

## 112-2 計算機組織 Midterm Project: ALU Design 作業說明

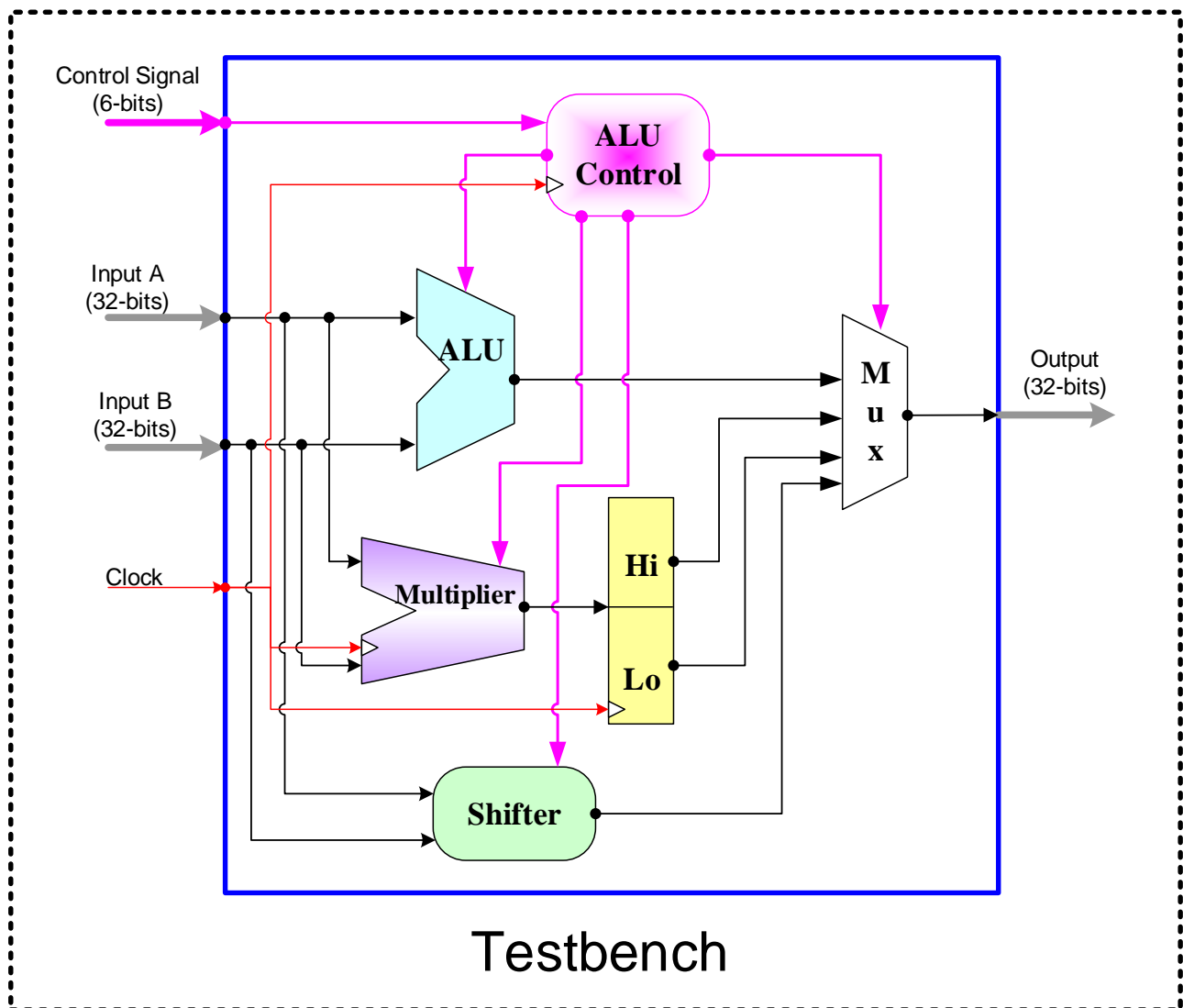
### 【Project 目的】

使用 Verilog HDL 與 Modelsim 模擬器，以計算機組織課程講義：

- [1] Chapter 3:Arithmetic for Computers Part 1: ALU (Add/Subtraction)
- [2] Chapter 3:Arithmetic For Computers Part 2: Multiplication and Division
- [3] Chapter 3:Supplement: Verilog Concepts

為基礎，設計 ALU 與乘法器，以供 Final Project 之用。

基本架構圖請參考「[Midterm Project 概略架構圖](#)」。



Midterm Project概略架構圖

## 【Project 說明】

### 1. 功能說明：本 Project 包含 AND, OR, ADD, SUB, SLT, SLL, MULTU 等 7 項功能，功能簡述如下：

- (1) AND, OR, ADD, SUB, SLT 之運算方式，請參考課程講義[1]P.12-P.28 所述。
- (2) MULTU 為無號數乘法，其運算方式，請參考課程講義[2]P.02-P.14 所述。
- (3) SLL 為邏輯左移，其運算方法，請參考課程講義[1]P.28 所描述之設計方式。

### 2. Midterm Project 概略架構圖之元件說明與設計要求：

其中，(1)為機測門檻，若不遵守設計規範或功能不正確，將不予評分，亦不能參與機測。

(1) **ALU:** 包含 32-bits AND, OR, ADD, SUB, SLT 等功能，並以課程講義[1]P.12-P.28 所述之設計方式，使用 Gate-Level Modeling 與 Data Flow Modeling (Continuous Assignments)，從 Full Adder 做起，以 Ripple-Carry 的進位方式，連接 32 個 1-bit ALU Bit Slice，成為 32-bit ALU。**不能直接用 '+' operator，亦不可使用 Always Block 或 Procedure Assignment 來設計。**本模組為組合邏輯(Combinational Logic)。

(2) **Multiplier:** 為 32-bits 無號數乘法 Sequential Multiplier，須採用 **Second Version Sequential Multiplier** 來設計，請參考課程講義[2] P.02-P.14。可使用 Always Block 或 Procedure Assignment 來設計，**但【不接受迴圈形式的設計】**；意即 Multiplier 內不能有 for/while 等敘述。本模組為循序邏輯(Sequential Logic)，因此須以 Clock 訊號同步。

(3) **Shifters:** 設計 32-bits Barrel Shifter，以完成邏輯左移運算。並以課程講義[1]P.28 所描述之設計方式，以 Data Flow Modeling(Continuous Assignments)完成，**不能直接用 '>>'或'<<' operator，亦不可使用 Always Block 或 Procedure Assignment 來設計。**請以 160 個 Mux 實現 Shifter。本模組為組合邏輯(Combinational Logic)。

(4) **HiLo 暫存器:** 為乘法器計算完後，儲存計算結果之 64-bit 暫存器。本模組為循序邏輯(Sequential Logic)，因此須以 Clock 訊號同步。

(5) **Mux:多工器:** 須以 Data Flow Modeling 設計。本模組為組合邏輯(Combinational Logic)。

**(6) ALU Control:** 根據輸入的 6-bit 控制訊號，決定該完成哪一種運算。控制訊號與功能對應如下：

Signal : 6-bit Value (Decimal)

```
-----  
AND      : 36  
OR       : 37  
ADD      : 32  
SUB      : 34  
SLT      : 42  
SLL      : 00  
MULTU    : 25  
-----
```

本模組為循序邏輯(Sequential Logic)，因此須以 Clock 訊號同步。

**(7) Testbench:** 為所設計之模組之測試平台，須以讀檔的方式，讀入測試資料。以驗證所設計之模組，功能正確性。簡單之讀檔測試範例與測試資料將由助教提供，但機測時將有不同的測試資料，以確認設計正確性與完整性。並藉此評分。

**(8)延遲(delay):** 前述(1)(2)(3)(4)(5)(6)(7)項設計，除了(7)項可包含延遲(#constant)，以作為 Clock 週期與 Reset 之用外，其餘設計均不可包含延遲(#constant)的敘述。

**(9)特別規定:** 本 Project 內所有 Verilog 程式，除了 Testbench 外，其餘部分皆【**不接受迴圈形式的設計**】；意即程式內不能有 for/while 等敘述。亦不能包含 Function/Task 敘述。亦不能包含 always @(\*) 敘述。

### 3. 設計注意事項：

- (1) 需分 Module 與檔案，一個 Module 一個檔案，並以 Module 名稱命名，所有不可寫在同一模組裡。
- (2) Datapath 與課本已有所不同，需自行修改相異之處。請以 Word 或 PowerPoint 重新繪製，並詳述設計方法。未以 Word 或 PowerPoint 繪製者，報告將扣部分分數。
- (3) Testbench 須依所提供之參考設計，將測試指令或程式轉換成指定格式，輸入 Testbench。不可寫死在 Testbench 裡。助教機測時亦將以機測專用輸入檔案加以測試。無法通過測試者，依通過之功能給分。
- (4) 所有設計須以課程所介紹的 Verilog 語法完成，且需通過 Modelsim 模擬。以 Schematic 繪製電路、非規定之 Gate-Level 設計、非課程所介紹的 Verilog 語法部分，將不予計分。

#### 4. 特別規定：

- (1) 若整個 Verilog 設計或報告內包括不屬於上述 7 項運算之任何其他電路，不論是否有功能，或是已註解，每一部份扣學期總成績 5%，並視為版本處理。
- (2) 若程式未達設計門檻，將不予機測或評分。

### 【報告撰寫格式與繳交說明】

#### 1. 報告撰寫格式：

需依照「計算機組織報告格式」撰寫，範本置於 i-Learning，至少 8 頁，除依照格式各章節所需之說明外，需包含下述項目：

- (1) 組別、學號、班別、姓名等資料。
- (2) Datapath 與詳細架構圖，並以 PowerPoint 或 Word 設計繪製。
- (3) 設計重點說明。
- (4) Modelsim 驗證結果與 Waveform 輸出圖形，並加以說明。
- (5) 心得感想。
- (6) 各組員分工方式與負責項目。

#### 2. 報告上傳：

**2024/05/22 PM 9:00 前**上傳至 **i-Learning**。上傳資料需包括：

- (1) 報告 Word 電子檔
- (2) 架構圖 Word 或 PowerPoint 電子檔
- (3) 所有程式檔案與其執行目錄

並將這些檔案置於一資料夾，予以壓縮。

檔案名稱為 CO\_Midterm\_班級\_組別\_組員學號\_組員姓名\_重傳次數.7z。

例如：二甲 第三組 10227000 王大明 重傳第一次

==> CO\_Midterm\_二甲\_第三組\_10227000\_王大明\_1.7z

#### 3. 機測時間：

機測：**2024/05/24 至 篤信 253** 機測。

各組機測時間請於 **2024/05/22 PM 9:00 前**，至 **電學 702** 門口填寫機測時間表。

除排定的機測時間，不接受其他時間機測。亦不接受當場修改。

注意：機測當天需同時繳交報告之書面資料。

#### 4. 注意事項：

(1) 評分標準：機測分數(書面+口試)，答錯者扣該分項分數，並請助教繼續詢問至回答正確或扣完為止。同組同學可能不同分。

(2) 嚴懲抄襲：不論抄襲或被抄襲者均為 0 分。

PS：如有問題請至 i-Learning 討論區或 E-Mail 詢問，我們將會儘快回覆。

如有補充說明會公佈在 i-Learning 討論區，請同學隨時注意消息

朱守禮 2024.04.29