

## Lab 7

學號: 108032053

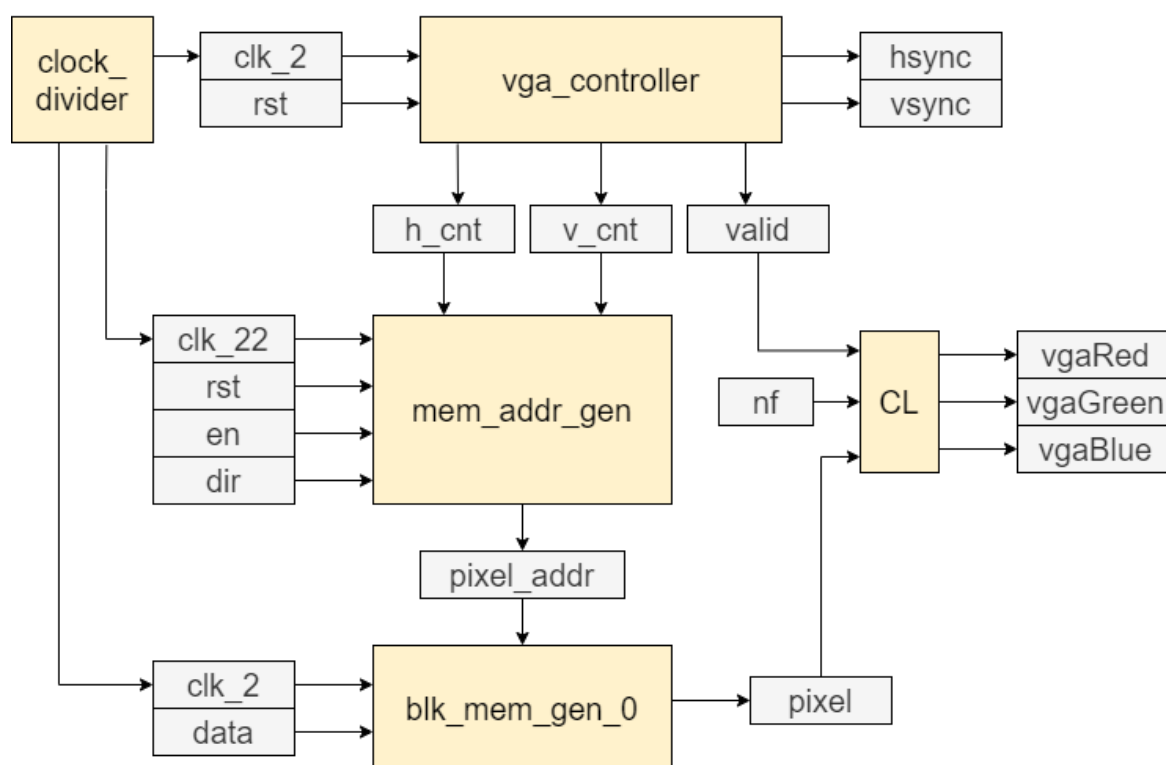
姓名: 陳凱揚

### 1. 實作過程

#### (1) lab7\_1

在此題中，我的 block diagram 如下圖 1 所示，在 mem\_addr\_gen module 中，會以 pos 來記錄上下移動了多少 pixel，並以 dir 來增減 pos 的值，如下圖 2 所示。接著以 pixel\_addr 去 memory 中取出 pixel 的 RGB 值，最後再以 nf 判斷要輸出原始 pixel 還是負片的 pixel。

### Block Diagram



▲ 圖 1

```
// pos
always @(posedge clk, posedge rst) begin
    if(rst) pos <= 8'd0;
    else pos <= next_pos;
end
// next_pos
always @* begin
    next_pos = pos;
    if(en) begin
        if(dir) next_pos = (pos==8'd0)?(8'd239):(pos-8'd1);
        else next_pos = (pos==8'd239)?(8'd0):(pos+8'd1);
    end
end
```

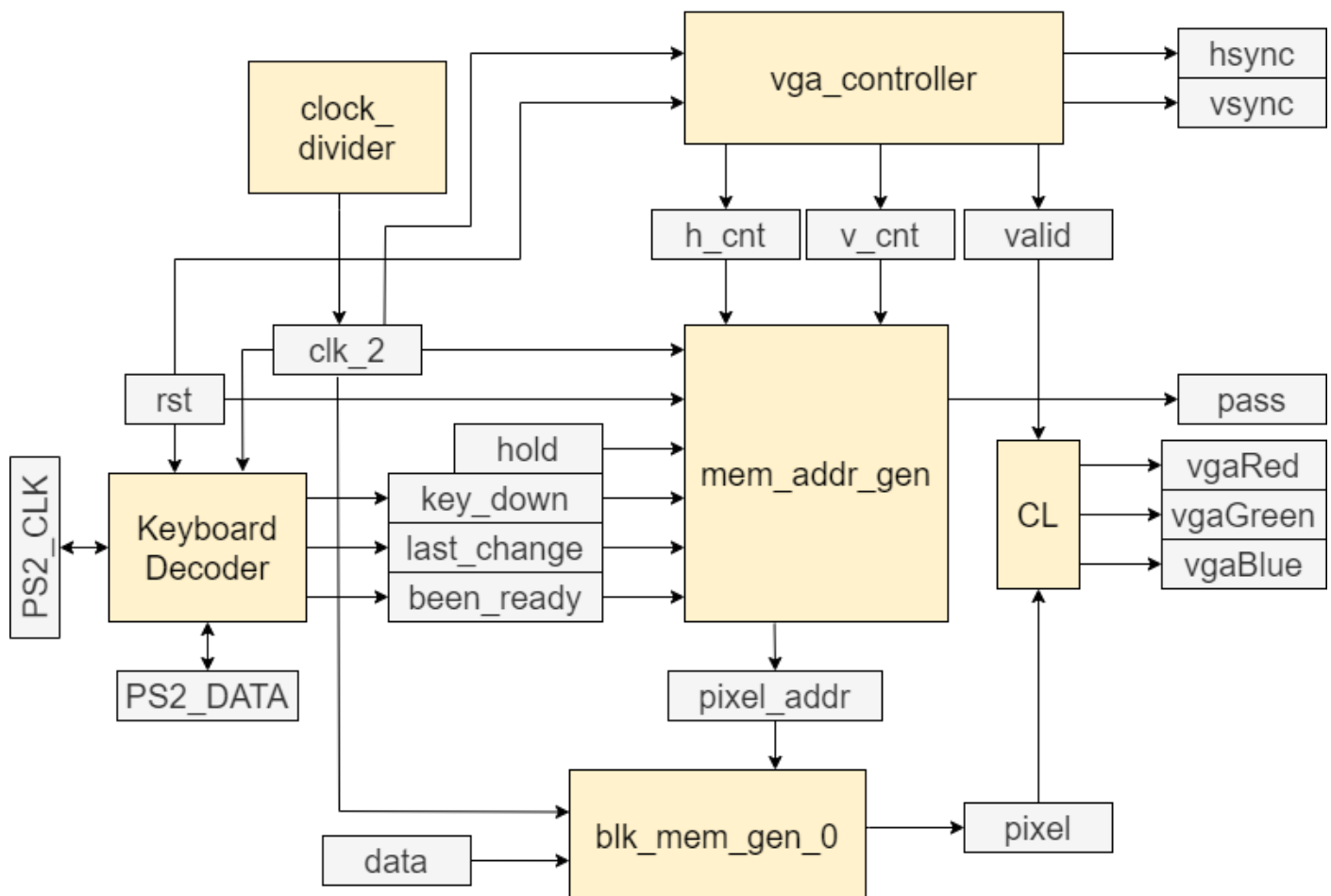
▲ 圖 2

## (2) lab7\_2

在此題中，block diagram 如下圖 3 所示，我會將鍵盤的訊號和 hold 傳進 mem\_addr\_gen module 中來判斷現在應該輸出哪個 pixel\_addr，就可以直接將此 pixel\_addr 傳入 memory，並輸出對應的 pixel，而 mem\_addr\_gen 裡也會判斷和輸出 pass 來代表是否通關，

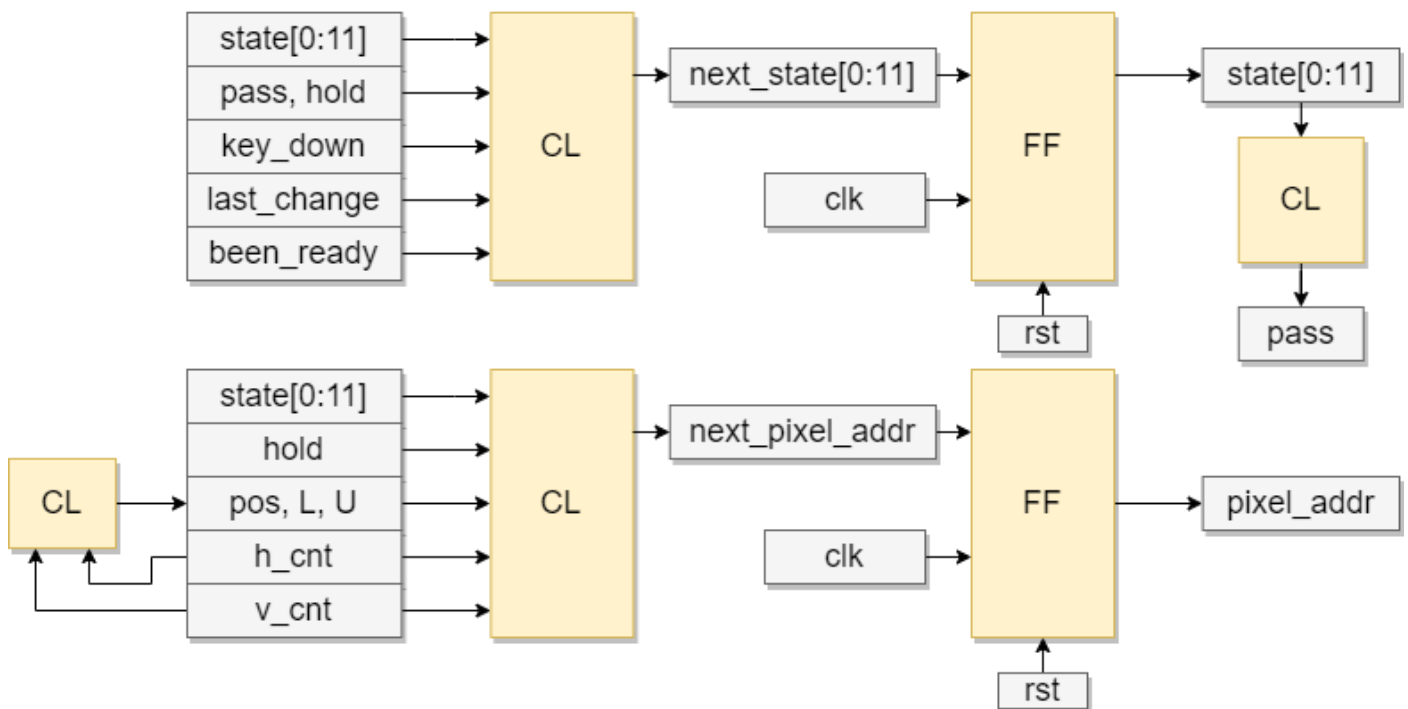
在 men\_addr\_gen module 中，其 block diagram 如下圖 4 所示，我用 12 個 state 分別儲存 12 個區塊目前的旋轉狀態，因此共有 4 種狀態代表了 4 個旋轉的方向，而 state diagram 如下圖 5 所示，會在 pass 和 hold 都等於 0 時，才根據鍵盤的輸入來改變狀態。此外，我使用 pos、L、U 分別儲存目前在哪個區塊、當前區塊的左界、當前區塊的上界，最後就可以根據這些資訊來決定要輸出的 pixel\_addr 為何，下圖 6 即為像素點在不同 state 的公式推導，將黑點對應到紅點的位置上，而圖 7 為轉換之程式碼。

## Block Diagram



▲ 圖 3

## Block Diagram

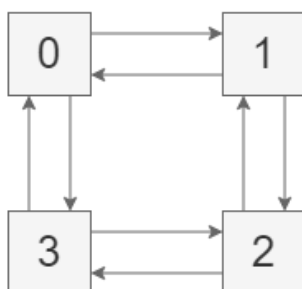


▲ 圖 4

## State Diagram

clockwise(for state[i])

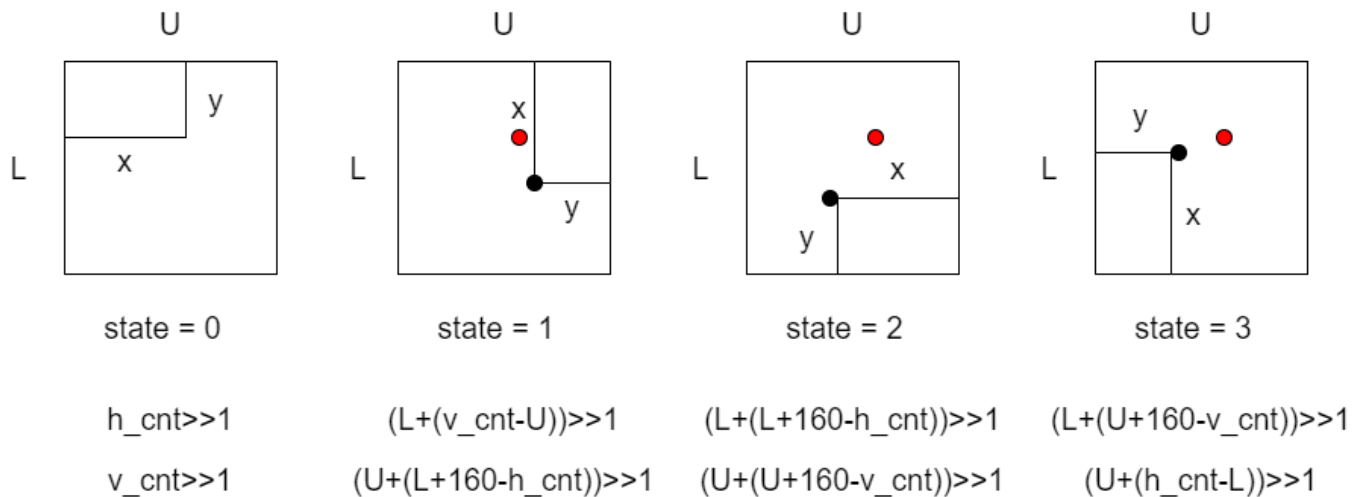
```
!pass && !hold && been_ready && key_down[last_change]
  && (last_change==KEY_CODES[i])
  && !(key_down[LEFT_SHIFT_CODES]
  || key_down[RIGHT_SHIFT_CODES])
```



counterclockwise(for state[i])

```
!pass && !hold && been_ready && key_down[last_change]
  && (last_change==KEY_CODES[i])
  && (key_down[LEFT_SHIFT_CODES]
  || key_down[RIGHT_SHIFT_CODES])
```

▲ 圖 5



▲ 圖 6

```
// next_pixel_addr
always @* begin
    next_pixel_addr = 17'd0;
    if(hold) begin
        next_pixel_addr = (h_cnt>>1)+320*(v_cnt>>1);
    end else if(pos < NULL) begin
        case(state[pos])
            2'd0: next_pixel_addr = (h_cnt>>1)+320*(v_cnt>>1);
            2'd1: next_pixel_addr = ((L+(v_cnt-U))>>1)+320*((U+(L+10'd160-h_cnt))>>1);
            2'd2: next_pixel_addr = ((L+(L+10'd160-h_cnt))>>1)+320*((U+(U+10'd160-v_cnt))>>1);
            2'd3: next_pixel_addr = ((L+(U+10'd160-v_cnt))>>1)+320*((U+(h_cnt-L))>>1);
        endcase
    end
end
```

▲ 圖 7

## 2. 學到的東西與遇到的困難

在這次 lab 中，我覺得最大的困難為旋轉後的對應座標要怎麼轉換，因為每一個區塊都必須獨自判斷，一開始會有點腦袋空白，不知道該如何起頭，後來畫在紙上觀察後才漸漸有了頭緒。感覺這種問題只能靠多實作不同的設計，才能慢慢培養出應對方法。而這次也學到了如何使用 memory 的 IP，對於關於螢幕的設計上相當實用，也開始要盡量節省 memory 的使用，以免超出資源上限。

### 3. 想對老師或助教說的話

有一天畢書盡去了夜店結果被抓走了，請問是誰抓的呢？

...  
...  
...  
...  
...  
...  
...  
...  
...  
...  
...  
...  
...  
...  
...

DJ，因為 DJ 抓了畢 ( DJ drop the beat )。