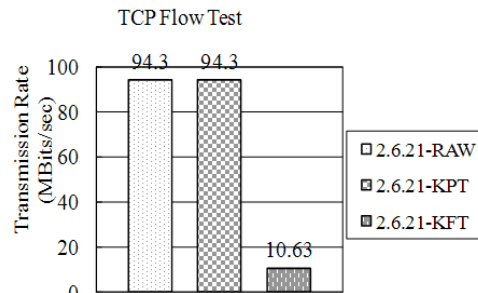
## ■ 核心與網路協定行為評比技術與工具 (Kernel Profiling Tool, KPT)

核心與網路協定行為評比技術與工具 (KPT) 為子計畫三所開發的工具，目的是用來追蹤嵌入式網路通訊裝置的網路應用程式在執行時，封包所經的核心函式、觸發的核心事件以及網路封包都能被記錄，並進一步分析出嵌入式系統的整體網路行為。本工具透過指令嵌入技術取得核心系統與網路協定互動的相關重要資訊，諸如時間標記、網路封包相關資訊和核心函式執行順序等等。之後將取得的核心行為記錄 (Kernel Behavior Log, KBL)，與網路封包擷取工具取得的網路協定行為記錄 (Protocol Behavior Log, PBL) 進行整併，即可剖析暨除錯核心系統內部的網路通訊協定堆疊，進一步地分析網路封包傳送與接收過程的效能。

由於核心系統可能受到追蹤與量測技術的干擾，因此 KPT 利用暫存機制與背景移轉機制，減少追蹤與量測技術對核心系統的影響。由圖九的超文件傳輸 (HTTP) 及檔案傳輸 (FTP) 測試可以看出對於加入核心分析功能的核心系統及沒有加入核心分析的核心系統，其網路效能差距約為 1.3% 與 1.7%，由此得知本工具對核心系統的影響很小。



圖十 核心系統之傳輸速率比較



圖十一 核心系統之傳輸速率比較

再從另一個測試角度來看，將此 KPT 與其他的核心系統追蹤工具 (在此列舉 Kernel Function Trace, KFT) 以及未經修改的 kernel，來比較 TCP 的傳輸速率，圖十一的結果可得知本工具對於核心系統行為並無顯著的影響。

圖十二為 KPT 實際測試的視覺化結果，透過核心事件擷取與自動核心函式追蹤的補釘嵌入技術就能夠取得網路封包在核心系統內的網路協定行為與核心函式互動的資訊，讓使用者可以探討待測裝置的核心事件發生的原因與時機，了解網路封包於核心系統內的網路協定堆疊之運作流程，並追蹤與量測一個網路封包從啟動到處理完畢，其間所經過的延遲與反應時間。

然而核心系統原始程式非常龐大且牽涉到太多與網路通訊協定無關的處理，因而提出兩個方法來找出相關程式片段，此兩方法為選擇性指令嵌入 (Selective Instrumentation) 與封包關聯 (Packet Association) 技術。圖十三為選擇性指令嵌入技術的架構，分成三個部分：

- (1) Polymorphism Module 負責調查原始程式碼，以及尋找 structsk\_buff 的所有變形，將結果儲存於 Polymorphism Table。
- (2) Direct use function Module 從 Polymorphism Table 取得 structsk\_buff 的所有變形後，負責搜尋與嵌入指令於 Direct use

function，將搜尋結果儲存於 Direct use Table。

(3) Indirect use function Module 從 Direct use Table 取得所有 Direct use function 的資訊，負責搜尋與嵌入指令於 Indirect use function，最後將搜尋結果儲存於 Indirect use Table。

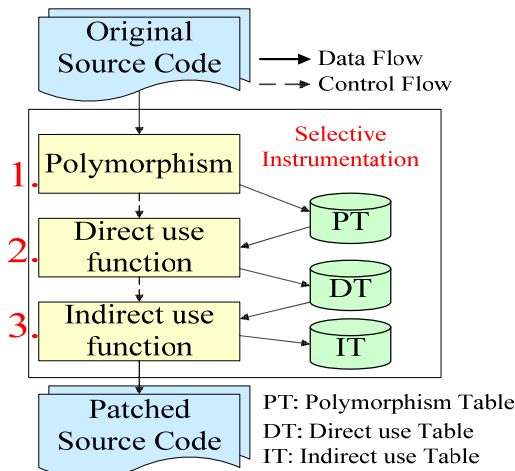
核心系統的原始程式碼經過這三個流程之後，產生嵌入剖析與除錯指令的原始程式碼。

No	Time	Source	Destination	Protocol	Ident
1	09:31:33.743698	140.113.24.65	140.135.104.56	UDP	Ident: 0000 S-Port: 47597 (Other), D-Port:
2	09:31:33.760409	140.135.104.56	140.113.24.65	UDP	Ident: CF25 S-Port: 37953 (Other), D-Port:
3	09:31:33.760427	140.113.24.65	140.135.104.56	UDP	Ident: 0000 S-Port: 47597 (Other), D-Port:
4	09:31:33.787289	140.135.104.56	140.113.24.65	UDP	Ident: CF26 S-Port: 37953 (Other), D-Port:
5	09:31:33.805059	140.113.24.65	140.135.104.56	TCP	Ident: C015 S-Port: 44362 (Other), D-Port:
6	09:31:33.805084	140.113.24.65	59.120.111.36	UDP	Ident: 0000 S-Port: 47597 (Other), D-Port:
7	09:31:33.823047	140.135.104.56	140.113.24.65	TCP	Ident: CF28 S-Port: 37953 (Other), D-Port:
8	09:31:33.823065	140.113.24.65	140.135.104.56	TCP	Ident: C016 S-Port: 44362 (Other), D-Port:
9	09:31:33.823072	140.113.24.65	140.135.104.56	TCP	Ident: C017 S-Port: 44362 (Other), D-Port:
10	09:31:33.841530	59.120.111.36	140.113.24.65	UDP	Ident: E705 S-Port: 31208 (Other), D-Port:
11	09:31:33.845678	140.135.104.56	140.113.24.65	TCP	Ident: CF2C S-Port: 37953 (Other), D-Port:
12	09:31:33.845698	140.113.24.65	140.135.104.56	TCP	Ident: C018 S-Port: 44362 (Other), D-Port:

No	Time	Event
1	09:32:52.275363	NETWORK wlan0 State: Link DOWN
2	09:33:04.279020	ROUTING Deleted 140.113.24.0/24 dev wlan0 src 140.113.24.65
3	09:33:04.279069	ROUTING Deleted default via 140.113.24.254 dev wlan0
4	09:33:04.291885	NETWORK Deleted wlan0 inet 140.113.24.65
5	09:33:04.292436	ROUTING Deleted broadcast 140.113.24.255 dev wlan0 src 140.113.24.65
6	09:33:04.292752	ROUTING Deleted broadcast 140.113.24.0 dev wlan0 src 140.113.24.65
7	09:33:04.292775	ROUTING Deleted local 140.113.24.65 dev wlan0 src 140.113.24.65
8	09:33:04.293287	NETWORK Deleted wlan0 inet fe80::222:68ff:fe17:684

圖十二 KPT 實際測試的視覺化結果



圖十三 選擇性指令嵌入技術之架構

而封包關聯技術是為取得核心行為與網路協定行為之間的互動資訊，用以整併核心行為記錄與網路協定行為記錄，於不影響

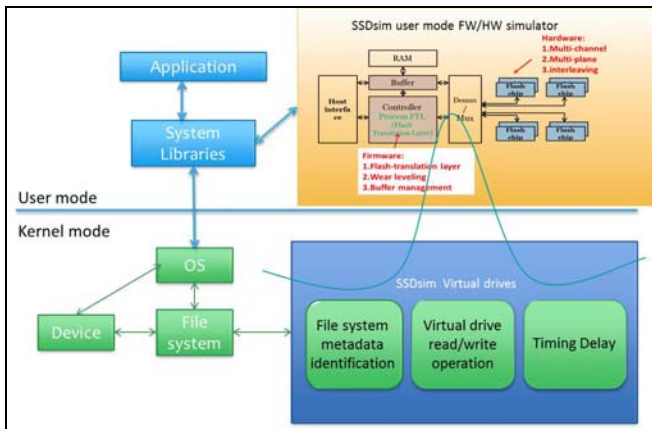
核心系統運作的行為情況之下，從核心系統內部取得最少的封包相關資訊，達成整併 KBL 與 PBL 的目標。

### ■ 嵌入式儲存裝置測試系統 (Embedded Storage Benchmark System, ESBS)

子計畫四提供嵌入式儲存裝置測試系統(ESBS)，主要是針對快閃記憶體(Flash Memory)為基礎之儲存系統，設計與實作出完整之效能評比程序、工具以及技術。目前已完成計畫中固態硬碟儲存裝置(Solid state disk, SSD)的效能評比與分析，透過收集各種 Workload 並針對固態硬碟的特徵進行分析，將相似特徵的集成評比套件 (Benchmark Suite)，並且歸納這些特徵對固態硬碟管理議題造成的影響。研究中，在於針對 NAND Flash-based 的儲存設備，提出一個韌體演算法( Firmware Algorithm )和硬體架構( Hardware Architecture )的標準制定原型與效能評估工具，使用比傳統硬體測試更加快速的軟體 simulator 技術來設計 Flash memory storage system。韌體模擬部分提供定義的標準模型總結了各種韌體演算法的抽象化概念並實作為 API，該 API 讓使用者能夠快速制訂各種 FTL( Flash Translation Layer )韌體演算法，硬體架構模擬部分則使用 SystemC 來模擬各種快閃記憶體的架構。

此外，本研究計畫也結合虛擬磁碟 ( Virtual Disk )技術作為使用者操作介面，使用者可通過對虛擬磁碟進行一般磁碟存取操作(如格式化、寫入檔案...等)，並實際感受所模擬的硬體架構及韌體演算法搭配後的時間延遲( Time delay )現象，結合 simulator 所產生之 result，可正確評估各種不同硬體架構及韌體演算法搭配的效能。因此本研究不僅能有效解決 NAND Flash-based 的儲存設備開發週期過長的問題，幫助開發者找到適合的設計，並設計與實作出完整之效能評比程序、工具以及技術。

但若用 ESBS 測試則更可以清楚看到就細部的管理機制，以圖十三的垃圾回收管理機制(Garbage Collection, GC)為例 (X 軸為發出要求順序, Y 軸為存取位址, Z 軸為回應時間), OCZ 的 GC 管理機制是優於 Transcend 的, 造成整體效能有極大落差的主要原因在於硬體條件的不同, Transcend 是採用單層式儲存 (Single Level Cell, SLC) 的技術, 而 OCZ 是用多層式儲存 (Multi Level Cell, MLC) 的技術, 因此如果是硬體條件在相似的狀況下, ESBS 效能評比測試工具比 IOMeter 更能適當的選擇出符合特定用途所需要的最佳管理機制儲存裝置。



圖十四 Flash memory storage system 系統架構圖

本研究提出一個韌體演算法 ( Firmware Algorithm ) 和 硬體架構 ( Hardware Architecture ) 的標準制定原型與效能評估工具, 使用比傳統硬體測試更加快速的軟體 simulator 技術來設計 Flash memory storage system (如圖十四)。韌體模擬部分提供定義的標準模型總結了各種韌體演算法的抽象化概念並實作為 API, 該 API 讓使用者能夠快速制訂各種 FTL ( Flash Translation Layer ) 韌體演算法, 硬體架構模擬部分則使用 SystemC 來模擬各種快閃記憶體的架構。

此外, 本研究計畫也結合虛擬磁碟 ( Virtual Disk ) 技術作為使用者操作介面, 使用者可通過對虛擬磁碟進行一般磁碟存取

操作(如格式化、寫入檔案...等), 並實際感受所模擬的硬體架構及韌體演算法搭配後的時間延遲 ( Time delay ) 現象, 結合 simulator 所產生之 result, 可正確評估各種不同硬體架構及韌體演算法搭配的效能。因此本研究不僅能有效解決 NAND Flash-based 的儲存設備開發週期過長的問題, 幫助開發者找到適合的設計, 並設計與實作出完整之效能評比程序、工具以及技術。

## [4] 研究成果與討論

總計畫已建立出測試平台並能夠導入網路流量進行測試, 各子計畫也已陸續完成各工具的開發。未來的一年除了持續將各子計畫的開發工具加強其完整合並和總計畫提供的測試平台進行結合, 並且也會陸續對產品在耗電、時間及 SSD 的成效增加測試案例來對嵌入式網路通訊裝置進行評比, 透過這一系列從黑箱到白箱的垂直整合測試方法來幫助開發商選擇適當的元件, 最後再對整合後的嵌入式裝置進行效能的評比, 讓研發工程師能夠進行問題追蹤及產品改善, 發揮嵌入式通訊裝置最大的優勢以滿足市場及消費者的需求。

## [5] 参考文献

- [1] "Using the NCTUns 2.0 Network Simulator and Emulator to Facilitate Network Researches", S.Y. Wang - Proc. 2nd International Conference on Testbeds and Research, 2006.
- [2] "Harpoon: a flow-level traffic generator for router and network tests", J. Sommers, H. Kim, P. Barford - ACM SIGMETRICS Performance Evaluation Review, 2004.
- [3] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs. Iperf 1.7.0 -- the TCP/UDP bandwidth measurement tool. <http://dast.nlanr.net/Projects/Iperf>, 2004.
- [4] Paul Barford , Mark Crovella, Generating representative Web workloads for network and server performance evaluation, Proceedings of the 1998 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems, p.151-160, June 22-26, 1998, Madison, Wisconsin, United States.
- [5] Charles R. Simpson, Jr. , Dheeraj Reddy , George F. Riley, Empirical Models of TCP and UDP End-User Network Traffic from NETI@home Data Analysis, Proceedings of the 20th Workshop on Principles of Advanced and Distributed Simulation, p.166-174, May 24-26, 2006.
- [6] Jelena Mirkovic , Brett Wilson , Alefiya Hussain , Sonia Fahmy , Peter Reiher , Roshan Thomas , Stephen Schwab, Automating DDoS experimentation, Proceedings of the DETER Community Workshop on Cyber Security Experimentation and Test on DETER Community Workshop on Cyber Security Experimentation and Test 2007, p.4-4, August 06-07, 2007, Boston, MA.
- [7] J. Sommers, P. Barford, Self-configuring network traffic generation, Proceedings of the 4th ACM SIGCOMM conference on Internet, 2004.
- [8] CoralReef: Passive network traffic monitoring and statistics collection. <http://www.caida.org/tools/measurement/coralreef>. Accessed August 2004.
- [9] Jing Cong , Bernd E. Wolfinger, A unified load generator based on formal load specification and load transformation, Proceedings of the 1st international conference on Performance evaluation methodologies and tools, October 11-13, 2006, Pisa, Italy
- [10] Joel Sommers , Vinod Yegneswaran , Paul Barford, A framework for malicious workload generation, Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, October 25-27, 2004, Taormina, Sicily, Italy
- [11] Michele C. Weigle , Prashanth Adurthi , Félix Hernández-Campos , Kevin Jeffay , F. Donelson Smith, Tmix: a tool for generating realistic TCP application workloads in ns-2, ACM SIGCOMM Computer Communication Review, v.36 n.3, July 2006
- [12] Endace measurement systems. <http://www.endace.com/>. Accessed August 2004.
- [13] SSFnet network simulator. <http://www.ssfnet.org>. Accessed August 2004.
- [14] UCB/LBNL/VINT Network Simulator - ns (version 2). <http://www.isi.edu/nsnam/ns>. Accessed August 2004.



- [15] Kashi Venkatesh Vishwanath , Amin Vahdat, Realistic and responsive network traffic generation, ACM SIGCOMM Computer Communication Review, v.36 n.4, October 2006
- [16] Network Emulation Testbed.  
<http://www.emulab.net>.
- [17] Spirent, <http://www.spirentcom.com/>.
- [18] IXIA, <http://www.ixiacom.com/>
- [19] Network Benchmarking Lab,  
<http://www.nbl.org.tw/>.
- [20] Toshihiro Kobayashi and Tetsuo Yamabe , “Linux DSP Gateway Specification , ” v3.3.1 , Nokia Corporation , 2006.
- [21] “OMAP5912 Applications Processor Data Manual , ” Texas Instruments , March 2004.
- [22] “DSP/BIOS LINK , USER GUIDE” Texas Instruments , March 2007.
- [23] “A. Sinha and A.Chandrakasan , “JouleTrack – A Web Based Tool for Software Energy Profiling , ” Proc. 38th Design Automation Conference , “June 2001
- [24] ”Energy Estimator/Analyzer for Embedded Dual Core Processor” , Chun-Hao Hsu , 2006” Measuring System Performance on the TMS320DM644x Platform” Texas Instruments , March 2007.
- [25] Chuanxiong Guo and Shaoren Zheng , “Analysis and Evaluation of the TCP/IP Protocol Stack of Linux”, International Conference on Communication Technology Proceedings, vol 1, pp. 444 –453, Aug. 2000.
- [26] Wikipedia – “GNU”, Available from: <http://en.wikipedia.org/wiki/GNU>
- [27] Ying-Dar Lin, and Ping-Tsai Tsai, “Trace
- [28] Linux TCP/IP kernel – Using remote debugging”, Available from: [http://speed.cis.nctu.edu.tw/~ydlin/miscpub/remote\\_debug.pdf](http://speed.cis.nctu.edu.tw/~ydlin/miscpub/remote_debug.pdf)
- [29] Peter Jay Salzman, Michael Burian, and Ori Pomerantz, “The Linux Kernel Module Programming Guide”, Dec. 2005.
- [30] Bryan Henderson, “Linux Loadable Kernel Module HOWTO”, Available from: <http://www.tldp.org/HOWTO/Module-HOWTO/>
- [31] Jongmoo Choi, “Kernel aware module verification for robust reconfigurable operating system”, Journal of Information Science and Engineering, Vol. 23 No. 5, pp. 1339-1347, Sep. 2007.
- [32] Moore, R. J., “Dynamic probes and generalized kernel hooks interface for Linux”, Proceedings of the fourth annual Linux showcase and conference. Atlanta, GA, USA, pp. 139-145, 2000.
- [33] Nicolas Lorient and Jean-Marc Menaud Generalized, “Dynamic Probes for the Linux Kernel and Applications with Arachne”, In Proc. of the 2007 IADIS Conference on Applied Computing, Feb. 2007.
- [34] R. Krishnakumar, “Kernel Korner: Kprobes - a Kernel Debugger”, Linux Journal, Jun. 2006.
- [35] Jeong-Won Kim, Young-Uhg Lho, Young-Ju Kim et al, “A memory copy reduction scheme for networked multimedia service in Linux kernel”, Lecture Notes in Computer Science, 2002, 2510: 188–195.
- [36] 曹敏峰, " Linux 網路通訊協定堆疊之高效率動態的指令嵌入平台之設計與實作

- ", 交大資工所碩士論文, 2008.
- [37] G. Irlam, "Unix File Size Survey," 1993.
- [38] <http://www.gordoni.com/ufs93.html>
- [39] A. Inoue and D. Wong, "NAND Flash Applications Design Guide", April, 2003.
- [40] Li-Pin Chang and Tei-Wei Kuo, "Efficient Management for Large-Scale Flash-Memory Storage Systems with Resource Conservation," ACM Transactions on Storage, Volume 1, Issue 4, 2005.
- [42] Intel Corporation, "Understanding the Flash Translation Layer (FTL) Specification".
- [43] SSFDC Forum, "*SmartMediaTM* Specification", 1999.
- [44] Compact Flash Association, "*Compact FlashTM* 1.4 Specification," 1998.
- [45] Linux MTD project and M-System, "NAND Flash Memory Translation Layer (NFTL)."
- [46] Nitesh Goyal and Rabi N Mahapatra "Energy Characterization of CRAMFS for Embedded Systems", Proceedings of International Workshop on Software Support for Portable Storage (IWSSPS), March 2005.
- [47] Siddharth Choudhuri, Rabi Mahapatra, "Energy Characterization of Filesystems for Diskless Embedded Systems", Design Automation Conference (DAC), June 2004.
- [48] Aleph One Company, "[Yet Another Flash Filing System](#) (YAFFS) ""
- [49] L. P. Chang and T. W. Kuo, "An Adaptive Striping Architecture for Flash Memory Storage Systems of Embedded Systems," 8th IEEE RTAS, September 2002, pp. 187-196
- [50] Han-joon Kim and S. Lee, "A New Flash Memory Management for Flash Storage System.", In Proceedings of the 23rd Annual International Computer Software and Applications Conference, pages 284–289, 1999.
- [51] Ousterhout, J., Da Costa, H., Harrison, D., Kunze, J., Kupfer, M., and Thompson, J., A Trace- Driven Analysis of the UNIX 4.2 BSD File System, Proceedings of the Tenth Symposium on Operating Systems Principles, Orcas Island WA, December 1985, pp. 15-24.
- [52] Atsuo Kawaguchi et al., "A Flash-Memory Based File System", in USENIX Technical Conference, 1995
- [53] Linux MTD Project, "Journaling Flash File System (JFFS), Journaling Flash File System 2 (JFFS2), and Journaling Flash File System 2 (JFFS3)."
- [54] Bray, T, "The Bonnie home page.",
- [55] <http://www.textuality.com/bonnie>, 1996.
- [56] Katcher, J., "PostMark: A New Filesystem Benchmark. Tech. Rep. TR3022, Network Appliance.", [www.netapp.com/techlibrary/3022.html](http://www.netapp.com/techlibrary/3022.html), 1997
- [59] William D. Norcott., Don Capps, "Iozone Filesystem Benchmark", [http://www.iozone.org/docs/IOzone\\_msword\\_98.pdf](http://www.iozone.org/docs/IOzone_msword_98.pdf)
- [61] N. Joukov, A. Traeger, CP Wright, Zadok,
- [62] "Benchmarking File System Benchmarks",
- [63] ETechnical Report FSL-05-04b, CS department Stony Brook University, 2005.
- [64] John H. Howard, Michael L. Kazar, Sherri G. Menees, David A. Nichols, M. Satyanarayanan, Robert N. Sidebotham, and Michael J. West., "Scale and performance in a distributed file system.", ACM Transactions on Computer Systems, 6(1):51–81, February 1988.
- [65] AIM Technology, "AIM Multiuser

- Benchmark - Suite VII Version 1.1. “,
- [66] <http://sourceforge.net/projects/aimbench> , 2001
- [67] Nathan Edel, Deepa Tuteja, Ethan L. Miller, and Scott A. Brandt, "MRAMFS: A Compressing File System for Non-Volatile RAM" ,IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2004)
- [68] R. Bryant, R. Forester, and J. Hawkes. "Filesystem Performance and Scalability in Linux 2.4.17." , In FREENIX '02, Monterey, CA, June 2002.
- [69] Sun, “File System Performance: The Solaris OS, UFS, Linux ext3, and ReiserFS”, White paper August 2004.
- [70] [http://sun.com/software/whitepapers/solaris10/fs\\_performance.pdf](http://sun.com/software/whitepapers/solaris10/fs_performance.pdf)
- [71] R. Bryant, R. Forester, and J. Hawkes. "Filesystem Performance and Scalability in Linux 2.4.17." , In FREENIX '02, Monterey, CA, June 2002.
- [72] Iometer Project. “Iometer” ,OPEN SOURCE DEVELOPMENT LAB. 2004. [www.iometer.org/](http://www.iometer.org/).
- [73] A. Kawaguchi, S. Nishioka, and H. Motoda, “A Flash Memory based File System,” Proceedings of the USENIX Technical Conference, 1995.
- [74] F. Dougliis, R. Caceres, F. Kaashoek, K. Li, B. Marsh, and J.A. Tauber, “Storage Alternatives for Mobile Computers,” Proceedings of the USENIX Operating System Design and Implementation, 1994.
- [75] Veritest. “NetBench.” , [www.veritest.com/benchmarks/netbench/](http://www.veritest.com/benchmarks/netbench/) , 2002
- [76] A. Tridgell, “dbench-3.03 README.” <http://samba.org/ftp/tridge/dbench/README>, 1999.
- [77] Rosenblum M, Ousterhout J. “The Design and Implementation of a Log-Structured File System.”, Proceedings of the 13th ACM Symposium on Operating Systems Principles, October 1991.
- [78] Mazieres, D. et al. , "Separating Key Management from File System Security", 17th ACM Symp. On Operating Systems Principles. (1999).
- [79] M. Kaminsky et al, “Decentralized User Authentication in a Global File System”, 19th ACM Symposium on Operating Systems Principles, 2003.
- [81] An-I Wang, Peter L. Reiher, Gerald J. Popek, Geoffrey H. Kuenning, “Conquest: Better Performance Through a Disk/Persistent-RAM Hybrid File System.”, USENIX Annual Technical Conference, General Track 2002: 15-28
- [82] Schmuck, F. and Haskin, R., ”GPFS: A Shared-Disk File System for Large Computing Clusters”, Proceedings of the Conference on File and Storage Technologies (FAST'02), 2002

# 國科會補助計畫衍生研發成果推廣資料表

日期:2011/10/28

國科會補助計畫	計畫名稱: 總計畫(2/2)
	計畫主持人: 林盈達
	計畫編號: 99-2220-E-009-043- 學門領域: 自由軟體暨嵌入式系統
無研發成果推廣資料	



99 年度專題研究計畫研究成果彙整表

計畫主持人：林盈達		計畫編號：99-2220-E-009-043-					
計畫名稱：嵌入式網路通訊裝置評比技術與工具之研發--總計畫(2/2)							
成果項目		量化			單位	備註(質化說明：如數個計畫共同成果、成果列為該期刊之封面故事...等)	
		實際已達成數(被接受或已發表)	預期總達成數(含實際已達成數)	本計畫實際貢獻百分比			
國內	論文著作	期刊論文	2	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	0	0	100%		
		專書	0	0	100%		
	專利	申請中件數	2	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	7	0	100%	件	
		權利金	4987	0	100%	千元	
	參與計畫人力 (本國籍)	碩士生	12	0	100%	人次	
		博士生	1	0	100%		
博士後研究員		0	0	100%			
專任助理		0	0	100%			
國外	論文著作	期刊論文	13	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	2	0	100%		
		專書	0	0	100%	章/本	
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力 (外國籍)	碩士生	0	0	100%	人次	
		博士生	0	0	100%		
博士後研究員		0	0	100%			
專任助理		0	0	100%			

<p>其他成果 (無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。)</p>	<p>無</p>
--	----------

	成果項目	量化	名稱或內容性質簡述
科 教 處 計 畫 加 填 項 目	測驗工具(含質性與量性)	0	
	課程/模組	0	
	電腦及網路系統或工具	0	
	教材	0	
	舉辦之活動/競賽	0	
	研討會/工作坊	0	
	電子報、網站	0	
	計畫成果推廣之參與(閱聽)人數	0	

# 國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

達成目標

未達成目標（請說明，以 100 字為限）

實驗失敗

因故實驗中斷

其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形：

論文： 已發表  未發表之文稿  撰寫中  無

專利： 已獲得  申請中  無

技轉： 已技轉  洽談中  無

其他：（以 100 字為限）

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

本計畫目的在於提供一系列完整的測試工具與測試方法，其中包括應用效能測試、耗電測試、分析與診斷機制、元件效能測試(例如：儲存裝置)，將其應用在目前主流的嵌入式網路通訊裝置，為其建造「嵌入式網路通訊裝置測試軟體發展中心(Embedded Benchmarking Lab, EBL)」來協助廠商及開發者評比嵌入式網路通訊裝置，進而對產品改良及修正，亦可做為日後學術相關研究時有實際的產品數據提供參考。於一般使用者更可依我們所提供的數據來參考及比較，選擇自己所需的產品。