



預測試的片上系統設計加快了可編程器件的開發

萊迪思半導體公司

2010年3月

Lattice Semiconductor
5555 Northeast Moore Ct.
Hillsboro, Oregon 97124 USA
Telephone: (503) 268-8000
www.latticesemi.com

引言

許多中等規模的可編程邏輯器件（PLD）設計，特別是控制方面的應用，通過片上匯流排互聯一些介面至微處理器，這些微處理器可以是片上晶片或外部的。雖然每個介面往往是相對簡單的，構建所有片上的互連和對它們進行調試的任務可能會非常耗時和令人沮喪。越來越多的設計人員使用預設計的基於處理器系統的開發板來加快開發過程。

這種方法有幾個好處。首先，沒有必要裝配許多IP或構建選擇的IP至片上匯流排的緩衝。此外，片上匯流排結構不需要選擇和構建。第二是沒有必要手工連接各種IP至片上匯流排。即使是本文中討論的相對較小的設計，使用約1000個查找表（LUT），對於頂層的連接大約需要300行的 Verilog代碼。第三，使用這種方法構建硬體的時間大大減少。設計人員可能需要花數天時間來組成基本的硬體部分，並進行所有重要的介面調試。一旦調試介面工作使設計已經處於穩定狀態，添加或刪除介面就變得非常簡單。此外，如果修改設計導致調試介面失敗，很容易返回到最近所知的成功設計版本。

典型的開發板

這裏以開發環境為實例，允許設計人員用預先設計的系統作為起點，然後修改它來構建自己的樣機，在將它移植到最終的系統之前使用萊迪思的MachXO迷你開發板。雖然適用於許多應用場合，這個電路板最適用於控制設計，這種設計往往用小型的非易失可編程邏輯器件來實現。圖1說明了典型的用可編程邏輯器件實現控制應用的功能。

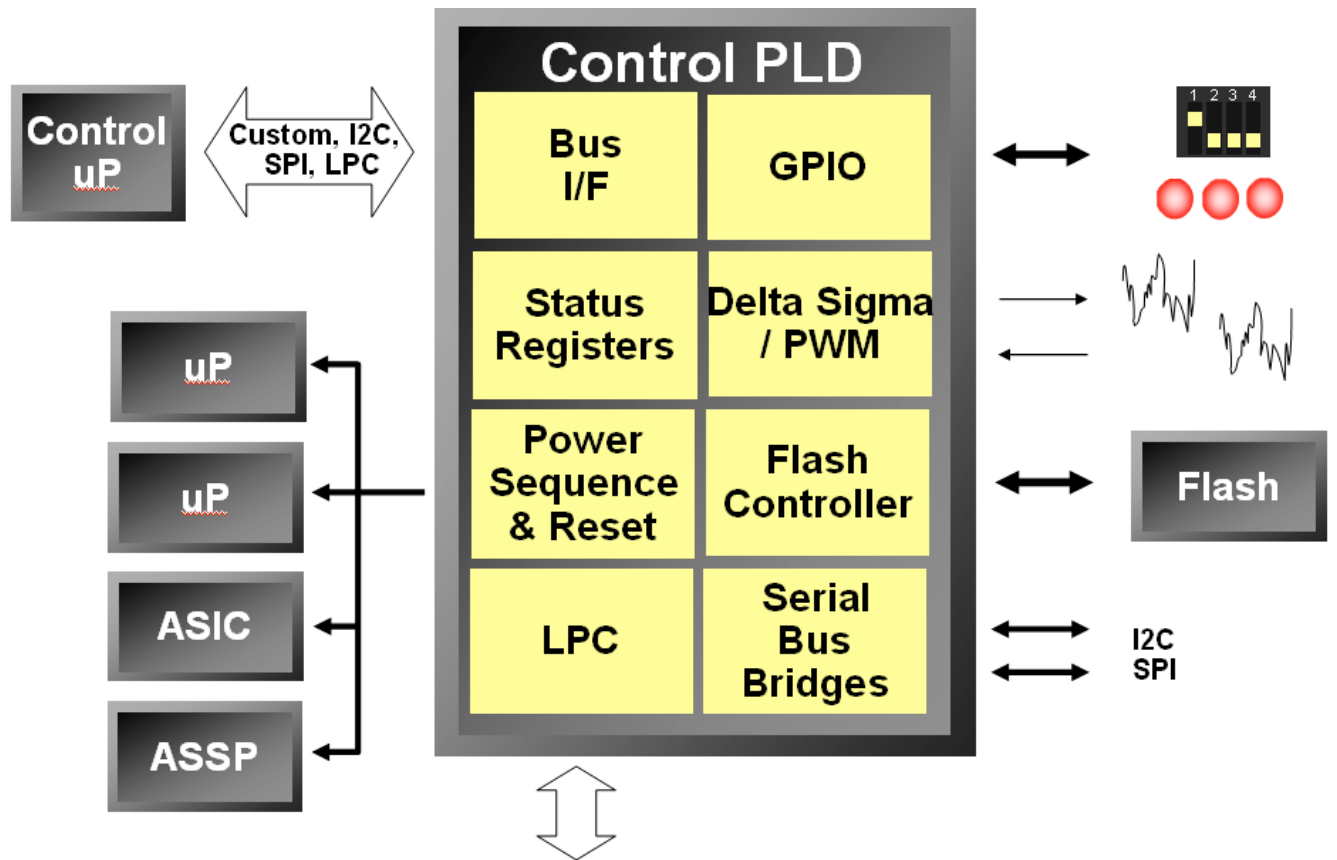


圖1用可編程邏輯器件實現的典型控制功能

迷你開發板的中心是MachX0 2280非易失、基於快閃記憶體擁有2280個LUT邏輯實現的可編程邏輯器件。圖2展示了MachX0迷你開發板，並對主要介面加了注釋。這個電路板提供五種主要的功能：

- 通過發光二極體組、擴展端、DIP開關和按鈕開關提供GPIO和狀態指示器。
- SPI快閃記憶體和I2C溫度感測器用作這些常用串列介面的例子。
- SRAM記憶體提供了額外的便箋式記憶體。
- 通過USB的RS232提供了一個調試介面。
- 通過USB的JTAG用作對器件編程的介面。

對於開發過程而言，通過USB介面至RS232的方式和通過USB至 JTAG的方式是至關重要的。利用這兩個小的“B”型的USB連接器，可以通過電纜連接到任何個人電腦的USB介面。一旦載入適當的驅動程式（適用於Windows XP及更高版本），通過USB介面至JTAG的方式允許萊迪思的ispVM軟體對所需修改設計的MachXO進行編程。採用相應的驅動程式，通過USB介面至RS232的方式對通過USB至RS232的方式進行仿真。驅動程式包括Linux的最新版本，也可用於Windows XP及更高版本。序列埠只作為主機的“COM”埠。此介面為片上設計提供了一個簡便的方法，用於發送狀態及適當修改、調試資訊至主處理器。

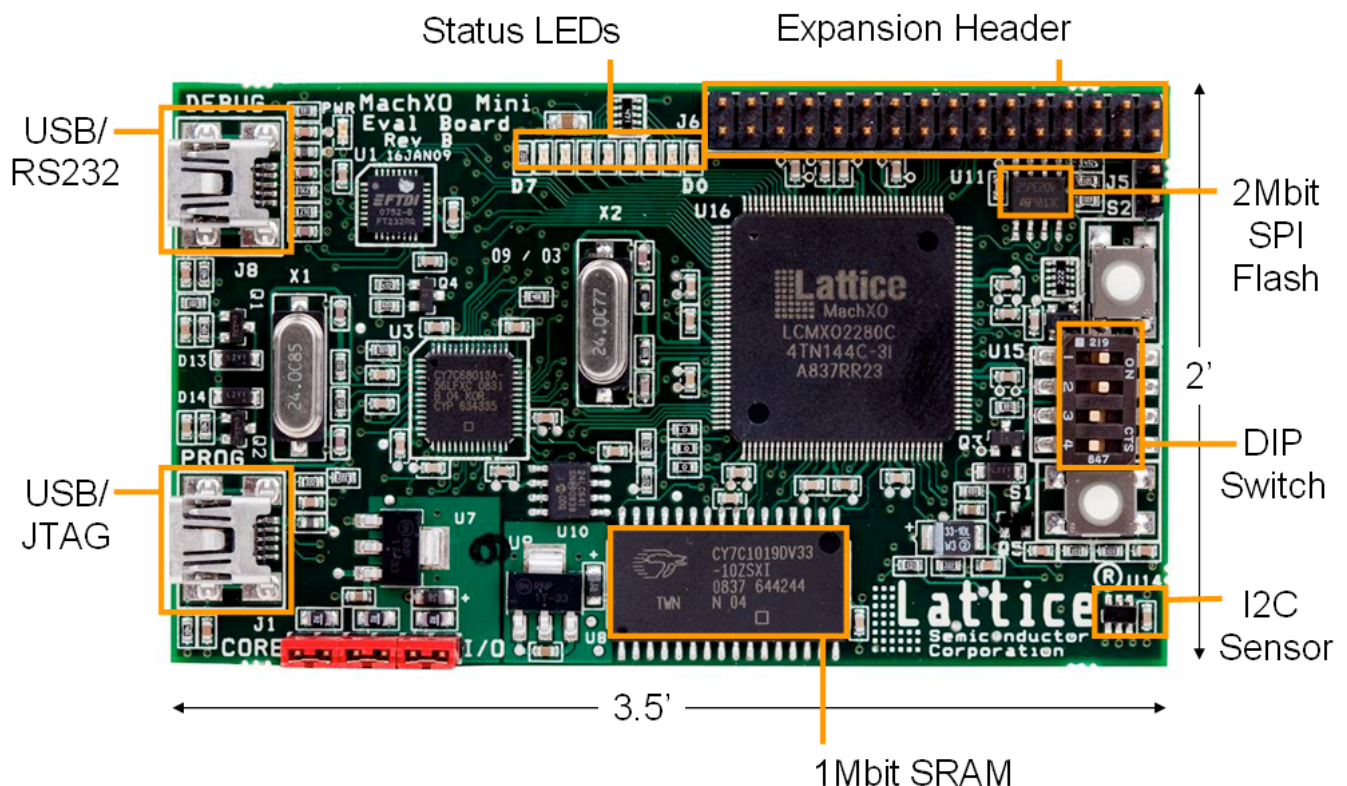


圖2 MachXO迷你開發板

預測試的片上系統設計

迷你開發板加速開發的關鍵是採用預開發的，預載入的片上系統（SOC）設計，可以作為用PLD實現控制功能的起點。下面是一個典型的開發流程和相關的時間，假設相應的軟體工具和驅動程式已被載入至PC主機：

- 1。驗證硬體和介面的正常運行如所期望的那樣。（通常為5分鐘。）此刻設計人員已經瞭解了硬體情況。
- 2。確認可以使用ispVM編程軟體將FPGA配置檔再次載入MachX0。（通常為5分鐘。）
- 3。使用ispLEVER設計軟體重新編譯設計，並確認所產生的配置如預期的情況工作。（通常為10分鐘。）此時，設計者有一個已知的良好起點。
- 4。首先修改設計，重新編譯並載入新的配置。（通常是60分鐘，但實際的時間取決於所修改的量。）

按照這個開發流程，初始開發需要幾個星期的時間可以減少到數小時。SoC設計的概況如圖3所示。可以看到設計由六個介面和一個微處理器組成，並通過一個片上總路線聯在一起。

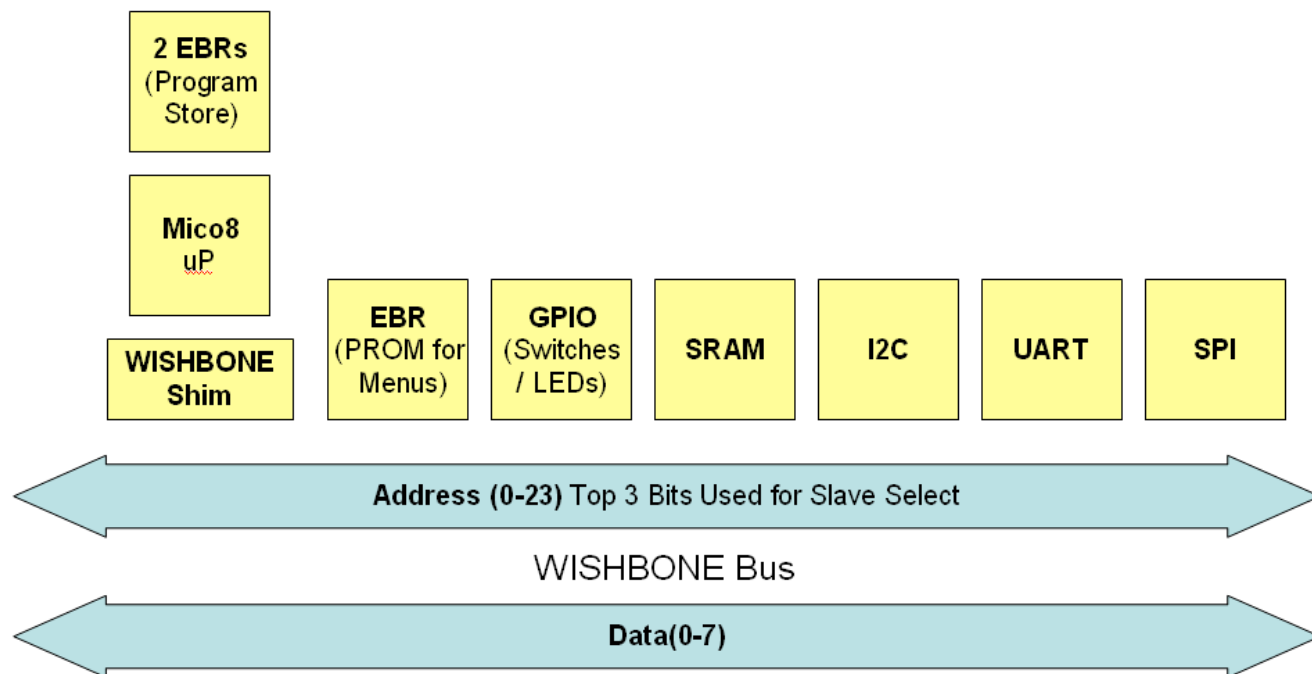


圖3 MachX0迷你板預裝載了SOC設計

這個設計的中心是片上匯流排。使用WISHBONE標準：WISHBONE是一個開放源代碼的匯流排標準，為SOC設計提供了一個理想的基本結構。WISHBONE標準的優點有以下幾個方面：

- WISHBONE是開放源代碼，所以可以用於各種FPGA或ASIC。
- 有大量的適用於WISHBONE介面開放源碼IP。
- 簡單但靈活的定義使設計人員能夠對性能的複雜性進行平衡。

由於許多用於控制的可編程邏輯器件的性能要求相對較低，Wishbone匯流排是與主微處理器作為主器件（**master**）的共用匯流排。位址匯流排有24位元，保留的前3位用於產生從器件的選擇信號。

LatticeMico8（參考設計RD1026）和相關的**WISHBONE**適配器（RD1043）用於構成設計核的處理器。Mico8具有開放源代碼的優點。這是非常重要的，因為設計人員通常要為嵌入式處理器開發大量的代碼。LatticeMico8開放源代碼的性質意味著它可以在各種FPGA或ASIC中實現，保護了嵌入式代碼的投資。

Mico8有8位元的資料通道，18位元寬的指令和16或32個通用寄存器。外部記憶體是分頁的，前面的8位位址包含在相關操作碼中。用R15和R16寄存器控制位址匯流排的其他16位元。程式記憶體可以設置成512和4096之間的深度。在這種情況下，使用18位元寬可達1024深度。程式記憶體存儲在兩個嵌入式RAM塊（EBR.）中，這些可以被初始化為器件啟動時所需的內容。請注意，每個MachX0 **EBR**支援9位元位元組，深度達1024。Mico8大約使用300個LUT。

通過修改彙編清單可以很容易地生成針對Mico8的新代碼，彙編清單是作為專案檔的一部分。於是通過修改後的代碼可以LatticeMico8工具鏈和生成的相應HEX檔。

ispLEVER工具可以使用這些HEX檔，合適地初始化EBR用於編程的存儲。Mico8工具鏈也是開放源代碼，可從萊迪思網站下載源代碼以及可執行檔。

其他的**EBR**記憶體連接到Wishbone匯流排。這個**EBR**配置成1024深度 8位元位元組。針對通過RS232介面的Mico8顯示的各種功能表，存儲空間是用來共用資料。

SPI介面利用了萊迪思的**SPI WISHBONE**控制器參考設計（RD1044）的修改版本。SPI介面連接外部2百萬位元的SPI快閃記憶體至Wishbone匯流排。如果需要的話，**SPI WISHBONE**控制器可以用來控制多達8個從器件（**slave**）。儘管在這種情況下使用了8位元寄存器，接收和發送寄存器可配置為1到32位元長度，。寄存器是雙緩衝的，當微處理

器服務回應中斷請求以清除其他緩衝區時，允許接收或發送資料。這個參考設計大約使用了113個LUT。這個I2C介面使用了Wishbone匯流排界面的萊迪思的I2C主器件（master）參考設計（RD1046）。這個設計支持7位或10位定址模式和一個8位接收/發送緩衝器。I2C參考設計大約使用了234個LUT。

UART的設計採用了**Lattice WISHBONE UART**參考設計（RD1042.）。這個設計用類似常用的獨立NS16450 UART功能實現了一個靈活的UART。設計佔用大約291個LUT的資源。SRAM介面相對比較簡單，是針對這個項目的代碼。這是一個簡單的**WISHBONE**外設的很好的例子。

整合所有的一切

MachX0迷你開發板的目標是為了加快小的、非易失FPGA設計的開發速度。為了實現這一點，所有在本文中討論的關於系統晶片設計的資訊可作為一個套裝軟體從萊迪思網站上下載。而且，除了檔和原始檔案，針對所討論的SoC設計，還提供了一個完整的ispLEVER項目，以及**TransFR**和使用睡眠模式以減少功耗的演示。提供的資訊包括專案檔和所有必要的優先選擇和約束，以便在第一時間編譯設計。

因此，如果要對你系統中的PLD控制設計進行全面的修改，請仔細看看這個開發系統和預測試的設計。它可以為你的下一個設計節省幾周的時間。

###