



小明和小強的故事

一直以來都在想寫這麼一個故事，這個故事的目的是為了說明清楚關於 timing 的一些概念。所以如果你對時序已經非常清楚的話，大可不必看下去以至於把自己反而搞糊塗了。

有一天天氣很好，所以小明決定去小強家裏。小明和小強家的門都是自動門，不是你想開，就可以開了的，而是在一定的時間後打開，而這時間間隔是固定的。為了讓時間變得簡單一點，我們假設它是一個小時。於是，小明在自己家門 4:00 打開的時候出去了。然後他就向小強家裏走過去，小強家的門是 5:00 打開的，小明在這之前到達小強家門口。這樣小明就順利到小強家裏了。

這個故事就講完了。讓我們重新把它推演一下，這個故事裏面關鍵的是兩點，一個就是小明到達小強家的時間，還有一個就是小強家的門的狀況。這兩個相對來說是兩件獨立事件。我們首先看一個小明從家裏到小強家門口這麼一個過程。首先是開門的時間點(Launch Edge)，開門需要一點時間，你不能在它剛開的時候就出去吧 (Tco)。最後就是從門口走到小強家門口的時間(Tdata)，這個時間是不一定的，因為有很多方式可以到小強家裏。我們會比較關心極端事件，就是最快的時間，和最慢的時間。

這還沒結束呢，最容易忽略的，其實就是時間的問題。好比小明以為他是四點點鐘出門的，但其實他家裏的鐘其實晚了 10(Tclk1)分鐘的。如果我們以一個標準時間來判斷小強和小明家的鐘。而他們家裏的鐘只有可能比標準時間要晚一些 (時鐘傳遞到門的延遲)。差不多可以了，我們看到這麼一個公式

到達時間 = Launch Edge + Tclk1 + Tco + Tdata

如果非要給這些值賦一個固定的值，那麼我來具體看一下。這裏面 Launch Edge, Tco 都是固定的，我們假設他們為 Launch Edge = 4:00, Tco = 5。如果我們說小明花了半小時走過了中間的路，那麼我們可以知道，當小明到小強家裏的時候應該是 4:00 + 5 + 10 + 30 = 4:45。

好了，我們再來研究一下小強家的門。這個比較簡單一點，就是小強家時間不准的時間，假如說為 5(Tclk2)分鐘好了，再加上開門時間點 Latch edge。這其實說明門是晚開了 5 分鐘。所以小強家開門的時間是：

開門時間 = 時間點 + Tclk2 + Latch Edge

這其實就是 5:05 了。

我們一直在說小明要在小強家門開之前到，那麼究竟多久之前到他就可以安全的進去呢？這裏面有一個建立時間 (Tsu) 好比要通過安檢啦什麼的。把這個值設為 3 好了，所以其實小明應該在 5:02 (5:05 - 3) 到是安全的。這點條件小明是符合了的 (4:46)。但是還有

一個條件。小強看到小明會很開心，所以他要和小明聊幾句才讓他進去，我們管這個叫做保持時間，在這段時間裏面小明不能走了，走了的話，也就進不去門了。我們假設這個時間也是三分鐘的話。就是說，從 5:02 到 5:08 之間，小明應該很安心，很穩定的呆在小強家門口他才能進去。小明進步了小強家可能不是什麼大不了的事情，不過如果你的設計時序不合格，就要打屁股了。所以我們來看看什麼情況下，小明進不了門。

首先我們倆安排幾個參數。

Fmax：什麼是最理想的狀況，就是小明走最長的路到達小強家門口的時候，恰好在小強家門口的建立時間點上。而我們把這中間的門開的時間間隔的頻率稱為 **Fmax**。以這個為例：小明 4:46 分到了，如果小強家的門在 4:49 分開就是最好的時間點了。而小強家的鐘的延遲有給了我們 5 分鐘的冗餘。所以最短的時間應該是 44 分鐘。這就是 **Fmax**。這也就是你的電路能夠達到的最快的頻率了。而我們規定的時間間隔大於 44 分鐘就可以保證小明可以在建立時間點前到達。

我們給這個差異一個名字 **setup slack**。就是建立時間點減去小明的到達時間點。結果如果是正的，說明小明可以順利到達，如果是負的，就不行。

剛才我們說過了，還有一個保持時間，我們來看看什麼情況下保持時間會失敗。我們給小明在找一個弟弟，這個弟弟呢會在下一次門開的時候出來，也來小強家。弟弟來的時候，小明就必須要離開了。如果我們把小強家的鐘再搞晚一點，晚了 20 分鐘的話，那麼開門的時間就會晚到 5:25。而保持時間點是 5:28 分。弟弟從 $5:00 + 10 + 1 + 5 = 5:16$ 出發，假設他走了一條近路，很快的就到了，比如只花了 10 分鐘，那麼他在 5:26 分到了，小明就不得不走了。所以我們這裏其實可以得到一個很有意思的結論，當發生保持時間問題的時候，往往是兩種可能。

1. 相互之間的時間誤差差距比較大
2. 中間的路徑非常的短。

所以我們往往需要分析電路的最快表現來分析保持時間。

這裏面也有一個 **hold time slack**。就是小明弟弟到的時間點減去保持時間點。

這個故事差不多講完了，大家可能已經暈了，其實我自己也差不多了。但是我們還要再接再厲。如果把兩個門開關的時間間隔變得不同。做時序分析的時候，工具會找到所有狀況中最糟糕的那個狀況來分析。不用擔心聽上去好像很複雜，這都是工具自己會做的事情。你知需要規定你想要的間隔時間，工具就會爲了你的目標而努力。對這個故事的整個的瞭解讓我們對電路的具體過程有更深入的理解。讓我們來看看我們可以做些什麼來規避 **Slack** 出現負值的情況。

首先看看 **setup slack**:

1. 把時間間隔變大，也就是降低頻率（這是廢話）
2. 把路徑縮短，就是把最長的那條路徑變短，使用各種優化手段。
3. 有的時候，我們會發現，其實小明根本不認識小強，所以我們完全沒必要管他是不是能

進小強家門。(false path) 我們可以告訴工具，不要去管它。

4. 如果在 5:00 到不了，6:00 到也可以，有的時候我們可能並不在乎他什麼時候到，只要能到就好。我們可以設置為 multi-cycle. 我們告訴工具說，沒關係不用太著急的，下次開門的時候再到也可以的。

再看看 hold time slack:

1. 把時間誤差變小。首先我們看看什麼情況下可能導致小明家的鐘差 5 分鐘，而小強家卻差 20 分鐘。如果這個時鐘是一個 gated clock，那麼他的驅動能力是有線的，所以時鐘傳遞的延遲會因此而變大。我們可以使用 PLL 來驅動時鐘，這樣的話驅動能力會加強，時鐘傳遞的延遲因此而變短。
2. 把最短路徑變長，這可以通過添加 LCell 來人為製造延遲。