行政院國家科學委員會專題研究計畫　期中進度報告

總計畫(1/3)

計畫類別：整合型計畫
計畫編號：NSC93-2213-E-009-110-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：國立交通大學電信工程學系(所)

計畫主持人：李程輝
共同主持人：孫雅麗，林盈達

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 94 年 5 月 30 日

行政院國家科學委員會補助專題研究計畫 □ 成 果 報 告
■期中進度報告

以內容為基礎之網路安全－總計劃

計畫類別：□ 個別型計畫　　■ 整合型計畫
計畫編號：NSC 93－2213－E－009－110－
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日

計畫主持人：李程輝教授
共同主持人：孫雅麗教授、林盈達教授

成果報告類型(依經費核定清單規定繳交)：■精簡報告　□完整報告

處理方式：本計劃可公開查詢

執行單位：交通大學電信系、台灣大學資訊管理系、交通大學資訊科學系

中　華　民　國　94　年　5　月　30 日

一、摘要

ABSTRACT

The progress of network speed and technology make network security tool workload increasing enormously. It seems hardly to solve this problem by software entirely, especially, on Internet content filtering. Although the software tool could cover almost entire network security jobs, software also is the performance bottleneck in nowadays network speed, including Antivirus, Web/URL Content filter and Anti-Spam etc. The purpose of this project is to implement the most consuming computation part by hardware(FPGA). Of course, the hardware must cooperate with embedded operation system, device driver and firmware to improve the performance.

In software, the embedded operation system had already work on Creator－PreSOC Board. The device driver for FPGA is developing in concurrent. In hardware, the prototype of string matching algorithm had already developed and upload to FPGA successfully, that includes both exact matching and regular expression matching. The more efficient and suitable algorithm for hardware implement is under construction. The firmware, that response for communication between hardware and software, will be developed in the next step. And debugging work will continue go on for hardware and software. Finally, we will do the whole system testing and debugging, the practical system architecture and performance will be presented.

KEYWORDS: FPGA, SOC, Embedded System, Network Security

中文摘要

隨著網路速度與技術不斷的提升，網路安全工具之工作量也越來越大。目前軟體的網路安全工具似乎已無法完全負擔沉重的工作量，尤其是網路內容過濾方面，包含病毒防護(Antivirus)、網頁內容過濾(Web/URL Content filter)和垃圾郵件過濾(Anti-Spam)等。雖然目前軟體工具可以涵蓋絕大部分之安全工作，但對於越來越快速的網路速度，軟體已成為瓶頸，而無法負荷如此龐大的工作量，因此將耗費大量計算資源的部分以硬體的方式達成，將是一種可行的方式，當然軟硬體的良好配合是效能提升的重要因素。本計畫之目的正是將最耗費計算資源的字串比對(String Matching)的演算法實現於 FPGA，並配合嵌入式作業系統(Embedded Operation System)以軟體，韌體與硬體互相整合之開發方法，以達到硬體加速軟體整合協調之目的。

軟體進度方面目前已經可將基本之嵌入式作業系統(Embedded Operation System)執行於實驗母版(Creator－PreSOC Board) 上，並正發展與硬體 FPGA 相關之驅動程式。硬體方面已經成功將字串比對(String Matching)演算法燒錄至 FPGA，其中包含精確比對(Exact Matching)與正規語言法比對(Regular Expression Matching)，並正發展更有效率且合乎硬體實現之字串比對演算法。下一步將同時進行軟體和硬體溝通協調的韌體(Firmware)發展工作與軟硬體的修正除錯工作，並發展合適之驅動程式，最後將進行軟硬體同步整合測試實驗，並實際展示此一系統之架構與效能。

關鍵字：嵌入式系統、字串比對、正規語言、網路安全

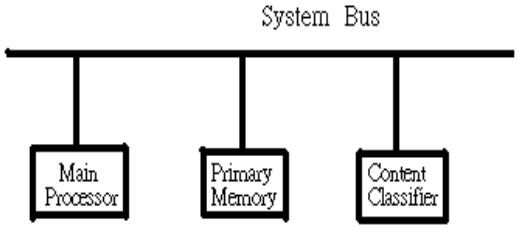
二、前言

網路技術日新月異，為工作、學術、生活等各方面帶來了莫大地幫助，但據統計，2000年企業及政府部門遭受駭客或病毒攻擊的高達 85%，因此網路安全逐漸成為人們重視的問題。網際網路、企業網路等網路應用的頻寬需求急劇上升，傳輸、檢查、拆解、組合、搜尋、內容比對、轉遞等 IP 封包運算處理動作，以往可以靠軟體程式在一般微處理器上執行，搭配以網路卡做封包出入口。但是近年來這些封包的運算處理越來越複雜，將資料輸入處理單元，完後再將結果送往輸出單元，慢速處理造成的時間延遲會嚴重影響到資料吞吐量，無法滿足線速率的操作需求。一般所用的網路安全工具主要是防毒、防火牆等軟體，但是在網路傳輸速度不斷增長的現在，單純只靠軟體負責網路安全已不足夠勝任，若能將網路安全軟體中最耗費資源的部份轉移到硬體上來實現的話，必能大幅度地增加網路安全工具的工作效率。

在大部分網路安全工具中，主要運算時間與資源是花費在字串比對(String Matching)上－將從網路上收到的資料與資料庫中(rule sets)的規則比對－因此要把此部分移至硬體上執行，以期達到加速的效果。而硬體開發最方便的方式就是採用 FPGA 的開發流程，而且若能配合現成的整合系統，就可解決軟體與硬體開發的困難處，並達成軟體硬體共同開發的效果。

三、計畫目標

本計畫的目標是研究適合硬體加速的高效率內容分類演算法與回溯機制，並以SoC(System of Chip)的架構製作雛型系統。此系統包括病毒與入侵偵測以及垃圾郵件與不當資訊過濾。目前公開程式碼(open source)已有偵測病毒與入侵以及過濾垃圾郵件與不當資訊的軟體模組，但是性能有限，當網路頻寬提升時，有必要採用硬體加速的技巧，避免本系統成為瓶頸，限制寬頻的應用。如圖一所示，本計畫採用共同處理器(co-processor)的架構，將內容分類機制獨立成個別處理器，分擔 CPU 的負荷，達到加速目的。



圖一　系統架構

四、子計畫間的合作情況

子計畫間的關係

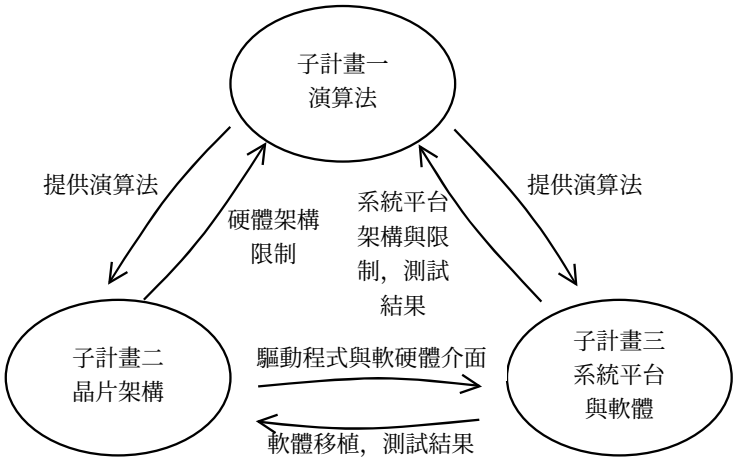
本計畫共分為三個子計畫，個別由孫雅麗教授(子計畫一)、李程輝教授(子計畫二)，與林盈達教授(子計畫三)執行。

子計畫一：「研究與實作在 SoC 環境下考慮狀態之網路內容分類的語言、編譯器與 執行引擎」負責內容分類演算法的設計。

子計畫二：「內容分類硬體加速器之設計與製作」負責 IC 硬體架構設計與實作。

子計畫三：「網路內容分類的系統架構：設計、實作、與評估」負責測試公開程式碼之性能、軟體移植與系統平台的建置。

三個子計畫的關連性表示於圖二。



圖二 子計畫之相關圖

子計畫一與子計畫二的互動是子計畫一提供演算法給子計畫二做硬體架構設計，而子計畫二提供子計畫一硬體的特性與架構設計的概念，協助子計畫一發展適合硬體加速的演算法。

子計畫一與子計畫三有類似的關連性。子計畫一提供子計畫三演算法來置換或植入現存於開放程式碼(Open Source Code)的軟體模組，而子計畫三提供子計畫一測試結果以及系統平台的環境與限制，以協助子計畫一最佳化演算法的設計。

至於子計畫二與子計畫三的關係，主要在共同製定軟、硬體的功能分割、介面的規格與驅動程式。

子計畫間的合作

由於本計畫擬開發雛型系統，因此三個子計畫之間必須密切聯繫、溝通。為了讓計畫

順利地進行，個別子計畫均有每個星期的固定討論會，讓各個子計畫中的成員們，報告自己的進展以及提出困難點加以討論，使每個人能夠在第一時間內了解自己所屬子計畫的最新進度。

另外每個月均有一次整體計畫的討論會(地點或在台北，或在新竹)。在整體計畫的討論會中，三個子計畫成員首先了解前次討論會中的結論與問題，由負責人就問題的現況、解決與否加以報告，再來分別對三個子計畫在這一個月來的進展提出報告。針對報告中所提出的問題，依照三個子計畫的個別性質，合力找出解決的方法。會議的最後，再訂出個別子計畫在未來一個月中的目標，相信透過充分的意見交流與互補有無的情況下，可以達到集思廣益的效果，提高成功開發雛型系統的機率。參與的研究人員在這種務實的討論之下，吸收了不少相關網路安全的新知識，並藉由開發雛型系統的機會，逐漸學習到如何開發一套系統與軟硬體的實務經驗。

五、設備器材購買

本計畫需要使用的設備是 SoC 平台，在計畫開始第一年之中，已先採購兩組供參與人員熟悉系統環境，預計在第二年中欲再採購一組(每個子計畫一組)，同時開始雛型系統的開發。

設備儀器名稱	數量
Creator－PreSOC Development Kit	X1
WINeZ ARM 模擬器 (ARM CPU ICE)	X1
Creator ARM922T-EPXA1 Board	X1

設備儀器簡介

簡介

1. Creator－PreSOC Development Kit：

Creator 是一個多用途的軟硬體發展平台，適合進行各項與”嵌入式系統”相關軟硬體設計之評估及學習之用。

Creator 使用 32 位元 RISC 架構之 ARM CPU 作為控制之核心，除了配置大量之記憶體空間供作業系統及應用程式使用之外，多樣化的周邊硬體設計舉凡網路、通訊、語音、影像、擴充界面支援大記憶體容量及 FPGA，多樣人機界面等，使得應用的範圍更多元。

除了上述硬體平台之外，Creator 的軟體套件包含了發展 Linux 應用程式所需之各種工具軟體如 Cygwin、GNU tools Chain、μClinux 核心原始程式碼、應用範例等等。如此一來使用者能夠很快的架設起該有的軟體發展環境，利用 Creator 的硬體平台及 ARM ICE 的輔助之下，直接進入應用程式設計的階段。

2. WINeZ ARM 模擬器(ARM CPU ICE)：

系統特點：

1. 採用 Stand Alone 設計，與 PC 傳輸採用並聯 Printer Port 界面。
2. **機型及支援 CPU**：ARM7 系列，可昇級為 ARM9 系列。
3. 可作同步 Reset/Trigger Match/Sync 輸出及 Sync Stop 等遠端控制。
4. 提供硬體匯流排中斷點，可設定 Address、Data Bus、CPU Status、4 External Trigger 及 256 pass counter，可作篩選及不限點數執行中斷點。

軟體模擬功能：

1. 高效率視窗導向跨多 CPU 平台之整合發展軟體－Domingo IDE。
2. 可直接下載 (Load Module) Bin、Hex、ELF、UBROF 等除錯程式。
3. IDE 整合發展環境可支援 ARM、IAR 等 C Compiler，可直接下載 GNU C compiler 之除錯格式並附操作流程。
4. 可作全螢幕線上行組譯、反組譯及可於程式視窗作 Symbolic 除錯，C 語言高階整合除錯。
5. 執行功能：Full Speed Running、Single Step Into/Over、Animate Step Into/Over、Go Here、Run & Trace Auto Stop 等。

3. Creator ARM922T-EPXA1 Board：

採用了 Altera Excalibur 系列的 EPXA1, 是一塊有成本效益且多用途的 SPOC 開發板。它不僅內建了 ARM922T，同時還有 100K 的邏輯閘可供設計之用。在軟體方面，可以在開發板上執行 Linux 作業系統，來驅動板子上的硬體周邊。另外，它還有 8MB Flash Memory、32MB SDRAM、工業用網路界面, 以及給 SOPC 使用的 AMBA lite function.

六、成果自評與未來展望

成果自評

目前根據現有資源設備，已經成功地在 Creator 上移植了 μClinux 2.4.x，其中更提供了網路功能，使得 Creator 開發版能夠連結到網路。同時也因為成功移植了 μClinux，讓我們可以在 Open Source 裡一些屬於 Linux 的軟體，從中尋找符合我們要求的軟體，或刪除我們不需要的軟體，進而免去了重新設計一套軟體的繁冗過程，並縮短驅動程式開發時程。

開放原始碼(Open Source Code)的網路安全工具中，如 Snort、ClamAV，在追究(Trace)完原始碼後，已經找出這兩套軟體所使用的字串比對原理，以及軟體如何引用字串比對的程式，這對於我們欲將字串比對的功能獨立在硬體上實現有著非常大的幫助。

這一年來，每個子計劃著眼在所擔負的領域內，蒐集了許多相關資料，訂定了各自的目標方向，並且在整體計畫討論會中提出來報告討論，大家集思廣益找出彼此的盲點，加以溝通解決。

所以就目前階段來說，不僅在軟硬體方面均有十足的進展，三個子計畫間更是合作無間，可以說是達成我們第一階段的目標。

未來展望

接下來一年之中，在軟體方面：將修改目前已有的網路安全工具，如 Snort、ClamAV，試著將其字串比對的部份分離出來，並配合新演算法，設計出所需要的參數，讓由硬體所負責的字串比對功能，能與軟體搭配合作。在硬體方面，研發尋找更有效率、更適於在硬體上實現的方法，使得硬體資源能夠充分地發揮，達到加速字串比對的目的。還有就是建立 FPGA 與實驗母板兩者的通訊介面，使得軟體與硬體能夠溝通無礙，達成兩者 Co-work 的目標。另外在子計畫合作方面，繼續維持每個月一次的討論會，讓彼此了解最新的進度與需求，以期在一年之後，能夠完成具有基本功能的雛型架構。

六、參考文獻

[1] T. Kojm. ClamAV. www.clamav.net, 2004.

[2] Aho, A. V., and M. J. Corasick, “Efficient string matching: an aid to bibliographic search,” Communications of the ACM 18 (June 1975), pp. 333-340.

[3] Boyer R. S., and J. S. Moore, “A fast string searching algorithm,” Communications of the ACM 20 (October 1977), pp. 762-772.

[4] S. Wu and U. Manber. A fast algorithm for multi-pattern searching. Technical Report TR-94-17, University of Arizona, 1994.

[5] R. Sidhu and V. K. Prasanna, “Fast Regular Expression Matching using FPGAs,” in IEEE Sysposium on Field-Programmable Custom Computing Machines (FCCM), (Rohnert Park, CA, USA), Apr. 2001.

[6] Gokhale, M., Dubois, D., Dubois, A., Boorman, M., Poole, S., Hogsett, V.: Granidt: Towards gigabit rate network intrusion detection technology. In: Proceeding of 12th International Conference on Field Programmable Logic and Applications. (2002) France.

[7] Moscola, J., Lockwood, J., Loui, R. P., Pachos, M.: Implementation of a content-scanning module for an internet firewall. In: Proceedings of IEEE Workshop on FPGAs for Custom Computing Machines. (2003) Napa, CA, USA.

[8] Young H. Cho, S.N., Mangione-Smith, W.: Specialized hardware for deep network packet filtering. In: Proceedings of 12th International Conference on Field Programmable Logic and Applications. (2002) France.

[9] F. Yu, R. H. Katz, and T. V. Laskhman, “Gigabit Rate Packet Pattern Matching with TCAM,” UCB technical report, UCB//CSD-04-1341, July 2004.

[10] I. Sourdis and D. Pnevmatikatos. Fast, Large-Scale String Match for a 10Gbps FPGA-Based Network Intrusion Detection System. In Proceedings of FPL2003, 2003.

[11] Sarang Dharmapurikar, Praveen Krishnamurthy, Todd S. Sproull, John W. Lockwood: Deep

Packet Inspection using Parallel Bloom Filters. IEEE Micro 24(1): 52-61 (2004).
[12] Nathan Tuck, Timothy Sherwood, Brad Calder, George Varghese: Deterministic Memory-Efficient String Matching Algorithms for Intrusion Detection. INFOCOM 2004.