

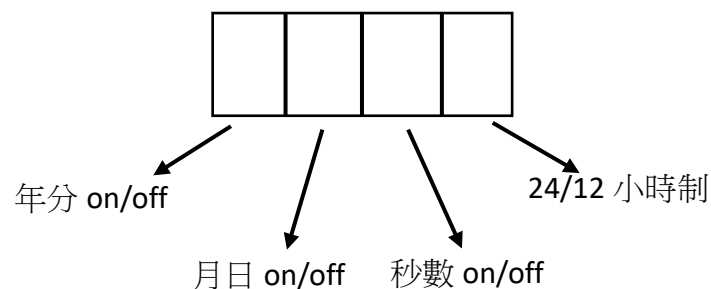
## Experiments

1. 一個時鐘最基本的需要時、分、秒，clk 一個週期為一秒，因此輸出值最少要有三個：時、分、秒，但為了配合 7 段顯示器的輸出，我將每個輸出分成兩位(十位、個位)，但在計算時還是先 reg 各一個時、分、秒，之後再用 assign 取其各位數。用 if、else 表示，當秒數到 60 時會歸零，而分數增加 1，同理在 60 分時時數會增加 1，分數歸零，時數則是在 24 做歸零。再增加兩個輸出值，以 always 和 if、else 將 AM/PM 模式的時數顯示出來，這麼一來總共有 8 個輸出值。

輸入值則是 clk、rst 及一位的 AM/PM 控制器。這時候需要一個大 module(clock\_disp.v)來將除頻器、clock、SSD 轉換器、SSD 控制器合起來，並依照輸入是 24 小時制或是 AM/PM 制來顯示不同的時數，再增加一個輸出 1 或 0 當作顯示 am/pm 的 led 燈，最後整合起來即可。(我有自己再寫一個按鈕來顯示秒數，有用到下一題的 FSM)。

2. 要增加年、月、日，因此必須多出三個輸出值，每個輸出值分兩個位數，接著在時間為 23:59:59 時，日期增加 1，再用 case 將 1、3、5、7、8、10、12 月的天數限制在 31，4、6、9、11 限制在 30，2 月限制在 28 天，超過便將月份加 1，並在 12 月 31 日 23:59:59 時在年分增加 1，就完成年月日的編寫。再以 assign 取其各位數字輸出。

但這時候多了許多控制，需要顯示年、月、日、時分(24 小時)、時分(12 小時)、秒數五種狀態，因此需要一個 FSM 來控制。我的輸入值是 4 個 lever，當都是 0 時是 24 小時制的時分，各個翻起來後會有不同的顯示，而左邊的排序比右邊的高，如下。



在 FSM 時，會輸出一個 0 至 4 的狀態分別是 24 小時、12 小時、秒、日月、年，而在最後的大 module 裡面，會用 case 依照輸出的 mode 來判斷要顯示的數值是甚麼，再用 clock 裡面的數值回傳 ssd。

3. 這題要將閏年寫入，就在 clock 裡面判斷 2 月月份的地方，加入條件：當年分%4 不等於 0 時以及年分%100 等於 0 時，天數最多 28，其餘狀態下為 29，其餘跟第二題一模一樣即可。