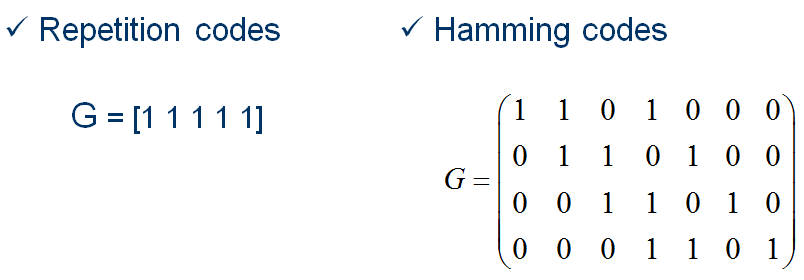
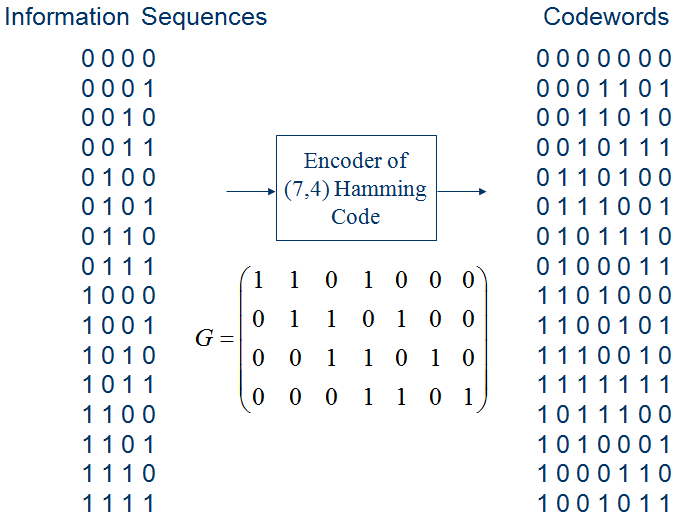
**實習單元三、cyclic code**

1. **原理介紹**

循環碼也是一種線性分組碼，最明顯的特色在於不論左移或右移循環幾位，所得到的結果永遠都是合法的，它們是具有代數性質的糾錯碼，方便進行有效的錯誤檢測和糾正。如果一個（n，k）線性具有下列性質，就成為循環碼（cyclic code）。一個線性區塊碼C在擁有長度n，finite field為F，則被稱做為循環碼，而如果對所有的i都符合 CiF 這個規則。





對於一個codeword　， 定義其編碼多項式標示為：

c(x)=

=()

=



說明:

1. g(X)屬於獨立。

2. 所有的codeword的多項式是g(X)的倍數。

3. g(X)| Xn-1

循環碼的解碼

假設C(X)的codeword，R(X)為接收端。

Error polynomial : E(X) = R(X)–C(X)

Syndrome polynomial : S(X) = R(X)(mod g(X)) = (C(X)+E(X))(mod g(X)) = E(X)(mod g(X))

接收方會接收到 r(x) + e(x) (Error Polynomial，錯誤多項式)，而若 e(x) 不存在，也就是沒有發生任何錯誤，則 r(x) + s(x) 應該能被 g(x) 整除，若沒有整除的話則餘數就稱為 s(x)，可用來診斷與更正錯誤。

1. **程式碼**

這次的作業我使用的軟體是C語言，共分為2個部分：

第一部分

根據一個由 建立出的 (7, 4) 循環碼，假設輸入訊息，則程式會模擬出編碼完的字碼為何，程式碼如下圖2.1所示：

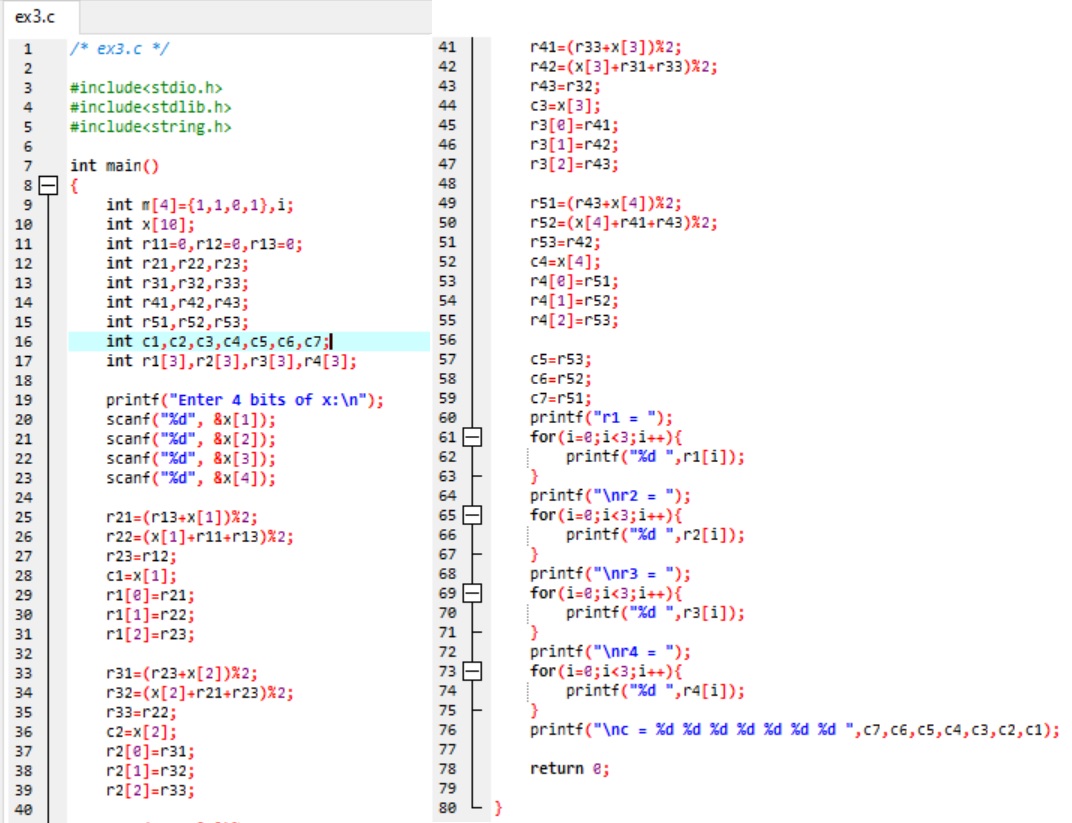


圖2.1

第二部分

根據一個由 建立出的 (7, 4) 循環碼，他的解碼電路圖如圖2.2和圖2.3所示。這個循環碼可以更正任何單一錯誤，以下是他7種single-error patterns以其對應的syndromes。

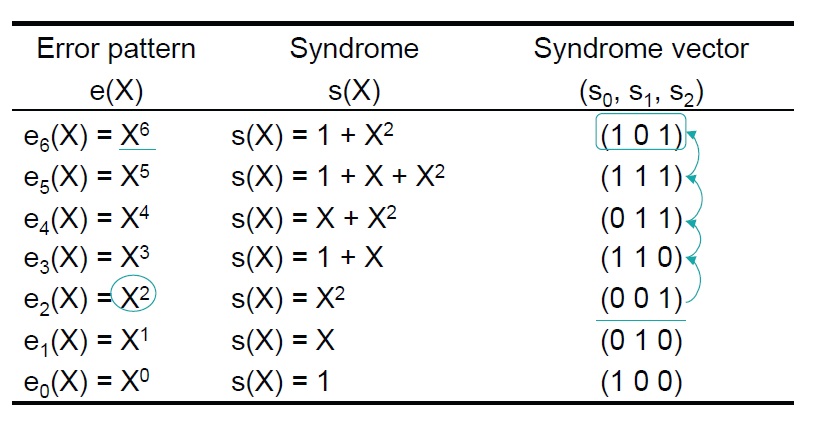


圖2.2

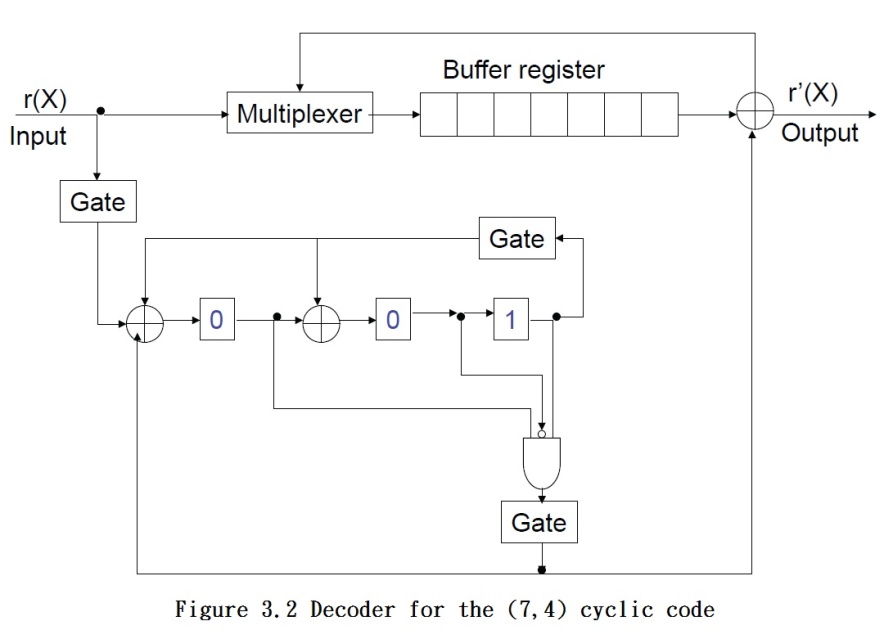
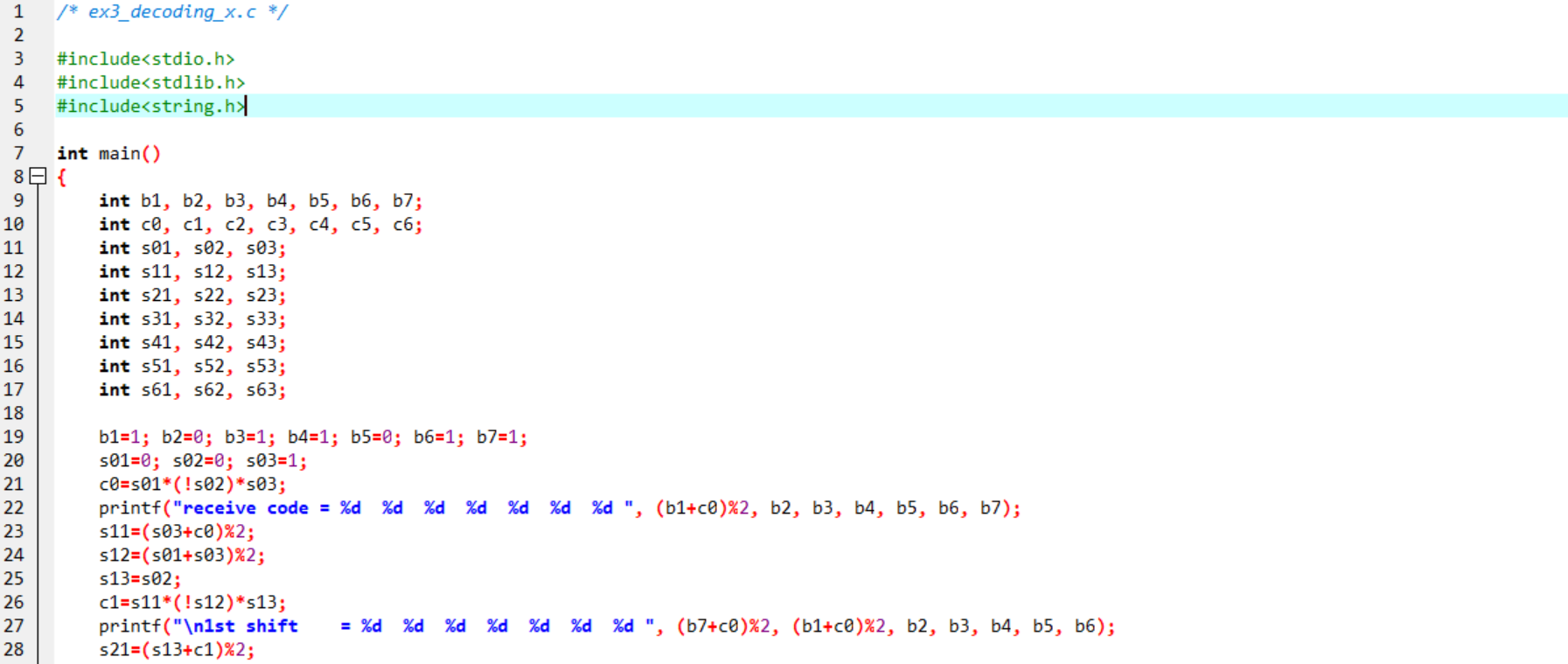


圖2.3

程式碼如下圖2.4所示：



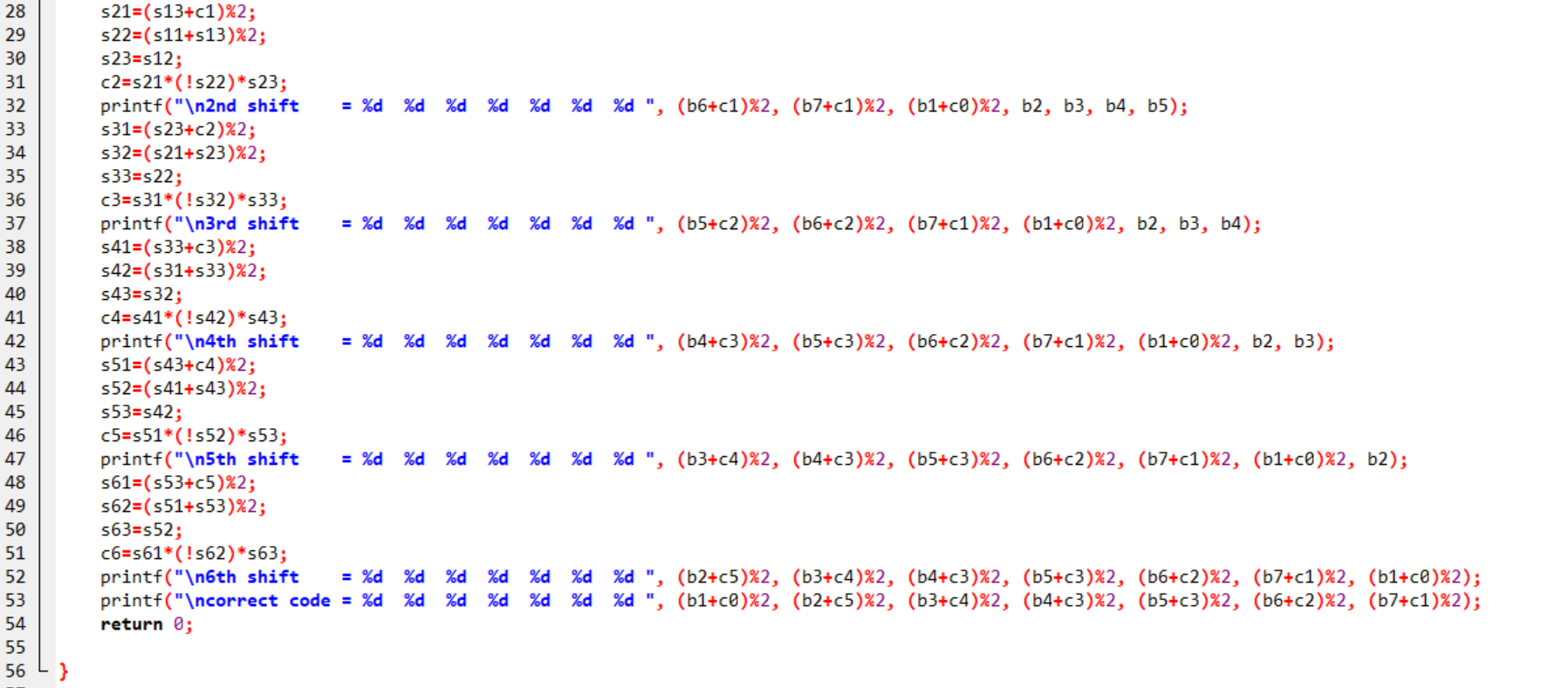


圖2.4

1. **執行結果**

第一部分

當我輸入訊息時，程式會直接執行編碼結果，如圖3.1所示，顯示正確的編碼結果1、0、0、0、1、1、1。

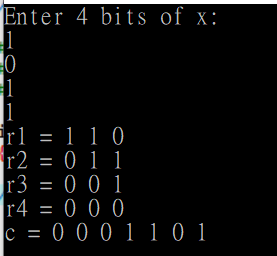


圖3.1

第二部分

假設傳輸的字碼為，收到的字碼為，我利用程式模擬出以下解碼完的字碼，如圖3.2所示。

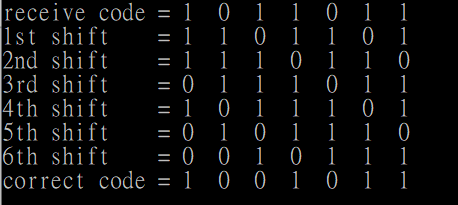


圖3.2