

PRACTICA DE SIMULACIÓN DE HARDWARE DE CODIGOS CICLICOS 2020

Fecha “límite” de presentación hasta el 15 de Diciembre 12hs

Control de error(Actividad Grupal)

1) MEDIANTE UNA HERRAMIENTA DE SIMULACIÓN, (POR EJ SIMULINK) REALICE LA IMPLEMENTACION DE HARDWARE CORRESPONDIENTE AL POLINOMIO (CODIFICADOR Y DECODIFICADOR) QUE EN CADA CASO REQUIERA MENOR CANTIDAD DE COMPONENTES Y EMULE EL FUNCIONAMIENTO DE UN CANAL EN EL QUE SE PUEDA INTRODUCIR UNA CANTIDAD DE ERRORES CORRESPONDIENTES A LOS PUNTOS 4 Y 5 Y VEA EL FUNCIONAMIENTO DEL DECODIFICADOR

2)OBTENGA LA DISTANCIA MINIMA Y CAPACIDAD DE CORRECCION DEL CODIGO.

3) ENCUENTRE LAS MATRICES GENERADORA Y CHEQUEADORA.

4) (MANUALMENTE) GENERE ERRORES EN LAS PALABRAS TRANSMITIDAS DENTRO DE LA CAPACIDAD DEL CODIGO Y VERIFIQUE LOS SINDROMES OBTENIDOS.

5) (MANUALMENTE)GENERE ERRORES EN LAS PALABRAS TRANSMITIDAS FUERA DE LA CAPACIDAD DEL CODIGO Y COMPRUEBE QUE LA DECODIFICACION SE PUEDE INTERPRETAR COMO OTRA PALABRA VALIDA.

Secuencia Pseudoaleatoria

1) DE ACUERDO A LA CANTIDAD DE FF UTILIZADA EN EL CIRCUITO CODIFICADOR-DECODIFICADOR IMPLEMENTE UN GENERADOR PSEUDOALEATORIO CON ALGUNA DE LAS SIGUIENTES COMBINACIONES:

[2,1]

[3,1]

[4.1]

[5,3] [5,4,3,2] [5,4,2,1]

[6,1] [6,5,2,1] [6,5,3,2]

[7,1][7,3,2,1][7,3][7,4,3,2][7,6,4,2][7,6,3,1][7,6,5,2][7,6,5,4,2,1][7,5,4,3,2,1]

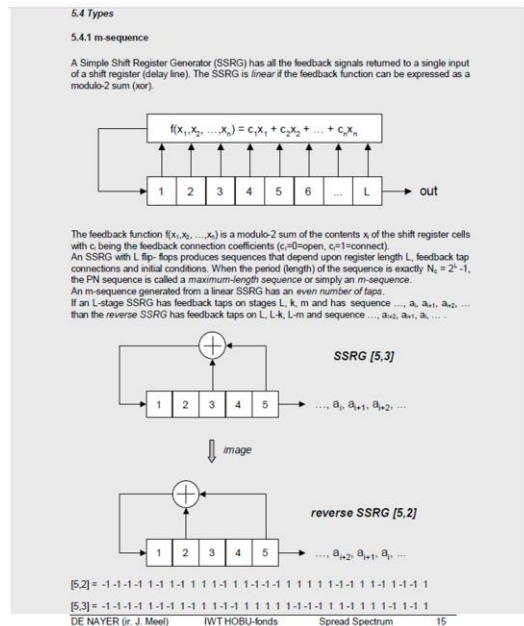
[8,4,3,2] [[8,6,5,3] [8,6,5,2] [8,5,3,1] [8,6,5,1] [8,7,6,1] [8,7,6,5,2,1] [8,6,4,3,2,1]

[9,4] [9,6,4,3] [9,8,5,4] [9,8,4,1] [9,5,3,2] [9,8,6,5] [9,8,7,2] [9,6,5,4,2,1] [9,7,6,4,3,1] [9,8,7,6,5,3]

[10.3] [10.8.3.2] [10.4.3.1] [10.8.5.1] [10.8.5.4] [10.9.4.1] [10.8.4.3] [10.5.3.2] [10.5.2.1] [10.9.4.2] [10.6.5.3.2.1] [10.9.8.6.3.2] [10.9.7.6.4.1] [10.7.6.4.2.1]

$[10,9,8,7,6,5,4,3] \quad [10,8,7,6,5,4,3,1]$

[11,2] [11,8,5,2] [11,7,3,2] [11,5,3,2] [11,10,3,2] [11,6,5,1] [11,5,3,1] [11,9,4,1] [11,8,6,2] [11,9,8,3] [11,10,9,8,3,1]



SE DEBE CUMPLIR QUE EL N° DE FF UTILIZADO PARA EL CODIFICADOR SEA IGUAL AL N° DE FF UTILIZADO EN EL GENERADOR PSEUDOALEATORIO.

2)ENCUENTRE LA SECUENCIA CORRESPONDIENTE ($g_1(x)$ para el grupo 1, $g_2(x)$ para el grupo 2....)

- $K=3$ $g_1(x) : x^{11} + x^9 + x^8 + x^7 + x^4 + x^2 + x + 1$

- $K=3 \quad g_2(x) : X^{11} + X^{10} + X^9 + X^7 + X^4 + X^3 + X^2 + 1$

- $k=4$ $g_3(x) : X^{17} + X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^3 + X + 1$

- $k=4$ $gg_4(x) : X^{17} + X^{16} + X^{14} + X^{10} + X^9 + X^7 + X^3 + X^2 + 1$

- $k=5$ $g_5(x) : X^{10} + X^8 + X^5 + X^4 + X^2 + X + 1$

TELECOMUNICACIONES II: AREA “TEORIA DE SEÑALES INFORMACION Y CODIFICACION” INGENIERIA ELECTRONICA. TRABAJO PRACTICO, CICLO LECTIVO 2020

- $k=5$ $g_6(x) : X^{10} + X^9 + X^8 + X^6 + X^5 + X^2 + 1$
- $k=5$ $g_7(x) : X^{16} + X^{12} + X^{11} + X^8 + X^6 + X^4 + X^3 + X^2 + X + 1$
- $k=5$ $g_8(x) : X^{16} + X^{15} + X^{14} + X^{13} + X^{12} + X^{10} + X^8 + X^5 + X^4 + 1$
- $k=5$ $g_9(x) : X^{23} + X^{20} + X^{19} + X^{18} + X^{17} + X^{15} + X^{14} + X^9 + X^6 + X^5 + X^4 + X^3 + X + 1$
- $k=5$ $g_{10}(x) : X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{17} + X^{14} + X^9 + X^8 + X^6 + X^5 + X^4 + X^3 + 1$
- $k=10$ $g_{11}(x) : X^{11} + X^8 + X^7 + X^2 + 1$
- $k=10$ $g_{12}(x) : X^{11} + X^9 + X^4 + X^3 + 1$
- $k=4$ $g_{13}(x) : X^{11} + X^8 + X^7 + X^5 + X^3 + X^2 + X + 1$
- $k=4$ $g_{14}(x) : X^{11} + X^{10} + X^9 + X^8 + X^6 + X^4 + X^3 + 1$
- $k=6$ $g_{15}(x) : X^{15} + X^{14} + X^{13} + X^{12} + X^{10} + X^9 + X^8 + X^5 + X^4 + X^2 + 1$
- $k=6$ $g_{16}(x) : X^{15} + X^{14} + X^{12} + X^9 + X^8 + X^5 + X^2 + 1$

Los generadores polinómicos más utilizados forman parte de estándares internacionales, y son los siguientes:

CRC-12: $x^{12} + x^{11} + x^3 + x^2 + x + 1$ **CRC-16:** $x^{16} + x^{15} + x^2 + 1$ **CRC-CCITT:** $x^{16} + x^{12} + x^5 + 1$

CRC-32: $x^{32} + x^{26} + x^{23} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ CRC-12 se utiliza en los códigos con longitud de carácter de 6 bits; CRC-16 y CRC-CCITT se utilizan en conexiones WAN, mientras que CRC-32 se utiliza en conexiones LAN.

Para todos los grupos Comprobar las siguientes aseveraciones:

En todos los generadores utilizados aparece $x+1$ como factor, ya que esto asegura la detección de todos los errores con un número impar de bits. Un código polinómico de r bits detectará todos los errores a ráfagas de longitud $\leq r$. Un generador como CRC-16 o CRC-CCITT detectará todos los errores simples y dobles, todos los errores con un número impar de bits, todos los errores a ráfagas de longitud 16 o menor, 99,997% de los errores a ráfagas de 17 bits y 99,998% de los errores a ráfagas de 18 o más bits.

En todos los casos realizar el correspondiente informe que incluya la consideración de cada punto.