

以狀態為基礎的電子系統層級功率分析方法

State-based ESL Power Analysis

陳昭宏 林士哲 林詠捷 王逸凱 鄭良加
Zhao-Hong Chen, Shih-Che Lin, Yung-Chieh Lin, Yi-Kai Wang, Liang-Chia Cheng

中文摘要

過去在電子系統層級設計技術中，主要都是著重在純功能(Functionality)驗證上，而逐步發展至今日已能夠提供效能(Performance)分析、功率(Power)分析、甚至於熱(Thermal)分析等，有助於系統設計者進行早期架構探索、系統規格制訂、與軟硬體協同設計與驗證，對於提升系統設計效率、縮短整體設計時程、建立可執行系統規格(Executable Spec.)等皆有明顯效益，已逐步為國內業界所認同並開始採用，然而，當國內業者在實際導入時面臨許多問題與挑戰，這些問題包括：(1) 缺乏系統層級功率分析標準流程、(2) 缺乏基於TLM功率分析平台之智慧行動軟體協同設計技術。為了解決上述問題，本研究將以具高系統複雜度之穿戴式裝置(智慧眼鏡)為應用載具，目標為開發下世代事務層級(TLM-based)功率分析平台與智慧行動軟體協同設計技術，並建立ESL Power Format業界標準。

Abstract

In the past, the electronic system-level (ESL) design techniques, it is mainly focused on a pure functionality verification. But today the technique has grown to provide performance analysis, power analysis, and even thermal analysis techniques. Those approaches can help system designers to perform design exploration, architecture specifications and HW/SW co-emulation at the early phases. Thereby improve system performance, shorten the overall design time and defining the executable specifications. The domestic companies began to recognize the potential and practicability of the concept. However, the domestic companies still faces various problems and challenges, these issues include: (1) the lack of standard processes for system-level power analysis, (2) the lack of transaction-level model (TLM) based VP with power analysis of smart mobile software development technology. To solve the above problems, this work target for the development of next-generation TLM-based platform with power analysis and smart mobile collaborative software development technology base on the high degree of complexity wearable device (smart glass), and then establish industry standards ESL power format.

關鍵詞(Key Words)

電子系統層級設計技術 (Electronic System Level ; ESL)
軟硬體協同設計與驗證 (HW/SW Co-emulation)
功率分析 (Power Analysis)

1. 介紹

根據工研院IEK的預估統計如圖1所示，2014年智慧型手機(Smart Phone)的銷售量將達13億支，已超越傳統手機成為全球銷售量最高的終端產品；而平板電腦(Tablet PC)的銷售量亦將於2013年超越桌上型電腦，並將於2014年達到3.3億台銷售量，超越筆記型電腦，成為銷售量最高的電腦產品，展望未來，物聯網(IoT)將是業界下一件大事(Next Big Thing)。



圖 1 智慧行動裝置數量成長趨勢

而在這波風潮下消費者對於可攜式設備的功能需求也愈趨多樣化，以手機晶片為例，目前已不再是單純的通訊作用而已，還需包含網路、音樂、GPS等多種應用，在處理資料量倍增應用層面更廣泛的情況下，手機所需之電力也必須跟著大幅提升才能滿足消費者使用時數之需求，以智慧眼鏡為例，目前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。然而受限於現今電池製造技術仍無法大幅提升電池蓄電力以及充電速度過慢等因素，高效能與低功率需求已成為現今晶片設計兩個主要的設計考量。另一方面，製程的縮小和晶片上的功能模組密度的提升，導致晶片功率密度過高，進一步衍生出可靠度以及冷卻包裝的問題必須同時被解決，而這些問題都將額外造成晶片製造成本的增加。換言之，設計者在設計過程中必須考量的設計條件將不再只是效能，如何有效地控制功率消耗亦是非常重要的。

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率平衡，

是目前產業最需解決的問題。由圖2可知，在不同的設計層級，採用的功率管理策略各不相同。在系統層級(System Level)藉由適當的軟硬體配置達到系統功率控制。而在硬體架構層級(Architecture)常使用動態電壓頻率調整(Dynamic Voltage Frequency Scaling, DFVS)技術或快算即關(Run Fast and Then Shut Down, RFTS)策略。在暫存器層級(Register-Transfer-Level, RTL)，由設計者設計微架構(Micro-architecture)並搭配clock gating或multi-Vth技術來降低動態功率。在邏輯閘層級(Gate-level)，則更換基礎邏輯閘或改變其大小(Gate Sizing)以達到功率最佳化。經由過去研究經驗可以發現，愈於設計先期進行功率最佳化，對於整體系統單晶片之系統功率將帶來愈大的效益。

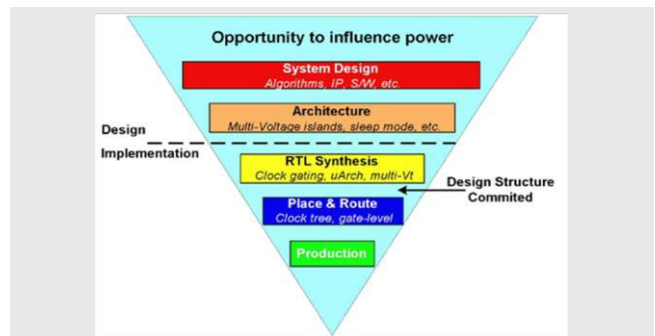


圖 2 不同設計層級採用的低功率策略

有鑑於此電子系統層級(Electronic System Level, ESL)設計技術被提出，其核心概念在於提高抽象層級(Abstract Level)的設計方法與軟硬體協同設計驗證技術 [1]，有效解決並加速現今系統晶片設計所面臨之複雜軟硬體規格與即時上市(Time-to-Market)的挑戰，並且可提早進行軟體測試與除錯工作，提升晶片設計之成功率。系統架構除了是產品差異性的一個重要指標外，更對系統效能與功率更有舉足輕重之影響，特別在於系統功率。透過圖3ITRS(International Technology Roadmap for Semiconductors)研究報告可以了解，未來系統功率的優化與調整80%將在ESL層級便完成 [2]。因此，如何即早於系統設計初期獲得效能與功率資訊，進行架構探索與最佳化，一直是

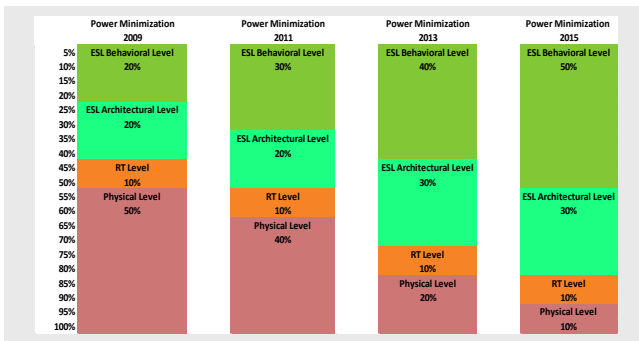


圖 3 系統層級功率分析對整體功率優化之影響

產業發展重要之議題。

針對 ESL 效能分析，可透過時序建模 (Timing Model) 技術將時間資訊加註於高階模型並整合效能評估技術藉此實現系統層級效能估測 [3]，目前 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]。

而針對功率議題，目前已有部分研究 [5, 6] 提出高階模型功率建模 (Power Model) 方法。一般非正式的分類，可以分成事件為基礎 (Event-based) 的方法或方程式為基礎 (Equation-based) 的方法。前者藉由統計特殊事件的次數，來估算可能的功率，運算上可以快速獲得結果，但是僅能獲得平均功率，無法獲得功率波型圖；後者必須對於電路有相當程度的了解，才能提供準確的方程式，但是方程式相關的參數萃取相當困難，因此也常常讓建模的工作受到阻礙。EDA 廠商如 Synopsys [7]、Docea [8] 亦提出類似的高階功率建模方式並提供功率分析平台供使用者進行系統層級的功評估。雖然這些方法已解決部分系統層級功率分析所面臨的問題，但與 RTL 設計流程的整合度仍然不足，目前系統層級功率技術仍有下列問題尚待解決：

- (1) 缺乏系統層級功率分析標準流程：通用功率格式 (Common Power Format, CPF) 與統一功率格式 (Unified Power Format, UPF) 在 RTL-to-GDS 流程已成熟支援，但在系統層級設計流程則仍缺乏完整支援，業界亦缺乏 ESL Power Format 標準。
- (2) 缺乏基於 TLM 功率分析平台之智慧

行動軟體協同設計技術：目前基於 ESL 平台之軟體開發主要以驗證功能是否正確為主，對於支援功率為導向之軟硬體協同優化設計仍不成熟。

2. 以狀態為基礎的功率分析

本研究將著重於「事務層級功率分析平台」之技術目標進行細部討論。其動機是為了解決缺乏系統層級功率分析標準流程提出「State-based ESL power analysis methodology」技術方法，該技術是以 ITRI 現有 ESL 功率分析引擎專利 [9] 為基礎進行擴充。有別於事件為基礎的方法或方程式為基礎的方法，以狀態為基礎的功率分析方法更符合電子系統層級的設計精神。屏除詳盡的硬體行為，只用有代表性的功率狀態 (Power State) 來計算功率。可以大幅改善模擬速度，也可以兼顧功率分析的準確度。其架構如圖 4 所示：

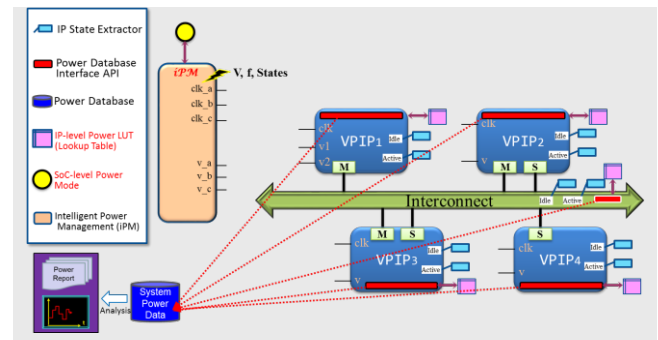


圖 4 State-based ESL power analysis

左邊所列項目為基礎功率分析元件，其部分技術已在 [10] 提出。本研究主要之概念是在虛擬平台上對每個高階矽智財各別加入功率分析元件用來監測其訊號變化狀況，並於內部插入兩個基礎功率分析元件分別為：(1) 功率資料庫介面 (Power Database Interface, PDI)、(2) 功率狀態提取器 (IP State Extractor)。當系統正常運作時，假設被觀察的埠產生變化便立即透過 IP State Extractor 通知 PDI 記錄當下時間，並至功率查找表 (IP-Level Power LUT) 擷取對應狀態的功率消耗值，最後儲存於功率資料庫 (Power Database) 中，如此便可完整記錄該高階矽智財於此次模擬中功率變化的情況。而為支

援CFP，本研究預計擴充二個功率分析元件，分別為：系統層級功率模式(SoC-level Power Mode)與智能功率管理元件(Intelligent Power Management, iPM)。前者用來制訂標準格式，提供使用者在系統層級規劃功率模式(Power Mode)如：低功率模式、高效能模式、關閉模式(Power Off)以及系統架構的電源域(Power Domain)與時脈域(Clock Domain)。同時使用者亦可透過已在RTL-to-GDS流程相當成熟之CPF/UPF檔案來制訂系統功率模式，如此便有效整合ESL功率技術與現今RTL功率技術。其已規劃之系統功率模式會經由本功率分析平台自動產生iPM元件，使系統層級支援動態電壓與頻率調整技術(DVFS)與快速執行隨後停止(RFTS)功率管理方法。此外，為使得高階矽智財能更容易整合本研究之技術，這邊亦會開發自動接線程序將上述功率分析元件、iPM元件與矽智財自動進行連接，讓使用者更容易使用。而為支援“混合型式高階建模設計方法”相容於晶圓業者如台積電、聯電之製程技術，本研究預計擴充原本IP-Level LUT，使其支援，甚至推廣至業界，成為ESL Power Format標準。

最後「State-based ESL power analysis methodology」可進行虛擬平台分析，使用者可於系統層級規劃Power Mode、Power Domain與Clock Domain，同時產生具功率分析能力之虛擬平台程式碼，並對其進行重新編譯與功能模擬，並得到整體系統功率消耗的情況，藉此去調整系統架構或是變更電源域之設定，達到在系統開發初期即可對整體功率進行優化與調整之設計需求。

3 · 功率分析方法實作

本章節將開始深入探討本研究所定義的基礎功率分析元件，首先將於3.1節中針對功率分析的基礎概念做說明，接著將依序於3.2節、3.3節、3.4節與3.5節分別介紹：(1) 功率查找表(IP-Level Power LUT)、(2) 功率資料庫介面(Power Database Interface, PDI)、(3) 功率資料庫(Power Database)、(4) 智能功率管理元件(Intelligent Power Management, iPM)等基礎功率分析元件。

3.1 功率分析基礎概念

本節內容中將明確定義何謂功率模型(Power Model)以及功率狀態(Power State)，讓使用者在功率分析這塊領域上有一定的認知，同時說明本研究在實作中對於功率狀態的定義與規範。

3.1.1 功率模型

功率模型簡單來說即是用來評估功率消耗的方式，主要目的是希望在晶片設計初期就能準確預測將來硬體實際運作的功率消耗情況，其模型的設計概念從電晶體層級至演算法層級皆有許多研究在作探討，而通常越往設計流程底部所得到的功率分析資訊準度越高，但缺點便在於分析速度相當緩慢，而越往設計流程頂部其準確度越低，但其分析速度越快。現今較為成熟之技術即為Spice所使用的電晶體階層功率模型與PTPX所支援的標準邏輯閘功率模型。

高抽象層級中存在著兩種常用的功率建模方式，其名稱分別為：分析方法(Analytical Method)與表徵法(Characterization-based Method)。前者使用數學函數的概念將該設計之特性作為功率計算參數，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率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，最後再透過功率特徵化程序分別為每個子區塊建出其功率模型，此種方法屬於白盒子(White Box)式分析方式，好處在於功率分析準確度較高，但需要去深入了解該設計內部架構。而本研究所提的功率分析主要是基於表徵法的概念來實現各元件的功率模型。

3.1.2 功率狀態

功率狀態的定義即是用來表示該電路當前

運作的情形，在本研究中將功率狀態定為功率模型內的最小單位，其每個功率模型中最少必須存在一個功率狀態，而每個功率狀態可對應到兩組功率消耗資訊，分別為：動態功率(Dynamic Power)與靜態功率(Static Power)。

在功率分析中除了功率模型的切割數量，其內部功率狀態的定義數量也將顯著影響到功率分析精準度。以圖5為例，元件DMA可先依照子模組切割出Kernel與MEM兩塊功率模型，接著依照電路運作情形可再區分出ACTIVE與IDLE兩種功率狀態於上述功率模型中，但這樣的分法最多只能看出兩塊子模組有運作與沒運作的功率情況。因此若想再提高功率分析精準度，可將Kernel的ACTIVE功率狀態在依據內部運作模式細分成PUSH與POP兩種，甚至於可依據單個事務(PUSH Transaction)之內容來定義功率狀態。值得注意的是，在本研究中功率狀態的定義與傳統做法略有不同，主要的差別在於「功率狀態之定義不一定要依照元件架構來做區分」，舉例來說在處理器的功率模型中，可以依照指令集類型來定出相對應的功率狀態，抑或者像影像編碼解碼的功率模型中，可以依照處理影像的數量來定義出不同的功率狀態。

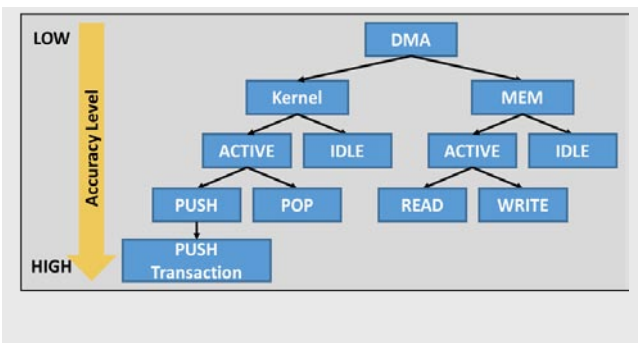


圖 5 系統層級功率分析對整體功率優化之影響

3.2 功率查找表

功率查找表即為元件自身的功率消耗資訊資料庫，內容主要紀錄該元件的各個功率狀態在不同操作條件下所量到的動態功率與靜態功率值，其操作條件通常可包含製程技術(Technology)、製程工藝(Technology Process)、閾值電壓(Threshold Voltage)、製程變異

(Process Corners)、操作電壓與頻率...等許多參數的設定。而本節內容將對功率分析所制定之功率查找表的檔案格式作詳細說明。

本研究提出之功率查找表是基於XML1.0 [11] 語言來做定義，該語言之規則如圖6所示，主要由元素(Element)來構成資料的描述，而元素自身可再細分成三個項目分別為：起始標籤(Start-Tag)、元素內容(Element Content)與結尾標籤(End-Tag)。起始標籤的表示規則是將字串描述於角括號內部，例如：<MEMBER>，與此相對的結尾標籤則是在角括號內部多出“/”符號，例如：</MEMBER>，而元素內容則單純為字串或數字之描述，例如：“Tom”與“Mary”。值得注意的是元素內容中允許再嵌入別的元素資訊，例如：<TEAM>與</TEAM>中包含著“ESL”與“RTL”兩筆元素，而“ESL”與“RTL”中又各自包含著兩筆“MEMBER”元素資訊，因此易於做到結構化資料的描述。

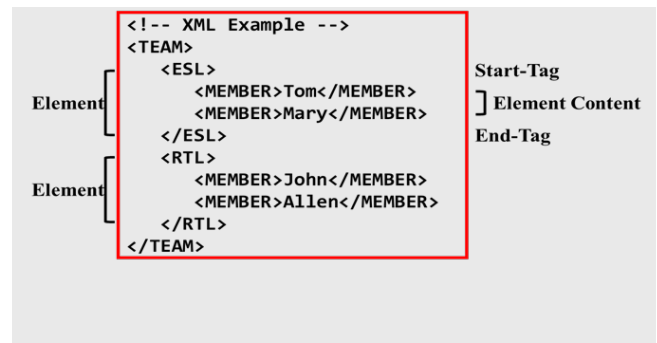


圖 6 XML語法規則範例

3.3 功率資料庫介面

功率資料庫介面的定義是希望提供有系統的方式將功能模型(Functional Model)、功率消耗計算及功率分析分別獨立出來各別開發，以利於整個分析系統未來的發展。透過此介面事務層級的模型便可以很容易跟不同精準度的功率模型作結合。底下內容將對本研究所制定之功率資料庫介面作詳細的介紹。

本研究提出之功率資料庫介面主要包含三項功能：

- (1) 透過功率查找表轉換當前功率狀態為功率消耗值、
- (2) 當功率狀態改變時計算先前狀態持續

的時間、

- (3) 計算先前功率狀態的總能量消耗。底下將透過圖7來解釋功率資料庫介面如何運作。

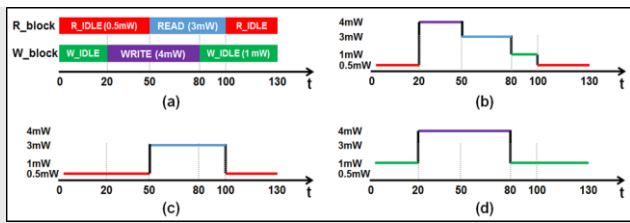


圖 7 功率資料庫介面技術概念

為方便說明，目前先單獨考慮圖7 (a)中的“R_Block”功率消耗行為，假設在初始時間時其功率狀態為“R_IDLE”，則功率資料庫介面便立即透過功率查找表找出相對狀態的功率消耗值(0.5mW)，接著當時間點來到50s時，其功率狀態轉變為“READ”，此時功率資料庫介面便立即計算“R_IDLE”狀態持續的時間($t=50$)並算出總能量消耗為25mJ，而同時間一樣透過功率查找表找出“READ”狀態的功率消耗值(3mW)。如此不斷地重複上述動作，便可完整分析出“R_Block”的功率消耗情形。

經由上述說明可以發現功率資料庫介面同時間只能管理系統元件中某個功率模型(即某個子區塊的運作行為)，因此對於擁有多個功率模型的系統元件而言，其功率資料庫介面的宣告數量就必須與功率模型數量相同，舉例來說若存在N個功率模型則功率資料庫介面也必須要有N組。接著再回顧圖7 (a)的範例，目前可以看到該系統元件內部存在著“R_Block”與“W_Block”兩個功率模型，假設若只使用一組功率資料庫介面來進行分析，其結果將如圖7 (b)中的功率波形，顯而易見的該結果與實際電路運作行為存在著許多誤差，舉例來說“R_IDLE”狀態理當持續到50s才會進行轉態，但因為“W_Block”在時間20s時，其功率狀態由“W_IDLE”轉變為“WRITE”，因此使得功率資料庫介面提前計算“R_IDLE”狀態的持續時間與總能量消耗，造成功率分析誤差產生。而若對各個功率模型分別給予功率資料庫介面，則在功率分析上將可完美的分析出圖7 (c), (d)中的

功率波形。

3.4 功率資料庫

功率資料庫的功能主要是用來記錄系統上所有“功率資料庫介面”所計算出來的功率消耗值，同時儲存各功率狀態轉換的時間點資訊。其技術概念主要有底下幾點重要事項：

- (1) 具備定時或定量將內部資料寫出之機制、
- (2) 轉換功率資訊為標準波形格式。前者是因為當系統複雜度較高或系統模擬時間較久時，便需要更多的記憶體空間來存放功率資料，因此伺服器將存在著記憶體空間是否不足的疑慮，所以需要有其機制負責將資料寫至外部硬體。而後者則是因為在系統層級的功率分析中，設計者除了想與底層電路進行功率誤差比較外，還會想要知道兩邊的功率趨勢是否相同，因此這邊將提供功率波形資訊讓設計者進行對照分析。

3.5 智能功率管理元件

智能功率管理元件的功能旨在提供ESL虛擬平台上各個元件所需的電壓(V)與頻率(MHz)，底下內容將對本研究所制定之智能功率管理元件與系統層級功率模式描述作詳細的介紹。

3.5.1 技術概念與設計結構

相較於傳統硬體設計會有SCU模型負責提供系統各種時脈來源；而電壓源則透過晶片上的PMU模型控制晶片外的PMIC模型來提供系統的電壓來源。假如是為了提供驅動程式的開發應用，則必須建立與硬體規格一致(暫存器記憶體配置一致)之ESL模型，而建立此種模型之開發時間冗長且只適用於此特定規格之PMU與SCU，一旦系統所選擇之硬體模型更改後，就必須重新開發一次ESL模型，缺乏彈性且曠日廢時。然而依照目前的應用而言，本研究需要的是能在虛擬平台進行模擬過程中，能藉由

CPU透過去設定iPM後，讓iPM提供特定的電壓與頻率輸出，使系統上各個IP跑在期望的電壓與頻率來得到正確的模擬結果。其架構如圖8所示。

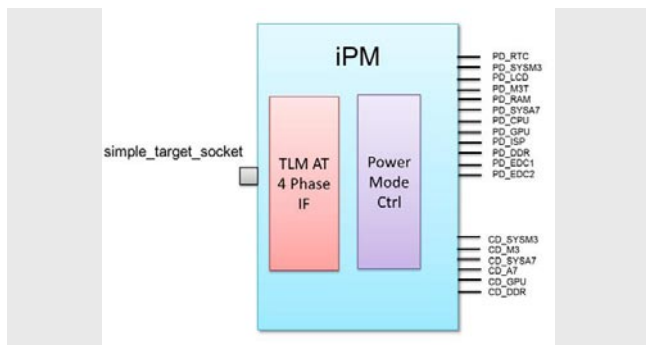


圖 8 智能功率管理元件設計架構

3.5.2 系統層級功率模式描述

在傳統的設計流程中，主要透過CPF或UPF檔案來描述系統有哪些電源域與時脈域，並且會制定出數個系統功率模式，而每個功率模式將規範每個電源域的電壓值，與每個時脈域的週期大小。本研究內容所開發的智能功率管理元件模型就是基於此種想法設計而成，這裡同樣實現了一個自動產生智能功率管理元件模型的程式，只要提供系統層級的DomainList描述檔，與PowerModeList描述檔，便可以自動產生出智能功率管理元件模型。

DomainList描述檔旨在定義系統中共有哪些電源域與時脈域，這也同時決定智能功率管理元件的clock/voltage輸出埠個數、名稱與其資料型態。PowerModeList描述檔則在定義系統有哪些功率模式，每個功率模式底下各個Domain相對應的數值為何，另外此檔案在PowerMode的描述順序上將會決定智能功率管理元件的Control Register：POWER_MODE之定義。以圖8舉例來說當POWER_MODE設定為0時則對應到第一個Power Mode宣告，即PM_OFF mode，設定為1則對應到PM_RUN1 mode。除此之外在產生智能功率管理元件模型時也必須指定域設功率模式(從PowerModeList中任選一個)，讓此智能功率管理元件在elaboration階段即有正常之電壓頻率輸出，才不會造成某些IP在模擬開始時無法正常操作。

4 . 實驗結果

本章節將透過智慧眼鏡虛擬平台作為測試載具來驗證虛擬平台分析軟體，內容主要呈現透過Cortex M3控制iPM藉此改變系統操作模式，而LCDC模組在這樣的設定情況下將從“IDLE”狀態轉變至“ACTIVE”狀態的運作流程。

最終功率分析結果如圖9所示，初始狀態系統功率模式為PM_OFF，然後透過Cortex M3控制iPM藉此改變系統操作模式為PM_RUN1，而LCDC模組在這樣的設定情況下便從“OFF”狀態轉變至“IDLE”狀態，接著Cortex M3再去設定LCDC內部暫存器要求開始動作，因此LCDC模組又從“IDLE”狀態轉變至“ACTIVE”狀態，最後LCDC不斷地讀取存放在LPDDR的圖片，因此LPDDR模組也開始有功率消耗情況發生。

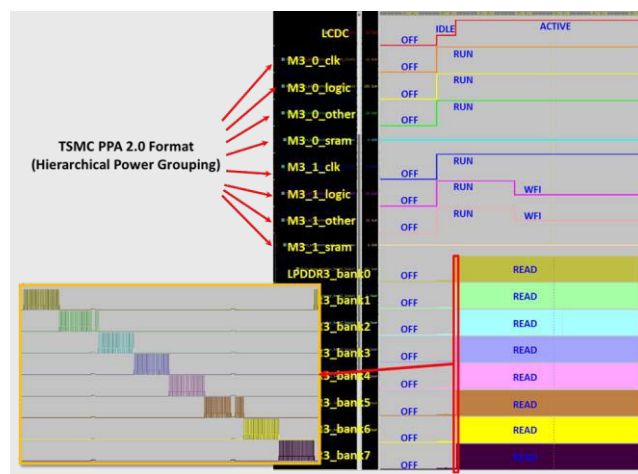


圖 9 智慧眼鏡虛擬平台功率分析結果

5 . 結論

本研究為解決「缺乏系統層級功率分析標準流程」之問題，提出「State-based ESL power analysis methodology」技術方法。該技術不僅提供系統層級功率模式標準定義(Top-Down Flow)，同時亦支援於RTL-to-GDS流程已相當成熟之CPF/UPF檔案格式(Bottom-Up Flow)，藉此讓設計者在系統層級得以即早規劃功率模式以及系統架構的電源域與時脈域，有效整合ESL與現今RTL功率分析技術。此外本功率分析平台將透過前面所定義的系統功率模式資訊自動產生iPM元件，促使原始虛擬平台支援動態

電壓與頻率調整技術，進而幫助系統設計者去開發動態功率管理演算法，藉此提升整體系統效能並有效降低功率消耗。

而最後本研究可有效提供使用者經由功率分析資訊得到整體系統功率消耗的情況，藉以去調整系統架構或是變更電源域之設定，達到在系統開發初期即可對整體功率進行優化與調整之設計需求。

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 Acepplorer. <http://www.doceapower.com>
- [9] W.-T. Hsieh et al., “Transaction Level System Power Estimation Method and System,” U.S. Patent US8,510,694
- [10] Y.-S. Chen, et. al, “Method for inserting characteristic extractor,” U.S. Patent 8 756 544, Jun. 17, 2014.
- [11] XML 1.0 Specification, World Wide Web Consortium, 2010

作者簡介

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，研究領域包括電子系統層級設計方法與高階合成技術。

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，研究領域包括電子系統層級設計方法與功耗分析技術。

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，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。

王逸凱



工研院資通所 / 設計自動化技術組 / 設計流程開發部工程師，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。

鄭良加



工研院資通所 / 設計自動化技術組 / 設計流程開發部經理，研究領域包括超大型積體電路電腦輔助設計與低功耗 IC 設計技術。