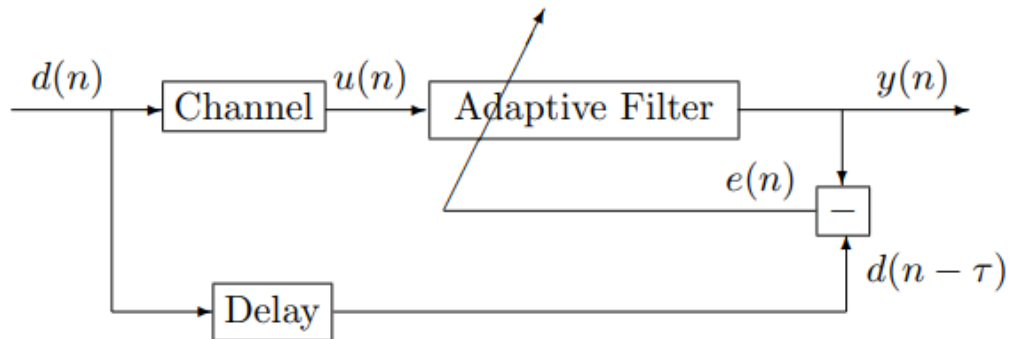


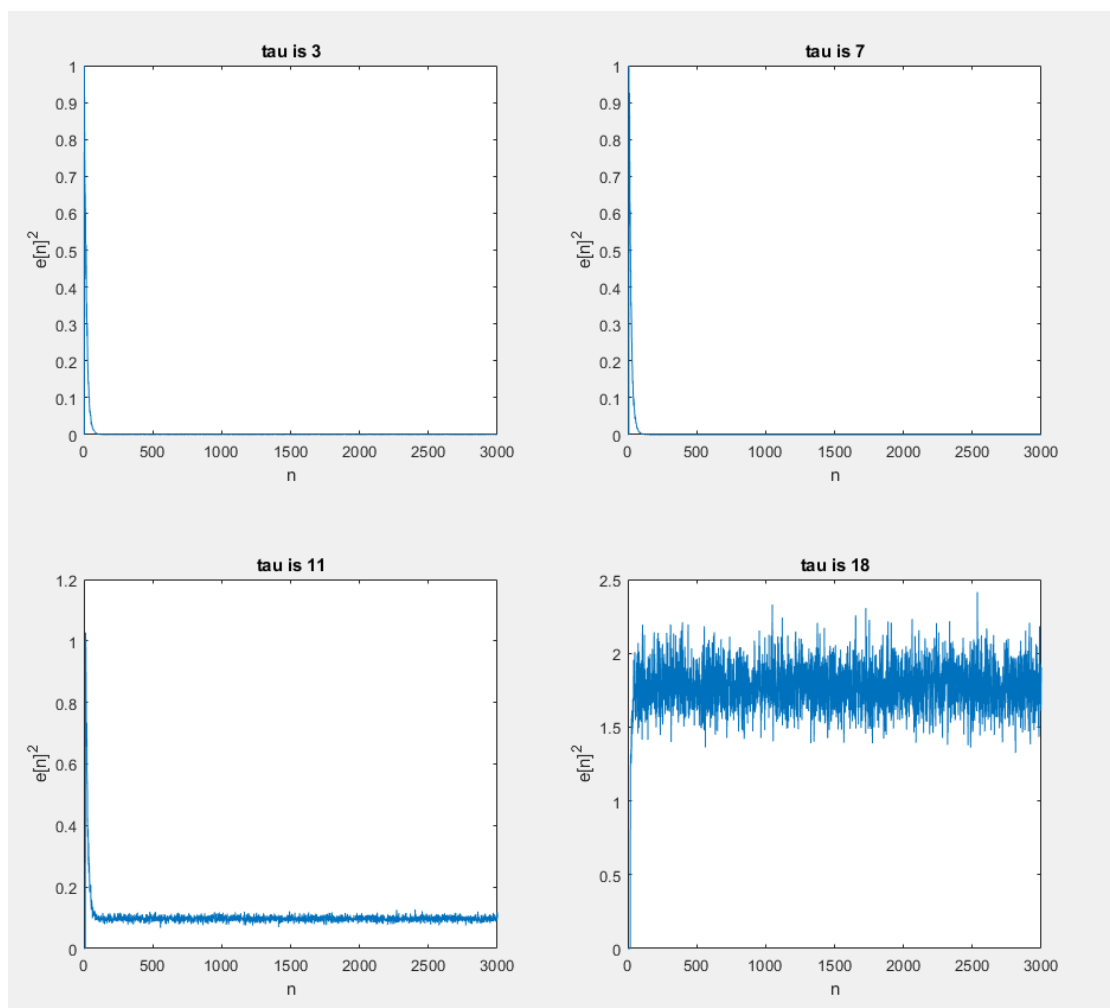
這次 final 我選擇撰寫程式

第一個是 adaptive equalizer 的其中一個應用 inverse modeling，模擬的檔案名為 AF.m，block diagram 如下：



$d(n)$ 為已知的 random sequence, 這邊使用 LMS 演算法去 train 他，這個 equalizer 有 11 個 taps，所以 delay 超過 11 會出現不正常的信號，channel 使用 raised cosine。

為了測試性能，我測試他的 delay 不同造成的差異，可以看到 delay 的不同並不會造成特別影響但 delay(τ) 超過 11 的話跑出雜訊。



第二個程式名叫 `getline.m`

內容是將圖片 (uint8 格式的 matrix)，用某些演算法偵測畫素差異，配合一個參數 (建議值為 10~50)，去將邊線描出來。

圖片需用 `imread` 讀取再用我寫的 function 'getline' 跑。

我們做一個範例，檔案名稱為 `example_of_getline.m`，圖片名稱為 `cat.jpg`，參數我用 25，而比對實做的結果如下



背景的線條會被偵測為雜訊，故沒有描繪出來。