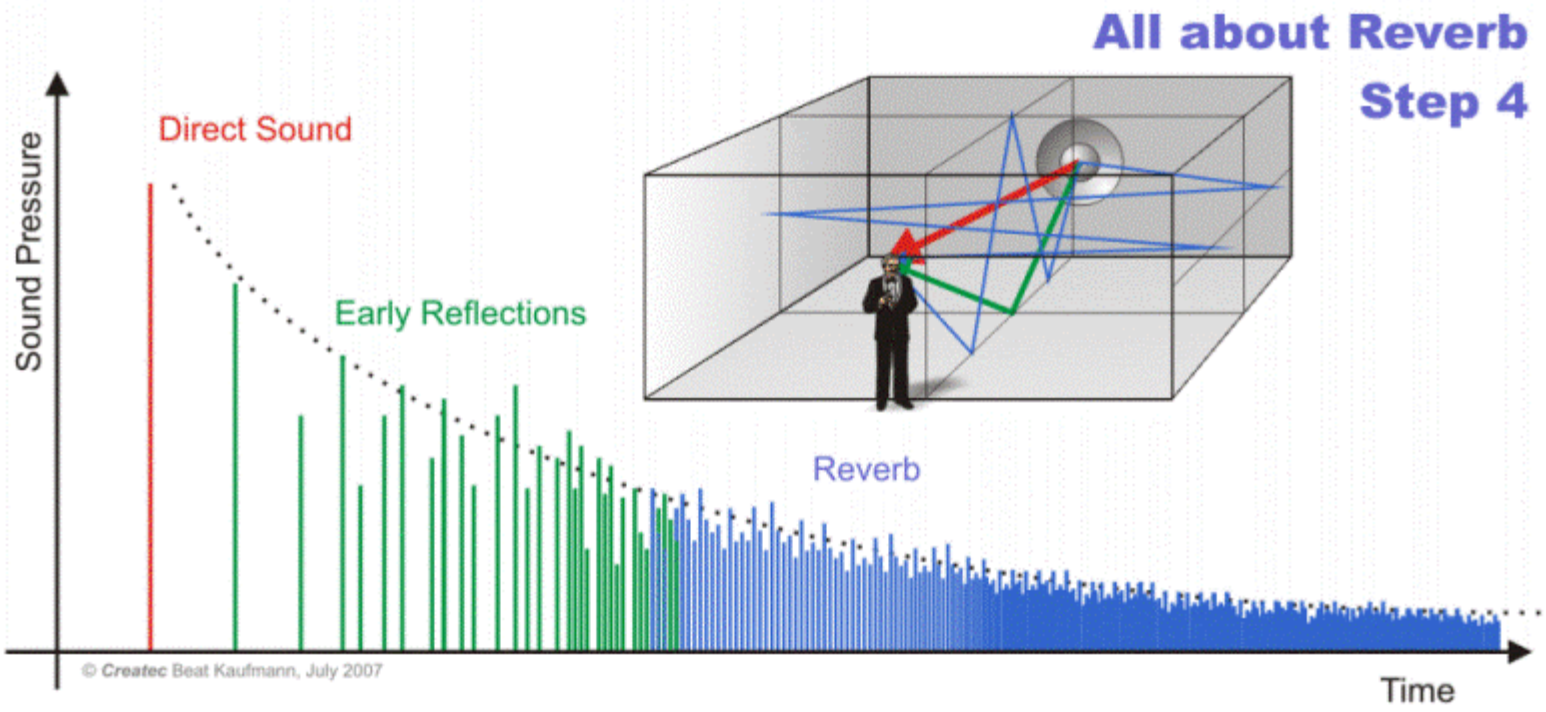
# Reverberation
# 殘響產生器

# Introduction

- Reverberation is the persistence of sound after a sound is produced.
- A reverb, is created when a sound or signal is reflected causing a large number of reflections to build up and then decay as the sound is absorbed by the surfaces of objects in the space – which could include furniture, people, and air.
- When the sound source stops but the reflections continue, decreasing in amplitude, until they reach zero amplitude.

# Direct Sound & Early Reflection & Reverb



# Feedback Delay Network

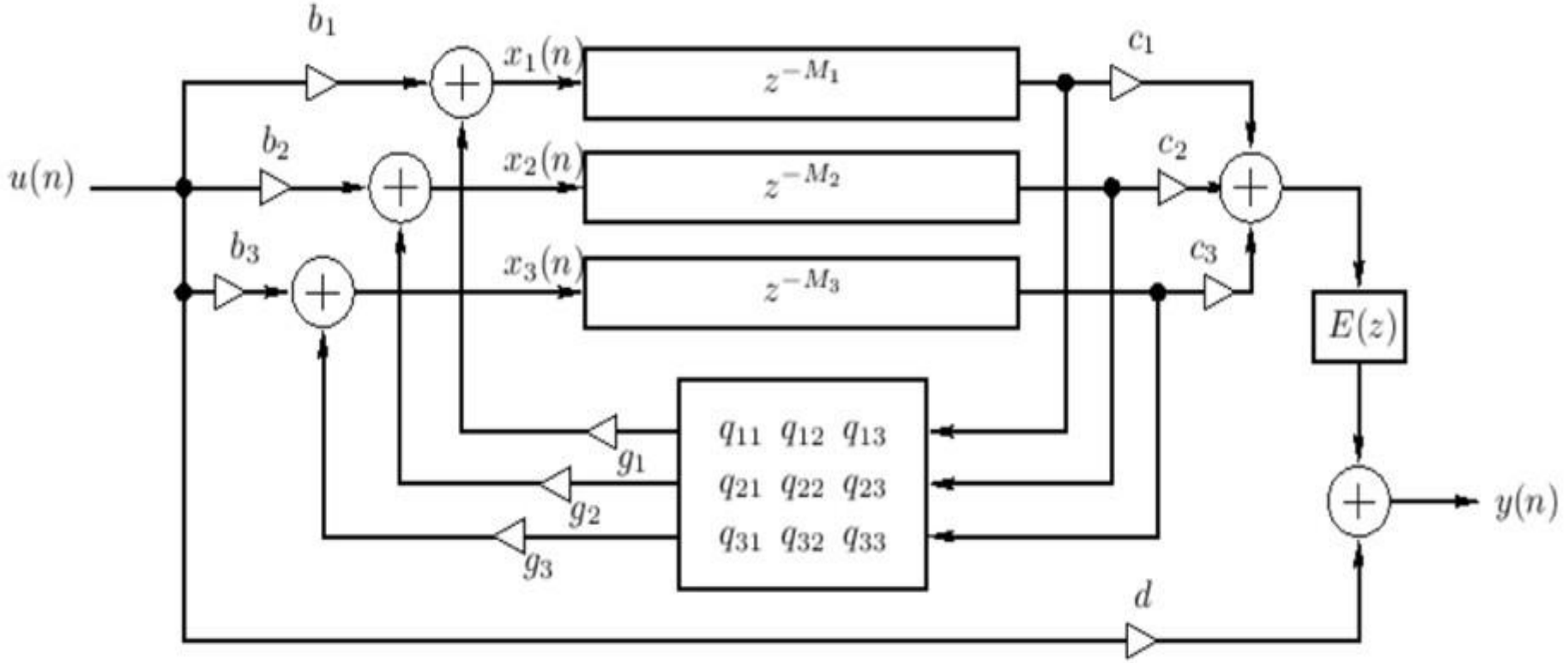



**Figure 3.10:** Feedback Delay Network (FDN) structure proposed for artificial reverberation by Jot.

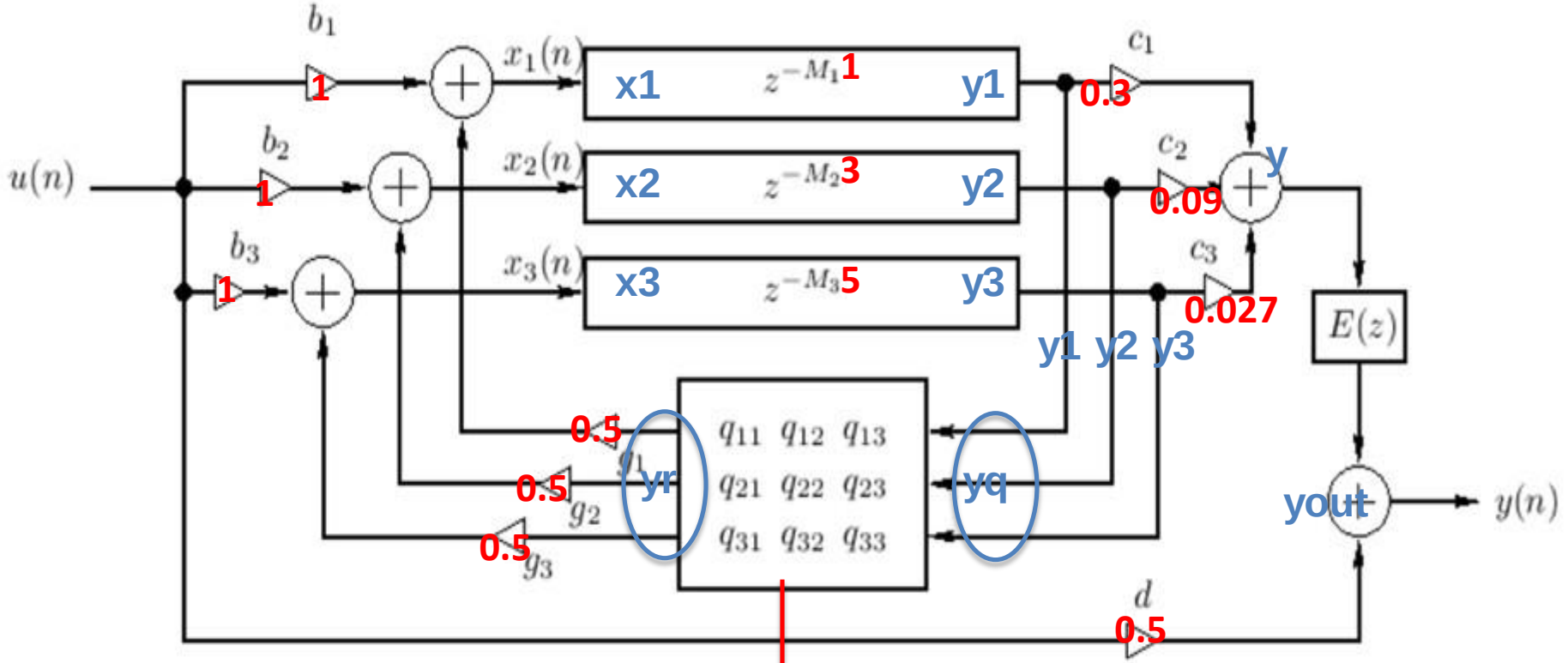



**Figure 3.10:** Feedback Delay Network (FDN) structure proposed for artificial reverberation by Jot.

$$q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

# Pseudo code

1. Fs,samples = readwav(‘2.wav’)
2. length = (len(samples))
3. multiple = 6
4. delay = Fs//10 * multiple  # 0.1s * miltiple
5. direct = zeros(length+delay*M3)  #直接訊號
6. y = zeros(length+delay*M3)  #回授訊號
7. direct[0:lenght] = samples
8. x1 = b1 * direct , x2 = b2 * direct , x3 = b3 * direct
9. y1 = delay(x1,delay*M1), y2 = delay(x2,delay*M2), y3 = delay(x3,delay*M3)

# Pseudo code

10. yq = zeros(length+delay*M3,3)  #matrix
11. yq[0] = y1, yq[1] = y2, yq[2] = y3
12. yr = yq * q
13. x1 = x1 + g1 * yr[0], x2 = x2 + g2 * yr[1], x3 = x3 + g3 * yr[2]
14. y = y + c1 * (x1 + y1) + c2 * (x2 + y2) + c3 * (x3 + y3)
15. yout = y + d * direct
16. writewav('result.wav',Fs,yout)

# 注意事項

- 繳交期限2018/5/23 17:00( 五點以前驗收繳交為 A 。Office Hour結束之前繳交為B。當周日午夜前繳交為C。之後以缺交論F。 )
- 作業請繳交至FTP : 140.116.82.230
  - username：signalsystem107
  - password：screamlab
- 格式
  - 所有程式限定使用Python，Matlab或是C語言
  - 命名規格(壓縮檔標題) :lab10_學號_姓名_vX (X為版本號)
    - Ex:lab9_F71234567_王大明_v1
  - 內容 :lab10.m or lab10.py、result.wav