|  |  |
| --- | --- |
| **Lab 5** | |
| 學號: 109062318 | 姓名: 簡弘哲 |

1. 實作過程

一張含有 文字, 白板 的圖片

自動產生的描述

在這次lab中的FSM總共有6個state，分別是IDLE,TYPE,AMOUNT,PAYMENT,RELEASE,CHANGE

，我覺得這些state就已足夠。至於block diagram由於變數有點多，所以我就用斜線上面標記數字7的方法來表示有7個input。

IDLE:

一張含有 文字 的圖片

自動產生的描述

上面的if(led\_clk\_1p)負責led與7segment的閃爍，其中led\_clk\_1p是每秒產生”將1維持一個clock cycle”的訊號。DARK則是DISPLAY的設定(DISPLAY=7'b111\_1111)，從右到左是GFEDCBA，讓它都是暗的。

下面的if負責state的轉換，並且在轉換之前先把7segment (my這個變數負責顯示7 segment) 該顯示的數字先設定好，以確保在進入下一個state的時候數字不會亂掉(其他state的轉換也是一樣的精神，先把7segment設定好再跳過去那個state，在report底下說明其他state的時候就不贅述)。其中ADULT,STUDENT,CHILD都是設定7segment的字母A,S,C，為了方便理解與debug才將它parameterize。

TYPE:

一張含有 文字 的圖片

自動產生的描述

在TYPE state就根據使用者按下哪個按鈕，7segment就顯示相對應的字母與價錢就ok了。

這裡的type\_next要存起來是因為之後RELEASE會用到這個資訊，然後led要關掉。

AMOUNT:

一張含有 文字 的圖片

自動產生的描述

在決定購票數量的時候，我有加上數量判斷(my[3]負責顯示amount)以防止超過3或小於1的情況發生，這裡變數amount要記住買了幾張，因為在顯示應付價格時需要知道這個資訊，led要全暗。

一張含有 文字 的圖片

自動產生的描述

值得一提的地方是在從AMOUNT轉PAYMENT的時候，因為沒辦法用/,%，所以十位數跟個位數要分開處理。只好先算價錢總共有幾種可能，算出來是6種(5,10,15,20,30,45)，就把相同價錢的組合一起寫在一個if裡面，像是child\*2跟student\*1的價錢一樣，那就把它們寫在同一個if裡。另外led要全暗。

PAYMENT:

一張含有 文字, 銀色, 匾額 的圖片

自動產生的描述

在付錢的時候須注意進位的情形，+1$要檢查個位數(my[1])是不是9，+5$的時候則要檢查個位數>=5，+10$什麼都不用檢查(因為價錢最多45$，不會再更高了)，直接十位數+1即可，led保持全暗。

RELEASE:

一張含有 文字 的圖片

自動產生的描述

這個state需要flashing( ~LED )以及數5秒再進到CHANGE，所以led\_clk\_1p所產生的訊號負責反轉LED燈、變數cycle負責數秒，每秒+1。其中onesec\_clk是一個每秒產生”持續一個clock cycle的1”之訊號。當cycle數超過5秒時，就可以進入CHANGE state了，反之就維持在RELEASE。

CHANGE:

一張含有 文字 的圖片

自動產生的描述

紅色框框是找完零錢的時候要準備進入IDLE的前置工作，將led,7segment都亮起來，其他變數都重置(圖中的amount\_next應改為1，因amount初始值是1)。

藍色框框則是找零的logic，先判斷要return的錢有沒有>=5，如果有就可以先return 5元再return 1元，只是這邊也需要注意退位的情況，找5塊的時候要注意個位數<5 ?如果是就要退位，退位的時候也要檢查十位數是否>0 ?如果十位數是0就繼續維持0，否則就-1。

return 1元的時候就不用檢查退位了，因為要return 1$只有4種情況(01,02,03,04)，所以可以直接個位數-1。

1. 學到的東西與遇到的困難

這次很幸運地幾乎沒什麼遇到困難，一天內就搞定了，只有在設計的時候花了一些時間想該怎麼處理票的數量、種類(child,student,adult)以及找零，其中算出找零該找多少是我花最多時間去思考的。

一張含有 文字 的圖片

自動產生的描述

(Index 0是7-segment最左邊的數字、3則是最右邊。reg [3:0] my[0:3]是用來顯示在7-segment上長度為4的陣列。remain則是記錄要找多少錢)

第一個if是投入的零錢比應付價格多的情況，如果是這樣就先設定好7-segment要顯示什麼數字然後再進到release state。else if則是使用者按下cancel的情形，那就要將最右邊的兩位數字設成使用者所投入的零錢量，因為該找零的錢就是使用者所投入的金額，最後再進入到change state。

1. 想對老師或助教說的話

為什麼傳入debounce跟onepulse的clk要跟用來操作fsm的clk一樣?

如果不一樣的話可能會有甚麼問題?

Ex. debounce跟onepulse module傳入100MHz/2^13，但fsm運作在100MHz底下。