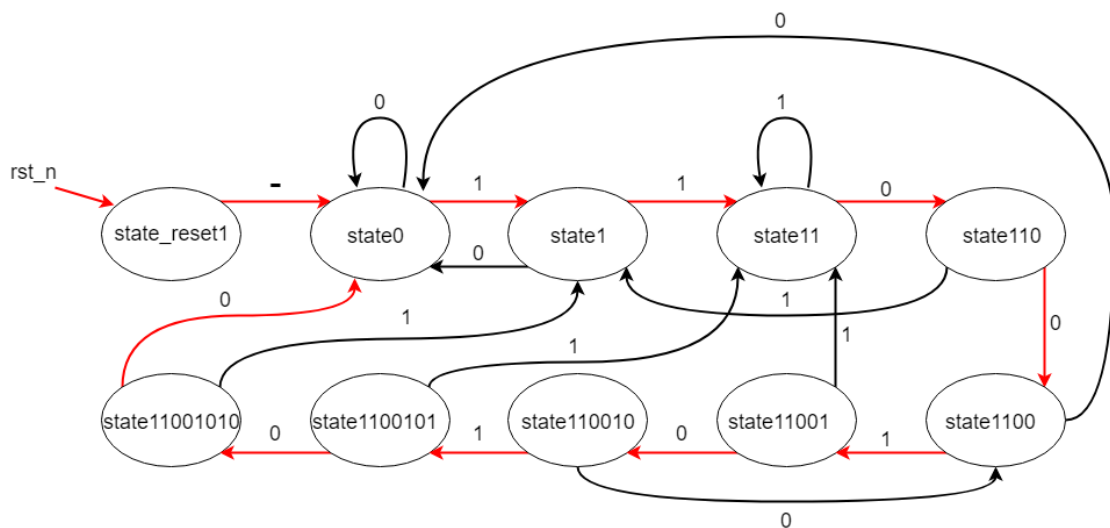


# Logic Design

210510210 詹其侁

# Lab3

## 1.state 的設計&解釋



- **State\_reset1**：一開始的 state，當 rst\_n 等於 0 的時候，會進到裡面。此時傳進來 addr=1023 的 data，這裡會將 data 設成 next\_length，並且將 next\_length 經由 DFF 存在 length。Next\_addr1 會被重置成 0，並且無條件進入 state0，state\_reset1 的作用就是讀取 length 並且給他一個 addr。這裡的 flag 跟 fin 都無條件是 0，因為他只是作為一個初始化的 state。
- **State0**：當從 state\_reset1 跑到這裡後，就會開始

讀取資料，當讀到的資料是 0 就會不停循環，當讀到的資料是 1 就會傳給下一個 state，此時還沒有讀出完整我們想要的信號，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。

- State1：這個 state 紀錄的是"1"，當他讀到 1 時，會傳給 state11，讀到 0 時，因為我們要的數列是  $110(010)^+10$ ，所以會重新回到 state0，此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。
- State11：這個 state 紀錄的是"11"，讀到 1 時，就會在裡面循環，讀到 0 時會傳給下一個 state，state110，此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。

- **State110**：這個 state 紀錄的是"110"，讀到 0 時，會進到下個 state，state1100，讀到 1 時，1101 會變成 1，所以要回到 state1。此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。
- **State1100**：這個 state 紀錄的是"1100"，讀到 1 時，會進到下個 state，state11001，讀到 0 時，會跳回去 state0 重新來過。此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。
- **State11001**：這個 state 紀錄的是"11001"，讀到 0 時，會進到下個 state，state110010，讀到 1 時，110011 會變成 11 所以會跳回 state11。此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。

- State110010：這個 state 紀錄的是"110010"，讀到 1 時，會進到下個 state，state1100101，讀到 0 時，1100100 會變成 0 所以會跳回 state0。此時信號依然不完整，所以 flag 是 0，假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。
- State1100101：這個 state 紀錄的是"1100101"，讀到 0 時，會進到下個 state，state11001010，同時由於信號已經完整了，所以就會將 flag 拉起變成 1。讀到 1 時，11001011 會變成 11 所以會跳回 state11。假如  $addr > length$  那 fin 就要拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度， $next\_length = length$ 。
- State11001010：這個 state 紀錄"110(010)+10"，進到這個 state 就代表已經找到我們要的信號，並且 flag 已經舉過手了，所以 flag 會是 0，同時會再讀新的 data，讀到 0 時，會回到 state0，讀到 1 時，會回到 state1。假如  $addr > length$  那 fin 就要

拉起來變成 1，不然就是 0。Addr 要加 1，同時我們也需要紀錄長度，next\_length = length。

## 2. Ncverilog Simulation Result:

- Before Design Vision synthesis:

```
GET ! addr = 7 , your_flag = 1 , ans_flag = 1
GET ! addr = 18 , your_flag = 1 , ans_flag = 1
GET ! addr = 44 , your_flag = 1 , ans_flag = 1
GET ! addr = 67 , your_flag = 1 , ans_flag = 1
GET ! addr = 147 , your_flag = 1 , ans_flag = 1
GET ! addr = 199 , your_flag = 1 , ans_flag = 1

!!!! ACCEPTED !!!!
```

- After Design Vision synthesis:

```
GET ! addr = 7 , your_flag = 1 , ans_flag = 1
GET ! addr = 18 , your_flag = 1 , ans_flag = 1
GET ! addr = 44 , your_flag = 1 , ans_flag = 1
GET ! addr = 67 , your_flag = 1 , ans_flag = 1
GET ! addr = 147 , your_flag = 1 , ans_flag = 1
GET ! addr = 199 , your_flag = 1 , ans_flag = 1

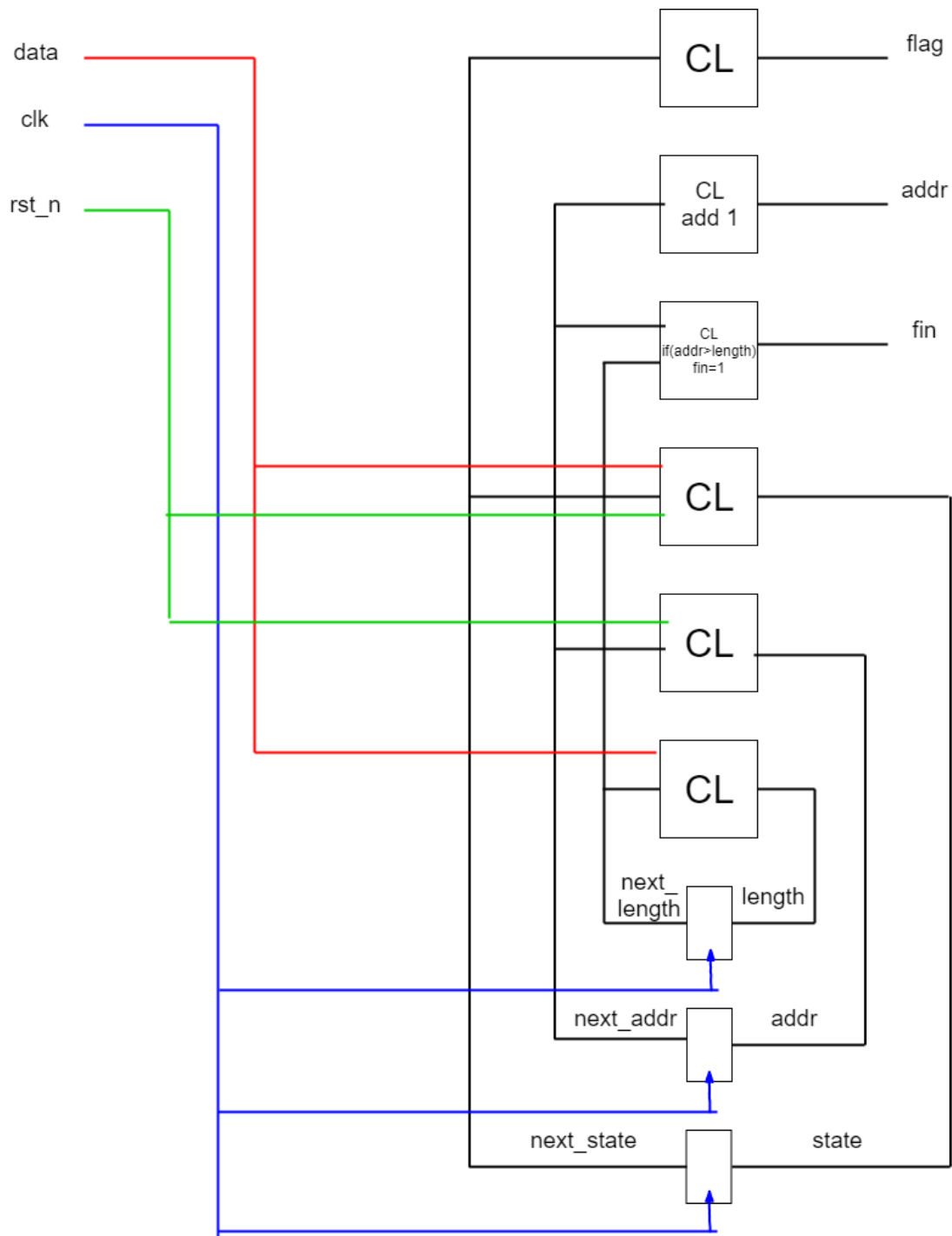
!!!! ACCEPTED !!!!
```

## 3. nWave Result:



- 紅色圈圈是 flag 拉起來的地方，rst\_n 在第二個 cycle 變成 0 其他都是 1，其他部分都沒什麼問題。

#### 4. Block Diagram:



這是一個 moore machine 的 block diagram，因為我的 input 是進到 state 去做判斷，再把結果給 output。這個圖用到三個 DFF 分別去記，length, addr, state，這

三個 output 判斷的方法，flag 是看他在哪個 state，fin 是看 addr 是不是大於 length，addr 則是要加一。

## 5. Problem encountered and discussion:

- 問題：這次我一開始就卡在怎麼從記憶體裡面先找到 1023 這個位置然後拿出 data，一開始我沒想到要用一個 state\_reset1 所以一直想不通，如果在某個 state 設值，那 length 不是就每次都會被重置嗎

解決辦法：後來去問助教，想到可以利用 state 的特性來處理，又加了一個 state。

- 問題：再工作站跑程式的時候，陷入到無窮迴圈。

解決辦法：一開始我跟助教都以為是 fin 沒拉好的問題，中間也改了很多次，有時候沒有無窮迴圈了，有時候還是一直卡住。改到大家都走了，才發現，是 addr 的問題，所以後來我再 reset 那邊加了一個給 addr 值的 assign 就成功解決了這個問題。

- 問題：我的結果跟答案差了一個 cycle。

解決辦法：本來我都是在最後一個 state 把 flag 拉成 1，可是我發現，他在倒數第二個 state 就應該要拉了，因為他接收到 0，就可以確定已經對了，沒必要再跑到最後一個 state 再拉。搞懂這個概念後，就能理解為什麼會差一個 cycle 了。

- 問題：設定 clock period 的問題，我在跑 make syn 的時候出現了 “too many module instance parameter assignment” 的錯誤。

解決辦法：我發現是在燒電路的時候，燒成了 DFF 的電路，所以會錯。在同學的幫助下，我燒了 PD 的電路就對了。

- 討論：這次的 lab 感覺 code 沒有上次的多，但是要理解的東西沒有比較少，在做的過程中，我一開始根本不知道 DFF 在做什麼，看了教學也不太清楚，後來在不停的討論中，了解他可以拿來記錄這次的結果，拿到下次用。這次要用到 nWave 來 debug，很好的幫助我在一次複習老師上課都



繪畫的圖形，這次真的蠻謝謝助教跟同學的，如果沒有討論可能會做不出來，因為下禮拜還有考試，所以我這次就沒有做 **bonus**，有點可惜。再次感謝助教跟同學的幫助。