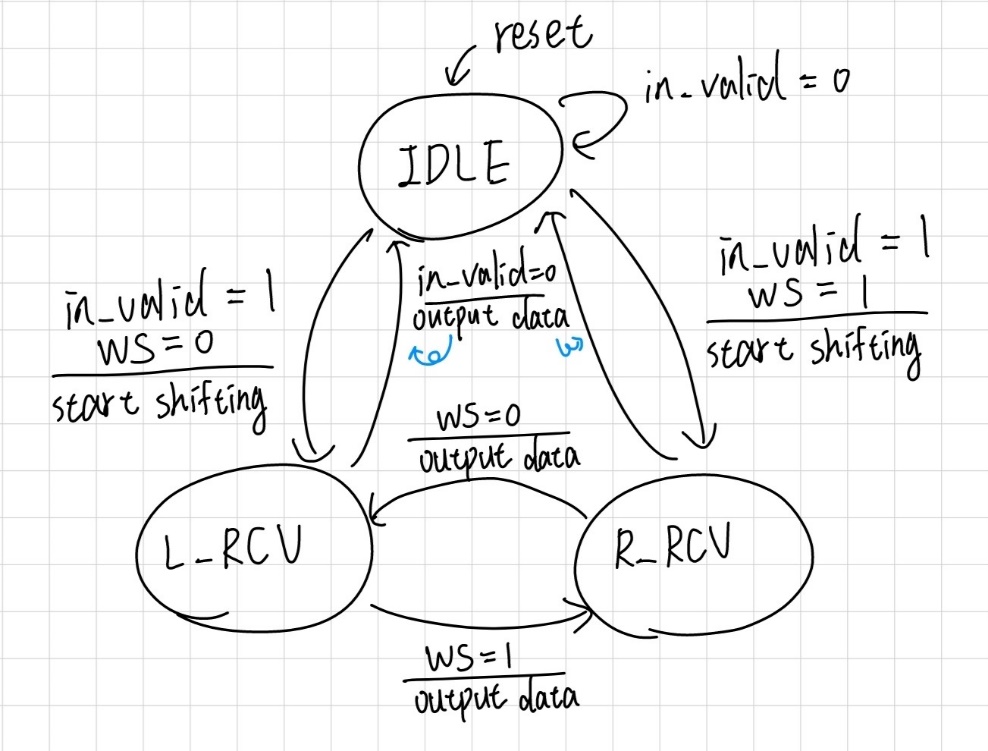
數位電路與系統 HW2 Report 110511233 李承宗

作業二是Sequential Circuit，因此免不了設計FSM，加上輸入輸出是serial-in-parallel-out的關係，所以可以先估計電路會有一組接收資料的shift register。左右聲道的部分我打算用demultiplexer來選擇輸出channel的方式來實現，這樣就可以讓輸出共用同一組register，減少資源用量(但可能會造成glitch，希望這樣設計出來的電路不會對使用者造成影響)。

有了基本的構思之後，接下來就是設計的部分：從FSM開始，由於這次作業的電路牽涉到收發資料，因此會有閒置、接收、結束的狀態。最後我設計的FSM有三個狀態，分別如下：



主要拿來控制是否接收送來的資料，以及要輸出的聲道，因此我設計出來的狀態分別有：不需要接收資料的狀態、輸出至左聲道的以及輸出至右聲道的，我發現接收結束之後好像不需要其他特殊處理，因此我把拔掉了。如此設計可以讓狀態切換的時候觸發輸出資料的事件，讓後續電路設計的難度降低。

在實現的時狀態控制的部分我還有多用一組2-bit的register，用來記錄現在暫存器的資料應該輸出到哪個聲道（或不輸出），作為輸出電路的選擇線、的輸出。

一張含有 圖表, 圖解 的圖片

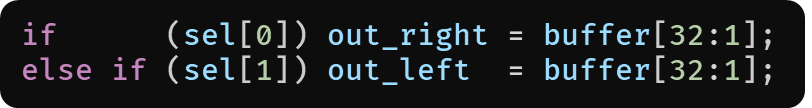
自動產生的描述

架構圖如上，有前面設計好的、輸出選擇線、輸出的邏輯電路以及一組的。其中會輸出、或，代表著沒有輸出、左聲道輸出以及右聲道輸出，因此只要將的兩條線做就可以得到的訊號，以及的訊號則是用的概念來做輸出，但是輸出的時候選擇線只會有一條是1，因此只需要兩組32bits的來選擇輸出，這樣一來就不用請出了。比較難設計的部分是，由於在左右聲道切換的當下需要輸出資料，還得將的資料讀進來變成下一次的輸出，因此的操作模式不是單純的的移位跟：

一張含有 文字 的圖片

自動產生的描述

藉由這段程式碼可以看出會執行向左移位的動作，然後會有清除前32bits（為）或是單純向左移位的動作，選擇方式為檢查當下是否有輸出（此頁右下為範例波形）：當下狀態有輸出的話，裡面的前32bits就已經使用過了，因此下一個state就要把裡面的東西清空。但是輸出當下以及觸發輸出的當下（即為輸出前一個state）的資料還沒使用過，需要存起來。因此我處理的方式是留下內的最後一個bit以及，這也是為什麼我的會有33bits而不是32bits。另外shift-in的資料是而不是，是為了避免為0時還有資料輸入而做的處理。輸出取內的前32bits，再用的值決定輸出頻道，至此整個電路設計完成。



一張含有 資料表 的圖片

自動產生的描述一張含有 文字, 計分板 的圖片

自動產生的描述