針對關鍵任務SoC設計增加韌性的自動化方法

作者

Mary Ann White
產品管理總監

摘要

採用抗輻射強化元件或冗餘 (redundancy) 系統，將安全措施導入系統單晶片 (SoC) 設計，對於航太與國防 (Aerospace and Defense, A&D)、雲端、汽車、機器人、醫藥和物聯網 (Internet-of-Things, IoT) 等產業領域相關的關鍵任務應用至關重要，有助於更具有彈性地因應隨機硬體故障。設計可靠且具韌性的功能確實會對半導體開發產生影響，而這些安全措施通常已經由 SoC 設計人員手動插入。然而，手動方法經常可能導致無法解釋的故障。有鑑於此，新思科技建立了一套完全自動化的實作流程，以插入各種類型的安全機制，從而實現更可靠、更具韌性的關鍵任務 SoC 設計。

本篇技術白皮書會探討在設計中實施安全機制/措施 (Safety mechanism/measures, SM) 的流程，讓機制和措施更具韌性，並分析從啟動設計到最終產品的有效性。

簡介

對於 SoC 設計，重要的是不僅要證明設計功能是正確的，還要確保設計在系統環境中運作正常，而且在目標使用壽命期間安全無虞。在半導體生命週期的各個階段，或許會遇到許多不同類型的錯誤或故障，無論是系統故障或隨機類型的故障，都可能在製造過程或現場運作時發生，而其中具有一些生產中且已部署在各種關鍵任務應用的元件。

在 IC 開發過程階段，系統故障 (systematic failure) 是可以被發現並修正的永久性故障類型。舉例來說，可製造性設計 (Design for Manufacturing, DFM) 工具，例如物理驗證，可以輕鬆識別可能無意中融入設計的訊號短路或開通。使用可靠的測試工作台 (test bench) 進行功能模擬 (functional simulation)，可望識別出設計的安全相關問題。在這兩種情況下，設計問題甚至可以在晶片進入製程之前，就予以修正，讓 SoC 更加穩定。

此外，必須確保用於設計 IC 的電子設計自動化 (Electronic Design Automation, EDA) 工具不會產生可能導致系統故障的安全問題。應遵循穩健的工具驗證及評估流程，以確保這些工具可以為關鍵任務半導體設計進行「安全」部署。

但是，在可能導致 SoC 設計在實地應用時暫時故障的意外事件發生時，該怎麼做呢？針對這些類型的故障，半導體裝置必須具備足夠韌性，才能修正或減輕故障所造成的後果。

可靠性 vs. 韌性

可靠性與韌性的概念和目標之間存在明顯差異，尤其是針對半導體裝置。可靠性是指 SoC 長期正常運行的能力 (例如，沒有出現故障或晶片裝置損害)。反之，韌性則是指半導體裝置從突如其來的故障中「修復」的能力，例如：由太陽閃焰 (solar flare) 引起的單粒子翻轉 (single event upset, SEU)。在這兩種情況下，SoC 的任何故障都可能導致嚴重後果，特別是針對SoC的關鍵任務設計若出現故障，後果更是不堪設想。

電子 IC 設計應在特定使用壽命內維持穩定運作。舉例來說，汽車裝置應維持至少 10 年的可靠度，方能符合 ISO 26262 的規範。在晶片開發階段，應將老化的模型和晶片故障率納入考量，以解釋日後老化效應的影響。此外，設計方法論可以藉由佈署使用 DFM、測試及模擬 EDA 工具，改善製造流程，並藉由嚴格的測試及驗證，實現可靠性。

然而，在設計流程的過程中，必須考慮到 α 粒子和太陽閃焰輻射等 SEUs 造成的隨機硬體故障，才能確保裝置具備足夠韌性。這就是安全措施或安全機制設計概念對關鍵任務設計而言不可或缺的地方，可將影響降到最低，並針對突如其來的SEUs 提供防護，讓設計更有韌性。

安全機制/措施使用冗餘 (redundant) 元件，作為「備援」系統插入，促進快速自動修復，以確保關鍵運作受到保護。這些安全機制/措施能讓實地應用(in-field) 的SoC 從任何意外的 SEUs 中修復，並維持全面運作。

在 SoC 設計中採用冗餘或抗輻射強化元素的安全機制/措施，可以監控並偵測隨機故障的存在，並在故障出現時協助系統達到安全狀態。此外，這些安全機制/措施通常會用於緩解造成目標故障的攻擊。

適用於 SoC 設計的備援系統

隨機硬體故障代表無法控制的事件，像是 SEU 可能會導致glitch或狀態遺失(loss of state)。這些故障無法徹底消除，因此必須透過冗餘和其他安全機制/措施來降低風險。

適用於晶片設計的冗餘有數不同種類型。一般來說，雙重模組冗餘 (Dual Modular Redundant, DMR) 系統會執行錯誤偵測，且需要設計可恢復到安全狀態或操作的情況。然而，三重模組冗餘 (Triple Modular Redundant, TMR) 系統則會執行錯誤修正，讓裝置在發生 SEU 故障的情況下仍可繼續運作。

在半導體設計實作的典型 DMR 和 TMR 系統可透過同位編碼 (parity encoding) 應用於不同元件，像是正反器 (flip flop)、核心和有限狀態機 (finite-state machine)。對於關鍵任務設計，抗輻射強化雙向互鎖儲存單元 (DICE) 正反器使用非常普遍，其包含電晶體 (transistor) 層級的冗餘，而非單元 (cell) 層級。

設計實作挑戰

採用半導體裝置韌性冗餘類型的安全機制/措施並非新鮮事。然而，插入安全機制/措施大部分仰賴手動進行。因此，晶片設計面臨的挑戰除了手動插入冗餘元件外，還有物理佈局的限制，比如：將其放置在相距特定距離的位置上，這樣 SEU 就不會對所有冗餘元件造成影響。此外，必須考量到繞線 (routing) 因素，防止級聯故障 (cascading) 或共模干擾 (Common Mode Interference, CMI) 對重置、功率或時脈網路訊號造成影響。

這些手動方法無法很有效地進行擴展，因為它們通常需要數百行甚至數千行的指令碼，而這些指令碼通常需要針對每個專案項目進行客製化。此外，手動插入安全機制/措施冗餘元件可能會影響功率、效能和面積 (PPA) 目標的平衡，而造成效益分析的難度。

新思科技合成 (synthesis) 及物理實作工具採用全自動插入安全機制/措施的方法，讓關鍵任務設計更具韌性。在合成階段可以自動插入元件，而佈局繞線 (place and route, P&R) 工具將有助解決物理實作挑戰，例如：訊號網的佈局距離和繞線獨立性。

自動化安全機制/措施插入的 EDA 流程

使用安全規範格式 (Safety Specification Format, SSF) 來記錄安全機制/措施，讓新思科技 EDA 工具得以掌握需要在關鍵任務設計中插入、分析和驗證哪些安全機制/措施。典型實作流程與插入安全機制/措施的新思科技自動化實作流程的比較，如下圖1 所示：

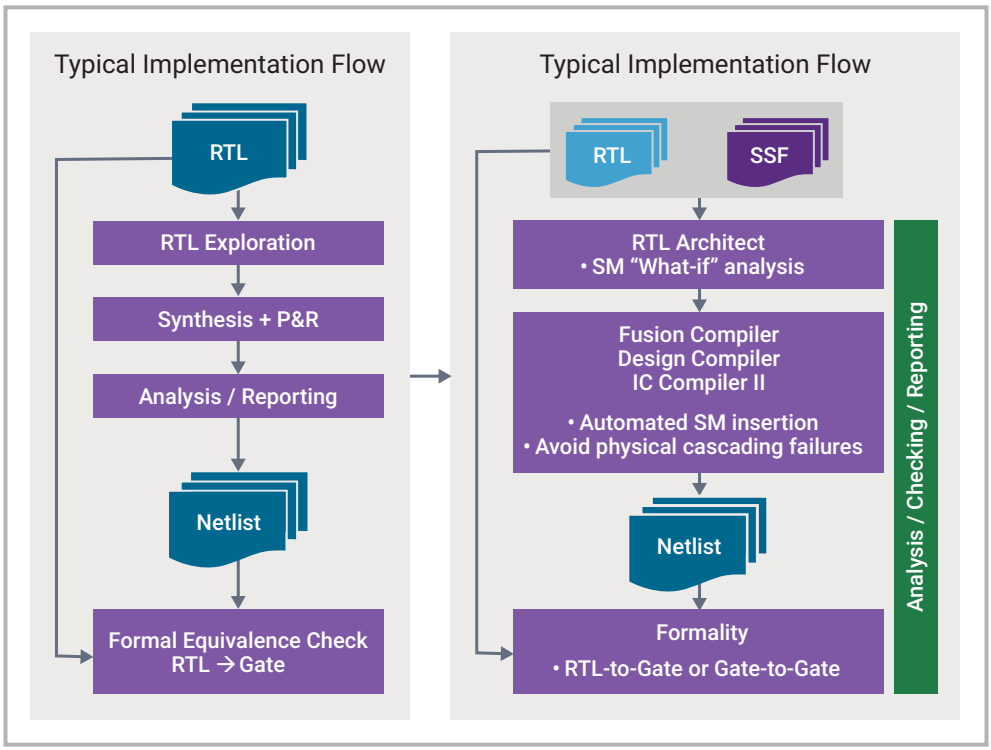


圖1 : 針對晶片設計韌性的安全流程

我們可以使用自動化安全流程識別、分析、插入和驗證安全機制/措施的需求。舉例來說，可以自動實作的冗餘安全機制/措施包括插入雙核心鎖步 (Dual Core Lock Step, DCLS) 以及雙重模組或三重模組冗餘 (DMR/TMR) 暫存器。同樣重要的是，物理實作工具應該要能夠識別並減緩潛在級聯故障，像是重置、功率或應適當分離的時脈網路。

A. 自動化安全暫存器冗餘插入

在正反器上實作冗餘的方法有許多種，而選擇何種方法則取決於韌性與 PPA 目標之間的平衡。

最具韌性的設計仰賴三重模組冗餘 (TMR) 正反器來實現。此系統由三個相同的正反器同步連接，並透過多數表決機制決定正確的輸出。基本上，資料透過這三個正反器進行複製，且能夠偵測並修正 SEU 造成的單一位元錯誤 (例如噪聲或輻射干擾)。在這種情況下，SEU 對所有三個正反器都產生影響的可能性很小，因此可以降低故障造成的後果，讓設計得以正常運作。在各種對策中，這種方式所消耗的晶片面積最大。

第二種替代方法是使用雙重模式冗餘 (Dual-mode Redundancy, DMR)；然而，這只能偵測錯誤，無法進行修正。需要其他邏輯來協助判定當錯誤發生時要採取何種應對措施。

關鍵路徑(critical path)應用於許多關鍵任務設計中，只需使用容錯暫存器進行強化，就可以減少故障的機會。在這種情況下，故障既沒有被修正，也不會被偵測到，而只是大大降低了因 SEU 導致故障的可能性。由於冗餘建立在電晶體層級，因此在替代選擇中消耗的晶片面積最小；但從統計資料來看，相較於三重模組冗餘正反器，其對 SEU 的韌性較低，特別是針對面積較小的先進製程。

另一種較新的方法是將同位錯誤保護方案(parity error protection scheme) 應用於一組暫存器中。相較於 DMR 和 TMR ，雖然這個方式提供的面積較小；然而，卻可以用較慢的頻率取得平衡。Synopsys Design Compiler® 或 Fusion Compiler™ 等合成工具可以自動插入圖 2 所示的任何以暫存器為基礎的安全機制/措施。

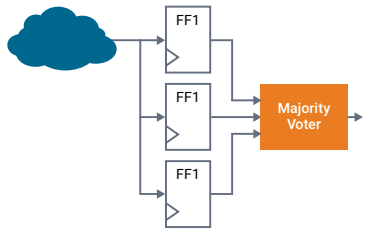
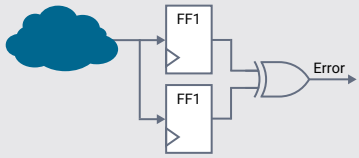
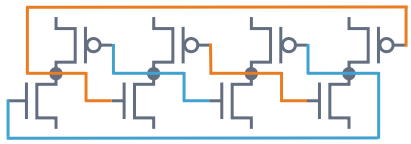
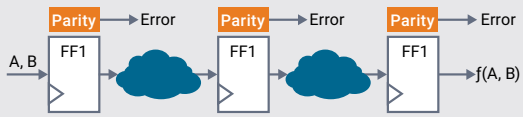
Safety Register Strategy	Description	Design Example
Triple Modular Redundancy (TMR)	• Three registers sample the input state • Majority voting logic • Output is self-corrected	
Dual Modular Redundancy (DMR)	• Two registers sample the input state • Error detection only	
Fault Tolerant (FT)	• Dual Interlocked Storage Cell (DICE) • Resist SEU event (rad-hard) • Reduced area • Available in special libraries	
Error Protection Scheme (Parity)	• Safety applied to register groups / busses • Types: Parity (odd/even), EDC, ECC • Reduced area vs TMR / DMR	

圖2 : 安全暫存器策略

B. TMRs 的物理實作

Synopsys Fusion Compiler 和 IC Compiler™ II 等佈局繞線工具使暫存器的佈局合理化(legalize)，並確保繞線符合無共模干擾的要求。這種工具按照 SSF 意圖所定義的物理距離佈局於暫存器中。作為一項額外的安全措施，這種工具可以在暫存器的兩側插入額外電源插頭，以進一步確保其具備抵禦任何 SEU 的韌性。

圖 3 說明了新思科技工具如何因應 TMR 的物理實作要求，讓 SEU 不會同時影響所有三個暫存器。在暫存器兩側插上電源插頭，作為額外的安全機制/措施，並顯示如何使用獨立物理繞線及緩衝，將安全暫存器時脈/重置分離，以避免網表上的級聯故障。

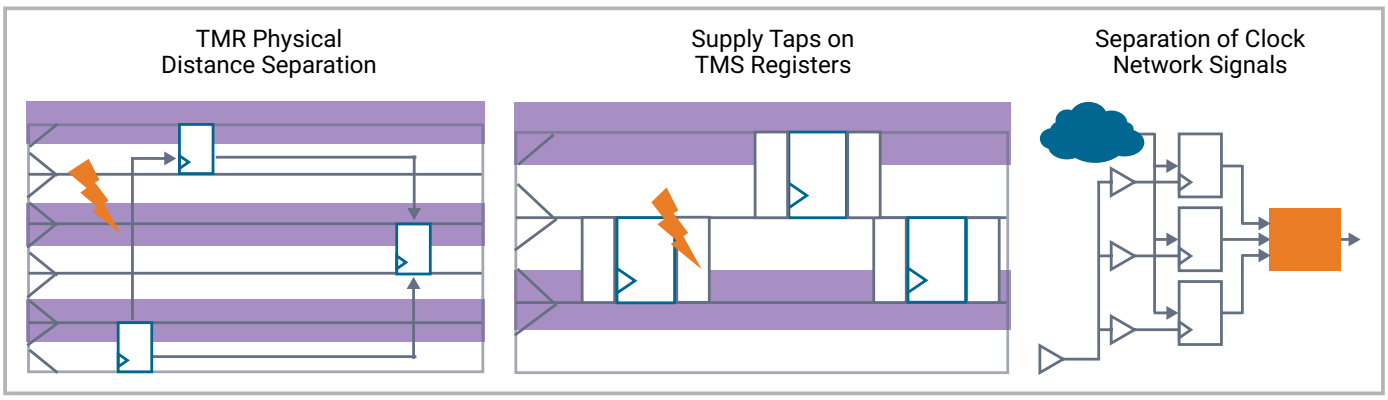


圖3 : TMR 暫存器的物理實作要求

Design Compiler、Fusion Compiler 及 IC Compiler II 均提供強大的錯誤檢測功能，確保安全機制/措施已適當地被採用。並使用視覺輔助工具，如圖 4 所示，將 TMR 組和多數決邏輯(voting logic) 用相同顏色標記出來。

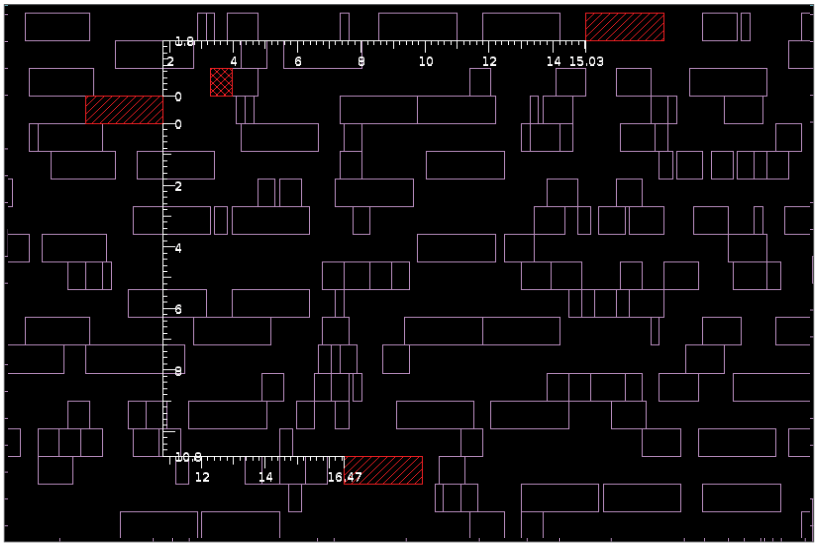


圖4 : TMR 和表決器 (voter) 的物理佈局

C. 冗餘核心設計實作

雙核心鎖步 (DCLS) 是另一種在 SoC 中設計冗餘的方法。DCLS 使用兩個可以同步執行的相同核心，基本上，就是同時執行相同的指令集。比較兩個核心的輸出，以確保這兩個核心產生相同的結果。任何錯誤或不協調 (例如：可能因 SEU 而發生的情況) 都會被偵測為錯誤。

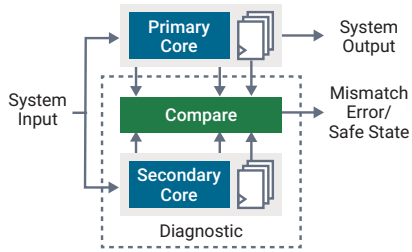
Redundant Core Strategy	Description	Design Example
Dual Core Lock Step (DCLS)	• Two cores run in lockstep (same instruction sets) • Output of cores produce identical results • Any mismatch from a fault gets detected as an error	

圖5 : DCLS 安全機制/措施

DCLS 通常用於關鍵任務應用，以確保半導體設計的韌性。如前所述，雙重冗餘通常用於錯誤偵測，也就是在 DCLS 的情況下，必須設計進一步的邏輯來修正錯誤，並讓核心同步恢復。與 TMR 暫存器一樣，除了錯誤偵測之外，還可以佈署三核心鎖步 (Triple Core Lock Step, TCLS) 進行錯誤修正；然而，因為核心設計為面積密集型，因此實際上不常使用三核心鎖步。

D. DCLS 的物理實作

與 TMR 暫存器相似，DCLS 物理實作的要求包括如下，以避免出現共模故障 (Common mode failure)，如圖 6所示：

- 主要與次要核心之間的物理距離間隔
- 在時脈樹 (clock tree) 合成及時脈網路生成過程中，避免在兩個核心中共用失敗群集
- 採用單一核心的物理繞線，盡可能不與其他核心共用

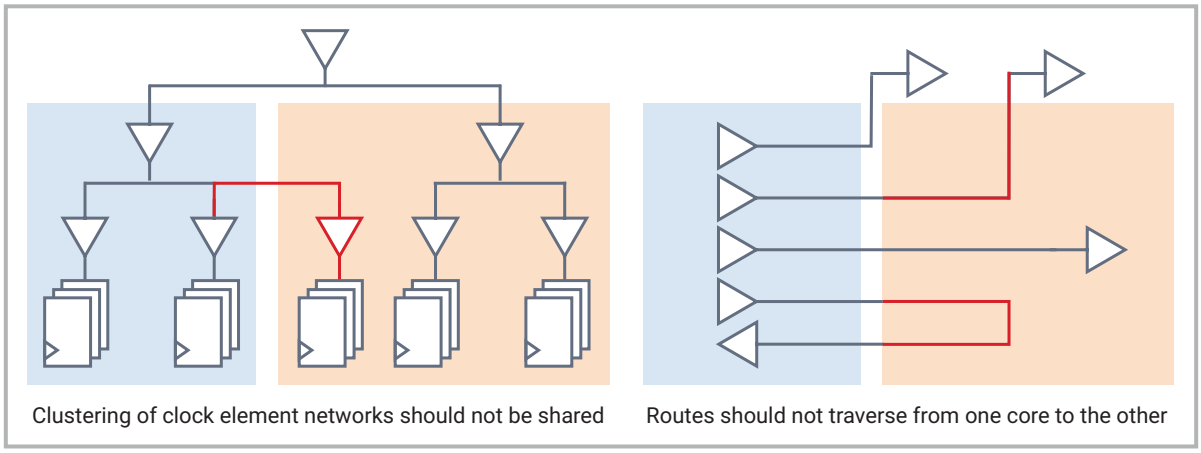


圖6 : DCLS 物理實作要求

將 DCLS應用在設計的範例中，核心不僅應該進行物理分離，而且任何訊號或時脈繞線都不應該從一個核心穿越到其他內核，因為任何共用繞線上的 SEU 可能導致兩個核心都出現故障。以新思科技 ARC 為基礎具有和不具有安全意識的 DCLS 實作範例，如圖 7 所示。核心不僅按照 SSF 指定的距離進行物理分離，而且繞線也會在每個核心內「本地化 (localize)」，因此從一個核心穿越到其他核心的可能性將被最小化。

Fusion Compiler 與 IC Compiler II 中的佈局、時脈樹合成、優化、繞線及回報引擎都已經更新並具有安全感知，以符合用於冗餘暫存器及核心的所有安全機制/措施 要求。

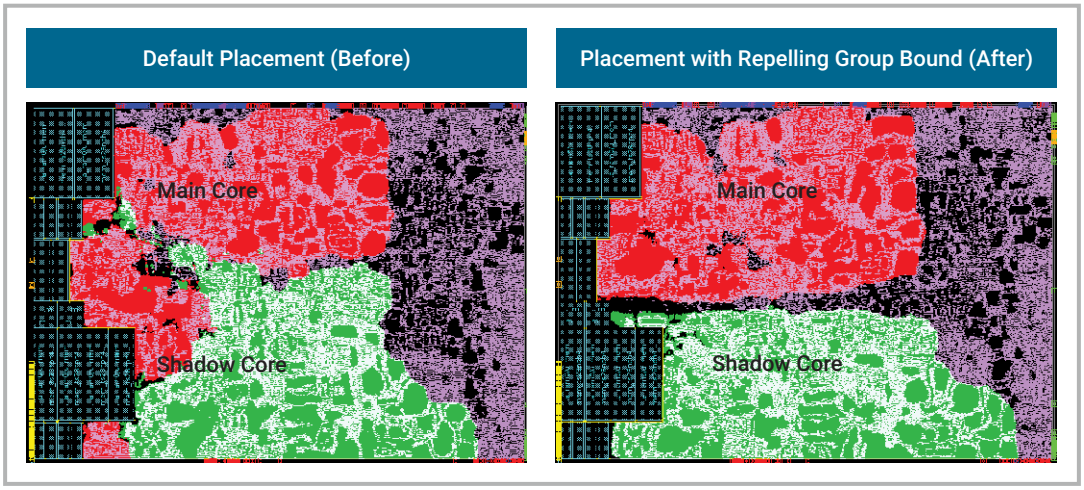


圖7 : DCLS 的安全感知實作

結論

總而言之，可靠性專注於避免半導體設計長期運作時出現故障，而韌性則確保可以從突如其來的事件及單粒子翻轉 (SEU) 導致可能發生的軟性錯誤 (soft error) 中快速修復。這兩個概念在任務關鍵型半導體設計中都很重要。需要實施不同的安全機制/措施對策及平衡，才能達到所需的效能和穩健性層級。

以往，這些安全機制/措施對策都是手動插入，可能需要數千行自訂指令碼，而且容易產生人為錯誤；但新思科技提供了一種完全自動化插入冗餘的方法，讓各個產業的關鍵任務設計都能更具韌性。