**Código Binario**

* **¿Qué es codificar?**

Se llama codificación a la transformación de la formulación de un mensaje a través de las reglas o normas de un código o lenguaje predeterminado. Todas las operaciones informáticas están cifradas en código binario, o bien, combinaciones más o menos complejas de unos y ceros que ocurren constantemente.

El mensaje para que sea posible codificarlo y decodificarlo debe que ser comprendido y entendido por el emisor y el receptor, El emisor envía o expresa un mensaje en la misma lengua que habla el receptor es decir el emisor envía el mensaje y se encarga de codificarlo ya sea desplazando las líneas cambiando letras, escribirla en varios idiomas o uso de algún código mientras que el receptor para decodificarlo o descifrarlo, consiste en analizar el texto y realizar los cambios necesarios para poder lograr su comprensión

* **Códigos binarios numéricos:**

1. Código BCD

Implementación - características

Representa a los dígitos decimales por secuencias de bits en sistema binario. BCD es un acrónimo de Binary Coded Decimal es decir, decimal codificado en binario, cada secuencia de 4 bits es un digito decimal del código, siendo en este la posición más a la izquierda o más significativa cuenta con peso de 8, en cambio el menos significativo se encuentra del lado derecho que cuenta con un peso de 1, las ponderaciones utilizadas en este código decimal binario es de base 2

Una de sus características es la distancia ya es aplicable a las combinaciones binarias siendo esta el número de bits que cambia de un número a otro

Este código se puede decir que es continuo ya que cumple que para todas las posibles combinaciones de código a su siguiente número solo cambia por un bit

-Aplicación

Algunas de sus aplicaciones pueden ser en la electrónica donde se debe mostrar un valor numérico, especialmente en los sistemas digitales no programados a igual que su uso en los sistemas digitales para las lecturas que sean correspondientes que solo pueden ser obtenidas en binario

1. Código reflejado

Implementación – características

Es un sistema de codificación que ordena un conjunto de 2^n números binarios, de tal modo que sólo uno de los bits cambie de una palabra a la siguiente este código también es conocido como código Gray, este código solo cuenta con una cantidad de tres dígitos que están en binario, este código fue implementado cuando surge la necesidad ya que cuando se hacían circuitos lógicos digitales con dispositivos electromagnéticos, estos en sus contadores necesitaban potencias más elevada a las entradas y esto generaba picos de ruido cuando barios bits cambiaban simultáneamente, al aplicar este código se logró garantizar que en cualquier transición lo único que variaba era solamente un bit

Aplicaciones

Este código se sigue aplicando en el circuito electrónico combinaciones mediante el uso de un Mapa de Karnaugh, ya que el principio de diseño de buscar transiciones más simples y rápidas entre estados, a pesar de que los problemas de ruido y cambios de potencia, siendo este uno de los más usados para poder cambiarlo a binario y de binario a digital

1. Código de Exceso-3

Consiste en que cada una de las unidades de a convertir se le toma o suma un 3 además de ser un código decimal en binario auto complementario y un sistema de numeración es una forma de representar valores con un número balanceado de positivos y negativos, los números son representados como dígitos decimales, y cada dígito es representado por cuatro bits como el valor del dígito más 3 de sus unidades

Esta forma de codificación fue muy utilizada en las computadoras antiguas al igual que otros dispositivos para realizar sus cálculos como las calculadoras portables

* **Código binario Alfa-Numérico**

1. Código ASCII

Este tipo de código permite representar las letras minúsculas y mayúsculas, dígitos decimales, signos de puntuación y caracteres especiales, la mayoría de dispositivos lo emplean para poder comunicar al ser humano con los dispositivos electrónicos

Este es un código de 7 bits que permite representar 128 símbolos, Los códigos binarios de las letras mayúsculas y minúsculas difieren en el estado de un bit.

El código ASCII se desarrolló en el ámbito de la telegrafía y se usó por primera vez comercialmente como un código de tele impresión impulsado por los servicios de datos

-Evolución

La Agencia de Estándares Estadounidense, que se convertiría más tarde en el Instituto Nacional Estadounidense de Estándares, publicó por primera vez el código ASCII en 1963, El ASCII publicado en 1963 tenía una flecha apuntando hacia arriba en lugar del circunflejo y una flecha apuntando hacia la izquierda en lugar del guion bajo.

La versión de 1967 añadió las letras minúsculas, cambió los nombres de algunos códigos de control y cambió de lugar los dos códigos de control ACK y ESC de la zona de letras minúsculas a la zona de códigos de control, ASCII fue actualizado en consecuencia y publicado como

ANSI X3.4-1968,

ANSI X3.4-1977]],

ANSI X3.4-1986

Este incluyendo su año de publicación de cada una de las versiones que en ella se obtenían cambio como los códigos de control y de escritura a lo largo de su evolución

El ASCII reserva los primeros 32 códigos para caracteres de control, códigos previstos originalmente para no llevar la información imprimible, sino para controlar algo los dispositivos como cancelar acciones o realizarlas, El carácter del “espacio” se considera un gráfico invisible más bien que un carácter de control

* **Códigos binarios de error**

Si en un código binario de longitud constante, se utilizaran todas las combinaciones posibles de sus “n” dígitos binarios (2^n), resultaría imposible detectar si se ha producido error, ya que una combinación del código se transformaría en otra que también pertenece a él. Quiere esto decir, que la posibilidad de detectar errores se logra no utilizando todas la combinaciones posibles, de forma que al recibir una determinada combinación se puede identificar como errónea si no pertenece al código

Al querer transmitir dígitos decimales para lo que se emplea el código BCD natural. En éste no se emplean más que 10 de las 16 combinaciones posibles con 4 dígitos binarios. Supongamos que se emite la combinación 0011 y tras producirse error en un dígito binario durante la transmisión se recibe la 1011; como este no pertenece al código utilizado, es inmediatamente detectado que este no pertenece al mensaje, por lo que el error será detectado.

1. **Código de Paridad**

-Características

Se dice que una combinación binaria tiene paridad par si el número de unos de esa combinación es par. De igual forma se dice que una combinación tiene paridad impar si su número de unos es impar, Un código de paridad se obtiene añadiendo a las palabras de un código de distancia mínima uno, un dígito que se denomina de paridad.

Si el código que se desea obtener es de paridad par, este dígito debe adquirir un valor tal que la paridad de cada combinación sea par. Igual criterio se aplica si el código deseado es de paridad impar, para verificar que los códigos de paridad son detectores de error en un dígito binario basta con probar que su distancia mínima es dos

* Un código de paridad par o impar posee una distancia mínima estrictamente mayor que uno
* En un código de paridad par o impar existen siempre dos combinaciones que distan dos unidades

-Implementación

Cuando la probabilidad de error no es muy alta, ha probado ser efectivo un sistema denominado Chequeo de paridad. Esta técnica emplea un bit extra, llamado de paridad, cuyo valor se selecciona de tal forma de dejar un número par de unos en la palabra de código, entonces se dice que el código es de paridad par. Si el bit se escoge tal que el número de unos en la palabra sea impar, se tiene un sistema de paridad impar, En aplicaciones digitales de movimiento de información, es importante detectar cualquier tipo de error en la información recibida

-Ejemplo de Aplicación

* Analizador y manejador de mensajes cortos para poder verificar en la decodificación respectiva si en la transición del mensaje no se generó algún error que pueda afectar a la mala recepción del mensaje
* Para detectar si existe o no error en la palabra de código recibida, se comprueba si ésta cumple el criterio de paridad preestablecido, si es así se supondrá que no ha existido error en la transmisión, si no es así es que algún dígito ha variado de valor, no se puede saber cuál es, pero sí que ha existido error en el.

1. **Código de CRC**

-Implementación

En éstos, las palabras consecutivas tienen la misma distancia, y esta distancia existe también entre la primera y la última palabra del código. En este sentido, la lista puede considerarse cerrada, es un método de control de integridad de datos de fácil implementación. Es el principal método de detección de errores utilizado en las telecomunicaciones.

-Características

La verificación de redundancia cíclica consiste en la protección de los datos en bloques, denominados tramas. A cada trama se le asigna un segmento de datos denominado código de control

El código CRC contiene datos redundantes con la trama, de manera a diferencia de la detección de paridad, los errores no sólo se pueden detectar sino que además se pueden solucionar. Consiste en tratar a las secuencias binarias como polinomios binarios, denotando polinomios cuyos coeficientes se correspondan con la secuencia binaria localizando y corrigiendo el error

-Aplicaciones

Se dice que M es el mensaje que corresponde a los bits de la trama que se enviará, y que M(X) es el polinomio relacionado con él. Supongamos que M'(x) es el mensaje transmitido, el mensaje inicialmente al que se concatena un CRC de n bits. El CRC es el siguiente: M'(X)/G(X)=0. Por lo tanto, el código CRC es igual al remanente de la división polinomial de M(X) (X) entre G(X), por lo tanto, si el destinatario del mensaje divide M'(x) por G(x), obtendrá un remanente de cero si la transmisión ocurrió sin errores, dando como resultado la satisfactoria recepción del mensaje al igual que su análisis

-Ejemplo de Aplicación

-En este proceso de detección de errores, un polinomio predeterminado, es conocido tanto por el remitente como por el destinatario. El remitente, para comenzar el mecanismo de detección de errores, ejecuta un algoritmo en los bits de la trama, de forma que se genere un CRC, y luego transmite estos dos elementos al destinatario. El destinatario realiza el mismo cálculo a fin de verificar la validez del CRC.

-Los polinomios más generados usualmente son:

-CRC-12: X12 + X11 + X3 + X2 + X + 1

-CRC-16: X16 + X15 + X2 + 1

-CRC CCITT V41: X16 + X15 + X5 + 1 (utilizado en procedimientos de HDLC)