系統層級功耗分析平台設計

The Design of System Level Power Analysis Platform

黃迅倫　王振傑　賴俊宏　陳耀華　紀美伶　陳昭宏　林士哲　林詠捷　盧俊銘
Hsun-Lun Huang, Chen-Chieh Wang, Chun-Hung Lai, Yao-Hua Chen, Mei-Ling Chi, Zhao-Hong Chen, Shih-Che Lin ,Yung-Chieh Lin, Juin-Ming Lu

中文摘要

本文介紹一套能夠進行全系統(Full System)分析之系統層級功耗模擬平台，平台中整合了多項目前在智慧行動裝置上使用的關鍵性IP模組，包含中央處理器(CPU)、繪圖處理器(GPU)、影像編解碼器(CODEC)、影像處理器(ISP)、記憶體控制器(LPDDR3 Ctrl)、晶片匯流排(NoC)及液晶顯示控制器(LCDC)等。這些模組的建置皆以C/SystemC程式語言撰寫，並支援事務層級模型標準介面(TLM 2.0)，各IP模組並整合了具工研院資通所自主專利之系統層級功耗模型介面，在此平台上不僅可以快速地進行Android作業系統開機外，更可以針對不同應用情境進行系統整體功耗剖析，讓系統開發者在設計初期能夠快速地進行架構探索、系統效能功耗分析，並進行軟硬體協同設計與優化等，幫助系統開發者能夠早期正確定義系統規格，並實現最佳化系統整體設計，強化系統晶片性能並縮短開發時程。

Abstract

In this paper we present the ITRI System-Level Power Analysis Platform, an electronic system level (ESL) virtual platform for evaluating smart mobile device architecture with real-world workloads. The virtual platform has integrated many ESL IP models of major components that are indispensable in mobile SoC design, including CPU, GPU, Video Encoder, Video Decoder, ISP, LPDDR3 controller, NoC and LCDC. The ESL IP models are designed based on C/SystemC programming language and are compatible with the standard TLM 2.0 interface with non-blocking, AT 4-phase protocol. The ESL IP models are also incorporated with ITRI-patented ESL power analysis interface. The platform can boot Android within minutes, which is hundred times faster than traditional simulation approach. Furthermore, the platform can support detailed IP-level power profiling based on different application scenarios, which provides very useful information for system designer to evaluate, explore, and optimize the system performance and power as earlier as possible.

關鍵詞(Key Words)

電子系統層級 (Electronic System Level；ESL)

事物層級建模 (Transaction Level Modeling；TLM)

功耗模型 (Power Model)

1．前言

數位時代隨著網路發展從先前的有線通訊、無線通訊進化到物聯網階段（如圖 1）。全球行動通訊系統協會（Global System for Mobile Communication Association，GSMA）指出，2020年的物聯網(Internet of Things；IOT)市場規模預估將高達4兆5千億美元，麥肯錫（McKinsey）也預估物聯網至2025年將每年有2兆7千億～6兆2千億美元之間市場規模，GE更預測到2030年時，物聯網的市場預估規模將高達10兆～15兆美元。由上述的資料看來，未來對於智慧行動裝置，包含穿戴式裝置的需求將會越來越大，因此對於低功耗的需求更勝其他資訊產品，以智慧眼鏡為例，Google Glass在正常使用下（即時錄影）僅能使用約45分鐘，離使用者可以接受的時間（3小時以上）有數倍的差距，如何從系統架構設計的初期，確保正確的軟硬體架構，從而達到最佳效能與功耗平衡，是目前產業最需解決的問題。電子系統層級(Electronic System-Level, ESL)設計技術提供抽象化設計方法與軟硬體協同模擬技術[1]，有效解決現今系統晶片設計所面臨之複雜軟硬體系統架構設計與即時上市(Time-to-Market)的挑戰。系統架構的好壞對系統效能與功耗影響十分深遠，如何在設計初期快速進行架構探索與規格制訂則是目前產業仍存在的技術缺口。ITRS (International Technology Roadmap for Semiconductors)報告中指出(見圖 2)，未來系統功耗的優化與調整80%將在ESL層級完成[2]。因此，如何即早於系統設計初期獲得效能與功耗資訊，進行架構探索與最佳化，一直是產業發展重要議題。

圖 1 數位時代的演進與物聯網市場規模趨勢 (資料

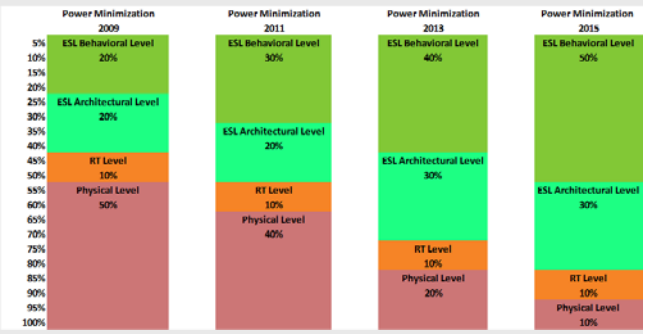

圖 2 系統層級功耗分析對整體功耗優化之影響 (資料來源：ITRS 2011)

針對ESL效能分析，可透過時序(Timing)建模技術將時序加註於高階模型，並整合效能評估技術以實現系統層級系統效能估測[3]，EDA廠商如Carbon亦提供工具針對RTL (Register Transfer Level) 模型進行高階模型轉換與時序建模[4]。針對功耗議題，目前已有部分研究[5][6] 提出高階模型功耗建模方法，並且EDA廠商如Synopsys [7]、Docea [8]亦提出類似的高階功耗建模方法並提供功耗分析平台供使用者進行系統功耗分析。目前基於ESL平台之軟體開發主要以驗證功能為主，對於支援以功耗為導向之軟硬體協同優化設計仍不成熟，尤其是缺乏基於TLM功耗分析平台之智慧行動軟體協同設計技術。因此在本篇論文中將介紹以TLM 2.0 [9] 標準為傳輸介面的ESL SoC 虛擬平台上整合由ITRI自行研發之功耗模型及建模技術，使複雜之系統晶片在開發初期便可以在電子系統層級上進行各IP及系統功耗之分析與評估，此功耗分析平台的面市將有助於加速行動裝置之系統開發與設計。

2．電子系統層級設計

隨著系統單晶片複雜度日益提升，系統設計者必須尋找新的設計方法，不但要能夠應付日益漸增的設計複雜度，也要能夠加速產品上市的時程，而最明顯的解決方案就是提高產品

的設計抽象層級，而ESL設計技術即是一種高抽象層級的系統設計開發方法，採用SystemC語言的設計方法，SoC架構工程師可以搭配虛擬平台（virtual platform），來模擬系統晶片的行為(Behavior)，另外，利用事務層級(Transaction-level)時序模型設計方法，可進一步模擬系統時序週期，此統稱為事務層級模型（Transaction Level Model；TLM）[9]。

TLM是由Accellera Systems Initiative所發表的公開建模標準，它是一種透過事務(Transaction)層級函數呼叫而非訊號或連接線來進行模組之間的溝通介面，此標準針對電子系統層級建模定義了三個基本的設計抽象層級，分別為：程式者觀點(Programmer's View；PV)層級、具時序之程式者觀點(Programmer's View with Timing；PVT)層級、以及週期精確(Cycle Accurate；CA)層級。一般而言，PV層級之建模方式，主要用來做軟體開發用，因為它擁有非常高的設計抽象層級，可提供非常快速的行為與功能性模擬；PVT層級則比PV層級多加了時間的概念，有了近似的時間資訊，可以用來進行架構分析與系統效能評估，且同時還保有可接受的模擬速度；CA層級則具備更詳細的時間與硬體設計時序資訊，主要目的是用於進行系統驗證或更精準系統評估所使用。

在本文提到的虛擬平台上所有ESL高階層級模組皆是以C/SystemC程式語言及TLM 2.0標準為傳輸介面來進行建模，在抽象層級的使用上，由於需要時序以便進行系統功耗分析，並且要能夠兼具Android作業系統開機，因此本中的虛擬平台是以PVT層級為基礎來進行建置。

3 系統層級功耗分析平台架構

在本章節將介紹由工研院資通所自行建置之電子系統層級功耗分析平台。此功耗分析平台將分為三部分來介紹：IP層級功耗模型、系統層級功耗分析模組以及系統平台整合架構介紹。在系統平台架構上整合了許多在行動裝置上會使用到的關鍵性IP，包含多媒體裝置之繪圖處理器、影像編解碼器及影像處理器等等。而在CPU上，則是以Google官方所釋放出來的emulator來進行整合，並模擬執行ARM v7指令集。在本文中，我們將以Encoder為例來介紹IP層級功耗模型，而在虛擬平台架構的介紹上，將著重在如何將Android emulator整合至虛擬平台上。

3.1 IP層級功耗模型

進行SoC系統功耗分析之前，必須先針對系統上每個IP模組建置及整合相對應的功耗模型。圖 3是整合了ITRI自行開發之IP層級功耗模型及功耗分析引擎(Power Engine)之事務層級模型。IP層級功耗模型中包含了功耗回報應用程序接口(Power Report Application Programming Interface)以及功耗狀態監控器(Power State Monitor)。此外還整合了一個功耗分析引擎(Power Engine)來記錄IP模型的功耗狀態及功率消耗。

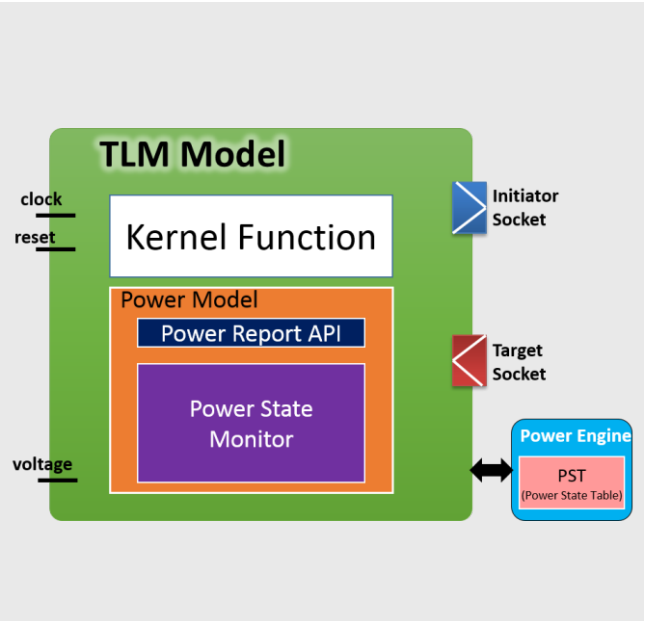

圖 3 IP 層級 TLM 功耗模型

在高抽象層級中，功耗模型的建置有兩種常用的方式，分析方法(Analytical Method)與特徵法(Characterization-based Method)。前者使用數學函數的概念將該IP之功能運作特性與狀態當作輸入計算參數，例如面積大小、輸入與輸出埠的數量、輸入與輸出埠的觸發參數(Switching Activity)等資訊，此種方法屬於黑盒子(Black Box)式分析方式，好處在於不需了解與修改該設計內部架構。後者的概念則是將設

計在細切出子區塊(Sub-Block)，而子區塊的定義依其所需功耗分析精準度可粗略到依子模組(Sub-Module)來做的切割，抑或可精確到依子函式(Sub-Function)來做的切割，最後再透過功耗特徵化程序分別為每個子區塊建出其功耗模型，此種方法屬於白盒子(White Box)式分析方式，好處在於功耗分析準確度較高，但需要去深入了解該設計內部架構，開發時間較長。

功耗狀態的定義即是用來表示該電路當前運作的情形，每個功耗模型中最少必須存在一個功耗狀態，而每個功耗狀態可對應到動態功率(Dynamic Power)與靜態功率(Static Power)兩組功率消耗資訊。然而在功耗分析中除了功耗模型的切割數量，其內部功率狀態的定義數量也將顯著影響到功耗分析精準度。以編碼器(Encoder)為例，如圖 4，以幀(Frame)為單位針對其功耗狀態定義，可以從只依照電路運作情形區分三種功耗狀態：OFF、IDLE及ACTIVE。若想再提高精確度，可以針對ACTIVE狀態再根據不同的編碼類型再區分為常見的H.264、H.263、MPEG4、MPEG2及JPEG等編碼格式(根據不同Vendor的編碼器，其能支援編碼的規格亦不同)，甚至H.264可以再細分為Baseline Profile、Main Profile及High Profile這三種功耗狀態。若仍想再更精確，又可以針對每種編碼類型的每張frame再細分為I frame、P frame及B frame。從Coarse-grained層級的三種功耗狀態可以一路細分到Fine-grained層級的十七種功耗狀態。甚至若想再得到更精確的功耗預估值，還可以再考慮到編碼器壓縮的位元率(bit-rate)等等因素。因此在電子系統層級功耗分析平台上，根據不同的IP型態及特性，會定義出適合其各別的功耗狀態，在此將不再多加詳述其他IP之功耗狀態的定義。而相關的功耗狀態資訊會被存到對應每個IP的功耗分析引擎，記錄平台於模擬過程中之功耗狀態轉換資訊。

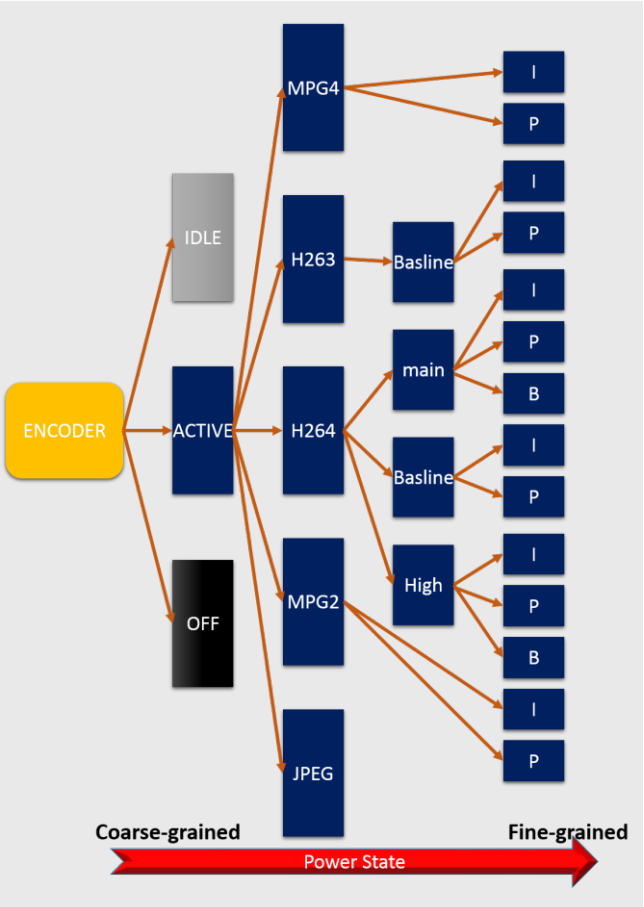

圖 4 編碼器功耗狀態定義

3.2 系統層級功耗分析模組

在每個IP模組建置及整合相對應的功耗模型並定義出相對應的功耗狀態之後，接下來便可以開始著手進行系統層級功耗分析模組建置。在真實的晶片上，通常會根據模組間不同的電壓供應以及不同的時脈頻率，會定義出整個系統的操作模式，甚至在同一個操作模式下，有些比較耗能的模組，如CPU，會有動態頻率電壓調整(Dynamic Voltage and Frequency Scaling; DVFS)機制。在電子系統層級功耗分析平台上，工研院資通所自行建置了一個電壓頻率控制單元(Voltage and Frequency Control Unit; VFCU)模組來對各IP模組及不同的系統操作模式來設定相對應的電壓及頻率，如圖 5。

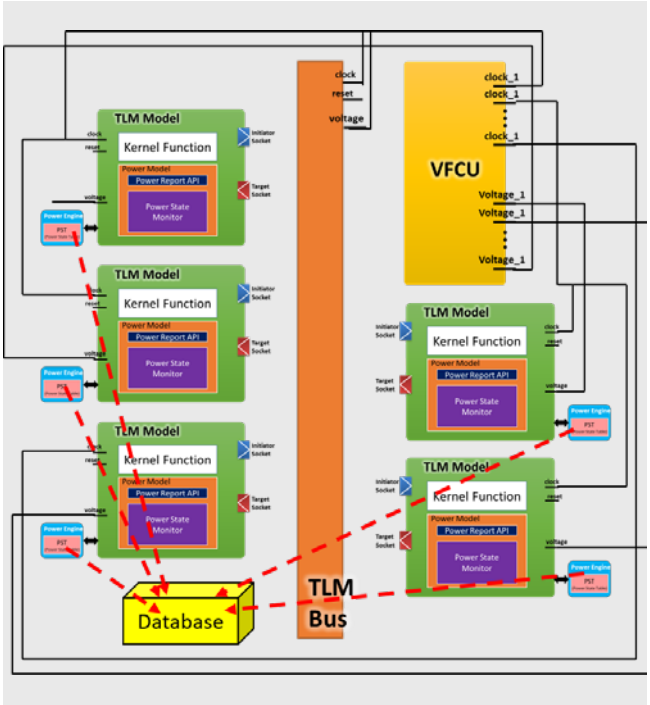

圖 5 系統層級功耗分析模組

程式可以根據系統不同的操作模式來設定電壓頻率控制單元，傳送給各IP需要的操作電壓及操作頻率。而各IP會在模擬過程中，動態地反應功耗狀態的變化，並將這些變化儲存下來傳送至資料庫(Database)中，最後將根據資料庫裡面的資料進行系統的功耗分析並畫出功耗波形圖。

3.3 系統層級功耗分析虛擬平台架構

電子系統層級功耗分析虛擬平台如圖 6，是以C/SystemC為主要程式語言來描述硬體核心行為及時序，並且以TLM 2.0 non-blocking AT 4-phase為傳輸介面的ESL高階系統虛擬平台，包含ARM Cortex-A7 (Andriod-QEMU)、繪圖處理器(GPU)、影像解碼器(DEC)、影像編碼器(ENC)、影像訊號處理單元(ISP)、液晶顯示器(LCDC)、通用非同步收發傳輸器(UART)、Mail Box、DRAM、SRAM、電壓頻率控制單元(VFCU)及事務層級匯流排(TLM Bus)等高階模型設計。

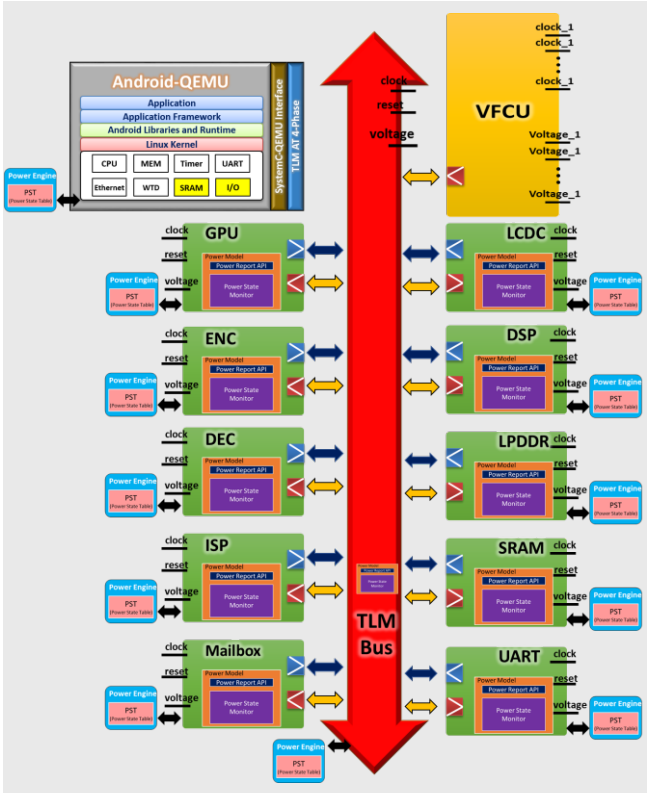

圖 6 電子系統層級功耗分析虛擬平台架構圖

電子系統層級功耗分析虛擬平台中最主要的核心模組CPU是採用Android-QEMU做為Simulation Model來模擬ARM v7指令集。為使QEMU能與虛擬平台上其他TLM-SystemC Mode溝通，必須為其定義SystemC-QEMU Interface API並針對此API設計TLM AT 4-Phase Wrapper。QEMU是一利用Dynamic Binary Translation (DBT) 技術之全系統功能模擬器 (full system simulator)。其架構主要如圖7右半所示，它實作了開機所必須硬體功能模組以及其他一些周邊硬體模型，因此可在不用修改軟體系統(Unmodified OS)的情況下直接開機，同時由於DBT技術動態地轉譯guest code到host code直接執行所帶來的高效率，整個模擬可以在近似於原生系統速度級之下執行。

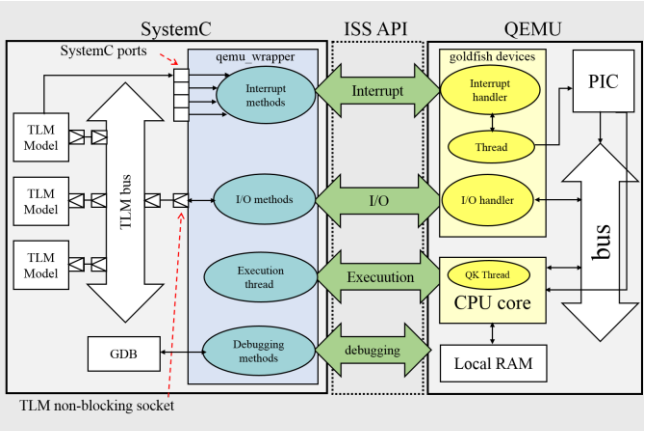

圖 7 QEMU 架構(QK: Quantum Keeper, PIC: Programmable Interrupt Controller)

在QEMU-SystemC執行控制上，比較好的方式是將控制權掌握在SystemC wrapper上，讓它決定QEMU要模擬多久的時間。QEMU原本在模擬時內部會有一個無窮迴圈，不斷地執行「fetch code block」與「dynamic binary transaction (DBT)」將guest code轉換成host code然後執行。我們將原本這個無窮迴圈打斷，讓每次執行一個遞迴(looping)後都會檢查一個quantum expired flag來決定是否做下一個遞迴。這個flag會被我們新增的一個quantum keeper thread，每隔固定時間設定，藉此使得每次的執行的時間固定成設定的time quantum，詳細請參考圖 8。如此QEMU wrapper中的execution thread 便可以透過呼叫 ISS API 中的 qemu_run_simulator() 使QEMU執行一固定時間。

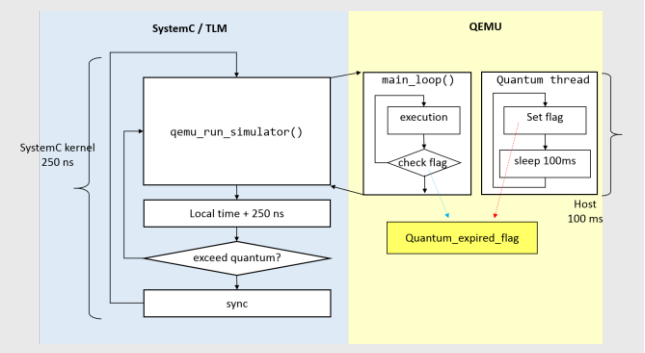

圖 8 QEMU-SystemC 執行控制

QEMU系統目前並沒有模擬CPU中的cache system，因此所有cacheable的transaction都會在QEMU內部一塊預先配置好的記憶體空間，由host CPU直接做load/store指令完成，至於non-cacheable的transaction則會透過呼叫I/O emulation function來模擬對於bus上的non-cacheable memory space的MMIO。在這裡，我們更改了I/O emulation function使得對於TLM bus上virtual devices memory space的存取都會使用到QEMU wrapper裡面定義的I/O methods。實作上，我們在透過 ISS API中的qemu_set_io_callback()，在初始化階段將QEMU內部的function pointer連結到wrapper中的I/O function實體。

在SystemC wrapper內，由於本平台使用TLM AT 4-phase protocol作為模組間溝通介面，因此我們需要在I/O methos內將QEMU發送來bus transaction的blocking function call轉換成TLM AT 4-phase的transaction以和TLM bus溝通，並在最後收到回應後(BEGIN_RESP階段)將得到的資料回傳如圖 9。

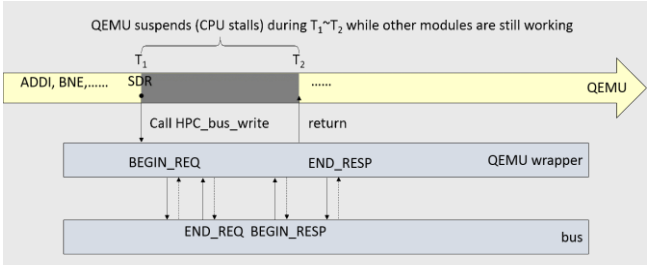

圖 9 QEMU wrapper TLM AT 4-phase protocol 流程

4．實驗結果

上一章節中描述到本電子系統層級功耗分析虛擬平台整合了Android emulator以及許多在行動裝置上會使用到的關鍵性IP，因此不但可以快速地進行Andorid作業系統開機(圖10)，更針對了不同的應用情境如錄影、多媒體播放、GPU Benchmark、收發郵件、上網、待機及睡眠等七種應用情境進行了模擬以及功耗分析。

圖 10 Android 作業系統開機

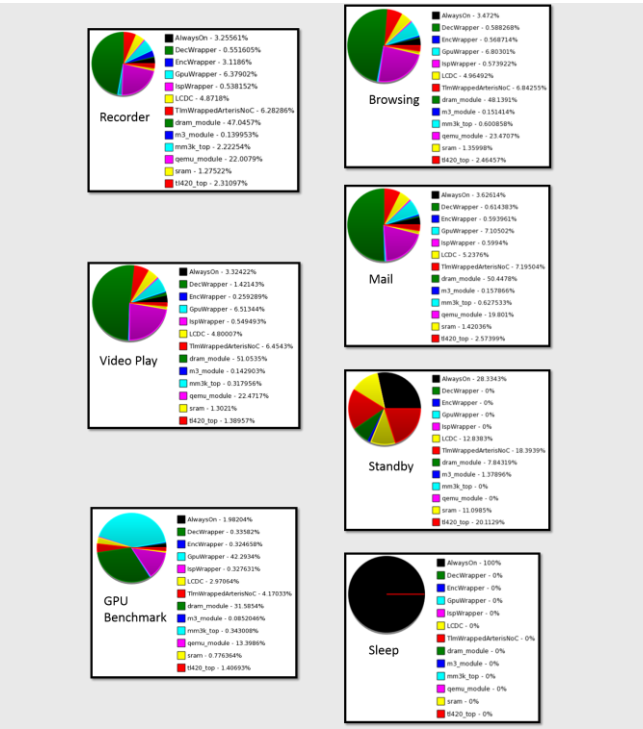

圖 11 系統虛擬平台功耗模擬結果 (圓餅圖)

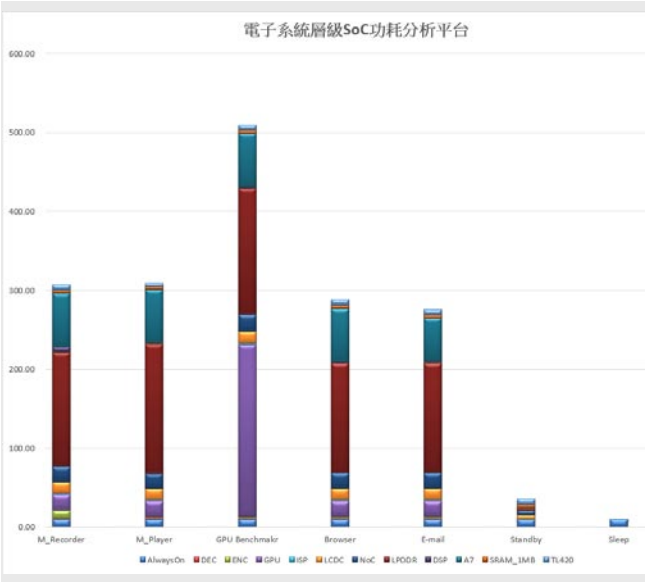

圖 12系統虛擬平台功耗模擬結果 (柱狀圖)

七種應用情境的模擬，在電子系統層級SoC功耗分析虛擬平台上不到三天的時間便可以得到模擬結果，如圖 11的圓餅圖及圖 12的柱狀圖。我們使用了圓餅圖觀察到SoC上各IP消耗功率所佔的比例，藉由這樣直觀的觀察，便可以很快的得知在每一種應用情境下，造成最大功率消耗的IP模組。而藉由柱狀圖的呈現，更可以馬上看出各個應用情境的功耗消長。如此直覺又快速的模擬及分析，可以讓系統開發者快速地進行軟硬體的調整，在SoC下線之前得到一個最佳的SoC系統架構。

5．結論

半導體產業下一波成長契機在智慧聯網應用，整體需求將包含智慧手持裝置、穿戴式裝置、感測節點裝置、網路連結、甚至是雲端伺服器與大數據等，其中，追求高效能與低耗能是始終不變的趨勢，因此能否在系統開發的初期便可以開始進行多項的系統效能分析以及系統功耗探索將是產品開發的決勝關鍵。因應此需求，導入電子系統層級設計方法應是勢在必行，目前國際大廠如Intel、Qualcomm、聯發科技等均已導入，已能分析基本功能與時序等特性，在功耗分析方面則尚未成熟，本技術可以補足這個部分的技術缺口，進一步強化國內半導體/IC設計業者系統層級研發能量，強化供應鏈產業競爭力。System feature, Show the results

參考文獻

[1] F. Poppen et al., “Impact Simulation of Changes to Development Processes: An ESL Case Study,” FDL, 2011.

[2] ITRS Roadmap 2011. http://www.itrs.net

[3] J. Cornet et al., “A method for the efficient development of timed and untimed transaction-level models of systems-on-chip,” DATE, 2008.

[4] Carbon SoC Designer. http://www.carbondesignsystems.com

[5] W.-T. Hsieh et al., “System Power Analysis with DVFS on ESL Virtual Platform,” SOCC, 2011.

[6] O. Mbarek et al., “Power-Aware Wrappers for Transaction-Level Virtual Prototypes: a Black Box Based Approach, “ VLSID, 2013.

[7] Synopsys Platform Architect. http://www.synopsys.com

[8] Docea Power Aceplorer. http://www.doceapower.com

[9] Accellera Systems Initiative. http://www.accellera.org/activities/working-groups/systemc-tlm

[10] 陳昭宏, “CPF-Aware TLM-based Power Analysis Framework”, 技資編號523A40452, 工業技術研究院, 20150831

作者簡介

黃迅倫

現任職於工研院資通所系統架構設計部技術副理。國立成功大學電機工程碩士。專長於電子系統層級設計流程開發。

王振傑

現任職於工研院資通所系統架構設計部，國立成功大學電腦與通信工程博士，專長於計算機架構、網路安全、軟硬體系統層級設計。

賴俊宏

現任職於工研院資通所系統架構設計部。國立中山大學資訊工程博士。專長於軟硬體協同設計、繪圖處理器架構設計及功耗分析。

陳耀華

現任職於工研院資通所系統架構設計部。元智大學通訊工程博士。專長於數位訊號處理及功耗分析。

紀美伶

現任職於工研院資通所系統架構設計部。逢甲大學資訊工程碩士。專長於軟硬體協同設計。

陳昭宏

工研院資通所 / 設計自動化技術組 / 設計流程開發部工程師，國立成功大學電機工程碩士。研究領域包括超大型積體電路電腦輔助設計與系統晶片驗證方法。

林士哲

工研院資通所／設計自動化技術組／設計流程開發部工程師，國立中央大學電機工程碩士。專長電子系統層級設計方法。

林詠捷

工研院資通所／設計自動化技術組／設計流程開發部工程師，私立元智大學資訊工程碩士。研究領域包括超大型積體電路電腦輔助設計與系統晶片驗證方法。

盧俊銘

現任職於工研院資通所嵌入式系統與晶片技術組副組長兼系統架構設計部經理。國立清華大學電機工程研究所碩士。專長為系統架構設計、超大型積體電路設計與測試。