以網路處理器實作無線安全語音閘道器

Implementation of Wireless Secure Voice Gateway in Network Processor-based Platform

周立德 陳志遠 曾柏嘉
國立中央大學 資訊工程所
Email: cld@csie.ncu.edu.tw

摘要

新世紀的網際網路 (Internet)面臨許多艱難的挑戰，如網路拓樸迅速地成長、即時 (real-time)多媒體應用需求大增、以及 VoIP 大符成長的需求…等等。舉例來說，在即時多媒體應用方面，視訊會議 (Video Conference)以及網路語音 (Voice over IP-VoIP)非常在乎資料傳輸的延遲時間，而 User Mobility 和網路安全性亦是日漸重視的議題，然而在傳統 VoIP Gateway 並沒有加裝 Security 和 Wireless 的機制，也無法保證其服務品質(QoS)，所以造成這類的應用服務無法完善地運作。為了解決現今 Network Device 上的窘境，我們實現嵌入式 Wireless Secure Voice Gateway，有效整合現存網路電話環境所需要的相關元件於嵌入式平台 IXP425 開發板上。此外為了達到 Voice Gateway 在 Data Plane 遞送封包的高效能，也必須移植(Porting)與編譯(Compiling) Codelets，使得 IXP425 的 Network Processor Engines (NPEs) 能夠支援 Voice Gateway 的相關運作流程。

我們在 IXP425 平台上完成了 Wireless Security Voice Gateway 所需的元件 (IEEE 802.11b/g Access Point、VPN IPSec Tunnel 和 Voice Gateway Function 等)，並且利用 IXP425 整合三個 NPEs 的高效能處理，使得 VPN IPSec Tunnel 並不會因為加上 Security 機制而造成傳輸效能上的流失，因此更能讓使用者在一安全性高且效能比一般 Voice Gateway 佳的網路環境下使用網路服務。

關鍵詞：*Network Processor, Voice Gateway, VPN IPSec, IEEE 802.11b/g*

1. 前言

一般來說，設備製造商為了要做出 Customer Premises Equipment (CPE) 和 Integrated Access Device (IAD)相關產品，需要具變動性佳且低成本的數位媒體信號處理(Digital Media Signal Processing)解決方式。然而目前市面上最常見到的解決方式就是透過獨立分開的晶片增加 Digital Signal Processing (DSP) Capability 而導致花費更高的 Bill of Materials (BOM) 成本，而且更增加了整合平台所需的成本和複雜性。有基於上述之理由，本論文研究的目的就在於：我們將 Voice Gateway 設計於嵌入式的開發板上，以 Intel IXP425 高效能處理網路封包(Intel XScale core at 533 MHz)及可程式化(Programmable)的特點，來解決一般在市面上的網路產品無法解決的窘境，如此一來無論在成本上或功能上的需求皆能因而獲得良善的解決。

除了移植 Voice Gateway 於嵌入式開發板上，我們亦將 Wireless 模組和 VPN Tunnel 成功移植，且編譯 Wireless Device Driver，運作 Wireless Access Point 的功能，提供 Wired Network 和 Wireless LANs 之間的連接點，讓 Mobile User 在一無線網路的環境下使用網路服務；而 VPN Tunnel 則是提供一安全的網路環境

2. 研究目的

2.1 欲解決之問題描述

本論文針對使用多媒體網路服務，如：網路電話 (VoIP) 與 WLAN (Wireless Local Area Network)等，將多功能之 Voice Gateway 成功開發於嵌入式的開發板上。為了解決目前市面 VoIP Service 硬體裝置先天上的限制，我們選擇 Intel IXP425 可程式化 (Programmable) 的特點，因應我們所需的功能(Wireless 802.11b/g Access Point, VPN IPSec Tunnel and Voice Gateway Function 等)成功 Porting 在嵌入式的開發板上，達到設計出降低開發成本及多功能之網路設備。

2.2 相關之解決方法

Intel 提供兩塊有關於 VoIP Service 應用的開發板，IXP421 以及 IXP425，本專題採用整合三個 NPEs 的 IXP425 當作開發之環境，Reference[8]為 IXP425 之詳細規格，每個 NPE 是一個具有獨立指令/資料記憶體的多執行緒(multi-threaded)處理器，NPEs 補強許多高度密集運算的 Data Plane 動作，包含：IP header inspection and modification, packet filtering, packet error checking, checksum computation and flag insertion and removal...etc. 此外 NPEs 在 Intel XScale core 上啟動高階的應用處理程序，這項獨特分散式處理架構使得本專題所採用之 Network Processor (NP) 足以滿足日漸成長的 VoIP Service 需求。並且 IXP425 有兩個 high-speed serial (HSS) ports 可以直接連接 Standard

Subscriber Line Interface Circuit/Coder-Decoders (SLIC/CODECs)，圖 2-1。此外在 IPSec 方面，IXP425 增加了一個嵌入式硬體的安全加速器(Security Accelerator)，應用於 IPSec 的加密、壓縮(Encapsulation)和認證(Authentication)演算法。而在軟體的解決方法，我們是撰寫修改 NPEs Codelets object code 達到 Voice Codec, Wireless AP, VPN IPSec…等功能，其中在 Voice Codec 部份，IXP4XX DSP Software 增加了 Software-based Voice and Telephony DSP Capability, 包含 G..729A Voice Codec 與 G..168-Compliant Echo Cancellation. 因此由上述可知我們將語音信號處理的工作搬移到 Intel XScale core 上，因此 NPEs 可以免除一般要做 DSP Signaling Process 程序所需之 DSP hardware，因此更達到減少 BOM cost.

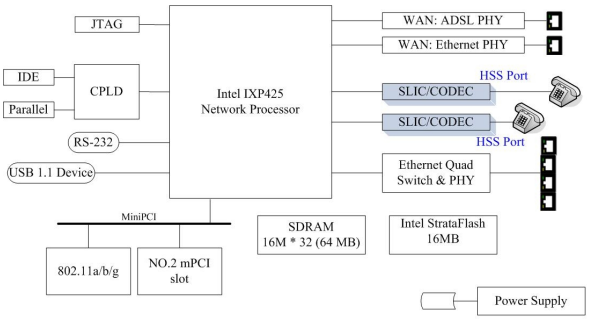

圖 2-1　Coyote IXP425 Network Processor Block Diagram

3. 重要貢獻

3.1 高度整合資料與語音功能

本專題透過 On-Chip 技術整合資料與語音功能，達到節省實作個別不同網路功能裝置的成本，而且更容易和其它硬體做結合。Intel XScale 核心包含了整合性 Multiply and Accumulate Functions 進而無需額外的硬體達到處理多媒體服務。以語音服務來說，Intel XScale 核心可以執行廣泛類型的 Speech Coding and Telephony Algorithms 因此無需一般額外的 DSP Chip. 對外連接裝置方面，NP 有兩個 HSS Ports 可以直接連接到 T1/E1 framers 或者是目前工業標準的 SLIC/CODECS. 在資料整合方面，PCI 2.2 host and option interface 提供直接連接其它裝置的變動性，包括：802.11x chips, PCMCIA Controllers and Cable MAC/PHYs. 而我們的無線網路裝置就是透過 Intel XScale core 這種高整合度的特性，將 Wireless 模組 802.11b/g Antenna 透過 Mini-PCI 介面加裝於 IXP425 上，成功 Porting Wireless AP 功能。

3.2 移植 DSP Software Library-無需額外的 DSP Chip

我們利用強大的 DSP Software Library，使得 IXP425 Network Processor 在 Intel XScale 核心上實作語音處理演算法 (Voice-Processing Algorithms)，並且可支援 1-Voice Interface 到 4-Voice Interface 範圍。Intel XScale 核心在一個非常低能源消耗的狀態下高速運算包含 G.711 and G.729a/b 等 Voice Codec. 將原本應透過 DSP Chip 處理的 Voice Codec 搬移至整合之 Intel XScale core 上，搭配 DSP Software Library 更加速語音演算法的處理，因而達到無需額外 DSP Chip 處理且可節省系統整體成本的一大利益。而 IXP4XX DSP Software Library 資料流動的範例應用可詳見圖 3-1。

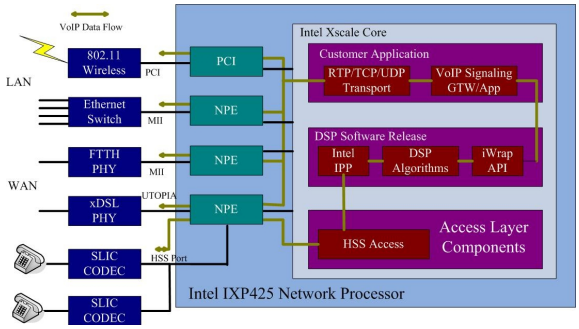

圖 3-1　DSP Software Library Data Flow Block Diagram

3.3 整合安全硬體加速元件

Intel IXP425 Network Processor 提供整合性硬體加速於網路安全性的應用，不但執行 DES, 3DES, and AES 資料加密演算法，除此之外更對於 VPNs, 802.11 and 802.11i 應用方面提供實作運算 SHA-1 和 MD5 的認證加密演算法。而且 Intel XScale 核心 API 可以使加密和認證的元件在 IXP425 Network Processor 的任何介面使用，對於網路介面多樣的現今環境來說，更是提供了一大彈性，尤其在處理無線傳輸上的安全議題；且經由 NPE 的高速處理，更可支援大量的加解密運算，其速率可達到 70Mbps. (For DES, 3DES and AES Algorithms)，由圖 3-2 可得知三個 NPEs 分工的 Block Diagram，其中 NPE B 就是 Intel IXP425 提供硬體加速的主要運算元件。

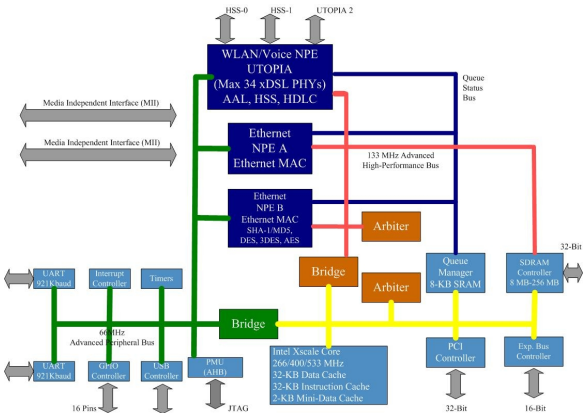

圖 3-2　IXP425 NPEs 分工狀態圖

4. 設計原理分析

4.1 系統功能架構

本專題採用 Intel IXP425 Network Processor 實作 Wireless Security Voice Gateway，改善以往 ASIC(Application Specific Integrated Circuit)所缺乏的變更性(Flexibility)和需長時間開發的缺點。並且 IXP425 整合三個 NPEs，使得 Intel XScale 核心能平行處理程式指令，NPEs 各司其職，讓網路封包的處理達到最佳化。圖 4-1 為 IXP425 Wireless Security Voice Gateway 的應用展示圖。

本專題將軟硬體功能模組分成三個部分：DSP Software: Voice Over Internet Protocol、Wireless 802.11b/g Access Point 與 VPN IPSec Tunnel，其 Software Stack 如圖 4-2 所示，以下針對各功能模組做一一描述。

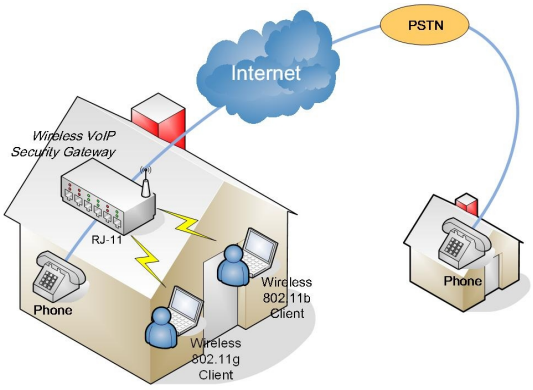

圖 4-1　Wireless Security Voice Gateway 應用展示圖

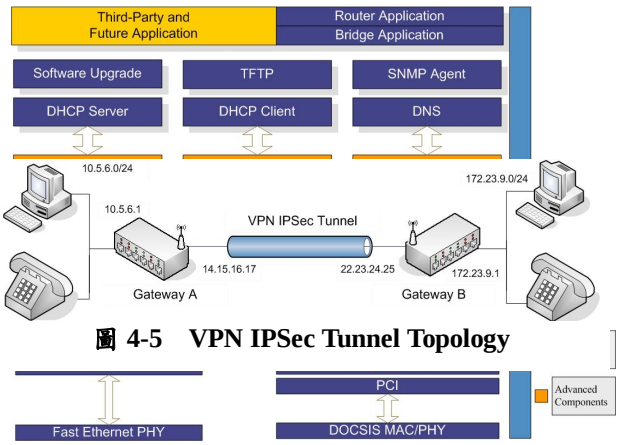

圖 4-5　VPN IPSec Tunnel Topology

圖 4-2　Wireless Secure Voice Gateway Software Stack

A. DSP Software: Voice Over Internet Protocol

(1)Overview：如圖 4-3 所示，我們用一個簡單的架構圖解釋一般 VoIP 應用的傳送流程。在圖中可見 A/D (Analog/Digital) Converter 提供一個語音壓縮的功能模組，這個語音壓縮模組每 10, 20, or 30 ms 壓縮一個 Voice Sample Block. 以不同型態的 Speech Encoder 決定壓縮的週期(G.729ab and G.711, G.723, etc.)而在輸出部份 A/D Converter 會將語音轉換成 binary bits. 並且轉換成 TCP/IP 封包格式傳送至一般網際網路。而同樣的另一接收端收到這 TCP/IP 語音封包後會轉送至語音解壓縮模組(Voice Decompression Module)做處理。而 Binary Bits 被轉換成語音格式並且送到 D/A (Digital/Analog) Converter. 而 Signaling block 則是處理 call set-up 和 tear-down 的功能。而本專題之 DSP Software 模組主要就是提供基本的語音和訊號處理功能，見圖中以黑色粗框標記起來的部份。

(2) Architecture and Features：DSP Software 主要包含八個 Resource Components 和三個 interfaces, 見圖 4-4。Resource Components 主要提供訊號處理的功能，如：Voice Compression/Decompression, Tone Generation/Detection, etc. High-Speed Serial (HSS) port 則是提供連接至一般家用電話的介面。Packet Interface 主要提供和 IP Stack 交換封包的介面。而 VoIP Application 透過控制介面(control interface)掌管操控整個 Resource Components. 綜觀上述 DSP Software 所有的運作是由圖中 Task Dispatcher 接收來自 HSS port 傳來的同步訊號(synchronization signal)來做工作的調節。因篇幅的關係，圖中其餘 Resource Components and Features 可詳見 Reference[3]有詳細的介紹。

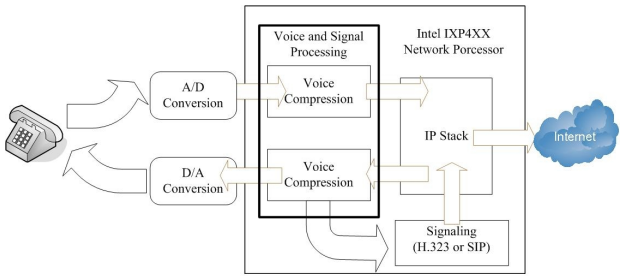

圖 4-3　VoIP 語音封包傳送流程圖

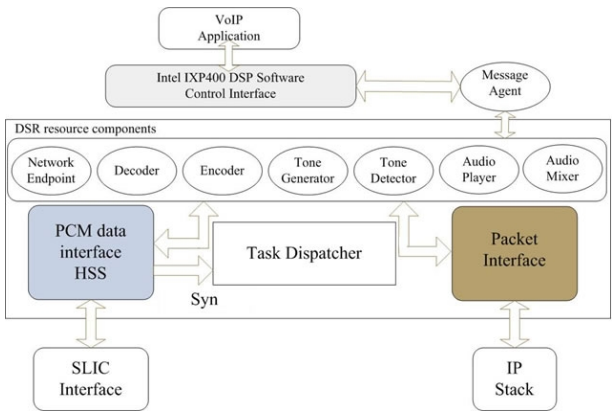

圖 4-4　IXP400 DSP Software Architecture

B. Wireless 802.11b/g Access Point

(1) 整合硬體架構減少成本：IXP425 包含一個高效能 PCI Interface, USB Controller, and four

10/100 Ethernet MACs. NP 的硬體模組不但結省了開發成本，將所需的硬體傳輸介面整合，而且 PCI Interface 更能讓本專題的 AP 功能連接多種 802.11x 的裝置、PCMCIA Controllers、Ethernet MACs 與 Broadband MAC/PHY devices.

(2) Flexible Software Architecture：Intel XScale 核心執行 ARM*V5 整數指令集架構(integer instruction set architecture)，其中包含 ARM V5T Thumb instructions and ARM V5E DSP extensions. 而本專題的 AP 模組就是利用 IXP4XX Software Library 實作 NPE 所需的指令和功能，因所有的 Library 都是用 ANSI-C 所撰寫，而且 IXP4XX Software Library 並無對其它種類的 OS Library Functions 做直接呼叫(Direct Calls)。更能使得本專題容易適用於其它作業系統，並不只侷限於某些系統程序(OS Library)。

C. VPN IPSec Tunnel

移植 Opensource Freeswan-1.99 於 IXP425 開發板上，包含以下主要特性：

- Security Architecture for the Internet Protocol
- Connection Type: Tunnel, Transport
- Key Management: Manual, Automatic, Internet Key Exchange
- Gateway Authentication: X.509, RSA signatures, pre-shared secret key,
- IP Protocol: ESP, AH
- Encryption: AES, 3DES, DES, Hardware encryption integration
- Authentication: MD5, SHA-1

主要選擇 FreeS/WAN 的原因，理由如下：

- FreeS/WAN 可支援多種平台，包含 Linux 及 Windows 作業系統
- FreeS/WAN 可支援 Routing 模式及 Bridging 模式的 VPN IPSec 建置
- FreeS/WAN 為開放源始碼，且授權自由使用。
- Montavista Linux 及 SnapGear Linux 皆有針對 FreeS/WAN 釋出 Patch，讓 Kernel Source 能完全整合 FreeS/WAN

5. 設計原理分析

本專題的實作環境為 Linux 與 Windows 作業系統，在 Linux 系統部份細分為開發端(Host)和目標板(Target Board)，如圖 5-1，在 Host 端我們安裝 Linux Red Hat 9.0 做為基本作業平台，再加裝 Montavista Linux Pro3.1 作為嵌入式開發軟體，而在目標板 IXP425 的部份，我們經由實驗測試現今的 OpneSouce SnapGear 與先前所提的 Montavista Linux 皆能於 IXP425 上正常運作，包括 Bootloader、Load Image 與 Cross Compile…等。在 Voice Gateway 方面，我們在 IXP425 上 porting DSP Software Library，使得 IXP425 無需額外的 DSP Chip 能達到 Voice Codec, Voice Compression/Decompression… 等功能。而在 Wireless Access Point 方面，我們加裝 802.11b/g 的 Antenna 模組於 IXP425 上，再將 IXP400 Software: IXP400AccessLibrary，經由其提供之 API 撰寫修改 Codelet，成功 porting Wireless 802.11b/g Access Point Functions。5.1 節為開發環境硬體環境詳細介紹，針對硬體的配備與功能都有詳細的說明。而 5.2 等為專題實驗系統研發工具介紹，包括 DSP Software Library, IXP400 Access Library Codelet, Freeswan, SnapGear, Montavista Linux, Linux Red Hat9.0…etc.

對於系統的開發，皆採用 C 語言來開發，如 5.2 節所示。

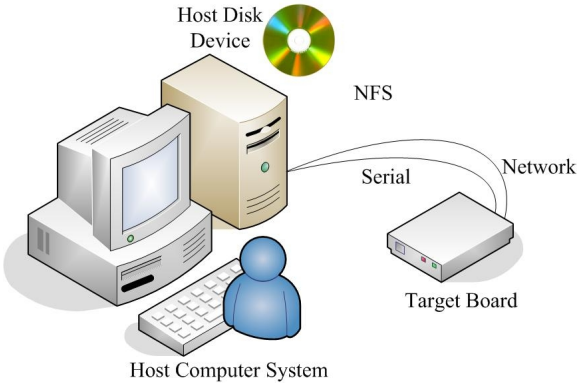

圖 5-1 專題實驗嵌入式開發環境

5.1 系統硬體配備

功能	無線網路環境下的使用者
配備	ASUS M5200 筆記型電腦，Intel P4 1.5G 中央處理器、512MB 記憶體、30GB 硬碟和 100Mbps 內建網路卡。

(1) Development Host

功能	負責嵌入式系統開發端，提供 TFTP、DHCP 及 NFS 功能，以及 Cross-Compilation
配備	ASUS M5200 筆記型電腦，Intel P4 1.5G 中央處理器、512MB 記憶體、30GB 硬碟和 100Mbps 內建網路卡。
系統軟體	RedHat Linux9, Montavista Linux Pro 3.1

(2) Target Board

功能	嵌入式開發板
配備	ADI Coyote IXP425、Intel IXDPG425、XScale Core 533MHz、 8 to 32MB SDRAM、3 NPEs
系統軟體	Montavista Linux

(3) 無線網路模組 (Wireless Modules)

功能	發送及接收無線網路資料(Beacon Frames)
介面	Mini-PCI

(4) WLAN Client

功能	Voice Codec 接收端
介面	RJ-11

(5) POTS Phone

5.2 開發工具

開發工具名稱	內容
C compiler	gcc compiler,gcc-3.3.1, GNU Cross Compiler
DSP Processing: Speech Codec	DSP Software Library
CSR Software	ixp400AccessLibrary-1.4 ixp400AccessLibrary-1.4-WithCrypto
Related library	glibc, gmp-4.1.2
VPN IPSec	Freeswan-1.99
Patch	Intel, Montavista, SnapGear,Freeswan…etc.
Driver	Atheros Device Driver

6. 實驗結果比較

6.1 實驗環境

系統的實作部份，主要為 Voice Gateway、Wireless 802.11b/g Access Point 以及 VPN IPSec Tunnel 三個部份。其功能已於之前敘述過，圖 6-1 為本專題運作的實驗架構圖。

在實驗中我們先一一將功能模組成功移植到開發板上，且每個功能模組一一做測試，最後將我們論文需求的所有模組同時運作，且能運作正常，

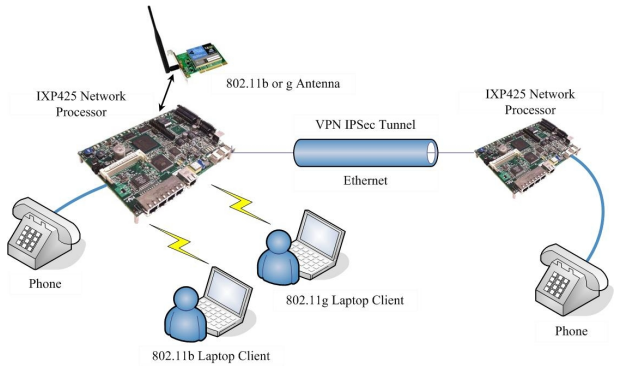

圖 6-1　專題實驗運作流程圖

底下就一一介紹我們實驗的過程及結果。

6.2 實驗結果

Wireless 802.11b/g Access Point

首先我們將利用 Wireless Extension 模組及 IXP425 建置無線網路環境，而在協定方面我們會採用 IEEE802.11b 或者是 IEEE802.11g。在硬體需求方面，主要列出底下四項：

1. 支援 Wireless Networking 的開發板
2. Wireless Extension 模組：Atheros Wirelss Card。
3. A Notebook with wireless support for 802.11b or 802.11g
4. Ethernet Cables

而在軟體方面，其實最主要的核心就在於，將原本適用於 x86 的 device driver 移植成適合 ixp425 核心的版本，因篇幅的關係，其中詳細移植過程我們不在論文中討論。成功登入 target 端後，我們將 Wireless 所需的模組：wlan.o, ath_hal.o, ath_pci.o 依序安裝至 Embedded Linux 上。成功安裝後可以看到 Wireless Card 詳細的參數及規格。

成功將 ath0 啟動之後，我們必需設定無線網路一些必要的參數，首先將 ath0 設定為 Infrastruture Mode，再指定使用何種協定做傳輸，本論文將實驗 IEEE802.11a/b/g 三種模式下，正常運作。

最後我們將 ixp0 (LAN)及 ath0 做成 bridge，我們命名為 br0，以 br0 為介面，設定 DHCP Server，再將 br0 和 ixp1 (WAN)做 NAT，設定 iptables，完成 Access Point 上應有的所有功能模組。

在 Wireless Client，我們分別測試 802.11a/b/g 三種不同模式的實驗，證明我們移植到 ixp425 上的 driver 能正常運作 Access Point 的功能，讓 mobile device 能在 wireless networking 底下使用網路服務。

Voice Gateway

本實驗實作 Voice Gateway 於 IXP425 開發板上，移植 Voice Codec 程式 IXP400 DSP Software，將一般 RJ-11 的家用電話接於開發板上的 POTS Port，在硬體需求方面，主要列出底下四項：

1. Developing board supporting VoIP (IXDPG425)
2. 1 Lan Computer, connected to the target
3. 2 POTS Telephones
4. 2 Ethernet Cables

而在軟體部份，Voice Gateway 最主要的核心程式就是 Intel DSP Software(DSR)，詳細功能及它內部的模組我們已在論文前半部詳述過了，底下列出幾項必需要程式模組：

1. dsr.o
2. IxDspCodeletApp
3. csr.o
4. ixp425_eth.o
5. csr_codelets_demoUtils.o
6. csr_codelets_dspEng.o
7. IxDspCodeletApp

VPN IPSec Tunnel

本實驗實作建置 VPN IPSec Tunnel 於 IXP425 開發板上，移植 Opensource Freeswan-1.99 及 ixp400AccessLibraryWithCrypto，其中針對 linux kernel 及 source 修改許多 patch，詳細清單已列於第五章的開發工具中，在硬體需求方面，主要列出底下兩項：

1. IXP425 開發板
2. Ethernet Cables

再在軟體方，是本實驗的重點，主要列出底下九項：

1. ixp400AccessLibraryWithCrypto
2. Freeswan-1.99
3. freeswan1_99_patch_IXP400_1_4_MVL3_0
4. gmp-4.1.2
5. freeswan-alg-0.8.0-BASE-common.diff
6. freeswan-alg-0.8.0-BASE-pluto.diff
7. freeswan-alg-0.8.0-BASE-klips.diff
8. freeswan-alg-0.8.0-enc-aes.diff
9. ixp425_1.4_freeswan-1.99_AES.patch

同樣的，詳細的移植過將不在論文中討論，以下開始介紹將 IXP425 移植 VPN IPSec Tunnel 的簡單流程。

一、Host 嵌入式開發端

1. 完整編譯
   make ixdpg425_config
   make menuconfig
   make dep (cross compile gmp 與 freeswan)
   make zImage modules
2. 複製 ixp400.o, ixp425_eth.o, ipsec.o, ipsec_alg.o 於 nfs_root/home/freeswan

二、Target 嵌入式開發版

1. 設定 IPSEC_FOLDER, IMAGE_FOLDER
   IPSEC_FOLDER=/home/freeswan/lib
   IMAGE_FOLDER=/home/freeswan
2. 載入模組
   insmod /$IMAGE_FOLDER/ixp400.o
   insmod /$IMAGE_FOLDER/ixp425_eth.o
   insmod /$IMAGE_FOLDER/ipsec.o
   insmod /$IMAGE_FOLDER/ipsec_aes.o
3. 設定介面
   ifconfig ixp0 $MAG_2_GON netmask 255.255.255.0
   echo 1 > /proc/sys/net/ipv4/ip_forward
   ifconfig ixp1 $MAG_2_PC netmask 255.255.255.0
   echo 0 > /proc/sys/net/ipv4/conf/ixp0/rp_filter
   echo 0 > /proc/sys/net/ipv4/conf/ixp1/rp_filter
4. 設定環境變數
   PATH=$PATH:/$IPSEC_FOLDER
5. 建立 IPSec Tunnel
   ipsec setup --start

7. 結論與未來發展

本文章題利用 Intel IXP425 網路處理器，設計並實作一應用於目前廣大 VoIP 市場的網路裝置，藉由 NP 可程式化的特點，開發 Wireless Security Voice Gateway，解決目前市面上網路設備先天上無法任意擴充功能的限制。而網路處理器的一特點就是，內含多個 NPEs 與其 Intel XScal 核心平行處理程式指令，更加速了網路處理的速度和效能。

在應用程式方面，我們採用 IXP400 Software Access Library, DSP Software Library 與 Freeswan，利用其內部程式無對一般 OS Library 作直接呼叫的特性，更是大大提升本專題的相容性。而 DSP Software Library，更是讓我們免除增設額外的 DSP Chip 來處理 Voice Codec 與 Telephony Algorithm，大大減低我們的開發成本和複雜性。

綜觀上述，我們將一般的網路設備的研究移至嵌入式開發板上，不但效能上得到改進，在應用層面上更是能達到多重服務(Multi-service)的效果。

未來我們將會朝更多元化的功能去努力，並將實際運作於網路處理器上的服務和同等功能的網路設備或一般 PC 做效能評比，讓現在的網路服務與效能大大提升執行的速率與效率。

參考文獻

[1] Intel IXP400 Software Release, v1.4, “Software Release Notes” Feb. 10, 2004

[2] Intel IXP400 Digital Signal Processing (DSP) Software, v.2.6.2, “Programmer’s Guide” Feb. 2005

[3] Intel IXP400 Digital Signal Processing (DSP) Software, “Voice Over Internet Protocol Application Note” Feb. 2004

[4] Intel IXP421 and IXP425 Network Processors Voice-Telephony Integration in CPE Devices, “Voice over IP Solutions” Sep. 2003

[5] Residential Gateway, “Cable Modem with Wireless LAN and Ethernet Ports”, white paper, May. 2004.

[6] “Intel IXP425 Network Processors Performance Analysis of VPN Devices” , The TOLLY Group, Jul. 2004.

[7] Intel Internet Exchange Architecture, “Addressing Netx-Generation CPE Challenges,” White Paper, Feb. 2002.

[8] Intel IXP425 Series,http://www.intel.com/.

[9] Montavista Linux Support, http://www.mvista.com/support/.

[10] SnapGearSourceForge, http://ixp4xx-osdg.sourceforge.net/.

[11] OpneSource Development Guide, http://sourceforge.net/projects/ixp4xx-osdg/

[12] FreeS/WAN Project Home Page, http://www.freeswan.org/