

DCLab1

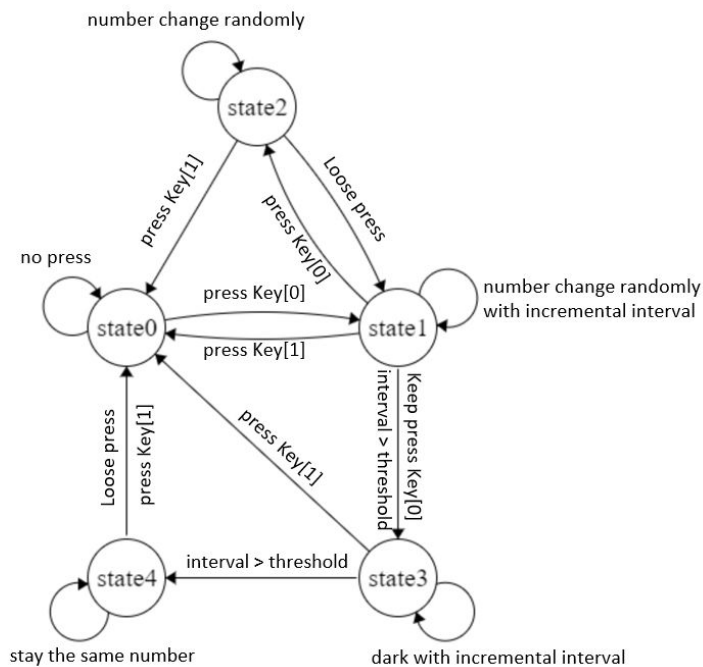
Team04 韓秉勳 蔡昕宇 解正平

User Manual

1. How to work

- 按下Key[0]
數字從00開始隨機變成00-15的數字，且變換間隔逐漸增加。
- 立即放開Key[0]
數字持續快速隨機改變，直到再按下Key[0]，才回到上一個步驟。
- 持續按著Key[0]
數字變換間隔增加，直到一段時間停止改變，閃爍一次表示完成點名。
- 按下Key[1]
數字回到初始狀態00，直到再按下Key[0]開始點名。
- Double Control
除了Key[0]和Key[1]可以點名，也可以使用Key[3]和Key[2]來點名。

2. Finite State Machine



3. User friendly design

- 在計時器停下時會閃爍一次，告訴使用者確切數字停下的位置。
- 可以利用 Key[0]~Key[4] 同時點兩個人的號碼，增加效率。
- 若誤短觸 Key[0]，系統會判定為無效功能而繼續隨機跳動。

Teaching Manual

I. 實驗目的

1. 了解 Verilog 的基本語法
2. 練習 Finite State Machine Diagram 設計
3. Quartus II 和 FPGA 的基本操作
4. 熟悉 nWave 來檢查程式
5. 了解隨機演算

II. 實驗方法

設計點名器控制器，數字快速隨機變化，變化漸慢完成點名。

III. 安裝Quartus II

1. DClab Website : NAS (<https://140.112.33.165:25251/>)
2. 下載支援自己作業系統的安裝檔，建議 NAS 可找到15.0版本，下載好執行。
3. 下載 Cyclone IV Library，建議 NAS 可找到463.9MB版本，下載好並設定。
4. 使用 NTU 的 IP 設定 License。
5. 安裝 USB 連接埠。
6. 搜尋 Installer 安裝 Library。
7. 完成Quartus II 即可支援本實驗所使用的 FPGA 開發板(DE2_115)。
8. 安裝步驟同學們都能自行完成，基本上以上步驟都完成即可使用。

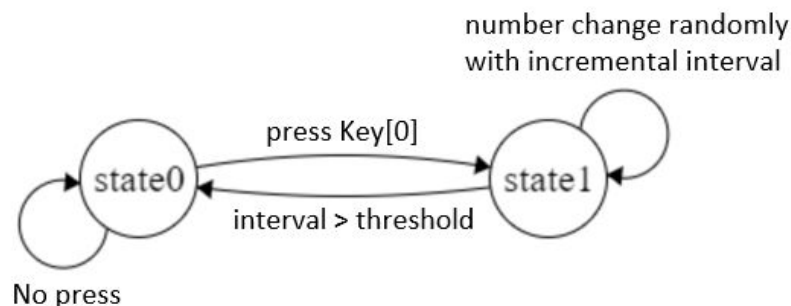
IV. FSM diagram 設計

1. 設計概念：

藉由狀態設定以及改變條件，完成構想的電路。以下舉例的狀態圖都沒有為狀態命名，同學可以自行練習根據該狀態的特性進行命名。

2. 實驗基本概念：

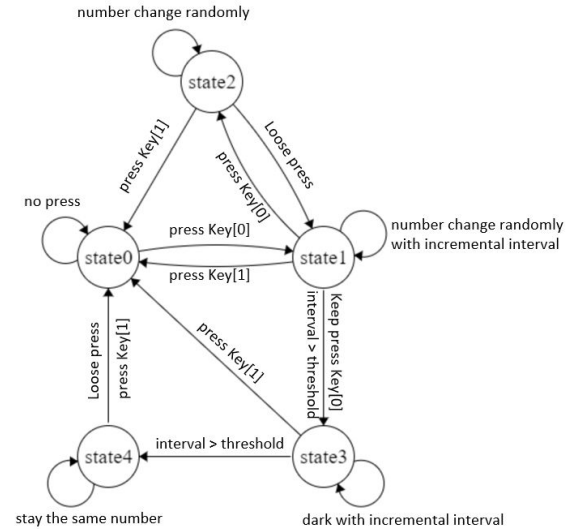
按下按鈕後開始切換數字，當數字切換間隔達到上限後便回到初始狀態，完成點名。設計兩個狀態，一個是初始狀態，一個是數字隨機變換的狀態，利用兩個狀態的變換去設計電路。



3. 實驗進階概念：

本實驗沒有明訂點名器需如何設計，只要維持實驗方法的規定，同學們可以自行新增功能，豐富點名器的應用，以下為舉例說明，右圖為設計概念圖。需要持續按著按鈕才可完成點名。

- 放開按鈕，數字會不斷快速切換，直到持續按下按鈕。
- 完成點名的時候會以閃爍提醒使用者完成點名。
- 若使用者想初始化數字為00，可以按下初始按鈕。
- 為了加速點名，使用者可以增加更多顯示器及按鈕來新增點名器。



V. Quartus II 和 FPGA 操作注意事項

- FPGA 請同學們插上電源並開機，然後以 USB 線連接電腦。
- 創建 project，檔名需與資料夾相同，路徑與 top.sv 相同。
- 嵌入所有 *.sv 及 *.sdc 檔案。
- Import 剛初始的 assignment “DE2_115.qsf” 不是創建 project 出來的新的。
- 按下 cltr+L 來 compile code
- Open programmer 並且加入 *.sof 檔，按 start 即可在 FPGA 使用。
- 完成實驗記得存下 *.sof 檔，下次可以直接燒錄使用。

VI. nWave 使用方式和如何 debug

- 於 sim/ 資料夾底下 type command **make -f ../Makefile Top**。
- 使用 nWave 程式並開啟剛生成的 *.fsdb 檔案。
- 加入想要檢查的 logic signal，觀察每個 clock 的變化。
- 調整顯示的進位模式，觀察數字變化合理性，EX : decimal, octal, binary。
- Happy debugging。

VII. 隨機演算法

本實驗並不要求點名器所使用的隨機演算法需要在統計上為真正的隨機。不過以下提供一些方法，給同學一點方向，也可以結合兩種方法，也許可以更增加點名器的隨機性。

- 參考 minstd_rand

http://www.cplusplus.com/reference/random/minstd_rand/

minstd_rand 使用了 Linear congruential generator (LCG)，是 LCG 的最簡單

例子，方法如下：

$$X_{i+1} = (aX_i + c) \bmod m$$

a : constant multiplier

b : increment

m : modulus

當 $c = 0$ 時，若最小的 k 滿足以下式子為 $m - 1$ 時，即有最長可能週期 $m-1$ ：

$$a^k - 1 \bmod m = 0$$

本實驗採用 $m = 2147483647$ ，故最長可能週期為 2147483646，雖非完全隨機，但應足夠滿足本實驗的隨機性。

2. 根據按下按鈕的時間決定

由於 FPGA 的 clock 高達 50MHz，故使用者按下按鈕的時間對機器而言也算是相當隨機，故也可以根據按下按鈕的時間來生成隨機數。

VIII. 功能應用

除了示範影片中的基礎功能外，同學也可以自行思考是否有其他功能可以加入到這個點名器中，例如更大的數字範圍、可變的數字範圍、指定數字等，或是有其他更有創意的功能都可以新增，例如樂透機，二十一點，一些與隨機數字相關的應用。本實驗希望同學們可以熟悉 FPGA 以及 verilog 的基本操作。