

(四)教學設計與規劃

4.1 「數位系統實務」教學目標

1. 訓練學生的電路學、電子學、數位系統之認識。
2. 著重於實務訓練，結合理論的應用。
3. 學習如何使用硬體描述語言(hardware description language, HDL)來設計數位電路，並學習使用 EDA 電腦輔助設計工具來設計電路。

2	必修	數位系統設計	本課程介紹數位系統之基本概念，以建立未來數位系統設計以及積體電路設計之基礎。
2	必修	數位系統實驗(一)	1. 訓練學生的電學、電子學、數位系統之認識。 2. 著重於實驗訓練，結合理論的應用。
2	必修	數位系統實驗(二)	學習如何使用硬體描述語言 Verilog 來設計數位電路，並學習使用電腦輔助設計工具來設計電路。

4.2 教學對象

本計畫之研究對象為元智大學資訊工程系大學部二年級學生，修習「數位系統實務」必修課程約160人，將招募1班80人，這些學生具備程式設計基本能力，並已修習部分核心基礎課程（如程式語言(一)、資料結構、電子電路學），適合參與FPGA與HDL實作。

4.3 教學場域

4.3.1 場域描述：

本計畫之研究場域為元智大學 3C 數位系統整合實驗室(位於元智一館 1008 電腦教室)，此場域專為數位系統課程設計，配備 80 台以上高階個人電腦，每台安裝 EDA TOOL 鏈軟體 (Quartus II)、30 片 FPGA 開發板 (DE0 系列)、教室 4 台大單槍投影機以及布幕、教師講桌直接配置教學廣播系統，確保學生聽課權益。

測試儀器 (TDS-220 示波器、6005 數位電表、1303AF 電源供應器、邏輯分析儀、MOTEC-FG 506 訊號處理器函數訊號產生器)，以及 5 面以上大白板，便於教師與學生進行實作討論與互動。此環境適合進行 FPGA 遊戲體驗、數位系統開發與實作課程，支持資工專業與跨領域學生修習。

4.4 教學方法與活動

本計畫以元智大學資訊工程系「數位系統實務」課程為核心並區分四個階段，教學規劃、進度與主要活動，如表1，具體課程安排 與教學重點，如表2。

教學設計與時程規劃區分 4 個階段，如表 1，分階段詳細說明如表 2，並特別強調每一階段的 GPBL 任務內容與其教學目標：

表 1「數位系統實務」教學設計與時程規劃表

階段	週次	課程進度/主要活動	備註
第一階段： 課程設計與 遊戲化任務 導入 (6 週)	1	課程介紹與數位邏輯設計概論	分組，27 組 (每 2-3 人)
	2	FPGA 開發環境建置與 HDL 基礎	
	3	DE0 I/O 控制基礎與前測實施	實施前測 (附件一) W3
	4	入門遊戲專案(GPBL I)： 基礎 HDL 專案與 Debug 指引	
	5	中階遊戲專案(II)：記憶體與複雜 FSM	實施期初學習評 量問卷 (附件 2) W5-7
	6	中階遊戲專案(II)：物件移動與碰撞	
第二階段： 遊戲化教學 與任務實作 (6 週)	7	中上階專案(III)：音訊輸出基礎	期初學習問卷停 止收集(附件 2)
	8	中上階專案(III)：多輸入與節奏時序	
	9	中上階專案 (III)：多輸入與節奏時序	
	10	進階專案(IV) 啟動與 VGA 基礎	
	11	進階專案 (IV) VGA 遊戲核心設計	
	12	進階專案(IV) VGA 遊戲邏輯 Debug	
第三階段： 期末專題設 計與整合應 用 (6 週)	13	期末專題 GPBL 任務 提案與架構審核	
	14	期末專題實作	
	15	期末專題實作	開始實施期末學 習問卷(附件 2) W15-17
	16	進階專題系統整合	
第四階段： 學習成效評 估與反思 (2 週)	17	期末專題發表會 (第一天) 同儕互評	期末學習問卷停 止收集(附件 2) 同儕互評 Rubric 評分表(附件 3) W17-18
	18	期末專題發表會 (第二天)、 同儕互評 與後測實施 數據整理與初步分析 研究結論撰寫與課程反思	Rubric 評分表收 集(附件 3) 後側(附件一) W18

表 2「數位系統實務」導入 遊戲專案課程活動與教學內容核心重點

週次	階段	課程安排	遊戲專案課程活動與教學內容核心重點		
1	一	FPGA 與 HDL	遊戲專案教學引導說明、課程目標、DE0 平台介紹、學生分組 27 組 (每 2-3 人)。		
2			Quartus II 流程、HDL 基礎語法		
3			介紹基礎 IP 模組：LED、七段顯示器、按鍵中斷。		
4		GPBL I	定時炸彈 1A2B	按鍵去抖、FSM 設計、精確時序邏輯應用。	
5		GPBL II	小恐龍跳跳接水果大作戰	亂數產生邏輯、數字比較判斷、LED/7Seg 顯示控制。	
6		GPBL II	俄羅斯方塊	Block RAM 應用、清除線路邏輯、複雜 FSM 管理。 物件座標計算、碰撞偵測邏輯。	
7	二	遊戲化教學 (GPBL) 與任務實作	GPBL III	Music Player	PWM 或 DAC 音訊介面驅動、波形 ROM 儲存。
8			GPBL III	MUSIC GAME	多按鍵輸入處理、精準時序判斷。
9			GPBL III	Quiz Buzzer Competition	教師與助教總結 RAM/ROM 應用、進階 FSM 設計，對既有專案進行優化 (資源/速度)。 GPBL III 第三階段總結與 Debug 專題技巧
10			GPBL IV	Snake Game	VGA 時序原理，進階專案啟動，以貪食蛇 (Snake Game) 解說。
11			GPBL IV	Pong Game	物件繪圖 IP 模組設計、座標更新、FIFO 應用。
12			GPBL IV	Space Invaders	複雜碰撞反射邏輯、多按鍵或 PS/2 鍵盤輸入介面整合。總結入門專案常見錯誤，強化 Debug 技巧。
13	三	期末專題設計與整合	學生分組提交期末專題 期末 GPBL 專題任務 提案 (需含完整系統架構圖、分工細節、技術挑戰預估)。		
14			VGA 輸入介面整合，教師與助教 Code Review 與實機驗證		
15			VGA 複雜碰撞整合，教師與助教 Code Review 與實機驗證		
16			專題核心功能整合與 System Debug。		
17	四	學成評與反思	期末專題發表會 同儕互評 GPBL 專案驗收，GPBL 成果展示：學生分組進行專題展示、功能 DEMO。		
18			期末專題發表會 同儕互評與後測實施 剩餘小組發表。 發表會結束後立即實施後測並收集所有 Rubric 評分表。		

階段一：課程設計與遊戲化任務導入 (Weeks 1-6)

本階段的核心設計是「打基礎，點燃興趣」。教學重點從傳統的理論知識傳授轉向以遊戲化的任務引導。

初期 (W1-W3) 專注於教學環境與基本技能的建立，如 FPGA 開發環境、HDL 基礎語法，以及 DE0 板上 I/O 的控制。此階段部署前測，為後續研究提供基準。GPBL 任務從 W4 開始：GPBL 任務 I-A (定時炸彈 Reaction Time Challenge) 訓練學生掌握精確時序邏輯、FSM 設計與按鍵去抖的實作；GPBL 任務 I-B (1A2B 數字猜謎) 則訓練亂數產生與組合邏輯判斷能力。

所有任務皆與積分排行榜掛鉤，激發初期競爭與參與感。

階段二：遊戲化教學與任務實作 (Weeks 7-12)

本階段的設計核心是**模組化進階與核心 IP 建立**，任務難度顯著提升，開始觸及複雜的數位系統核心。

GPBL 任務 II (俄羅斯方塊) 訓練學生應用 Block\ RAM 作為盤面記憶體與複雜 FSM 的設計；**GPBL 任務 II (小恐龍/接水果)** 則聚焦於**物件移動與碰撞偵測**的座標邏輯。

中上階任務進入多媒體應用：**GPBL 任務 III (Music Player)** 要求學生設計音訊 DAC/PWM 驅動與波形 ROM 應用；**GPBL 任務 III (節奏遊戲/Quiz Buzzer)** 則考驗**多輸入處理和精準時序判斷**。

W10 則開始引入 VGA 基礎，啟動進階專案的準備。透過(Snake Game) 說明 VGA 遊戲核心 IP 模組，W11 實作 **GPBL 任務 IV (Pong Game)**訓練 VGA 座標更新與 FIFO 緩衝等技術。隨後的 W12 (Space Invaders) 著重於**複雜碰撞與輸入介面整合** (如 PS/2 鍵盤) 並總結與 Debug 專題確保學生能回顧並優化 HDL 程式碼。

階段三：期末專題設計與整合應用 (Weeks 13-16)

此階段是 GPBL 的「整合應用與獨立創作」。學生運用前兩階段所學的 HDL 模組庫，開始設計一個完整、複雜的數位系統專題。W14 的專題提案與架構審核是**關鍵節點**，確保小組的分工與技術路線可行。W15 開始的實作強調跨模組的系統整合與 Debug。

期末問卷於此階段開始收集，以評估學生在高壓整合挑戰下的學習動機與滿意度。整個階段的目標是將分散的 HDL 知識，通過一個有創意且具挑戰性的遊戲專案進行高度整合。

階段四：學習成效評估與反思 (Weeks 17-18)

本階段旨在「成果展示與科學驗證」。W17-18 的期末專題發表會是學生展現 **GPBL 成果的高峰**，通過同儕互評 Rubric 評分表 (功能性、創意、團隊合作) 進行全面且量化的評估。專題發表後立即實施**後測**，與前測數據進行比較分析，對所有量化數據進行整理、統計分析 (如 t-Test) 與研究報告撰寫，完成整個教學實踐的完整迴圈。

研究分析階段：對前/後測成績、期初/期末問卷、以及 Rubric 評分進行數據整理、統計分析 (如 t-Test)，以科學方法驗證 GPBL 策略對提升學生**學習成效與動機**的有效性，並撰寫最終的教學反思報告。

第一階段到第二階段融入 GPBL 任務設計，區分入門、中階、中上、進階 4 種不同等級與核心 IP 模組訓練 (詳如 4.3 **FPGA 遊戲專案與核心硬體 IP 模組**)，以此為遊戲式專題學習 (Scaffolding) 鷹架。

此架構確保 GPBL 作為學習鷹架，逐步引導學生從基礎到進階，促進知識建構與實作應用。

GPBL 教學設計是一個循序漸進、層層遞進的結構，從 **興趣激發（階段一）** 到 **核心知識建立與模組化訓練（階段二）**，再到 **高階應用與系統整合（階段三）**，最終以 **成果展示與科學研究驗證（階段四）**。

課程設計將 HDL 知識、數位系統實作和遊戲化任務緊密結合，並在關鍵節點部署了前(W3)/後測(W18)、期初(W5-W7)/期末(W15-W17)問卷、和 Rubric 評分(W17-W18)等研究工具，以嚴謹的方法論來驗證 GPBL 對提升學生學習成效與動機的假設。

本「數位系統實務」課程的 GPBL 教學設計採用「打基礎鷹架 to 模組化進階 to 系統級整合 to 實證分析」策略。

設計上將知識點拆解為一系列具有挑戰性與趣味性的 GPBL 任務（從 I/O 控制到 VGA 複雜遊戲），確保學生在實作中掌握 HDL 語法、FSM 設計、記憶體應用、時序邏輯等核心知識。

透過貫穿始終的量化評量工具（前/後測、期初/期末問卷、Rubric 評分），本計畫旨在系統化地驗證這種遊戲化專題學習模式，在提升學生對硬體描述語言的學習興趣、實作能力與最終成效上的顯著影響。

學生以組為單位，80人分為27組（每組2-3人），

「數位系統實務」每週授課時數3小時，

每個GPBL單元評分說明與IP模組解釋，搭配核心IP模組實作應用；首先

1. GPBL任務，包含各單元遊戲與核心硬體 IP 模組 (GitHub)
2. 說明GPBL單元遊戲評分、積分與排名方式(前一周公告)
3. 各組進行GPBL單元遊戲，即時更新積分與排名(1小時)
4. 解釋GPBL單元遊戲IP模組(0.5小時)
5. 各組搭配核心IP模組實作GPBL單元遊戲(1.5小時)

上述方法流程將理論與實作緊密結合。

4.5 成績考核方式

作業（50%）：三份作業，每份佔15%，以能否跑出作業要求結果計分。

共4個等級GPBL任務11個遊戲單元，配分分別為10、10、15、15

透過積分表評分、透過排名。

期末專題GPBL任務（40%）：專題發表將由同儕使 Rubric 評分表進行同儕互評，評分標準包含功能性(40%)、創意性 (30%)、團隊合作 (30%)。(附件 2)

分組與編組設計

- 專題總組數：學生將被分為 27S 組。

- **編組方式：** 採用亂數 $9 \times 9 \times 9$ 的編組概念，以確保互評時各組別的隨機性和公平性。
- **同儕互評編組：** 互評環節是以 9 對 9 的模式進行。這表示在互評時，每組將參與到一個包含 9 個評分者和 9 個被評者的群組中。
- **最終分數計算：** 專題的最終成績（ 40% 權重）將由該組收到的 9 個同儕評分平均值來決定。

期末專題同儕互評設計（佔 40% ）

1. 分組與編組設計

- **專題總組數：** 學生將被分為 27 組。
- **編組方式：** 採用亂數 $9 \times 9 \times 9$ 的編組概念，以確保互評時各組別的隨機性和公平性。
- **互評群組：** 互評環節是以 9 對 9 的模式進行。這表示在互評時，每組將參與到一個包含 9 個評分者和 9 個被評者的群組中。

2. 被評分與評分別組

- **每組被評次數：** 每個小組的專案將會收到來自其他 9 個不同小組的評分。
- **每組評分次數：** 每個小組同時也是評分者，需要對**其他 9 個專案（組別）**進行評分。
- **評分組別：** 分組將採亂數編組的方式，確保評分者和被評者之間的分佈是隨機的。

3. 評分計算方式

- **專題總權重：** 期末專題佔總成績的 40% 。
- **評分依據：** 專題發表將由同儕使用 **Rubric 評分表** 進行互評。
- **評分標準：** Rubric 評分表涵蓋三個維度：
 - 功能性 (40%)
 - 創意性 (30%)
 - 團隊合作 (30%)

課堂參與（5%）： 根據同學出席、課堂參與度評分。

後測成績（5%）： HDL考試，每次佔5%，測試程式設計課程理論知識瞭解程度。

4.6 FPGA 遊戲專案與核心硬體 IP 模組

將每個遊戲拆解為多個核心的硬體 IP 模組。這些模組通常以 HDL 程式形式存在，並在 FPGA 設計中相互連接，以實現遊戲功能。拆解方式契合數位系統設計課程中「模組化設計」和「硬體合作開發」的教學目標。

一、入門遊戲專案 GPBL I (聚焦 I/O 和基本時序邏輯)

遊戲名稱	核心 IP 模組拆解
定時炸彈 (Reaction Time Challenge)	<ol style="list-style-type: none"> 1. 時脈與分頻模組：產生遊戲所需的工作時脈 (e.g., 50 MHz to 1 KHz)。 2. 計時模組：精確計量反應時間 (毫秒等級)。 3. 輸入偵測模組：處理按鍵去抖 (Debouncing) 與中斷信號。 4. 遊戲控制 FSM：管理遊戲狀態 (等待、開始、計時中、結束、顯示結果)。 5. 七段顯示器驅動：將計時結果轉換為七段顯示器上的數字。
1A2B 數字猜謎遊戲	<ol style="list-style-type: none"> 1. 亂數產生模組：根據時脈或特定演算法生成四位不重複的亂數。 2. 指撥開關讀取模組：讀取玩家輸入的猜測數字。 3. A/B 判斷邏輯模組：比較猜測數字與答案，輸出 A 數和 B 數。 4. 遊戲控制 FSM：管理輸入、判斷、顯示 A/B 數量的流程。 5. 七段顯示器驅動：顯示 A 數和 B 數。

二、中階遊戲專案 GPBL II (聚焦更複雜的顯示與狀態管理)

遊戲名稱	核心 IP 模組拆解
小恐龍跳跳 (Dino Jump)	<ol style="list-style-type: none"> 1. 時脈與分頻模組：控制遊戲更新速度與障礙物移動速度。 2. 輸入控制模組：處理跳躍按鈕的輸入，觸發跳躍狀態。 3. 物件位置計算模組：計算恐龍和障礙物的 Y 軸 (跳躍) 和 X 軸 (移動) 座標。 4. 碰撞偵測模組：判斷恐龍與障礙物是否重疊 (遊戲結束)。 5. 計分模組：根據時間累積得分。 6. 七段顯示器驅動：顯示分數。
俄羅斯方塊 (Tetris Mini)	<ol style="list-style-type: none"> 1. 時脈與分頻模組：控制方塊下降的速度。 2. 輸入控制模組：處理移動、旋轉、加速下降等按鍵輸入。 3. 方塊形狀與旋轉邏輯：儲存七種方塊的形狀，並實現正確的旋轉變換邏輯。 4. 遊戲盤面記憶體模組：使用 Block RAM 儲存 10\times 20 等級的盤面狀態。 5. 清除線路邏輯：偵測並清除滿行線路，同時處理上方方塊的下落。 6. 七段顯示器驅動：顯示分數和等級。
接水果大作戰 (Catch Game)	<ol style="list-style-type: none"> 1. 時脈與分頻模組：控制水果 (或物件) 掉落速度。 2. 輸入控制模組：控制接取籃子 (或主角) 的左右移動。 3. 物件產生與移動模組：隨機產生掉落物件的座標，並控制其 Y 軸移動。 4. 接取與碰撞偵測模組：判斷籃子與物件的座標是否吻合 (得分) 或錯過 (扣分/遊戲結束)。 5. 七段顯示器驅動：顯示分數、生命值。

三、中上階遊戲專案 GPBL III (聚焦音訊處理與多媒體 I/O)

遊戲名稱	核心 IP 模組拆解
------	------------

Music Player 音樂播放器	<ol style="list-style-type: none"> 1. 音訊輸出介面驅動 (PWM/DAC)：根據 DE0 硬體，驅動音訊輸出介面（如 GPIO 上的蜂鳴器/外接喇叭）。 2. 音訊資料儲存 ROM：使用 M9K 或 Block\RAM 儲存預設音高或波形資料。 3. 取樣率控制模組：精確控制波形的輸出頻率，以決定音高和音色。 4. 輸入控制模組：處理播放、暫停、選曲等按鍵操作。
MUSIC GAME 節奏遊戲	<ol style="list-style-type: none"> 1. 節奏時序產生器：根據歌曲的 BPM（每分鐘節拍數）產生精確的音符落下時序。 2. 音符判斷邏輯：在正確的時機（Window）內，比對玩家按鍵輸入與預設音符。 3. 音訊輸出模組：負責在成功/失敗時播放回饋音效。 4. 計分與連擊 (Combo) 模組：計算分數並追蹤連擊數。 5. 七段顯示器驅動：顯示分數與 Combo 數。
Quiz Buzzer Competition 測驗蜂鳴器賽	<ol style="list-style-type: none"> 1. 輸入鎖定模組：偵測多個玩家的按鍵輸入，並鎖定第一個按下的玩家（消除按鍵去抖問題）。 2. 優先權編碼模組：將按下的按鍵編碼為玩家 ID。 3. 音效輸出模組：輸出「搶答成功」或「搶答失敗」的蜂鳴器音效。 4. 遊戲控制 FSM：管理「等待問題」、「搶答中」、「鎖定玩家」、「得分計算」等狀態。 5. 七段顯示器驅動：顯示目前得分和搶答玩家 ID。

四、進階遊戲專案 GPBL IV（聚焦 VGA 圖形處理）

遊戲名稱	核心 IP 模組拆解
VGA 顯示之貪食蛇 (Snake Game)	<ol style="list-style-type: none"> 1. VGA 時序產生模組：產生 Hsync/Vsync 訊號與行/列計數，確保畫面顯示時序正確。 2. 畫布繪圖模組：根據 x, y 座標計算，輸出正確的 RGB 顏色訊號。 3. 蛇體座標記憶體：使用 Block\RAM 儲存蛇身每一節的 x, y 座標，需以 FIFO 或環狀緩衝區邏輯管理。 4. 食物與邊界偵測：判斷蛇頭是否撞到邊界、蛇身或吃到食物。 5. 輸入控制模組：處理方向鍵輸入與防止 180 度轉向。 6. 遊戲控制 FSM：管理遊戲速度、狀態與得分。
VGA 乒乓球 (Pong Game)	<ol style="list-style-type: none"> 1. VGA 時序產生模組：產生正確的 VGA 時序。 2. 球/球拍座標計算模組：計算球的速度向量和即時 x, y 座標（包含浮點數模擬的數位化）。 3. 碰撞偵測與反射邏輯：判斷球與球拍、邊界的碰撞，並計算新的反射角度和速度。 4. 畫布繪圖模組：繪製球、球拍、邊界和分數文字。 5. 音效模組：輸出碰撞音效（結合音訊輸出介面）。 6. 輸入控制模組：控制球拍的上下移動。
VGA Air-Hockey / 小蜜蜂 (Space Invaders)	<p>(通用模組，結合上述)</p> <ol style="list-style-type: none"> 1. VGA 時序與繪圖模組：基礎顯示核心。 2. 多物件移動與碰撞模組：（核心難點）處理多個物件（飛彈、敵人、玩家）的移動與複雜的碰撞矩陣。 3. 聲音/音效模組：處理射擊、爆炸等音效。 4. 輸入控制模組：處理複雜的移動與射擊按鍵。

4.7 教學活動角色情境與期末 GPBL 專題任務

4.7.1 教學活動角色任務與教學現場情境

階段	學生任務	主要活動	教師與 TA 角色	預期成果
模組探索期	模組閱讀與理解	學生閱讀現有 IP 模組 (VGA 控制器、音訊 PWM、七段顯示器等)，分析結構與邏輯。	教師透過 GPBL 任務拆解各 IP 模組作為鷹架。講解並提供設計說明文件與範例波形。	學生能正確描述模組功能與訊號流程。
模組改寫期	模組功能優化	學生改寫模組功能，如提升顯示解析度、改變音樂節奏控制、擴充遊戲關卡。	教師引導 Debug 技巧與模組整合方法。	學生能理解他人程式並完成自主功能擴充。
整合驗證期	遊戲系統整合	組合多模組完成完整遊戲專案，如「音樂節奏遊戲」或「貪食蛇」。	教師協助進行實體板端測試與除錯。	學生掌握多模組整合、時序同步與驗證流程。
學習成果展示期	分組期末專題製作	學生自選題目，改造或延伸現有遊戲模組，開發具創意之新遊戲。	教師擔任顧問，協助設計評分規準。 同學分組互相評分 期末專題評分 Rubric 構成：專題發表將由同儕使用 Rubric 評分表進行互評，評分標準包含功能性(40%)、創意性(30%)、團隊合作(30%)。(附件 2)	完成可於 DE0 板上執行之創新遊戲作品。

4.7.2 期末 GPBL 專題任務 遊戲式模組化進階任務範例 @GPT

專題模組特性整理表

專題名稱	主要模組組成	核心學習重點	難度
VGA 迷宮探險	VGA + FSM + ROM Map	地圖與碰撞邏輯	★★★★
VGA 打磚塊	VGA + 碰撞 + 計分	運動與反彈邏輯	★★★★
VGA 貪食蛇	VGA + FSM + 緩衝記憶體	動態座標與資料結構	★★★★★
VGA 乒乓球	VGA + 雙控制 + Timer	雙方互動與時序控制	★★★★
VGA 空中戰鬥	VGA + Sprite + 碰撞	多物件繪製	★★★★★
VGA 塊狀拼圖	VGA + 陣列 + 消除 FSM	二維邏輯控制	★★★★★
VGA 賽車遊戲	VGA + 滾動背景 + 碰撞	Scroll 特效與事件處理	★★★★★
VGA 打地鼠	VGA + 隨機 + 計時	隨機事件處理	★★★★
VGA 記憶配對	VGA + 匹配 + FSM	邏輯判斷與同步顯示	★★★★
VGA 節奏反應	VGA + 時序 + 判斷	節奏同步與延遲偵測	★★★★★

GPBL I-A 定時炸彈(Reaction Time Challenge)

定時炸彈

我們會預設20秒的時間，在按下button後開始做倒計時，

密碼利用四個指燈開做輸入，

所以密碼可能是0000、0001、0010、0100等，

猜對的話計時就會停止，沒猜到且時間歸零的話就會亮LED燈。

- 預設60秒時間
- SW0~7為輸入密碼(如:11111111、11111010)，對應的LED:1亮0暗
- LED9亮:輸入密碼比標準密碼小；LED8亮:輸入密碼比標準密碼大；兩個皆亮:答案正確
- 時間內猜對，倒數計時停止並在7-Segment Display顯示77表示成功 (LED8、9兩個皆亮:答案正確)
- 時間到未猜對，時間歸零並在7-Segment Display顯示88表示失敗
- button2為reset

使用一個button當作開關，按下就開始倒計時

撥動四個switch當作輸入的密碼，

如0000、0001、0010等等。

利用7-Segment Display顯示秒數、成功、失敗。

□預設60秒時間做倒計時



□LED9亮:輸入密碼比標準密碼小；LED8亮:輸入密碼比標準密碼大



LED9亮:比密碼小

LED8亮:比密碼大

□SW0~7為輸入密碼(如:11111111、11111010)，

對應的LED:1亮0暗

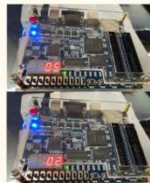


□時間內猜對，倒數計時停止並在7-Segment Display顯示77表示成功 (LED8、9兩個皆亮:答案正確)



照片

□時間到未猜對，時間歸零並在7-Segment Display顯示88表示失敗



照片

□button2為reset



GPBL I-B 1A2B 數字猜謎遊戲

Let's Play The 1A2B Game

START

How To Play

1. 使用者利用switch輸入三碼數字，每一位介於0-9(因為switch有限)並且對input值顯示在七段顯示器上ex: 123 則 001 010 011
2. 確認答案後按下button，即顯示答案，並在顯示答案後顯示幾秒
3. 使用者共有十條命，假設輸入錯誤，除了顯示提示 (XA XB) 之外，LED燈會在退出答案後熄滅一顆
4. 假設在十次內猜對(3A0B)，LED燈將會像馬燈一樣輪流亮起
5. 如果想再玩一次 (Reset)，則按下另一顆button到新的畫面

當我們在思考要製作什麼主題時，首先從日常生活中的應用開始思考，我們想到了許多好玩的小遊戲，其中1A2B猜數字遊戲是一個經典的遊戲。這個遊戲需要一定的思考但不會太困難，因此非常有趣且容易上手。

玩家需要猜測一組數字，如果猜錯，就會得到提示。這些提示告訴他們有多少數字是對的，並且這些數字的位置是否正確。基於這些提示，玩家需要繼續猜測直到猜對為止。

我們相信這個遊戲非常適合在DE0板上製作，因為可以使用七段顯示器、LED燈、switch和按鈕等元件實現豐富遊戲玩法。

GPBL II-A 小恐龍跳跳(Dino Jump)

恐龍跳跳

小恐龍是大家耳熟能詳的遊戲，當沒有網路的時候，Chrome的小恐龍遊戲可以消磨時間並訓練反應能力。

將小恐龍遊戲實現現在七段顯示器上面，以方格代替小恐龍，並設置上下障礙物。Switch控制恐龍的上下移動，若碰到障礙物會顯示LOSE，贏了則顯示PASS。

Input



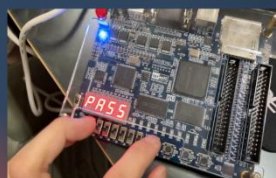
- Switch #0 重新開始遊戲
- Switch #3 模擬滑鼠單擊的動作

Output



- 七段顯示器：
 - 顯示地圖
 - 輸贏的提示
- LED*10 用來顯示幕前的分數 (分數起始至1023)

影片 (LOSE)



影片 (WIN)

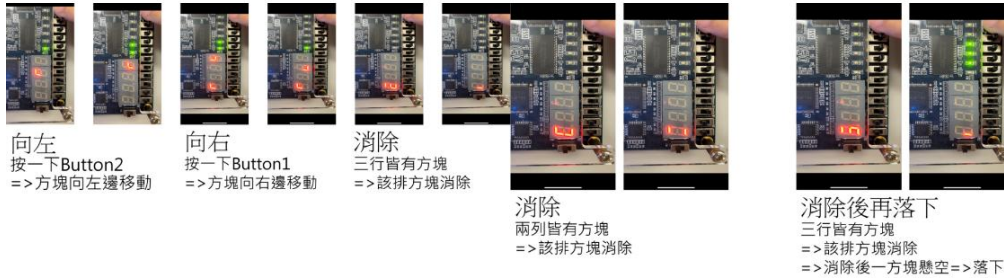
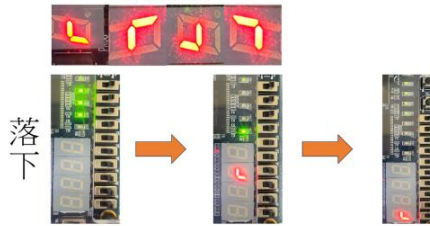


GPBL II-B 俄羅斯方塊(Tetris Game)

俄羅斯方塊

- Button0 =>重置遊戲
 - Button1 =>控制方塊向右
 - Button2 =>控制方塊向左
 - SW8 =>控制速度
 - SW9 =>控制速度
 - LED6~9 =>顯示預備落下
 - 七段顯示器 =>遊戲畫面、結束畫面
1. 隨機生成方塊。
 2. 從LED燈開始顯示落下，利用按鈕控制落下之方塊的左右，方塊落到七段顯示器最底部或下面有其他方塊時即停止掉落。
 3. 若一整排都是亮的即消除整排方塊，消除後若有方塊可以掉落則隨即落下。
 4. 若是有方塊停留在最上面無法再落下則結束遊戲並顯示LOSE字樣。

生成方塊



GPBL II-C 接水果大作戰(Catch Game)

水果大作戰

- 遊戲中，玩家需要透過button控制籃子移動，接住從頂部掉下來的水果。
- 每接到一個水果，玩家獲得1分但如果水果掉落在地上，遊戲結束。
- 遊戲的難度會逐漸提高，掉落速度會變快，需要更敏捷地反應和操作。

Input:
button: 按鈕輸入訊號，用來控制移動位置
button_rst: 按鈕輸入訊號，用來重置遊戲與顯示分數
clk: 時脈輸入，控制掉落速度

Output:
base_out: 3bit的輸出訊號，代表最底下控制移動的3個位置
l_out: 3bit的輸出訊號，代表左邊掉落位置的顯示
m_out: 3bit的輸出訊號，代表中間掉落位置的顯示
r_out: 3bit的輸出訊號，代表右邊掉落位置的顯示
l: 12bit的輸出訊號，代表七段顯示器顯示部分，用以輔助呈現遊戲開始前數數與遊戲結束後顯示分數

初始畫面

遊戲順序從左至右，遊戲開始前會先顯示3、2、1，接著開始遊戲。

遊戲進行

遊戲開始時，水果會從最上方最右邊三點位置隨機生成，玩家的目標是透過按鈕控制籃子移動接住水果，避免水果掉落在地上。遊戲中，籃子位置，玩家可以移動籃子，調整接水果的位置。

結束介面

最終數到1如果沒有成功接到水果時，遊戲將會結束並顯示上打贏片數時間來遊戲。

此時，玩家可以按下最右方的“RESTART”按鈕，重新顯示新的分數。

再次按下“RESTART”按鈕，則遊戲將會重新開始，回到初始畫面。

GPBL III-A 音樂播放器(Music Player)

音樂播放器

設計動機&理念

學術需求

設計一個音樂播放器可以讓學生深入了解數位系統的基本原理和實踐技巧。

學術需求

學術需求

應用需求

現代生活中常常需要用到音樂播放器，如運動時聽音樂、開車時聽音樂、工作時聽音樂等等。

創新設計

在設計音樂播放器的過程中，可以加入一些創新元素，如：觸摸感應、無線、雲端服務 (Reverb) 等。

使用的IO

謝謝聆聽

GPBL III-B 節奏遊戲(MUSIC GAME)

GPBL III-C 電子琴

Music is important

電子琴



實驗器材



DE0板



喇叭

大致步驟

1. 用Arduino開發板寫一段電子琴的code一段歌曲的code
2. 燒錄到DE0板
3. 連接喇叭的腳位並接入DE0板由sw0~sw7的音樂輸入
4. 在DE0板上輸出音樂

電子琴code

```

#include <Arduino.h>
#define SW0_PIN 2
#define SW1_PIN 3
#define SW2_PIN 4
#define SW3_PIN 5
#define SW4_PIN 6
#define SW5_PIN 7
#define SW6_PIN 8
#define SW7_PIN 9

// Piano keys
const int keyNotes[] = {
  440, 492, 554, 617, 688, 750, 814, 880,
  938, 1014, 1093, 1176, 1260, 1350, 1445, 1545,
  1650, 1761, 1878, 1999, 2126, 2260, 2399, 2543,
  2693, 2849, 3011, 3180, 3356, 3539, 3728, 3923,
  4125, 4333, 4548, 4770, 4999, 5235, 5478, 5728,
  5985, 6249, 6520, 6800, 7087, 7381, 7682, 7990,
  8305, 8627, 8956, 9292, 9635, 9985, 10342, 10706,
  11077, 11455, 11839, 12229, 12625, 13027, 13435, 13848,
  14267, 14691, 15120, 15555, 15995, 16440, 16891, 17347,
  17808, 18274, 18746, 19223, 19705, 20197, 20699, 21211,
  21733, 22265, 22807, 23359, 23921, 24493, 25075, 25667,
  26269, 26881, 27493, 28115, 28747, 29389, 30041, 30693,
  31355, 32027, 32709, 33391, 34083, 34785, 35497, 36219,
  36951, 37693, 38445, 39207, 39979, 40761, 41553, 42355,
  43167, 43989, 44821, 45663, 46515, 47377, 48249, 49131,
  49993, 50875, 51757, 52639, 53531, 54433, 55345, 56267,
  57199, 58141, 59093, 60055, 61027, 62009, 62991, 63983,
  64985, 65997, 67019, 68051, 69093, 70145, 71207, 72279,
  73361, 74453, 75555, 76667, 77789, 78921, 80063, 81215,
  82377, 83549, 84731, 85923, 87125, 88337, 89559, 90791,
  92033, 93285, 94547, 95819, 97101, 98393, 99695, 101007,
  102029, 103061, 104103, 105155, 106217, 107289, 108371, 109463,
  110565, 111677, 112799, 113931, 115073, 116225, 117387, 118559,
  119741, 120933, 122135, 123347, 124569, 125801, 127043, 128295,
  129557, 130829, 132111, 133403, 134705, 136017, 137339, 138671,
  140013, 141365, 142727, 144099, 145481, 146873, 148275, 149687,
  151109, 152541, 153983, 155435, 156897, 158369, 159851, 161343,
  162845, 164357, 165879, 167411, 168953, 170505, 172067, 173639,
  175221, 176813, 178415, 180027, 181649, 183281, 184923, 186575,
  188237, 189909, 191591, 193283, 194985, 196697, 198419, 200151,
  201893, 203645, 205407, 207179, 208961, 210753, 212555, 214367,
  216189, 218021, 219863, 221715, 223577, 225449, 227331, 229223,
  231125, 233037, 234959, 236891, 238833, 240785, 242747, 244719,
  246701, 248693, 250695, 252707, 254729, 256761, 258803, 260855,
  262917, 264989, 267071, 269163, 271265, 273377, 275499, 277631,
  279773, 281925, 284087, 286259, 288441, 290633, 292835, 295047, 297269,
  299491, 301723, 303965, 306217, 308479, 310751, 313033, 315325, 317627,
  319939, 322261, 324593, 326935, 329287, 331649, 334021, 336403, 338795,
  341197, 343609, 346031, 348463, 350905, 353357, 355819, 358291, 360773,
  363265, 365767, 368279, 370791, 373313, 375845, 378387, 380939, 383501,
  386073, 388655, 391247, 393849, 396461, 399083, 401715, 404357, 406999,
  409651, 412313, 414985, 417667, 420359, 423061, 425773, 428495, 431227, 433969,
  436721, 439483, 442255, 445037, 447829, 450631, 453443, 456265, 459097,
  461939, 464791, 467643, 470505, 473377, 476259, 479151, 482053, 484965,
  487887, 490819, 493761, 496713, 499675, 502647, 505629, 508621, 511623,
  514635, 517657, 520689, 523731, 526783, 529845, 532917, 535999, 539091,
  542193, 545305, 548427, 551559, 554691, 557833, 560985, 564147, 567319,
  570491, 573673, 576865, 580067, 583279, 586491, 589713, 592945, 596187,
  599439, 602691, 605953, 609225, 612507, 615799, 619101, 622413, 625735,
  629067, 632409, 635761, 639123, 642495, 645877, 649269, 652671, 656083,
  659505, 662937, 666379, 669831, 673293, 676765, 680247, 683739, 687241,
  690753, 694275, 697807, 701349, 704901, 708463, 712035, 715617, 719209,
  722811, 726423, 730045, 733677, 737319, 740971, 744633, 748305, 751987, 755679,
  759381, 763093, 766815, 770547, 774289, 778041, 781793, 785555, 789327,
  793109, 796891, 800683, 804485, 808297, 812119, 815951, 819793, 823645,
 
```

GPBL IV-A VGA 顯示之貪吃蛇(Snake Game)

SNAKE VIDEO GAMES	MOTIVATION	GAME SETTING
<p>1.娛樂性：貪食蛇遊戲可以提供玩家無盡的娛樂，讓他們放鬆心情並享受遊戲的樂趣。</p> <p>2.挑戰性：貪食蛇遊戲可以提供不同難度的遊戲模式，從而讓玩家挑戰自己的反應速度和技巧。</p> <p>3.懷舊感：貪食蛇遊戲是一個經典的遊戲，許多人在年少時都曾經玩過這個遊戲。對於這些人來說，再次玩貪食蛇遊戲可以讓他們重溫過去的回憶。</p> <p>4.簡單易懂：貪食蛇遊戲的操作非常簡單，只需要控制貪食蛇前進吃食物即可，這讓任何人都可以輕鬆上手，無需太多學習成本。</p>		<p>1. 貪食蛇每次吃到一個方塊，身體長度會增加一格</p> <p>2. 當貪食蛇碰到邊界時，遊戲結束，顯示lose</p> <p>3. 當貪食蛇得到15分時，遊戲結束，顯示End</p> <p>4.左邊7段顯示器顯示遊戲時間，右邊7段顯示器顯示分數</p> <p>5. 超過10分後，蛇的速度越來越快</p> <p>6. 每次方塊的位置隨機產生</p>
FUNCTIONS		
<p>BUTTON0 - 順時鐘方向轉向</p> <p>BUTTON2 - 逆時鐘方向轉向</p> <p>BUTTON1 - start game</p> <p>HEX0 & HEX1 - 顯示遊玩時間(0-99)</p> <p>HEX2 & HEX3 - 顯示分數(0-15)</p> <p>VGA - 連接螢幕，顯示遊戲畫面</p> <p>SWITCH0 - reset game</p>		<p>影片:遊玩主畫面 https://youtu.be/LD-ncHX-HIM</p>  <p>影片:計時&&得分畫面 https://youtu.be/8Fftwlf4zrl</p>

GPBL IV-B VGA 乒乓球(Pong Game)

Pong-game

使用課堂中老師所教授之
verilog designing 來模擬經典
街機遊戲**Pong-game**



需要IO設備

VGA螢幕(用來輸出畫面)
PS/2滑鼠(用來控制球拍移動)
SW0切換模式/SW1暫停

遊戲功能介紹























































遊戲畫面



GPBL IV-C Air-Hockey (Space Invaders)



GPBL 積分排名 @1125v3 有分

GPBL 11 遊戲總積分表														
組別	隊伍成員	GPBL I-A	GPBL I-B	GPBL II-A	GPBL II-B	GPBL II-C	GPBL III-A	GPBL III-B	GPBL III-C	GPBL IV-A	GPBL IV-B	GPBL IV-C	總分	排名
第 1 組	  	4	95	89	58	80	91	72	76	24	91	14	694	3
第 2 組	  	82	44	14	13	56	10	73	11	10	20	42	375	27
第 3 組	  	99	13	62	4	9	90	70	16	94	15	96	568	10
第 4 組	  	88	9	38	77	48	5	92	48	62	70	45	582	7
第 5 組	  	29	32	95	72	72	59	11	7	74	51	31	533	12
第 6 組	  	41	22	28	96	80	36	88	17	18	95	77	598	6
第 7 組	  	12	59	91	24	12	42	54	99	33	42	39	507	18
第 8 組	  	67	33	15	76	86	97	15	63	0	24	43	519	14
第 9 組	  	64	92	22	48	1	92	27	31	96	57	75	605	5
第 10 組	  	13	62	41	30	44	65	12	9	34	98	60	468	21
第 11 組	  	68	13	46	95	69	89	0	35	54	36	28	533	13
第 12 組	  	92	67	14	16	48	0	74	7	52	42	44	456	24
第 13 組	  	70	12	83	32	58	58	62	7	73	34	52	541	11
第 14 組	  	19	44	38	65	9	98	26	35	62	12	3	411	26
第 15 組	  	100	33	84	15	17	44	65	11	74	14	55	512	15
第 16 組	  	61	68	58	76	94	96	91	16	71	80	5	716	2
第 17 組	  	82	8	55	17	36	79	84	12	41	52	44	510	17
第 18 組	  	53	31	49	45	63	9	73	90	24	63	80	580	8
第 19 組	  	34	32	16	78	68	19	81	41	5	38	54	466	22
第 20 組	  	58	6	69	75	9	25	88	11	21	52	51	465	23
第 21 組	  	73	23	62	23	1	45	25	56	27	40	39	414	25
第 22 組	  	6	95	59	52	32	1	45	56	5	66	73	490	19
第 23 組	  	21	64	86	74	46	21	84	9	2	23	50	480	20
第 24 組	  	86	26	20	1	75	47	16	62	49	57	73	512	16
第 25 組	  	41	74	90	55	96	54	53	65	61	50	43	682	4
第 26 組	  	71	80	23	88	95	72	63	98	26	77	29	722	1
第 27 組	  	58	95	65	54	36	42	20	22	86	9	91	578	9

附件 1 HDLFPGA 知識能力評量

The set of pre/post-test questions for the Digital System Practice course, focusing on Hardware Description Language (HDL) syntax and FPGA Design. The questions are categorized into Basic (B), Elementary (E), and Advanced (A) levels, with 10 questions each, for a total of 30 questions.

Digital System Practice HDL & FPGA Knowledge Assessment (30 Questions)

Advanced

1. The primary goal of running Timing Analysis in the FPGA development tool (like Quartus) is to:
A. Ensure uniform coding style B. Check for correct I/O pin assignment **C. Verify if the circuit can operate reliably at the target clock frequency** D. Estimate power consumption
2. In the context of a team hardware project, the most critical element to define as a Standard Interface between two independently developed modules is the:
A. Module name B. Code line count **C. Clock and Reset signals and I/O bus protocol** D. Designer's name
3. The technique of Pipelining is commonly used in hardware accelerators on an FPGA primarily to:
A. Reduce power consumption B. Reduce LUT resource usage **C. Increase clock frequency and throughput** D. Simplify the HDL code
4. When the synthesis tool reports a "Setup Time Violation", the most probable underlying hardware cause is:
A. Logic depth is too deep B. HDL syntax error C. Input voltage instability D. Too few flip-flops were used
5. Finite State Machine (FSM) design often utilizes One-Hot Encoding for its states. The main benefit of this encoding scheme is:
A. Saves LUT resources **B. Simplifies state decoding and increases transition speed** C. Reduces routing congestion D. Easier to debug
6. To implement efficient Digital Audio Output on an FPGA, the core HDL module must focus on:
A. Optimizing VGA timing B. Introducing a CPU core C. Using more LUTs **D. Precise DAC or PWM driving and sample rate control**
7. In a Verilog Testbench, which key block is used to generate the initial input stimulus signals for the Design Under Test (DUT) within the simulator?
B. always B. module **C. initial** D. Data Valid signals
8. To minimize logic complexity and ensure proper synthesis in FPGA design, which HDL description practice should generally be avoided in high-performance paths?
A. if-else statements **B. Unoptimized floating-point arithmetic** C. Shift registers D. Case Statement
9. The primary purpose of using a FIFO (First-In, First-Out) buffer in a digital system is to:
A. Store a large amount of static data **B. Coordinate data transfer between two different clock domains** C. Increase VGA display brightness D. Handle button debouncing
10. The most common method used to mitigate issues related to Clock Domain Crossing (CDC) in an FPGA design is by using:
A. Two-stage or multi-stage DFF synchronizers B. Logic Analyzer C. JTAG Interface D. Tri-state Buffer

Elementary

1. When designing a Finite State Machine (FSM), which mechanism is primarily used to manage the sequence of states and transitions?
A. Assign statements B. Always @(*) for everything C. Initial block
D. Case statement or multi-always blocks with posedge clk
2. The primary goal of implementing Debouncing Logic for button inputs is to:
A. Increase input speed B. reduce circuit noise **C. Eliminate multiple transitions from a single press**
D. Lower FPGA power usage
3. The Clock Divider module in FPGAs, essential for slowing down the clock frequency for I/O devices, is most commonly implemented using a:
A. Multiplexer **B. Counter** C. Encoder D. Decoder
4. When driving a Seven-Segment Display on the DE0 board, the technique used to display multiple digits by rapidly switching between them is called:
A. VGA Driving B. PLL Locking **C. Multiplexing** D. DAC Conversion
5. The primary functional difference between Blocking (=) and Non-blocking (<=) assignment in synthesizable HDL is the type of circuit they infer:
A. Whether wire can be used B. Whether they can be used in combinational logic **C. Combinational vs. Sequential Logic** D. Whether they require a clock
6. For storing large amounts of game data (e.g., game board status or image data) on an FPGA, the most efficient resource is:
A. Combinational Logic (LUT) B. I/O Pins **C. Block RAM** D. Phase-Locked Loop (PLL)
7. When designing a VGA Driver circuit, which two timing signals must be precisely generated by the HDL module?
A. R,G,B color signals B. Read,Write signals **C. Hsync,Vsync signals** D. Data Valid signals
8. The main function of the PLL (Phase-Locked Loop) component found in FPGAs is to:
A. Generate VGA timing B. Generate high-speed random numbers **C. Generate multiple clocks with precise frequency and phase** D. Store large datasets
9. In FPGA design, which design practice is most likely to lead to Race Conditions or non-deterministic behavior if used incorrectly?
A. Using synchronous reset **B. Mixing blocking and non-blocking assignments to the same signal** C. Using Case Statement D. Using Assign Statement
10. For a basic VGA display (e.g., 640×480), if the display requires 3 bits for Red, 3 bits for Green, and 3 bits for Blue, how many total data bits are needed for each pixel's color information?
A. 1 bit B. 4 bits C. 8 bits **D. 9 bits**

Basic

1. Which keyword in Verilog is typically used to describe Combinational Logic?
A. always @(posedge clk) B. always @(*) C. initial D. task
2. Which data type in Verilog is primarily used to model a physical connection or net that does not store a value?
A. reg B. integer C. wire D. time
3. Which type of logic circuit is the D-Flip-Flop (DFF) classified as in FPGA design?
A. Combinational Logic B. Sequential Logic C. Arithmetic Logic D. Tri-state Logic
4. When connecting two Verilog modules together via the module ports, the signal carrying the data between them is typically declared as?
A. reg B. integer C. wire D. real
5. The core programmable element of an FPGA does NOT typically include which of the following?
A. Look-Up Table (LUT) B. Block Random Access Memory (Block RAM) C. Microprocessor Core (CPU Core) D. Routing Resources
6. What is the correct term for creating an instance of a sub-module within a top-level module in Verilog?
A. assignment B. declaration C. definition D. Instantiation
7. To describe a 2-to-1 Multiplexer using procedural HDL, which construct is most appropriate?
A. while B. for C. if-else D. forever
8. A key advantage of using Asynchronous Reset in digital circuits is that it provides:
A. Easier routing B. Lower power consumption C. Faster reset response time D. Less signal delay
9. What is the primary function of the Synthesis step in the FPGA development flow?
A. Checking syntax errors B. Translating HDL into a logic gate netlist C. Generating test waveforms D. Programming the chip
10. Which Verilog assignment operator is typically used inside an always @(*) block when describing Combinational Logic?
Blocking Assignment

期末專題發表與同儕互評 Rubric 評分表

本 Rubric 評分表適用於「數位系統實務」課程期末專題發表與同儕互評。每組學生進行 10 分鐘展示(實機 Demo+ 架構說明)，由其他組學生使用此表評分。評分分為三個主要類別：功能性 (40% 權重)、創意 (30% 權重)、團隊合作 (30% 權重)。每個類別包含 5 個具體評分項目，共 15 項。每項 評分採用李克特式量表 (1-5 分)，其中 1 分為最低 (需大幅改進)，5 分為最高 (優秀)。總分計算方式：功能性得分 $\times 0.4$ + 創意得分 $\times 0.3$ + 團隊合作得分 $\times 0.3$ ，滿分 100 分。評分後，請提供簡短文字回饋。

功能性 (40% 權重)：評估專題的技術實現與效能

- **HDL 語法正確性**：專題中 Verilog/VHDL 語法是否無錯誤，並正確實現邏輯功能？
 1. 語法錯誤頻繁，功能無法實現。
 2. 多處語法錯誤，功能部分實現。
 3. 少數語法錯誤，功能基本實現。
 4. 語法正確，功能穩定。
 5. 語法優化，功能高效無誤。
- **FPGA 模組整合**：專題是否有效整合多個 FPGA 模組 (如 I/O、時序邏輯、VGA)，並實現系統整體功能？
 1. 模組整合失敗，系統無法運作。
 2. 模組鬆散整合，功能不穩定。
 3. 模組基本整合，功能可運作。
 4. 模組良好整合，功能流暢。
 5. 模組創新整合，功能卓越。
- **實機 Demo 表現**：專題實機展示是否順利運行，無明顯技術故障？
 1. Demo 失敗或頻繁當機。
 2. Demo 部分運行，有多處故障。
 3. Demo 基本運行，少數故障。
 4. Demo 順利運行，無明顯故障。
 5. Demo 完美運行，效能優異。
- **除錯與穩定性**：專題是否具備有效除錯機制，並在展示中維持系統穩定？
 1. 無除錯機制，系統不穩定。
 2. 除錯機制弱，穩定性差。
 3. 除錯機制基本，穩定性一般。
 4. 除錯機制良好，穩定性高。
 5. 除錯機制先進，系統極穩定。
- **效能優化**：專題是否優化資源使用 (如 FPGA 資源、時序)，提升整體效能？
 1. 資源浪費，效能低劣。
 2. 資源使用一般，效能不足。
 3. 資源基本優化，效能可接受。
 4. 資源良好優化，效能良好。
 5. 資源高度優化，效能卓越。

創意 (30% 權重)：評估專題的創新設計與應用價值

- **遊戲設計原創性**：專題遊戲是否具備獨特設計元素，而非簡單複製既有遊戲？
 1. 完全抄襲，無原創性。
 2. 少數原創，設計平庸。
 3. 基本原創，設計一般。
 4. 良好原創，設計吸引人。
 5. 高度原創，設計創新。
- **跨領域整合**：專題是否有效整合音訊、圖形或其他跨領域元素 (如感測互動)？
 1. 無整合，單一領域。
 2. 整合嘗試失敗。
 3. 基本整合，效果一般。
 4. 良好整合，效果顯著。
 5. 創新整合，效果卓越。
- **使用者體驗設計**：專題遊戲是否提供良好互動體驗 (如易用性、娛樂性)？
 1. 體驗差，難以使用。
 2. 體驗一般，互動不足。
 3. 體驗基本，互動可接受。
 4. 體驗良好，互動流暢。
 5. 體驗優異，互動創新。
- **問題解決創新**：專題是否運用創意方法解決 FPGA 設計挑戰 (如複雜邏輯)？
 1. 無創新解決，依賴基本方法。
 2. 少數創新，解決一般。
 3. 基本創新，解決可行。
 4. 良好創新，解決有效。
 5. 高度創新，解決卓越。
- **應用潛力**：專題設計是否具備實務應用價值或擴展可能性？
 1. 無應用價值。
 2. 應用潛力低。
 3. 應用潛力一般。
 4. 應用潛力良好。
 5. 應用潛力高，具擴展性。

團隊合作 (30%權重)：評估團隊協作與溝通表現

- **任務分工合理性：**團隊成員是否合理分工，並有效完成各自任務？
 1. 分工混亂，任務未完成。
 2. 分工不均，部分完成。
 3. 分工基本合理，任務完成。
 4. 分工良好，任務高效。
 5. 分工優異，任務完美。
- **溝通與協調：**團隊在專題開發過程中是否有效溝通與協調衝突？
 1. 溝通失敗，衝突頻發。
 2. 溝通不足，協調一般。
 3. 溝通基本，協調可行。
 4. 溝通良好，協調有效。
 5. 溝通卓越，協調完美。
- **展示準備與表現：**團隊在發表中是否充分準備，並有效呈現專題？
 1. 準備不足，表現差。
 2. 準備一般，表現平庸。
 3. 準備基本，表現可接受。
 4. 準備良好，表現吸引人。
 5. 準備優異，表現出色。
- **回饋接受度：**團隊是否開放接受同儕回饋，並展現學習態度？
 1. 拒絕回饋，態度消極。
 2. 接受有限，態度一般。
 3. 基本接受，態度可接受。
 4. 良好接受，態度積極。
 5. 高度接受，態度學習導向。
- **團隊貢獻均衡：**團隊成員是否均衡貢獻，並展現集體責任感？
 1. 貢獻不均，責任感低。
 2. 貢獻一般，責任感不足。
 3. 貢獻基本均衡，責任感可。
 4. 貢獻良好均衡，責任感高。
 5. 貢獻極均衡，責任感卓越。

總分計算範例：假設功能性平均得分 4.2 (滿 5)、創意平均得分 3.8、團隊合作平均得分 4.0，則總分 = $(4.2 \times 0.4) + (3.8 \times 0.3) + (4.0 \times 0.3) = 1.68 + 1.14 + 1.2 = 4.02$ (換算為百分制約 80.4 分)。此 Rubric 確保評分客觀公正，並促進學生反思與改進。

學習動機相關評量題目（元智大學教學評量卷－李克特五點量表）

類別	評量項目 (Assessment Item)	學習動機關聯性
教師教學態度與方法	這門課的教師教法能引發學習興趣，激勵思考。	（核心） 直接評量教學法對學生興趣與思考動機的影響。
教師教學態度與方法	這門課的教師鼓勵同學發問、討論或其他的互動。	評量參與度與互動性，鼓勵主動探索的環境。
教師教學態度與方法	這門課的教師關心學生學習成效。	
教師教學態度與方法	這門課的教師教學態度與方法之綜合表現良好。	
教師教學態度與方法	這門課的教師對本課程講解清楚且條理分明（如講解理論時能適時舉例、適切提供指引、以協助學生了解教學內容）。	評量理解支持度，理解是持續學習動機的前提。
課程內容	教師採用的教材(教科書、參考資料或講義等)有效輔助學習。	
課程內容	整體而言，這門課的評量方式（例如：作業、考試、報告等）適當。	
學生自評	我對這門課的上課態度、學習精神、課前課後研習等各方面綜合考量後，我自認為對這門課：	評量學生自身對課程的投入程度（學習精神與研習意願）。
學生自評	本課程使我獲益良多（如專業知識、技能、態度、學習方法或價值觀等）。	評量學習滿足感與自我效能，間接影響持續探索的意願。
核心能力	創新與應用 資訊科技、工程及數學知識的能力。	評量課程能否激發學生主動應用與創新的意願。

備註說明：

- 五點李克特量表 (Likert Scale): 建議採用以下五個選項：
5: 非常同意 (Strongly Agree) 4: 同意 (Agree) 3: 普通 (Neutral) 2: 不同意 (Disagree) 1: 非常不同意 (Strongly Disagree)
- 這些題目結合了教師的教學設計（是否引發興趣）、環境的支持（是否鼓勵互動）以及學生的自我感知（學習精神、獲益程度），能夠較全面地評估 GPBL 策略對學習動機的影響。