



**I
N
A
O
E**

**Instituto Nacional de Astrofísica,
Óptica y Electrónica.**

**Decodificación para Detección
Coherente de Espectros
Dispersos**

Sirgei Garcia Ballinas

Tesis sometida por requisito parcial para obtener
el grado de maestría en ciencias en astronomía
en el Instituto Nacional de Astrofísica, Óptica y
Electrónica.

Supervisada por:

Dr. Rogerio A. Enríquez Caldera

©INAOE 2007

Derechos Reservados

El autor otorga al INAOE el permiso de
reproducir y distribuir copias de esta tesis en su
totalidad o en partes.



Agradecimientos

Le agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca concedida número 182272.

A mi asesor por su paciencia y dedicación.

A mis sinodales.

Dedicatoria

A mis padres y a mi esposa.

CONTENIDO

LISTA DE FIGURAS.....	VII
LISTA DE TABLAS.....	IX
CAPÍTULO 1 INTRODUCCIÓN	1
1.1 MOTIVACIÓN	1
1.2 JUSTIFICACIÓN.....	2
1.3 OBJETIVO.....	2
1.4 METODOLOGÍA	3
1.5 CONTRIBUCIONES.....	4
1.6 ORGANIZACIÓN DE LA TESIS.....	5
CAPÍTULO 2 DESCRIPCIÓN DE TÉCNICAS DE ACCESO MÚLTIPLE	6
2.1 INTRODUCCIÓN	6
2.2 FDMA.....	6
2.3 TDMA.....	7
2.4 CDMA.....	8
2.5 RESUMEN	9
CAPÍTULO 3 FUNDAMENTOS DE GPS.....	10
3.1 INTRODUCCIÓN	10
3.2 SISTEMA GPS	10
3.3 ECUACIONES BÁSICAS DE GPS.....	12
3.4 ERRORES DEL GPS.....	14
3.5 SEÑALES GPS.....	15
3.6 RESUMEN	16
CAPÍTULO 4 CIRCUITOS DE GENERACIÓN Y DETECCIÓN DE CÓDIGOS GOLD	17
4.1 INTRODUCCIÓN	17
4.2 BPSK.....	18
4.3 CÓDIGOS GOLD	20
4.4 CORRELACIÓN	25
4.5 RUIDO GAUSSIANO	28
4.6 DEMODULADOR DE LAZO DE COSTAS	30
4.7 DELAY LOCKED LOOP	32
4.8 RESUMEN.....	34
CAPÍTULO 5 GENERACIÓN DE SEÑALES	35
5.1 INTRODUCCIÓN	35
5.2 IMPLEMENTACIÓN DEL CÓDIGO GOLD.....	35
5.3 MODULACIÓN BPSK	39
5.4 RESUMEN	42
CAPÍTULO 6 RECEPCIÓN DE SEÑALES.....	43
6.1 INTRODUCCIÓN	43
6.2 DETECCIÓN.....	43
6.3 FUNCIONAMIENTO EN CONJUNTO	50
6.4 RESUMEN	51
CAPÍTULO 7 RESULTADOS EXPERIMENTALES	52
7.1 INTRODUCCIÓN	52
7.2 RESULTADO DE GENERACIÓN DEL CÓDIGO GOLD.....	52

7.3 ADQUISICIÓN DEL CÓDIGO GOLD	55
7.4 TIEMPO DE EJECUCIÓN	60
7.5 RESUMEN	61
CAPÍTULO 8 CONCLUSIONES Y TRABAJO FUTURO	62
8.1 CONCLUSIONES	62
8.2 TRABAJOS FUTUROS	63
BIBLIOGRAFÍA	64
ANEXO 1	66
PROGRAMA PRINCIPAL	66
PROGRAMA PARA GENERAR LOS CÓDIGOS GOLD	69
PROGRAMA PARA MODULAR EN BPSK LOS CÓDIGOS GOLD	70
PROGRAMA PARA EL COSTAS LOCKED LOOP Y EL DELAY LOCKED LOOP	71
PROGRAMA PARA LA CORRELACIÓN	74

LISTA DE FIGURAS

Figura 1.1 Diagrama de adquisición y detección de códigos Gold	4
Figura 2.1 Acceso múltiple por división de frecuencias	7
Figura 2.2 Acceso múltiple por división de tiempo	7
Figura 2.3 Acceso múltiple por división de código.....	8
Figura 3.1 Segmentos que componen al sistema GPS.....	10
Figura 3.2 Estaciones del segmento de control, cortesía de Leica geosystem..	12
Figura 3.3 Intersección de dos esferas	12
Figura 3.4 Intersección de tres esferas	13
Figura 3.5 Diagrama a bloques de un Receptor GPS	16
Figura 4.1 Esquema del transmisor BPSK.....	18
Figura 4.2 Constelación de señales para BPSK.....	19
Figura 4.3 Datos binarios y BPSK en el tiempo.....	19
Figura 4.4 Autocorrelación a) y correlación cruzada de dos.....	21
Figura 4.5 Configuración de Fibonacci.....	22
Figura 4.6 Configuración de Galois.....	22
Figura 4.7 Generación del código Gold.....	23
Figura 4.8 Primeros Chips de código Gold.....	25
Figura 4.9 Ruido (a), autocorrelación (b) y densidad espectral (c).....	30
Figura 4.10 Lazo de Costas	32
Figura 4.11 Diagrama del Delay Locked Loop.....	33
Figura 5.1 Generador de códigos Gold.....	36
Figura 5.2 Diagrama de flujo de la generación del código Gold (Parte 1).....	37
Figura 5.3 Diagrama de flujo de la generación del código Gold (Parte 2).....	38
Figura 5. 4 Diagrama de flujo de la función de modulación.....	40
Figura 5.5 La señal modulada de todos los códigos, es sumada	42
Figura 6.1 Diagrama de flujo del Costas	45
Figura 6.2 Diagrama de flujo de la correlación cruzada	46
Figura 6.3 Diagrama de flujo del Delay Locked Loop	48
Figura 6.4 Referencia (a), atraso (b) y adelanto de la señal.....	49
Figura 6.5 Toma de decisión en el límite del código.....	49
Figura 6.6 Sistema de adquisición e identificación.....	50
Figura 7. 1 Fracción del código Gold generado del satélite 1	52
Figura 7. 2 Modulación del código Gold del satélite 1	53

Figura 7. 3 Señales de los doce satélites más ruido gaussiano.....	54
Figura 7. 4 Señal con amplitudes diferentes de los códigos Gold.....	54
Figura 7. 5 Primera etapa de la adquisición.....	55
Figura 7. 6 Diagrama de residuos	56
Figura 7. 7 Salida del Costas	56
Figura 7. 8 Código Gold Recuperado del satélite 1 con ID 2, 6	57
Figura 7. 9 Función de autocorrelación del código Gold empleando el satélite 1.	58
Figura 7. 10 Correlación cruzada entre dos satélites diferentes	58
Figura 7. 11 Autocorrelación del satélite 1 con 1023 chips a 16 muestras-chip.....	59
Figura 7. 12 Se observa el tiempo de ejecución del DLL, Costas.....	60

LISTA DE TABLAS

Tabla 3.1 Característica de los satélites GPS.....	11
Tabla 4.1 Sumador de Módulo 2	23
Tabla 4. 2 Códigos Gold de los satélites.....	24
Tabla 4.3 Propiedades de la correlación cruzada de Códigos Gold.	29
Tabla 7.1 Tiempo de ejecución de los circuitos	61

CAPÍTULO 1

Introducción

1.1 MOTIVACIÓN

El ser humano siempre ha sido un explorador, por lo que tiene la necesidad de identificar y recordar lugares que es la forma más simple de navegar, ésta era la técnica que el hombre de la edad de piedra usaba para abrirse paso a través de la selva, el bosque o el desierto teniendo como punto de referencia a un determinado árbol, río o montaña.

Con el paso de los siglos, poco a poco la técnica se fue sofisticando haciendo que la navegación fuera con las estrellas, la luna y el sol, puntos de referencia que estando en medio del mar, eran de gran ayuda, por ejemplo para determinar la posición de un barco se utilizaba como referencia la constelación de la osa mayor y apoyados en instrumentos ópticos se realizaban cálculos geométricos lo que generó las cartas de navegación. De esta forma el hombre sabía donde estaba ubicado y hacia donde iría aunque con un error de algunos kilómetros.

En la época moderna, y en la mitad del siglo pasado surge la medición por radio frecuencia (LORAN acrónimo de LOng RANGE Navigation), la cual consistía en poner varias antenas y programar un dispositivo con la posición de éstas y así calcular la ubicación del usuario con un error de aproximadamente 300 metros. Con la era espacial también la navegación tiene una gran transformación y así teniendo como base a LORAN se crean los satélites de GPS (acrónimo de Global Positioning System), de esta forma tenemos la posición no solo en latitud

y longitud sino también en altitud, con un error de menos de 100 metros, con cobertura mundial y de libre acceso.

1.2 JUSTIFICACIÓN

En la actualidad los sistemas de GPS son muy importantes ya que son de gran utilidad estratégica para los militares y civiles, por ejemplo un avión sabe en que punto de la tierra se encuentra y es parte fundamental de su sistema de navegación. Otros ejemplos los encontramos en: i) camiones de carga, ii) búsqueda y rescate de personas que por medio de un dispositivo, que envía la posición de la persona para que en caso necesario la auxilien o iii) en el sistema penitenciario para saber donde se encuentra cada reo.

Los receptores modernos de GPS tienen hasta 12 canales lo cual significa tener 12 bloques de cálculo, uno para cada satélite (se puede observar, máximo 12 satélites en un mismo instante), con ello, el tiempo de ejecución se reduce pero se elevan los costos de los receptores y las dimensiones de estos. Los algoritmos empleados son patentados por cada empresa, entonces hay que desarrollar la tecnología para tener al alcance un GPS con alta precisión y económico. Por lo tanto hay que comprender su funcionamiento y crear nuestros propios algoritmos.

1.3 OBJETIVO

Desarrollar un algoritmo de modulación y detección de códigos Gold [2] para que se pueda conocer el tiempo de retardo entre la señal generada por los satélites GPS y un receptor con lo cual posibilitar el cálculo de la posición de un usuario.

Se empleará un solo canal para la detección debido a que el algoritmo empleado deberá ser altamente eficiente para que el tiempo de ejecución sea muy pequeño.

1.4 METODOLOGÍA

Para lograr nuestro objetivo se plantea la siguiente estrategia:

Para probar si se ha alcanzado el objetivo, se implementa un generador de códigos Gold (ver Figura 1.1). Posteriormente para simular la señal enviada por un satélite GPS, estos códigos son enviados a un modulador Binary Phase Shift Keying (BPSK), luego, como en la realidad la señal recibida son los datos modulados de 12 satélites posibles de detectar en un instante, se sumarán 12 señales con diferentes códigos Gold para luego, adicionar con ruido gaussiano y completar la señal GPS total. Esa será la señal que recibirá nuestro sistema.

Con lo anterior, se está en capacidad de detectar la señal, para lo cual, se generan los algoritmos de implementación de los circuitos de seguimiento y detección, el circuito Costas, y demodulación BPSK, del algoritmo de dispersión del espectro detectado.

Luego, se continúa con los algoritmos de implementación del circuito de retardo para obtener el tiempo de vuelo de la señal desde un satélite hasta el receptor. Esto se obtiene por calcular el desfase que sufre la señal desde que se originó hasta la llegada al usuario final.

Todo lo anterior considera además los acondicionadores de señales para emplear los algoritmos adecuadamente en el procesamiento digital.

Esta metodología se esquematiza en la siguiente figura:

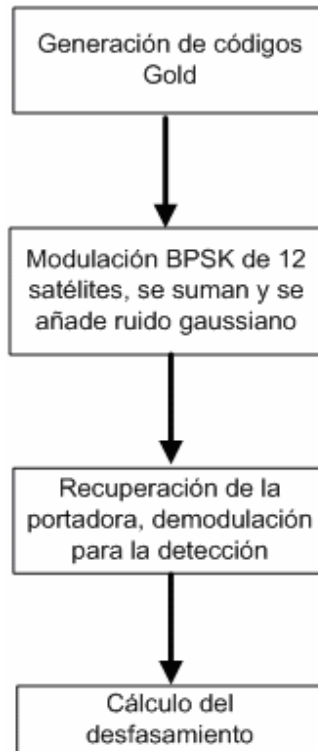


Figura 1.1 Diagrama de adquisición y detección de códigos Gold

1.5 CONTRIBUCIONES

Un generador de códigos Gold para GPS que trabaja para 32 satélites puestos en órbita (hasta el 2006) modulados en BPSK.

La implementación de algoritmos del Costas Locked Loop (CLL) y el Delay Locked Loop (DLL) modificado, que tienen como función recuperar la portadora de la señal entrante e identificar los códigos Gold respectivamente.

Poner en el dominio público el funcionamiento de un receptor GPS capaz de identificar los códigos Gold de satélites GPS.

1.6 ORGANIZACIÓN DE LA TESIS

En el primer capítulo se presenta la introducción de la tesis, la justificación de la misma, el objetivo que se pretende alcanzar y la metodología planteada para su realización, en el segundo capítulo se presenta una breve descripción de técnicas de acceso múltiple. En el tercer capítulo abarca una introducción del sistema de posicionamiento global (GPS). En el cuarto capítulo se muestran los circuitos involucrados los cuales son un modulador BPSK, un generador de códigos Gold, también hacemos uso de la correlación así como un demodulador de lazo de Costas y un Delay Locked Loop. En el quinto capítulo llamado generación de señales consta de la implementación del código Gold y su modulación. El sexto capítulo es la recepción de las señales y su detección. En el séptimo capítulo se muestran los resultados experimentales obtenidos a lo largo de la tesis. Y finalmente en el octavo capítulo las conclusiones y el trabajo a futuro para la tesis.

CAPÍTULO 2

Descripción de técnicas de acceso múltiple

2.1 INTRODUCCIÓN

A mediados del siglo pasado el hombre lanzo el primer satélite que poseía tan solo dos transmisores para un solo usuario pero en años recientes ha habido la necesidad de comunicación entre un mayor número personas. Por lo que nacen los satélites de acceso múltiple.

El acceso múltiple implica que más de un usuario tiene acceso a uno o más canales de radio dentro de un canal de comunicaciones vía satélite [13]. Los accesos múltiples de uso más común son acceso múltiple por división de frecuencia, acceso múltiple por división de tiempo y acceso múltiple por división de código.

2.2 FDMA

El acceso múltiple por división de frecuencia FDMA es un método de acceso múltiple que restringe un determinado ancho de banda de RF y éste se divide en bandas menores de frecuencia, llamadas subdivisiones [13], donde se ejecuta el enlace.

En la Figura 2.1 podemos apreciar el funcionamiento básico del sistema FDMA, donde f_1 , f_2 y f_3 representan diferentes frecuencias de las estaciones terrenas (en los enlaces satelitales) inmersas en un determinado ancho de banda.

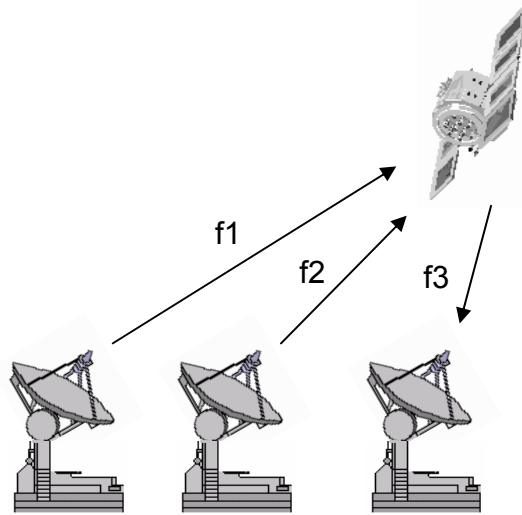


Figura 2.1 Acceso múltiple por división de frecuencias

2.3 TDMA

Acceso múltiple por división de tiempo (TDMA). (Figura 2.2) es un método de multiplexado por división de tiempo que multiplexa portadoras moduladas digitalmente entre estaciones terrestres en una red satelital [13]. Una estación terrestre transmite una ráfaga de información digital en un determinado tiempo y de esa forma evitar colisiones.

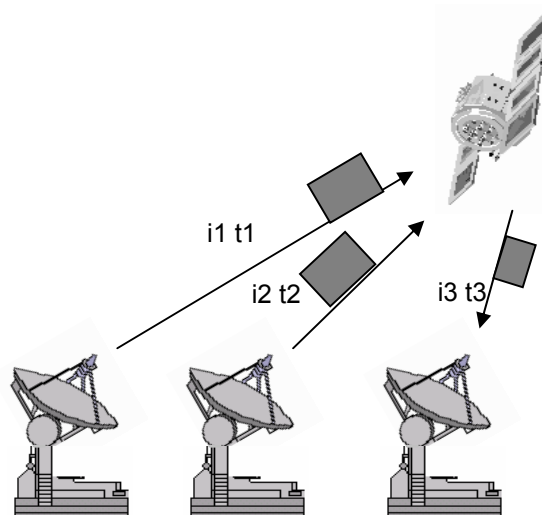


Figura 2.2 Acceso múltiple por división de tiempo

2.4 CDMA

En el acceso múltiple por división de código (CDMA), no existe una restricción en cuanto al ancho de banda ni el tiempo en que se realiza una transmisión de los datos. Cualquier transmisor puede cumplir su función en el momento que lo requiera y puede emplear cualquier banda o todo el ancho de banda de un canal [13].

Como en CDMA todos pueden transmitir en cualquier tiempo, se emplean técnicas de codificación para identificar los datos, cada receptor tendrá su propio código, estos son llamados códigos pseudos aleatorios.

En la Figura 2.3 se observa el funcionamiento de la transmisión CDMA en sistemas satelitales. Se puede apreciar que comparten una misma frecuencia y pueden transmitir datos en un mismo tiempo. Para recibir los datos de una estación terrena en particular, el receptor tiene que conocer el código pseudo aleatorio de la estación transmisora para su decodificación.

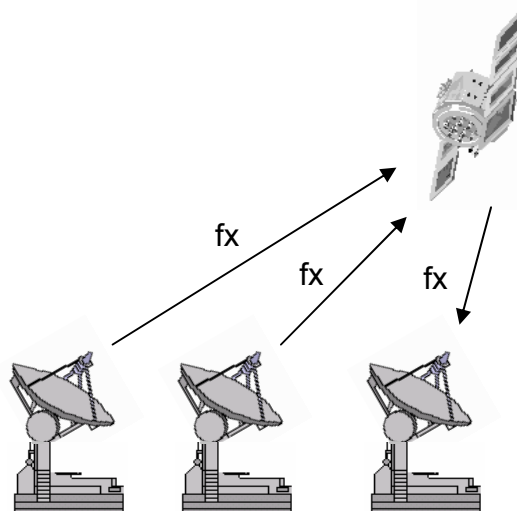


Figura 2.3 Acceso múltiple por división de código

Existen dos técnicas muy utilizadas en CDMA [13], siendo estas: espectro disperso y espectro por cambio de frecuencia.

La técnica de espectro disperso es cuando una señal expande otra señal más allá del ancho de banda necesario para transmitir la información. Los sistemas de espectro disperso sufren muy poca interferencia [14] dificultando su interceptación por tal motivo es ampliamente utilizado en comunicaciones militares y los sistemas GPS por ser creados para uso militar emplean este tipo de modulación que mas adelante lo retomaremos.

El espectro por cambio de frecuencia emplea un código digital para realizar cambios en la frecuencia de portadora en forma continua. La portadora es modulada con los datos del mensaje y se eleva su frecuencia con un oscilador local que es regido por un código pseudo aleatorio. Básicamente es la transmisión de datos sobre una banda limitada de frecuencias, durante un corto tiempo para posteriormente cambiar a otra banda de frecuencias [13].

2.5 RESUMEN

En este capítulo se presentó una introducción a las técnicas de acceso múltiple como TDMA, FDMA y CDMA. Por lo que se concluye se concluye que para aplicaciones de GPS, la mejor opción es CDMA ya que se puede transmitir a una determinada frecuencia y no hay una restricción de tiempo, además los datos son codificados. Con respecto al hardware, el hecho de tener una frecuencia de transmisión, conlleva a disminuir el peso del receptor.

CAPÍTULO 3

Fundamentos de GPS

3.1 INTRODUCCIÓN

En éste capítulo se muestran los segmentos en que se encuentra dividido el sistema de posicionamiento global, así como las ecuaciones fundamentales de los receptores GPS.

3.2 SISTEMA GPS

El sistema de posicionamiento global está dividido en tres segmentos (ver Figura 3.1) que son: Segmento espacial, segmento de control y segmento de usuario.

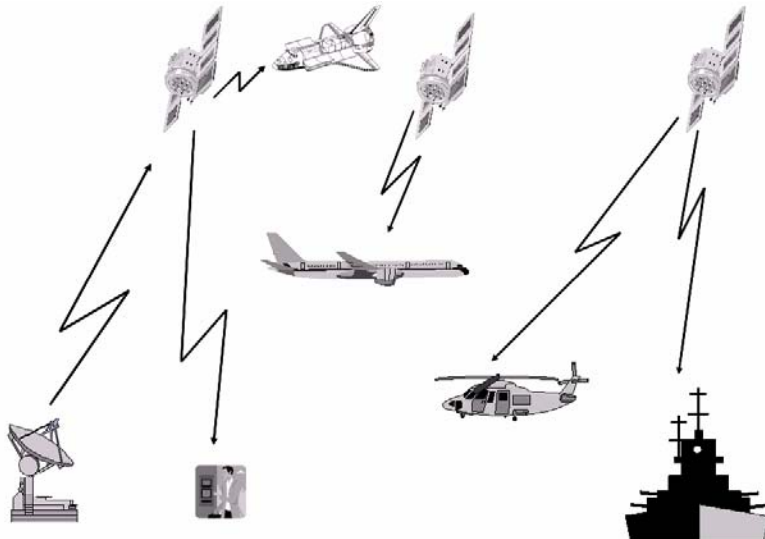


Figura 3.1 Segmentos que componen al sistema GPS

En la anterior figura se puede observar que el segmento espacial esta representado por todos los satélites que componen el sistema GPS, el sistema de control son las estaciones terrenas representadas por la antena parabólica y finalmente el segmento de usuario, que puede ser un barco, una persona, avión, incluso una nave espacial.

Como ya se mencionó el segmento espacial está compuesto por todos los satélites GPS que orbitan la tierra, incluyendo los de respaldo.

En condiciones normales 24 satélites se encuentran en operación con un periodo aproximado de 12 horas (ver Tabla 3.1).

Tabla 3.1 Característica de los satélites GPS

Satélites GPS	
Número de satélites	24
Número de planos orbitales	6
Número de satélites por órbita	4
Inclinación orbital	55°
Radio orbital	26560 Km
Periodo	11 hr 57 min 57.26 seg

El segmento de control se encarga del monitoreo de cada uno de los satélites, realiza las correcciones de posición en el espacio y actualiza sus efemérides. La estación principal del segmento de control esta ubicada en Colorado Springs (en los Estados Unidos de América), existen cuatro estaciones de control a lo largo del mundo, como se muestra en Figura 3.2.

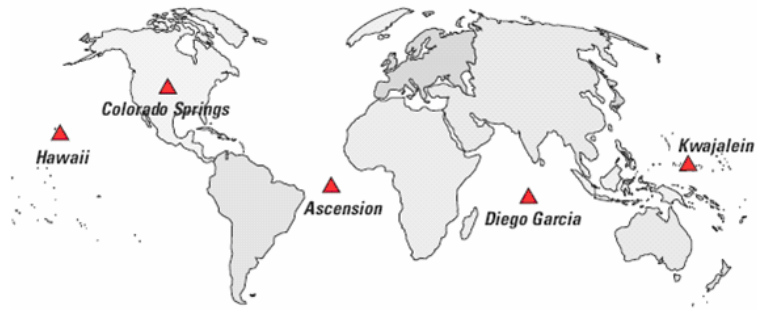


Figura 3.2 Estaciones del segmento de control, cortesía de Leica geosystem.

El segmento de usuario es compuesto por cualquier persona o sistema que requiera del servicio de GPS en cualquier parte de la Tierra o el espacio exterior.

3.3 ECUACIONES BÁSICAS DE GPS

Si se desea saber la posición de un objeto es necesario conocer algunos puntos en el espacio. Por ejemplo si empleamos un solo punto para referenciar nuestro objeto, solo conoceremos la distancia del punto con respecto a nosotros y eso no es suficiente pero si empleamos dos puntos y suponemos que estos realizan una trayectoria esférica, entonces la intersección de estos es donde se localiza el objeto (ver Figura 3.3).

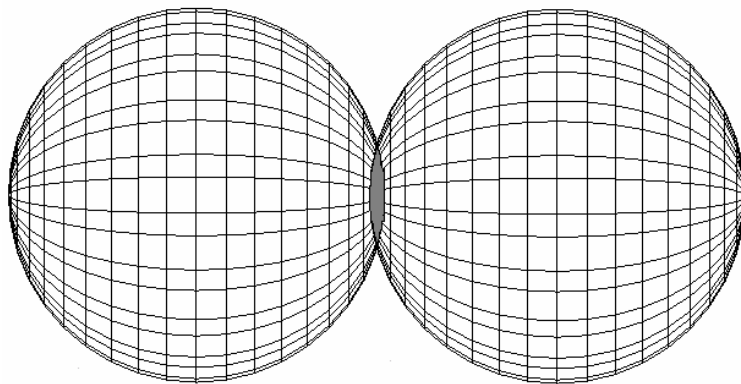


Figura 3.3 Intersección de dos esferas

Aun mejor si empleamos dos puntos y se calcula la distancia del objeto con respecto a éstos, entonces el objeto se ubica en dos posiciones posibles que son los puntos donde se intersectan las tres esferas (ver Figura 3.4).

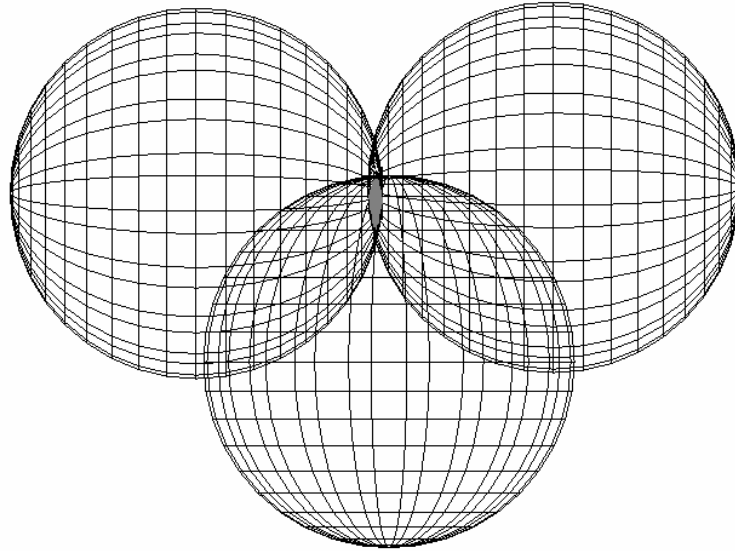


Figura 3.4 Intersección de tres esferas

Suponemos que tres satélites es condición suficiente para conocer un punto, entonces si r_1, r_2, r_3 representan a los satélites en un punto determinado en el espacio y r_u es el punto del receptor GPS (objeto buscado), entonces la distancia de los satélites al punto desconocido es definida como ρ_1, ρ_2, ρ_3 . Matemáticamente podemos expresar de la siguiente manera [15]:

$$\begin{aligned}
 \rho_1 &= \sqrt{(x_1 - x_u)^2 + (y_1 - y_u)^2 + (z_1 - z_u)^2} \\
 \rho_2 &= \sqrt{(x_2 - x_u)^2 + (y_2 - y_u)^2 + (z_2 - z_u)^2} \\
 \rho_3 &= \sqrt{(x_3 - x_u)^2 + (y_3 - y_u)^2 + (z_3 - z_u)^2}
 \end{aligned}
 \tag{2.1}$$

En teoría, resolviendo la ecuación anterior obtenemos la posición del receptor a partir de tres satélites [15].

3.4 ERRORES DEL GPS

Los satélites GPS (trasmisor) emplean osciladores atómicos de rubidio o cesio pero los receptores GPS utilizan osciladores de cuarzo, menos precisos pero de bajo costo. Existe una dependencia del oscilador tanto del satélite como del usuario lo que conlleva a que la distancia entre el usuario y el satélite sea

$$\rho_{iT} = c(t_u - t_{si}) \quad (2.2)$$

donde c es la velocidad de la luz, t_u es el tiempo del receptor, t_{si} es el tiempo del satélite y ρ_{iT} es el valor verdadero de la distancia del satélite al usuario. Sin embargo resulta casi imposible obtener el tiempo real tanto del tiempo del satélite como del usuario, por lo que resulta la siguiente expresión

$$\begin{aligned} t'_{si} &= t_{si} + \Delta b_i \\ t'_u &= t_u + \Delta b_{ut} \end{aligned} \quad (2.3)$$

donde t'_{si} es el tiempo actual del reloj del satélite, t'_u es el tiempo actual del usuario, Δb_i es el error del reloj del satélite y Δb_{ut} es el error del reloj del usuario. No solo el error de los relojes esta presente, si no también existen otros factores como el error producido por la ionosfera, troposfera, incluso la posición del satélite para solucionarlos existen modelos de ionosfera y de la troposfera por lo tanto la ecuación 2.1 tiene la siguiente forma incluyendo el error [15].

$$\rho_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} + b_u \quad (2.4)$$

Donde $i = 1, 2, 3, \dots, n$ y b_u es el error del usuario expresado en distancia.

3.5 SEÑALES GPS

Los satélites y los receptores GPS tienen osciladores que generan un mismo código de identificación ya que emplean modulación CDMA. Cuando el satélite envía el código este es comparado con el que se genera en el receptor y se puede calcular el desfase que ha sufrido el código por el tiempo que tarda en llegar desde el satélite hasta el usuario.

Existen dos tipos de señales: "grueso" (C/A) y preciso (P). Este último no es para uso civil sino militar (código clasificado), solo el gobierno de los Estados Unidos de América, quizás otros países y algunos civiles (con un permiso especial) tienen acceso a él.

Lo que hace clasificado a este código es que contiene las correcciones por ionosfera y troposfera, haciéndolo más preciso al receptor. El código C/A es para uso civil y todos podemos tener acceso a este pero no tiene corrección.

Las frecuencias fundamentales de los GPS's son:

$$L1 = 1575.42 \text{ MHz y}$$

$$L2 = 1227.6 \text{ MHz,}$$

éstas frecuencias son múltiplos de un reloj de 10.23 MHz [3, 4, 15]. La portadora de uso civil es L1 y de uso militar es L2.

Como ya se citó, las señales son transmitidas en CDMA en modulación BPSK y con un código de identificación para cada satélite, este último es llamado Código Gold.

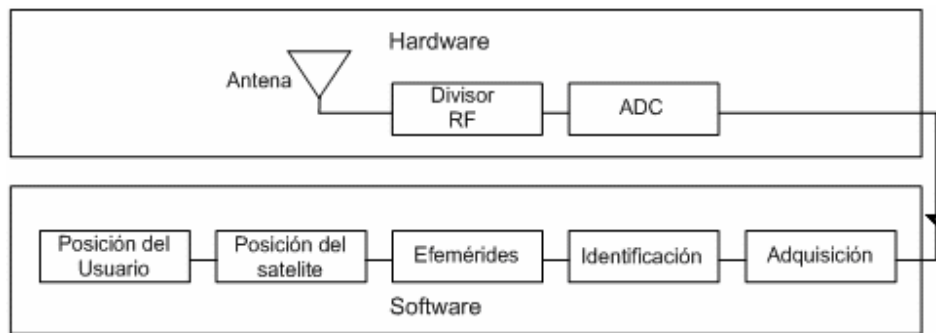


Figura 3.5 Diagrama a bloques de un Receptor GPS

En la Figura 3.5 se aprecia el diagrama básico de un receptor GPS, el bloque superior es de hardware que esta formado por la antena, divisor de RF y el convertidor analógico digital; el bloque inferior muestra el software que esta constituido por la adquisición, identificación, efemérides, posición del satélite y posición del usuario. En esta tesis se abarcan la adquisición y la identificación de los códigos GPS ya que es fundamental para calcular la posición de un usuario. Como podemos observar la posición del usuario puede ser calculada mediante las efemérides y las ecuaciones anteriormente descritas.

3.6 RESUMEN

En este capítulo se vió los principios básicos del sistema de posicionamiento global, los diferentes segmentos y la señal de GPS, así como los errores a los que es expuesta las señales GPS. También se vió las ecuaciones básicas para el cálculo de la posición de un usuario. Finalmente se presentó un diagrama bloques de un receptor GPS.

CAPÍTULO 4

Circuitos de generación y detección de códigos Gold

4.1 INTRODUCCIÓN

En este capítulo se verá la modulación Binary Phase Shift Keying (BPSK), los códigos Gold, la correlación, el ruido gaussiano, el demodulador de lazo de Costas, para la detección coherente y el Delay Locked Loop (DLL) para la detección de espectro disperso.

Como ya se dijo, los satélites GPS transmiten en CDMA y para identificarlos se emplean códigos Gold modulados en BPSK. Conforme viaja la señal desde el espacio hasta el usuario final en la tierra, ésta sufre distorsión, que en este caso es simulado por ruido gaussiano. Esto constituye la señal de espectro disperso y es a partir de ella que es necesario recuperar la portadora de la señal para luego, realizar la decodificación y finalmente la identificación de los satélites.

Para lo anterior, empleamos un demodulador de lazo de Costas (Costas Locked Loop) y para identificar el satélite a partir de su código asignado y simultáneamente calcular el desfase que tiene la señal desde que se generó hasta el usuario final, se emplea un circuito Delay Locked Loop (DLL). Todos estos circuitos son los que explicaremos a continuación.

4.2 BPSK

Binary Phase Shift Keying (BPSK), es un método de modulación digital angular de amplitud constante, con este tipo de modulación tenemos dos fases de salida para una sola frecuencia portadora, una fase representa “1” lógico y otra representa “0” lógico [10,12, 13]. Cuando los datos de entrada cambian de estado, inmediatamente la fase cambia dependiendo del estado en el que se encuentre, este cambio es de π radianes tomando como referencia el oscilador de la portadora. En la Figura 4.1 podemos ver un esquema del funcionamiento y las partes que componen a un modulador BPSK [13].

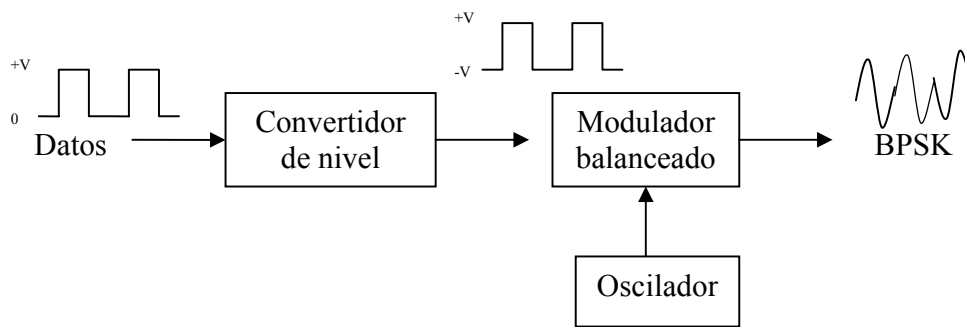


Figura 4.1 Esquema del transmisor BPSK

La representación matemática de la modulación BPSK es la siguiente

$$S_1 = A \cos 2\pi f_c t \quad 0 \leq t \leq T, \text{ para } 1 \quad (4.1)$$

$$S_2 = -A \cos 2\pi f_c t \quad 0 \leq t \leq T, \text{ para } 0 \quad (4.2)$$

donde T es el tiempo de duración del símbolo digital empleado (siendo en este caso los bits 0 y 1) y f_c es la frecuencia de la señal portadora en Hz.

Estas dos señales antípodas se pueden representar gráficamente por medio de una constelación de señales en un sistema bidimensional lo que podemos apreciar en la Figura 4.2.

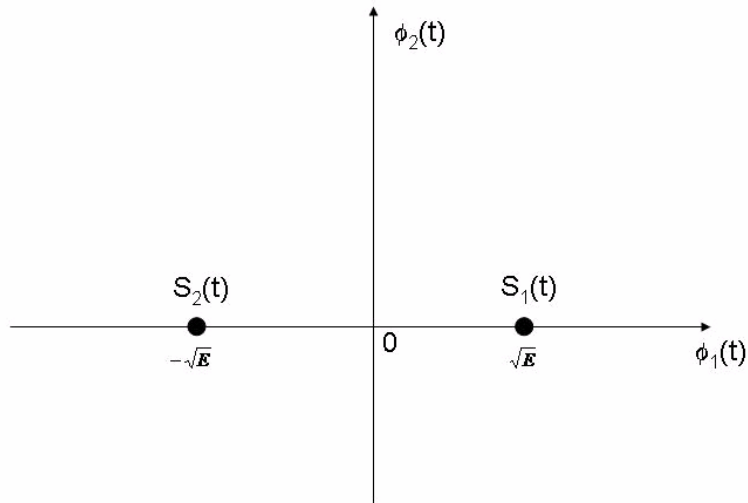


Figura 4.2 Constelación de señales para BPSK

con

$$\phi_1(t) = \sqrt{\frac{2}{T}} \cos 2\pi f_c t \quad 0 \leq t \leq T \quad (4.3)$$

$$\phi_2(t) = -\sqrt{\frac{2}{T}} \sin 2\pi f_c t \quad 0 \leq t \leq T \quad (4.4)$$

La Figura 4.3 presenta una señal típica BPSK en el tiempo [10], pudiéndose observar las variaciones en fase dependiendo de los datos binarios.

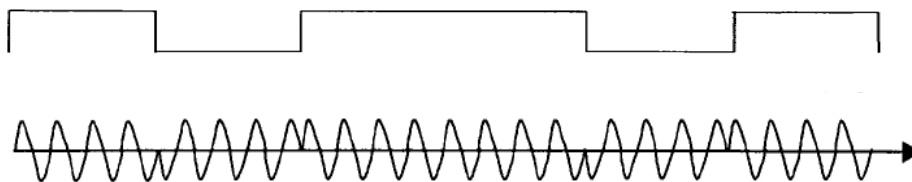


Figura 4.3 Datos binarios y BPSK en el tiempo

4.3 CÓDIGOS GOLD

Para la transmisión de datos en CDMA es requisito tener un código de identificación. En la actualidad los satélites GPS transmiten los datos en CDMA y utilizan como código de identificación el llamado código Gold [5, 11, 12].

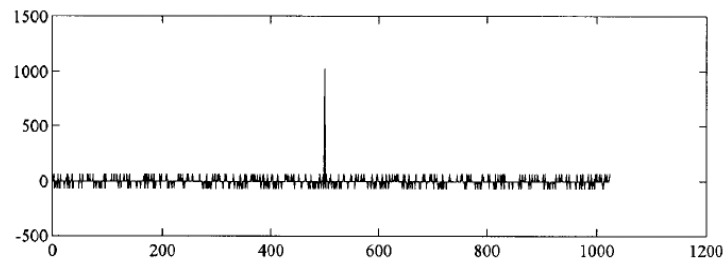
Los códigos Gold tienen como base los códigos pseudoaleatorios, estos códigos como su nombre lo indica, son una secuencia “semi-aleatoria”, implicando con ello que si se observa su comportamiento aparenta ser generado aleatoriamente pero esto no sucede ya que existe un método para generar los códigos y que estos sean replicados cuantas veces sea necesario.

Su comportamiento estadístico al realizar la correlación cruzada es mostrado en la Figura 4.4. En esta figura se muestra su comportamiento tomándose como ejemplo los denominados códigos Gold para GPS. Pero lo que más interesa por el momento es que estos códigos son programables y por eso han sido ampliamente utilizados como medio de identificación para la multicanalización en secuencia directa denominada CDMA.

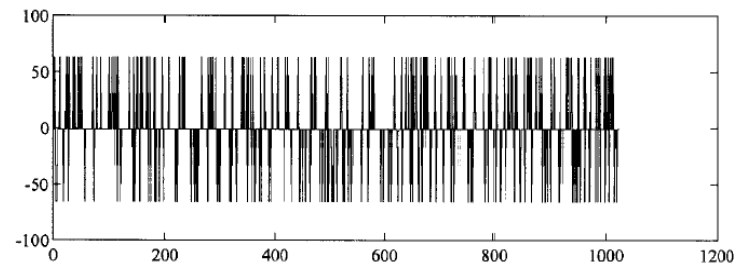
Para generar un código pseudoaleatorio se emplea un “Linear Feedback Shift Register” (LFSR) [7] que consiste en un registro de corrimiento de lazo cerrado. El LFSR tiene las siguientes propiedades [7, 8].

- Balance. En un periodo completo de secuencia pseudoaleatoria los números de 1s difieren del número de 0s por al menos 1.
- Corrida. Existen 2^{n-1} corridas de números consecutivos tanto 1s como 0s y la mitad de las corridas son de longitud 1, 2^{n-2} corridas de longitud 1s, 2^{n-3} corridas de longitud 2, 2^{n-4} corridas de longitud 3, ..., 2^{n-n} corridas de 0s de longitud n-1 y 1 corridas de 1s con longitud n.

- **Correlación.** Si una secuencia completa es comparada bit a bit, cualquier cambio de ésta, el número de aciertos menos el número de fallas es siempre -1.



a)



b)

Figura 4.4 Autocorrelación a) y correlación cruzada de dos diferentes códigos Gold b) [15]

Un LFSR puede ser descrito matemáticamente a través de la denominada configuración de Fibonacci (Figura 4.5) o Galois (Figura 4.6). Ambas configuraciones tienen una misma finalidad, proporcionar una máquina de eventos finitos, descrita en la siguiente forma polinomial [8].

$$f(x) = x^n + c_{n-1}x^{n-1} + \dots + c_2x^2 + c_1x + 1 \quad (4.5)$$

donde x es un estado del LFSR, n es el orden del mismo y c_i son constantes.

El polinomio anterior es algunas veces llamado polinomio primitivo irreducible, y cuando $c_i=1$ significa una conexión y cuando $c_i=0$ significa que no existe conexión en el LFSR.

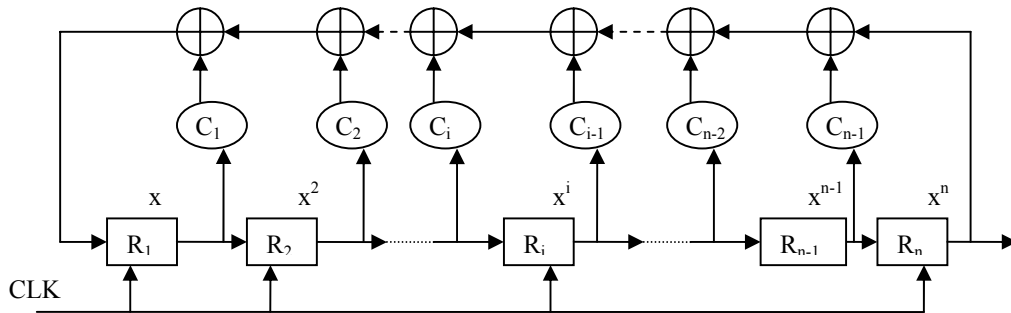


Figura 4.5 Configuración de Fibonacci

El comportamiento pseudo aleatorio proviene de que, para un generador de secuencia de registro de corrimiento con n eventos, la secuencia de salida siempre será periódica (periodo de código) puesto que, teniendo las condiciones iniciales del LFSR existe un número finito de pulsos donde las condiciones iniciales son nuevamente reproducibles.

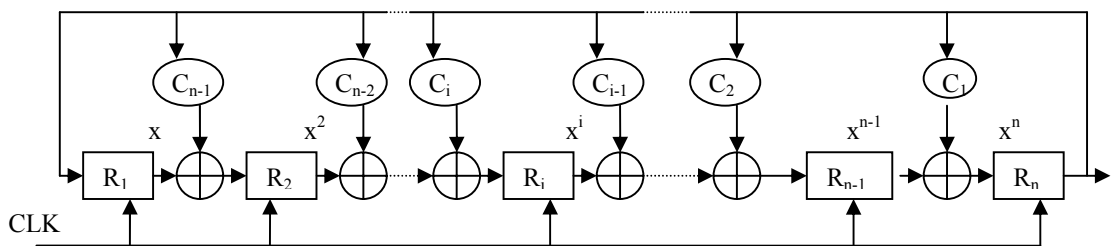


Figura 4.6 Configuración de Galois

El máximo número de eventos posibles para un generador de secuencia LFSR es de $2^n - 1$, a esto se le conoce como longitud máxima del LFSR [7, 8, 15].

Tabla 4.1 Sumador de Módulo 2

Entrada 1	Entrada 2	Salida
0	0	0
0	1	1
1	0	1
1	1	0

El código Gold es creado a partir de 2 generadores de secuencia LFSR de la misma longitud (Figura 4.7), el pulso de reloj es el mismo para los dos generadores y son conectados mediante un sumador de módulo 2 cuyo comportamiento se observa en la Tabla 4.1. Obviamente, el código Gold también hereda el periodo de código de los generadores LFSR.

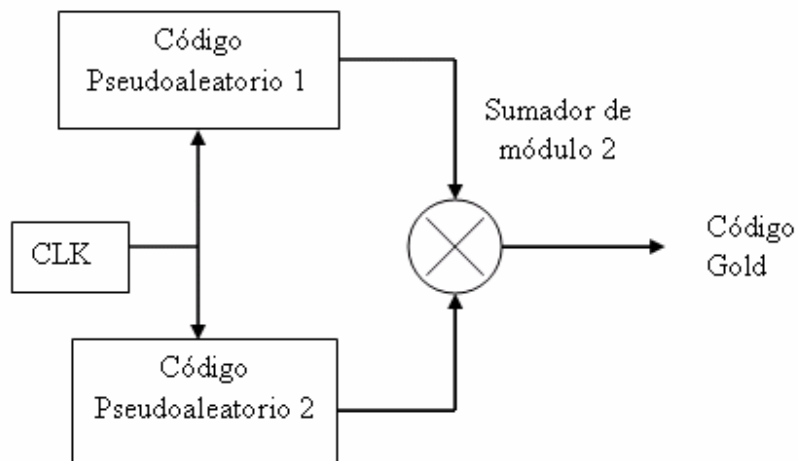


Figura 4.7 Generación del código Gold

La Figura 4.8 representa la generación de un código Gold, cuyo código ha sido desfasado 2 chips y corresponde al satélite 2 (Tabla 4. 2).

Tabla 4. 2 Códigos Gold de los satélites

Número de satélite	Selección de código	Códigos Gold en Hexadecimal (primeros 10 chips)
1	2 y 6	320
2	3 y 7	390
3	4 y 8	3C8
4	5 y 9	3E4
5	1 y 9	25B
6	2 y 10	32D
7	1 y 8	259
8	2 y 9	32C
9	3 y 10	396
10	2 y 3	344
11	3 y 4	3A2
12	5 y 6	3E8
13	6 y 7	3F4
14	7 y 8	3FA
15	8 y 9	3FD
16	9 y 10	3FE
17	1 y 4	26E
18	2 y 5	337
19	3 y 6	39B
20	4 y 7	3CD
21	5 y 8	3E6
22	6 y 9	3F3
23	1 y 3	233
24	4 y 6	3C6
25	5 y 7	3E3
26	6 y 8	3F1
27	7 y 9	3F8
28	8 y 10	3FC
29	1 y 6	257
30	2 y 7	32B
31	3 y 8	395
32	4 y 9	3CA

En la Tabla 4. 2 se observan los valores de los códigos Gold generados por satélites GPS en hexadecimal. Debe notarse que la columna final de ésta tabla, se refiere a los primeros 10 chips [2] de la secuencia del código correspondiente.

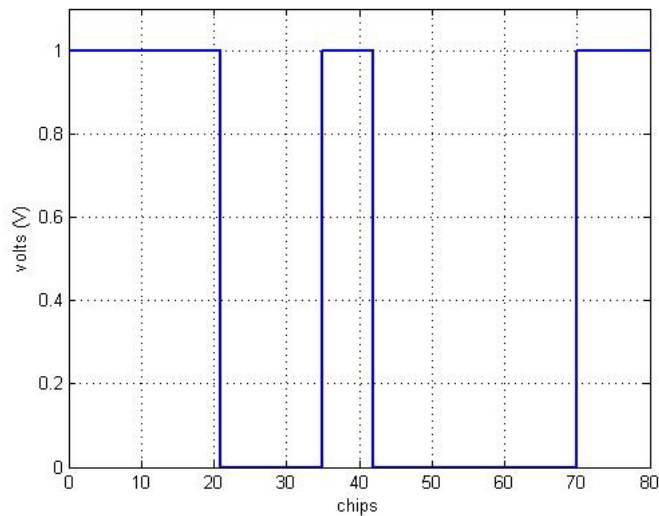


Figura 4.8 Primeros Chips de código Gold

Los códigos Gold tienen las siguientes propiedades.

- Siempre se utilizan 2 LFSR.
- Lazos de retroceso programables.
- La longitud del código es 2^{n-1} donde n es el número de eventos.
- La correlación cruzada es inversamente proporcional a la longitud del código.
- La longitud del código reduce el tamaño del pico de la correlación.

4.4 CORRELACIÓN

Considerando una señal $x(t)$ continua y determinista sea real o compleja su energía se define como [16]:

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 dt \quad (4.6)$$

la cual es finita. De acuerdo al teorema de Parseval [16] esta energía la podemos escribir como

$$E_x = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\omega)|^2 d\omega \quad (4.7)$$

donde $X(\omega)$ es la transformada de Fourier de $x(t)$. Por tanto, se tiene que $|x(t)|^2$ es la distribución de energía en el tiempo y $|X(\omega)|^2$ es la distribución espectral en la frecuencia.

Si se escribe $S_{xx}^E(\omega) = |X(\omega)|^2$ y definiendo la autocorrelación [6, 16]

$$r_{xx}^E(\tau) = \int_{-\infty}^{\infty} x^*(t)x(t+\tau)dt = x^*(-\tau) * x(\tau) \quad (4.8)$$

se tiene

$$S_{xx}^E(\omega) = \int_{-\infty}^{\infty} r_{xx}^E(\tau)e^{-j\omega\tau} d\tau \quad (4.9)$$

es decir, existe una correspondencia bien definida entre el par de transformadas de Fourier $S_{xx}^E(\omega) \longleftrightarrow r_{xx}^E(\tau)$ [15].

La autocorrelación es una medida de cuanto es la similitud de la señal consigo misma.

De igual forma se define la correlación cruzada

$$r_{xy}^E(\tau) = \int_{-\infty}^{\infty} x^*(t)y(t+\tau)dt \quad (4.10)$$

obteniendo

$$S_{xy}^E(\omega) = \int_{-\infty}^{\infty} r_{xy}^E(\tau) e^{-j\omega\tau} d\tau \quad (4.11)$$

es decir,

$$S_{xy}^E(\omega) \longleftrightarrow r_{xy}^E(\tau) \quad (4.12)$$

Por tanto, la correlación cruzada es una medida de que tanto se parecen dos señales.

Si las señales se describen con muestras discretas en el tiempo, se tiene

$$E_x = \sum_{-\infty}^{\infty} |x(n)|^2 \quad (4.13)$$

Por lo que el teorema de Parseval de tiempo discreto se expresa

$$E_x = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega \quad (4.14)$$

El término $|X(e^{j\omega})|^2$ se llama densidad espectral de energía [16], por lo que

$$S_{xx}^E(e^{j\omega}) = |X(e^{j\omega})|^2 \quad (4.15)$$

En la anterior ecuación el primer término es la densidad espectral de energía de la transformada de Fourier en el tiempo discreto de la autocorrelación

$$r_{xx}^E(m) = \sum_{n=-\infty}^{\infty} x^*(n)x(n+m) \quad (4.16)$$

entonces

$$S_{xx}^E(e^{j\omega}) = \sum_{m=-\infty}^{\infty} r_{xx}^E(m)e^{-j\omega m} \quad (4.17)$$

de forma análoga tenemos que la correlación cruzada es

$$r_{xy}^E(m) = \sum_{n=-\infty}^{\infty} x^*(n)y(n+m) \quad (4.18)$$

entonces

$$S_{xy}^E(e^{j\omega}) = \sum_{m=-\infty}^{\infty} r_{xy}^E(m)e^{-j\omega m} \quad (4.19)$$

por lo tanto

$$S_{xy}^E(e^{j\omega}) = \sum_{m=-\infty}^{\infty} r_{xy}^E(m)e^{-j\omega m} \quad (4.20)$$

En la Tabla 4.3 se muestran **¡Error! No se encuentra el origen de la referencia.** las propiedades que tienen los códigos Gold [5, 11, 12, 15]. Por ejemplo tomando el código Gold de los GPS es de 2 LFSR de $n = 10$ eventos (par) entonces tiene un periodo de 1023 si empleamos la relación existente en la tabla los valores de la correlación cruzada es de: $-65/1023$, $-1/1023$, $63/1023$.

En la Figura 4.4 (a) aparece un pico de 1023 y valores de -65 y 63 los cuales corresponden a al nivel de correlación cruzada sin normalizar.

4.5 RUIDO GAUSSIANO

También llamado ruido blanco, se define como un proceso aleatorio estacionario con una densidad espectral constante. El término “blanco” proviene

Tabla 4.3 Propiedades de la correlación cruzada de Códigos Gold.

Periodo del código	Número de eventos del registro de corrimiento	Nivel de correlación cruzada normaliza	Probabilidad
		$-\frac{2^{(n+1)/2} + 1}{P}$	0.25
$P = 2^n - 1$	n = impar	$-\frac{1}{P}$	0.5
		$-\frac{2^{(n+1)/2} - 1}{P}$	0.25
		$-\frac{2^{(n+2)/2} + 1}{P}$	0.125
$P = 2^n - 1$	n = par	$-\frac{1}{P}$	0.75
		$\frac{2^{(n+2)/2} - 1}{P}$	0.125

de la óptica ya que la luz visible contiene todas las frecuencias visibles, pero el ruido blanco es un proceso sin correlación con la misma energía en todas las frecuencias [6, 16]. Entonces un ruido que tiene la misma energía en todas las frecuencias, esto es, desde $-\infty$ hasta $+\infty$, tendría necesariamente energía infinita por lo que es un concepto teórico. Pero si nos limitamos a una banda de frecuencias, entonces tiene ya sentido práctico porque su energía ya no es infinita.

La función de autocorrelación en el tiempo continuo de ruido blanco de media cero y varianza σ^2 es una función delta [16], esto es,

$$r_{NN}^E(\tau) = E[N(t)N(t + \tau)] = \sigma^2 \delta(\tau) \quad (4.21)$$

entonces su densidad de energía tomando la transformada de Fourier es

$$P_{NN}(f) = \int_{-\infty}^{\infty} r_{NN}(t) e^{-j2\pi ft} dt = \sigma^2 \quad (4.22)$$

por lo que podemos decir que el ruido blanco tiene una potencia constante. Todas las características anteriores de un ruido blanco se pueden observar en la Figura 4.9.

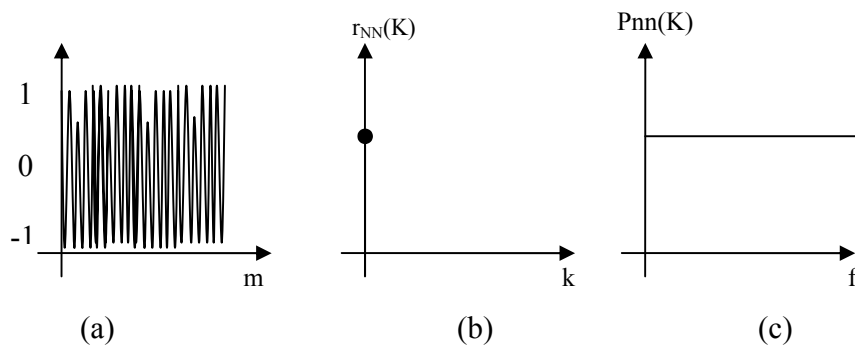


Figura 4.9 Ruido (a), autocorrelación (b) y densidad espectral (c)

Para las pruebas de esta tesis, se utilizó el comando *awgn* de Matlab el cual mide la potencia de la señal y posteriormente le suma el ruido blanco gaussiano con media cero y varianza 1 a una señal que se defina previamente. Lo anterior se hizo para asegurar que el ruido contiene frecuencias dentro del ancho de banda de nuestra señal de entrada.

4.6 DEMODULADOR DE LAZO DE COSTAS

El demodulador de lazo de Costas, también conocido como demodulador I Q fue creado en 1956 por Costas, este demodulador es ampliamente utilizado en receptores BPSK de espectro disperso [2].

La configuración básica del lazo de Costas (Figura 4.10), incluye un oscilador controlado por voltaje (VCO) que tiene como función, el proporcionar la frecuencia de referencia de la portadora. La señal de referencia producto del

VCO, es multiplicada simultáneamente en dos ramas, una rama denominada en fase (I) y otra rama denominada en cuadratura (Q) con la señal de entrada. El resultado se entrega a un filtro pasa bajas que proporcionará una señal de error al VCO para que la fase de referencia siga a la fase de entrada.

Matemáticamente puede escribirse de la siguiente manera. La señal recibida es multiplicada por la componente $\cos(2\pi ft + \phi)$ y también multiplicada por la componente $\text{sen}(2\pi ft + \phi)$ de la salida del VCO. Entonces se tiene la señal en fase, $y_I(t)$ y en cuadratura $y_Q(t)$

$$\begin{aligned} y_I(t) &= [s(t) + n(t)] \cos(2\pi ft + \phi) & (4.23) \\ &= \frac{1}{2} [A(t) + n(t)] \cos \Delta\phi + \frac{1}{2} n(t) \sin \Delta\phi + \dots \end{aligned}$$

$$\begin{aligned} y_Q(t) &= [s(t) + n(t)] \text{sen}(2\pi ft + \phi) & (4.24) \\ &= \frac{1}{2} [A(t) + n(t)] \sin \Delta\phi - \frac{1}{2} n(t) \cos \Delta\phi + \dots \end{aligned}$$

donde los puntos suspensivos son los términos de doble frecuencia los cuales tiende a desaparecer con el filtro pasa bajas de cada rama. Así, la resultante después cada filtro, tiene un error de fase dado por $\Delta\phi = \hat{\phi} - \phi$.

La salida de los filtros de cada rama, se multiplican y nos da la señal de error

$$e(t) = \frac{1}{8} \{ [A(t) + n(t)]^2 - n^2(t) \} \text{sen} 2\Delta\phi - \frac{1}{4} n(t) [A(t) + n(t)]^2 - n^2(t) \cos 2\Delta\phi \quad (4.25)$$

Luego, la señal de error es obtenida por medio de un filtro pasabajas que cierra el lazo de control sobre el VCO. Con ello, el VCO corrige su oscilación hasta seguir adecuadamente a la fase de la señal que se desea detectar.

Hay que observar que cuando ϕ es pequeña $\cos(2\phi)$ es 1, esto sucede cuando el Costas se “amarrá” a la señal por lo tanto la salida es $A^2/4$ siendo este valor la amplitud de nuestros datos recuperados.

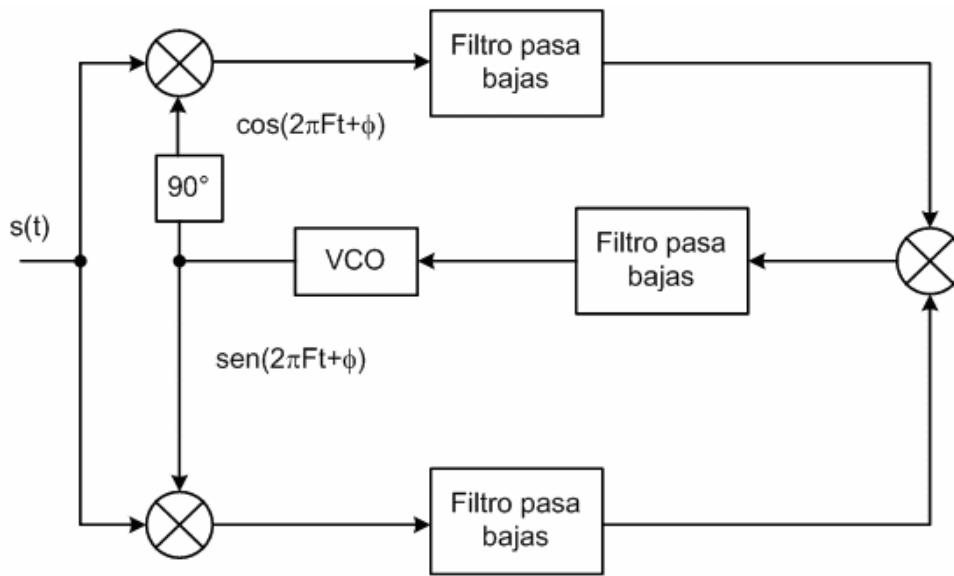


Figura 4.10 Lazo de Costas

4.7 DELAY LOCKED LOOP

EL Delay Locked Loop (DLL) tiene como función dar una estimación del tiempo que transcurre entre la señal recibida y otra generada localmente. En el caso de señales GPS la señal local será un código Gold. En la Figura 4.11 se puede apreciar el sistema básico de un DLL [12].

Su funcionamiento es el siguiente: la señal recibida es multiplicada por una réplica local que se encuentra tanto adelantada como atrasada (medio chip de adelanto y atraso). Se suman y pasan a través de un filtro pasa bajas al cual se

le aplica un filtro de mediana estadística cuyo resultado alimenta al VCO (reloj del LFSR)

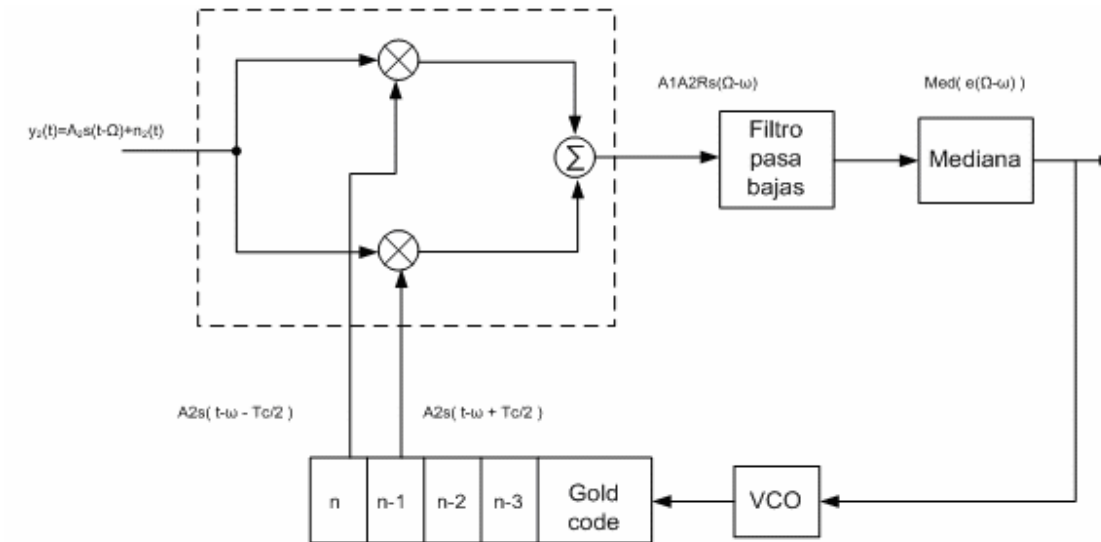


Figura 4.11 Diagrama del Delay Locked Loop

Para analizar el funcionamiento del DLL, se define la señal principal como:

$$y_1(t) = A_1s(t) + n_1(t) \quad (4.26)$$

y la señal de entrada como:

$$y_2(t) = A_2s(t - \Omega) + n_2(t) \quad (4.27)$$

Se suma un retardo ω a y_1 , entonces se tiene

$$E[y_1(t - \omega)y_2(t)] = E[A_1s(t - \omega)A_2s(t - \Omega)] \quad (4.28)$$

$$= A A_2 R_s(\Omega - \omega) \quad (4.29)$$

se obtiene un máximo, en $\Omega = \omega$

Para obtener el error entre $y_1(t-\omega)$ y $y_2(t)$, la señal resultante se envía a un filtro pasa bajas $H(\omega)$ cuya salida es proporcional a

$$e(\Omega - \omega) \quad (4.30)$$

Se toma la mediana de este error

$$\text{Med}[e(\Omega - \omega)] \quad (4.31)$$

Ahora pasa al VCO, indicándonos el avance de los códigos Gold de referencia. Este es un ciclo el cual se va repitiendo y por tanto, conforme avanza el algoritmo, el error (e) va disminuyendo hasta que el código generado es igual al analizado.

$$e(\Omega - \omega) = 0 \quad (4.32)$$

Es en este punto cuando el DLL hace converger al código de referencia con el código buscado.

4.8 RESUMEN

En este capítulo se presentó la teoría de los circuitos como el modulador BPSK, la generación de los códigos Gold, la correlación, el ruido gaussiano, el “Costas Locked Loop” (CLL) y el “Delay Locked Loop” (DLL), en los dos siguientes capítulos se verá como estos bloques se unen para formar, tanto un sistema generador de señales de espectro disperso, como un sistema de detección, por un lado dispersando el espectro y por otro realizando una detección coherente de la señal portadora.

CAPÍTULO 5

Generación de Señales

5.1 INTRODUCCIÓN

En este capítulo se presenta el sistema generador de señales de espectro disperso, utilizando los circuitos descritos en el capítulo anterior. Primero se generan los códigos Gold los cuales entran a un modulador BPSK, se suman doce señales moduladas, se desfasa intencionalmente y al resultado se le suma un ruido gaussiano.

5.2 IMPLEMENTACIÓN DEL CÓDIGO GOLD

El primer paso es la generación de los códigos Gold para ello se emplean dos LFSR de 10 eventos (ver Figura 5.1). Comúnmente se emplea la condición inicial de que todos los eventos deben de estar en 1's para iniciar el proceso [4, 7].

La ecuación matemática de cada uno de los dos LFSR es la siguiente:

$$G1(x) = 1 + x^3 + x^{10} \quad (5.1)$$

$$G2(x) = 1 + x^2 + x^3 + x^6 + x^8 + x^9 + x^{10} \quad (5.2)$$

Con estas dos ecuaciones obtenemos los códigos Gold para GPS [15], los cuales tienen un periodo de $2^n - 1$, esto es 1023 bits los cuales, al no contener información son denominados chips. Así, si tenemos 10 eventos, esto es $n=10$

con un periodo de código de 1023, esto es $P=1023$ en la Tabla 4.3 y de acuerdo a Spilker [11], se pueden calcular los tres valores de la correlación cruzada los cuales son: $-65/1023$, $-1/1023$ y $63/1023$.

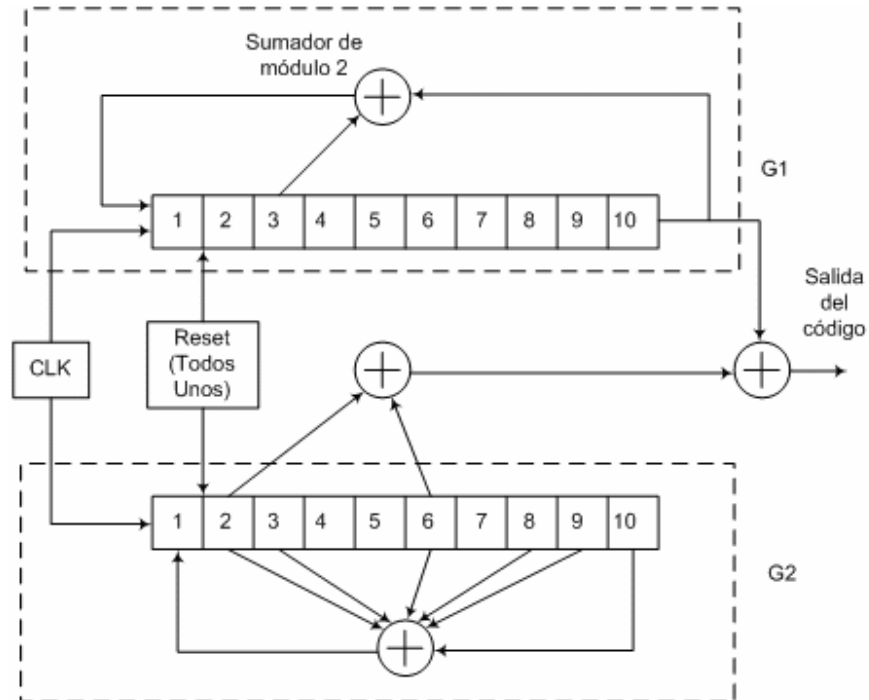


Figura 5.1 Generador de códigos Gold.

La Figura 5.1 es el diagrama a bloques del generador de códigos Gold. Producto de esta figura, surge el siguiente diagrama de flujo (Figura 5.2 y Figura 5.3) y en donde se observa el funcionamiento de un generador de códigos Gold para los satélites GPS.

Este funcionamiento fue programado en Matlab de acuerdo al siguiente algoritmo:

Los eventos del generador de números pseudos aleatorios o LFSR les llamaremos G1 y G2. A estos se les asigna el valor de 1's, esta es la condición inicial del generador de códigos Gold para GPS.

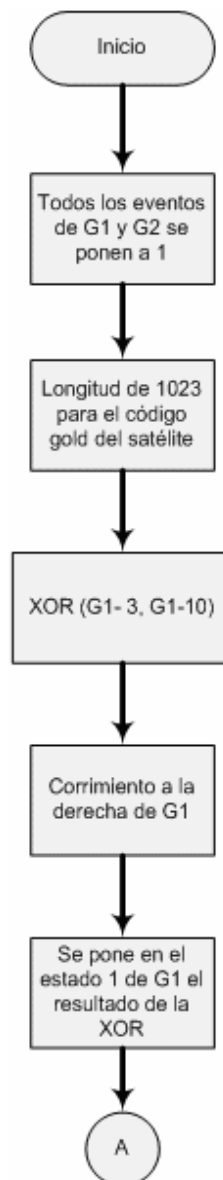


Figura 5.2 Diagrama de flujo de la generación del código Gold (Parte 1)

Después de indicar las condiciones iniciales, se crea un arreglo el cual contiene el código Gold generado. Obviamente, el tamaño del arreglo es del periodo del código Gold el cual es de 1023.



Figura 5.3 Diagrama de flujo de la generación del código Gold (Parte 2)

Se calcula el valor de G1 por lo que se procede a realizar el cálculo del sumador de módulo 2 el cual es hecho por una XOR (Tabla 4.1) entre los eventos 3 y 10 de G1 cuyo resultado pasa a formar parte del evento 1 en G1 por lo que es necesario realizar antes un corrimiento a la derecha para que los valores de G1 sean desplazados.

Continuando con el diagrama de flujo en la Figura 5.3, se realiza una XOR entre los eventos 2 y 3 el resultado se vuelve a ejecutar un XOR entre este último y el evento 6 de tal forma que los eventos 2, 3, 6, 8, 9 y 10 sean las entradas de una operación XOR, el resultado de esta operación se guarda en una variable temporal.

En el esquema (Figura 5.1) se observa que hay dos valores que son la entrada al sumador de módulo 2 (evento 2 y 6), estos valores son los códigos de identificación de cada satélite GPS.

El código así generado, nos ayuda a diferenciar cada satélite ya que al variar los valores de entrada como se especificó, nos produce un código Gold diferente para cada satélite. Siguiendo con el diagrama de flujo los valores de los eventos (2 y 6) entran a una XOR, cuya salida es una de las entradas de un sumador de módulo 2, mientras que la otra entrada es el valor del evento 10 de G1 y finalmente tenemos el primer valor del código Gold.

Por último se realiza un corrimiento a la derecha de G2 y en el evento 1 se guarda el valor temporal de la XOR inferior de G2.

5.3 MODULACIÓN BPSK

Como ya se había mencionado la modulación de los códigos Gold para GPS es en BPSK.

De manera similar a la generación de códigos Gold, se realizó un programa que realizaba la modulación de acuerdo al siguiente algoritmo:

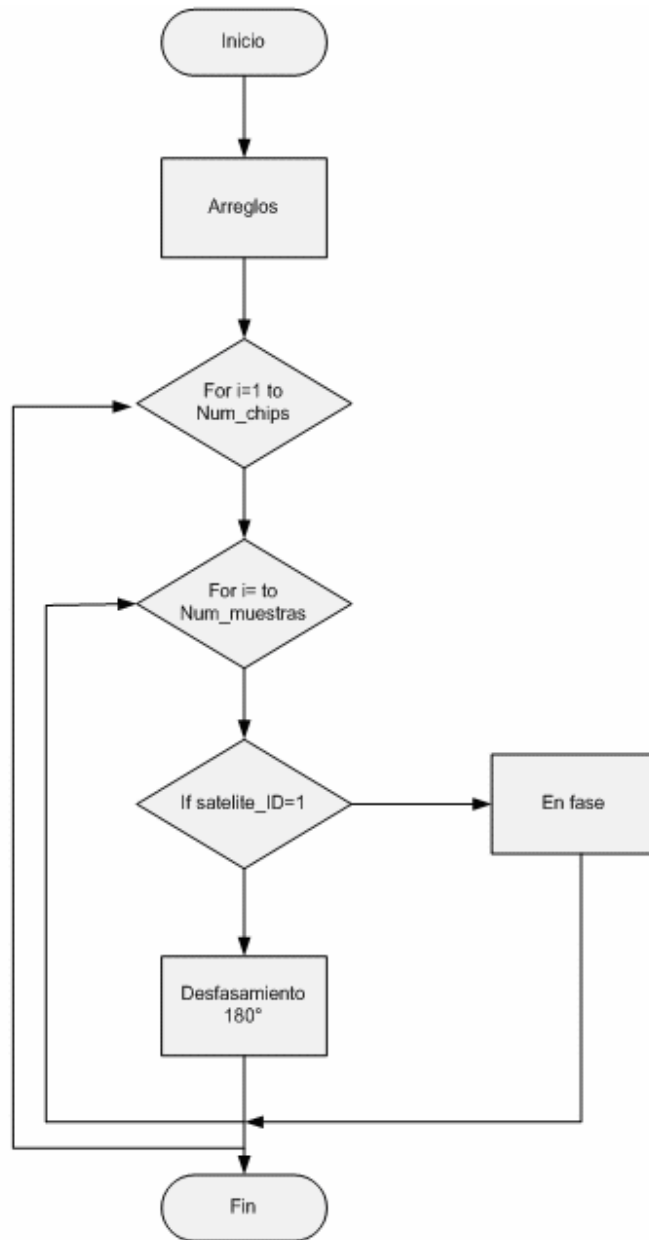


Figura 5. 4 Diagrama de flujo de la función de modulación

Primero se definen las variables que contendrán los valores de la modulación los cuales forman un arreglo. El tamaño de un arreglo va a depender de la longitud de los códigos Gold y el número de muestras que se quieren, en este caso se emplean 8 muestras entonces nuestro arreglo estará constituido por 8 muestras multiplicadas por 1023 chips el cual es la longitud máxima de Código Gold por lo tanto tenemos un arreglo con 8184 valores.

Ya que se ha generado el arreglo (Figura 5. 4) que contendrán los datos, es necesario definir una onda sinusoidal para el proceso de modulación BPSK. La generación de la sinusoidal es creada con un ciclo *for* anidado y un operador de comparación el cual decide si la onda será en fase o desfasada.

Cuando se detecta un 1 lógico no hay cambio alguno en la fase pero al ser diferente de este, la fase cambia 180° .

Después de generar la modulación del código, se procede a realizar la modulación de los demás códigos de la misma manera, posteriormente se suma la modulación de cada uno de los códigos de los satélites, que en este caso son 12, ya que es el número máximo que puede observar un receptor GPS, posteriormente se suma ruido gaussiano, con lo que se tiene un grupo de señales moduladas en BPSK que sufren distorsión, la señal es desfasada simulando el desplazamiento que la señal sufre respecto al tiempo local tanto del emisor (satélite), como del receptor ya que en los satélites GPS tienen relojes atómicos mientras que los receptores cuentan con un reloj de cuarzo ya que este último es muy barato.

Para tener una mejor comprensión de lo anteriormente mencionado podemos apoyarnos en el diagrama a bloques de la Figura 5.5, se puede observa que los códigos Gold son generados y se modulan, posteriormente, se suman y se induce ruido gaussiano, y la señal resultante es la que se envía al receptor GPS.

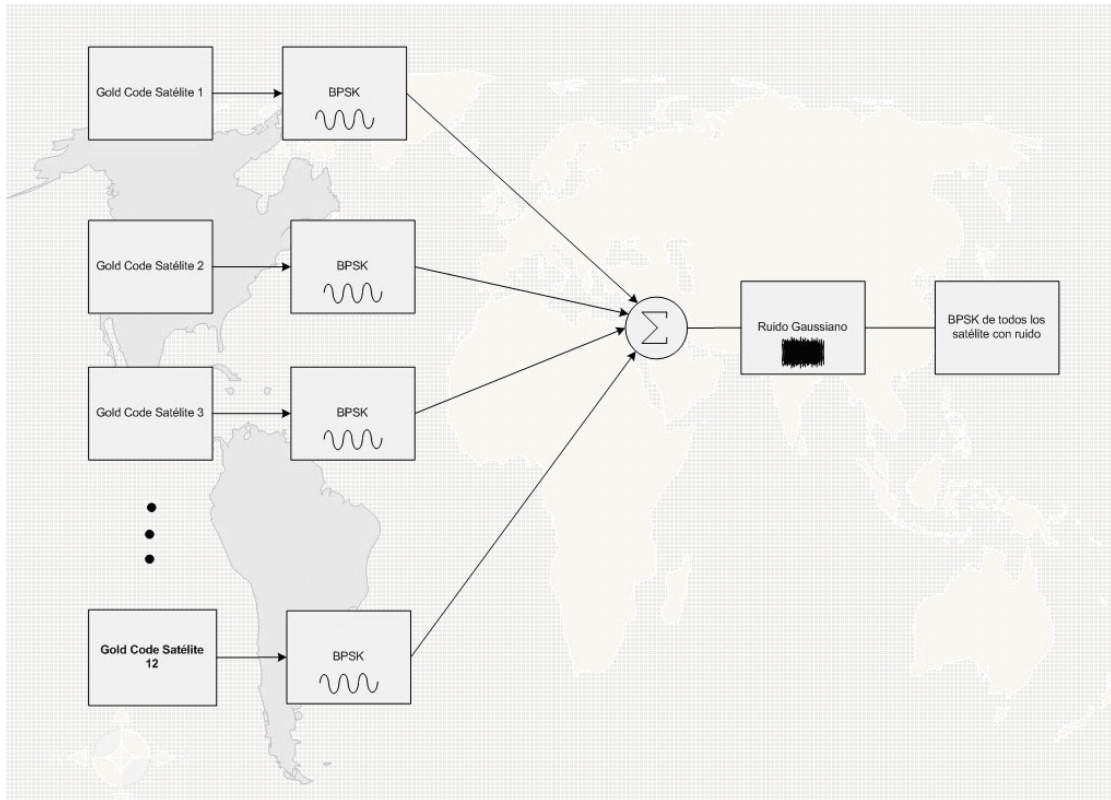


Figura 5.5 La señal modulada de todos los códigos, es sumada y se le añade ruido gaussiano

5.4 RESUMEN

En este capítulo hemos visto la generación de los códigos Gold y el comportamiento que éste presenta. También se vió la modulación BPSK de estos códigos los cuales fueron combinados junto con ruido para generar una señal que tenga la información de los 12 satélites.

CAPÍTULO 6

Recepción de Señales

6.1 INTRODUCCIÓN

En este capítulo se verá la técnica empleada para la detección de los códigos Gold, y el cálculo del desfaseamiento que este tiene con respecto a un código generado localmente. Para ello se hace uso de un Costas Locked Loop, ampliamente utilizado en comunicaciones para la recuperación de la portadora y la posterior demodulación de la señal entrante. También se verá el Delay Locked Loop para la determinación del desfaseamiento que sufre la señal desde que ha sido generada hasta la entrada a nuestro sistema.

6.2 DETECCIÓN

Para realizar la detección de los códigos Gold, la señal de entrada a la cual se le ha sumado ruido gaussiano, pasa a través de un filtro pasa bajas para delimitar el contenido espectral del ruido, posteriormente se envía a un CLL para recuperar la portadora con lo cual se demodula la señal, realizando una correlación cruzada.

La correlación cruzada se realiza para tener la certeza de que el código buscado existe en la señal analizada, para luego enviarla al DLL, el cual nos da como resultado el desfaseamiento que sufre nuestra señal recibida con la generada originalmente.

En el capítulo anterior se realizó la modulación de los códigos Gold los cuales son enviados desde los satélites GPS hacia el usuario en la tierra. Cuando la señal llega al receptor este tiene como función acondicionar la señal para luego pasarla por un filtro y disminuir el ruido blanco gaussiano, tanto en magnitud como en contenido espectral.

Para disminuir el ruido se emplea un filtro Butterworth de 5° orden, ya que, por su selectividad en la detección de satélites, arrojó mejores resultados en comparación con otros filtros Butterworth de menor orden.

La función de transferencia del filtro de 5° orden es la siguiente

$$H(s) = \frac{1}{1.0240e^3 s^5 + 0.8284e^3 s^4 + 0.3351e^3 s^3 + 0.0838e^3 s^2 + 0.0129e^3 s + 1e^{-3}} \quad (6.1)$$

La señal resultante del filtro anterior constituye la entrada a nuestro circuito CLL, teniendo como función, recuperar la portadora de la señal modulada en BPSK (Figura 6.1).

Para implementar el CLL, se toma como base el diagrama de flujo de la siguiente figura utilizando un filtro Butterworth de orden 3, este es ampliamente utilizado en aplicaciones inalámbricas y comunicación satelital [1] donde el efecto doppler esta presente. Se elige una condición inicial del VCO para iniciar el algoritmo que en este caso es cero.

La señal entrante se multiplica por su correspondiente en fase (rama I) y una desfasada 90 grados (rama Q), ambas son filtradas por separado, luego se multiplican entre sí, el resultado de esto es nuevamente filtrado y se envía al VCO, esta iteración se realiza 8183 veces por que es el producto de la longitud

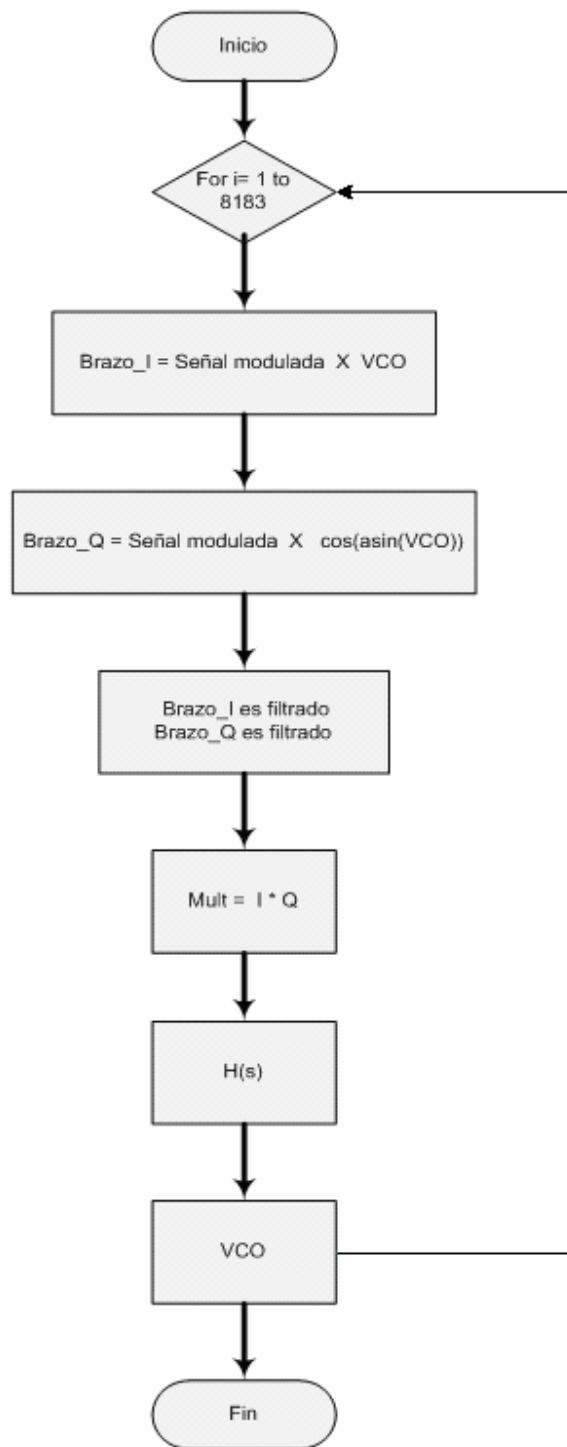


Figura 6.1 Diagrama de flujo del Costas

del código Gold por el número de muestras menos uno (con ocho muestras), de esta forma recuperamos la portadora.

Cuando se obtiene la portadora, ésta es multiplicada por la señal modulada dándonos como resultado la demodulación del BPSK. Sin embargo, en ésta etapa del procesamiento, el código binario no es fácilmente identificable por la cantidad de ruido que aun contiene, por lo que ésta señal aún tiene que pasar por un módulo de acondicionamiento, para que se obtenga una señal binaria bien definida.

La existencia de variaciones muy rápidas entre valores positivos y negativos, provoca incertidumbre en la detección de cada chip, por lo que se diseñó un algoritmo capaz de acondicionar la señal, de tal manera que sea posible su reconstrucción:

Primero se verifica que una muestra sea positiva, luego si la siguiente muestra es negativa y si la posterior a esta última también lo es entonces el valor es negativo, pero si la primera muestra es negativa, la segunda es positiva y la tercera es negativa entonces es negativa.

Se calcula una correlación cruzada mediante el par de transformadas de Fourier (Figura 6.2) para detectar si es el código buscado.

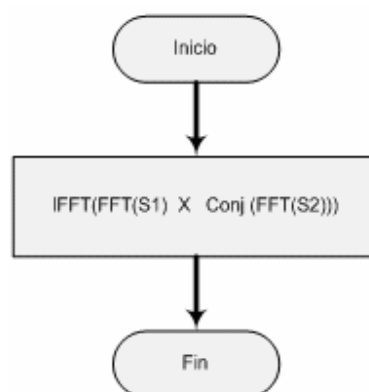


Figura 6.2 Diagrama de flujo de la correlación cruzada

La

Figura 6.3 representa el diagrama de flujo del DLL el cual se implementó de la siguiente manera:

Primero tenemos un código Gold seleccionado (buscado) como referencia, el cual es adelantado medio chip (recuérdese que un chip, en este caso, se compone de 8 muestras), también se pueden hacer adelantos de un cuarto u octavo de chip, pero esto implicaría mayor tiempo de ejecución para la adquisición del código.

El código adelantado se multiplica por el código demodulado. También hacemos algo similar pero en vez de adelantar se atrasa medio chip y se multiplica por el código demodulado.

Las señales resultantes de cada una de las multiplicaciones son sumadas y filtradas mediante un filtro Butterworth de segundo orden con la siguiente función de transferencia

$$H(s) = \frac{1}{16s^4 + 5.66s + 1} \quad (6.2)$$

Finalmente se aplica la mediana a los datos filtrados, y se almacena en un arreglo y se vuelve a repetir el adelanto y el atraso.

Posteriormente se compara punto por punto hasta encontrar el valor máximo de la mediana y el resultado es el desfaseamiento que tiene la señal entrante con la que tenemos de referencia.

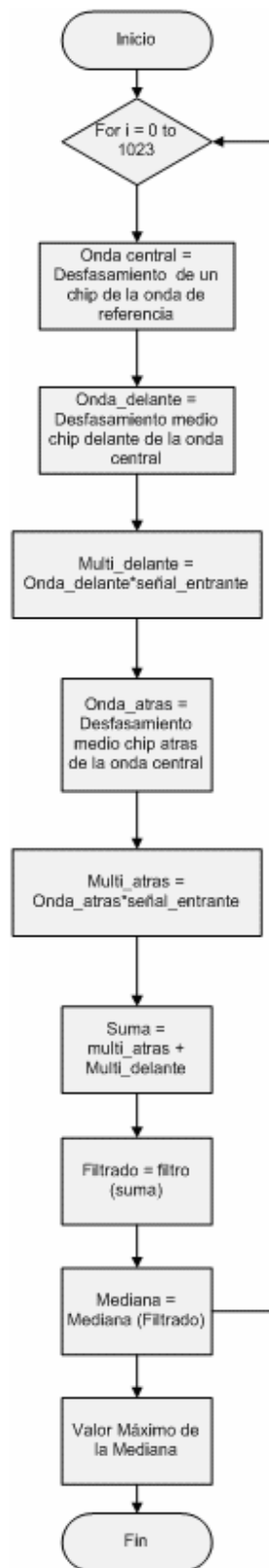


Figura 6.3 Diagrama de flujo del Delay Locked Loop

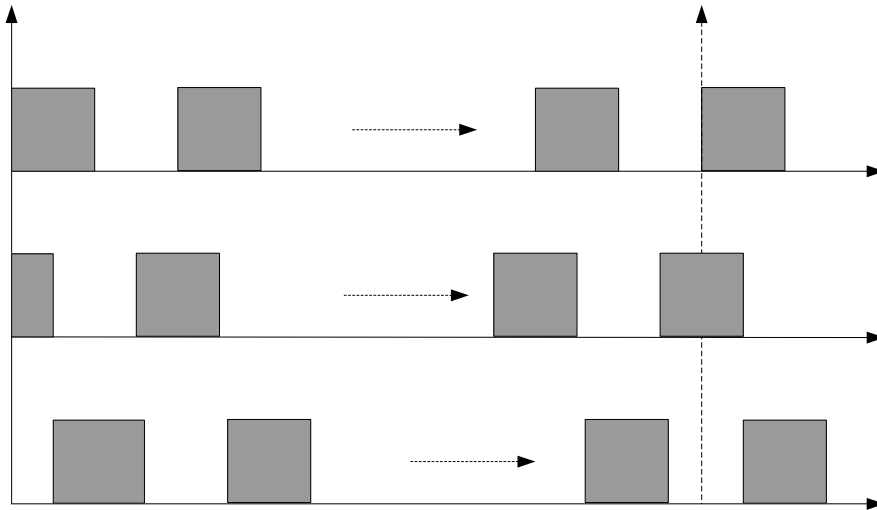


Figura 6.4 Referencia (a), atraso (b) y adelanto de la señal

La Figura 6.4 representa el comportamiento que tiene la señal cuando se adelanta o se atrasa medio chip.

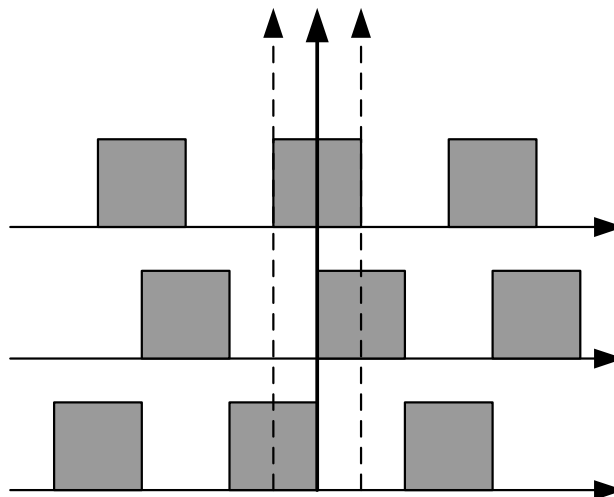


Figura 6.5 Toma de decisión en el límite del código

Para la realización del DLL se toma 1022 cálculos posibles, debido a que si se toma, x como referencia se observa que los pasos son de medio chip (Figura 6.5) entonces el último medio chip es el adelanto del inicio debido a que los códigos se repiten infinitamente por lo tanto solo se ejecutan 1022 cálculos.

6.3 FUNCIONAMIENTO EN CONJUNTO.

El sistema de detección e identificación funciona con base en el diagrama a bloques de la Figura 6.6 Inicia el CLL con la señal de los doce satélites y recupera la portadora. Con esa portadora se demodula y se realiza la correlación, después inicial el DLL el cual nos proporciona una estimación del tiempo de retardo que sufre la señal y que es precisamente el desfaseamiento descrito en la sección anterior

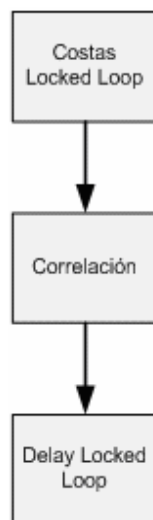


Figura 6.6 Sistema de adquisición e identificación

6.4 RESUMEN

Se realizó el algoritmo para la recuperación de la portadora mediante un CLL, el cual es ampliamente utilizado en el proceso de recuperación de la portadora en modulación BPSK. Se demodularon los datos a partir de la portadora recuperada para con ellos identificar el código Gold. Esto se logra con la correlación cruzada, si el código buscado, existe, éste se envía al DLL, el cual tiene como función calcular el desfaseamiento de la señal buscada con respecto a otra generada localmente.

CAPÍTULO 7

Resultados Experimentales

7.1 INTRODUCCIÓN

En este capítulo se verán los resultados obtenidos del sistema propuesto para la generación y detección de códigos Gold para 32 satélites GPS.

7.2 RESULTADO DE GENERACIÓN DEL CÓDIGO GOLD

Como se vio en el quinto capítulo, se desarrolló un algoritmo el cual arroja como resultado los códigos Gold para los 32 satélites que actualmente están en operación.

En la Figura 7.1 se observa el resultado de la generación del código Gold para el satélite 1.

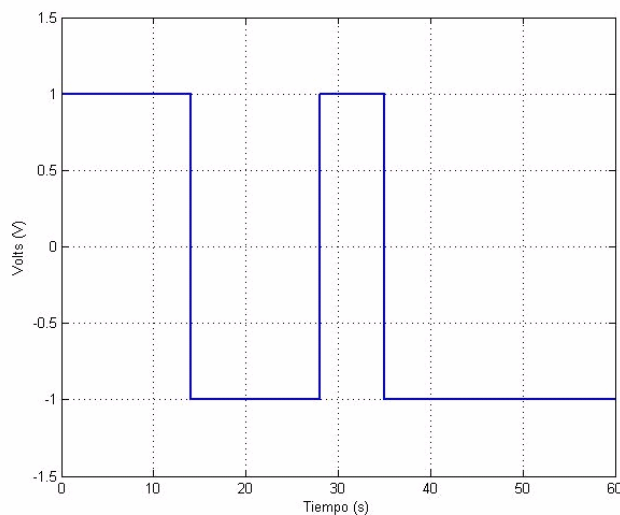


Figura 7.1 Fracción del código Gold generado del satélite 1

Este código Gold es modulado en BPSK con una portadora, tal que, ésta corresponde a un periodo de chip. La Figura 7.2 muestra la señal modulada resultante.

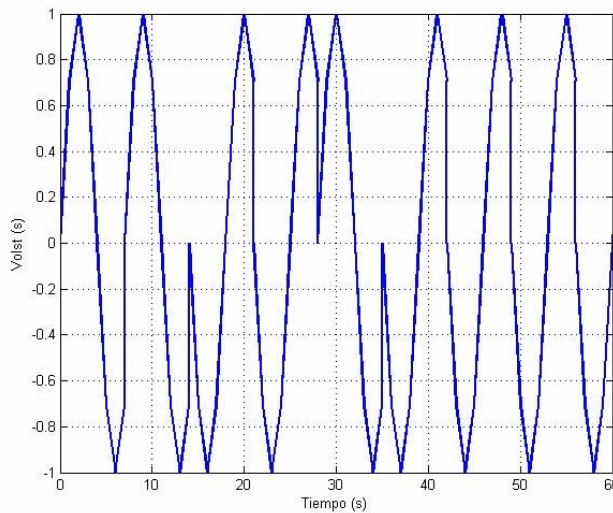


Figura 7.2 Modulación del código Gold del satélite 1

La señal tiene un valor de 1 Vpp y solo se muestran parcialmente los datos modulados para que se pueda tener una clara visualización, ya que la longitud total es de 8184 valores para un solo satélite.

Los receptores GPS solo pueden recibir la señal de doce satélites debido a la curvatura de la tierra en combinación con el número de satélites GPS, se define el número máximo permitido para un usuario.

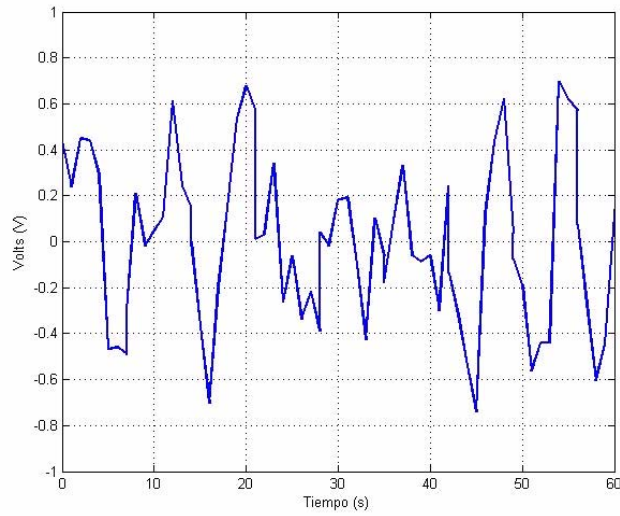


Figura 7.3 Señales de los doce satélites más ruido gaussiano

En la Figura 7.3 podemos ver la suma de estos doce satélites (modulados), más la suma del ruido blanco gaussiano, se eligió este tipo de ruido por ser el ruido más común y afecta en todas las frecuencias, la razón señal a ruido es de 0.09 dB. También se hizo un retardo por que las señales GPS son generadas en un tiempo local del satélite y al llegar a la tierra, éstas sufren un retardo con respecto al tiempo local del usuario (tierra).

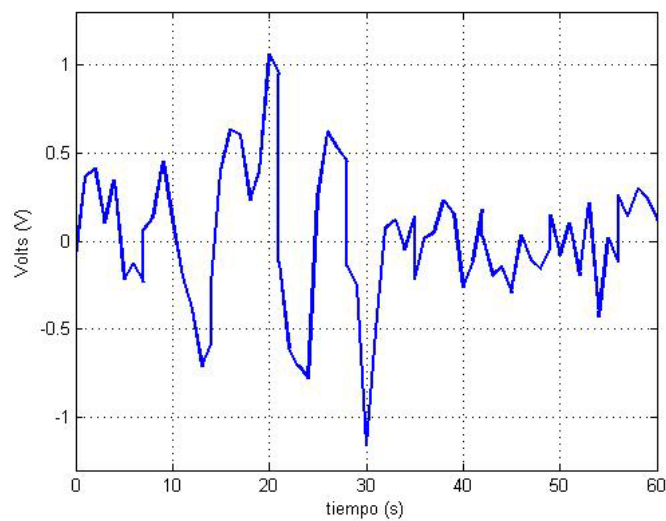


Figura 7.4 Señal con amplitudes diferentes de los códigos Gold

Cada satélite tiene una potencia diferente por su ubicación en cada instante, por lo que la señal de cada uno de los satélites varía en amplitud. La Figura 7.4 muestra la suma de doce satélites con amplitudes diferentes.

7.3 ADQUISICIÓN DEL CÓDIGO GOLD

La señal de los doce satélites entra al filtro con el fin de reducir el ruido y dejar pasar solo la frecuencia del código. En la Figura 7.5 se observa la señal ya filtrada.

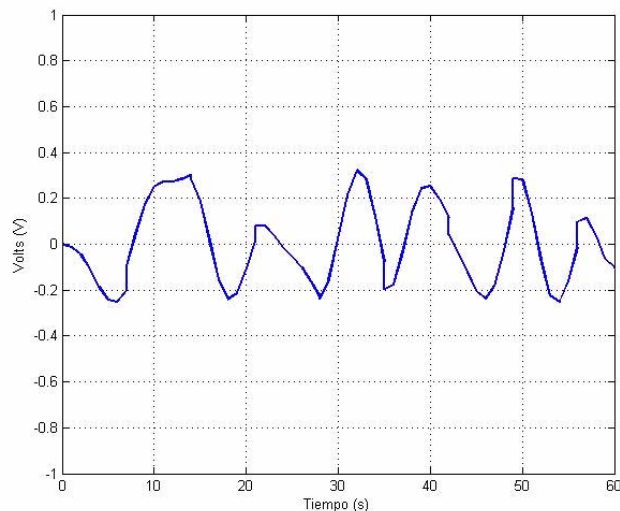


Figura 7.5 Primera etapa de la adquisición

La señal filtrada se envía al CLL, el cual como ya se mencionó anteriormente nos sirve para recuperar la portadora de la señal modulada en BPSK.

Al hacer una comparación de la salida del Costas con una portadora ideal, da como resultado un diagrama de residuos (Figura 7.6).

La siguiente gráfica indica que existe muy poca diferencia entre la portadora recuperada y la ideal. Las diferencias presentadas por la portadora recuperada

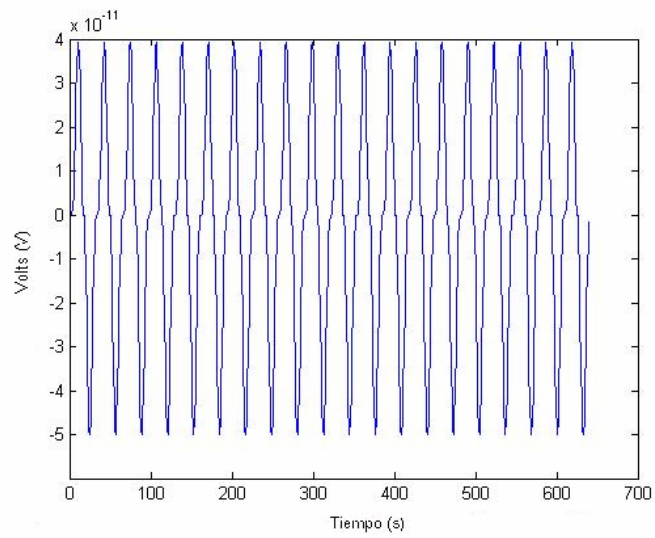


Figura 7.6 Diagrama de residuos

es del orden de 40×10^{-12} . En la gráfica no se observan todas las muestras ya que si se incluyeran no se podría distinguir su comportamiento.

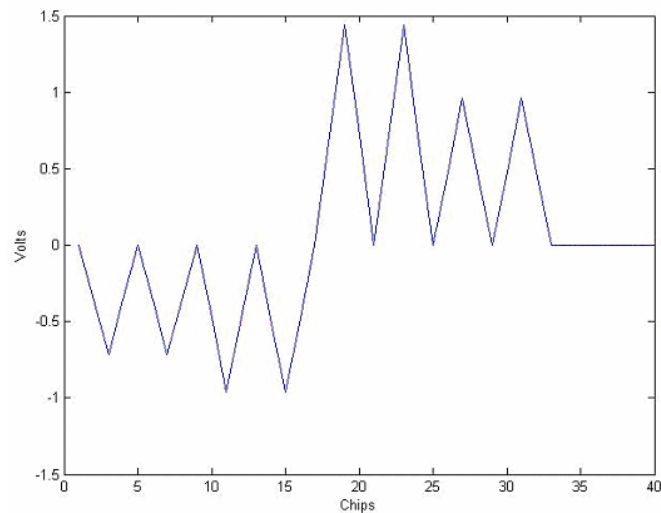


Figura 7.7 Salida del Costas

Después la señal se remodula, esto se logra multiplicando la señal entrante con la portadora recuperada, teniendo como resultado una señal como la presentada en la Figura 7.7

La señal demodulada se pasa al acondicionador de señal para que queden bien definidos los niveles digitales y podamos trabajar con ella ya que, como puede apreciarse en la figura anterior, ésta no es apropiada para el algoritmo del DLL.

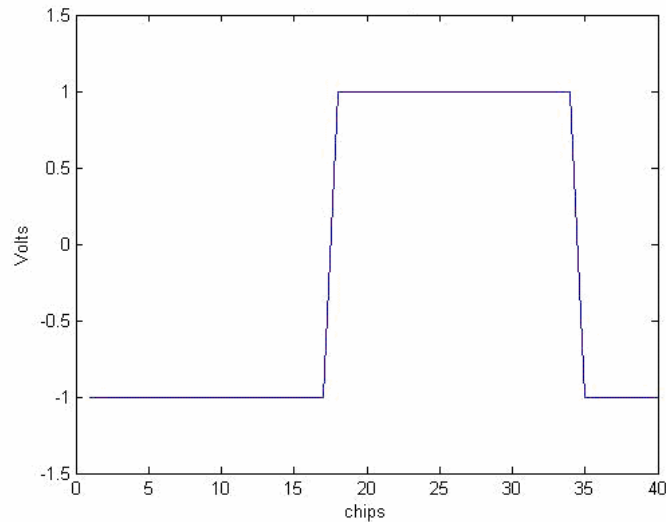


Figura 7.8 Código Gold Recuperado del satélite 1 con ID 2, 6

En la Figura 7.8 se observa el tren de pulsos recuperado después de la demodulación y el acondicionamiento de la señal. En éste instante la señal tiene los niveles de voltaje apropiados.

Como se vió en el capítulo cuatro, para la identificación de los códigos Gold se emplea la transformada de Fourier, con este fin se realizaron pruebas de correlación entre diferentes códigos Gold y la autocorrelación de uno de ellos.

Para el cálculo de la autocorrelación se utilizaron los datos del satélite 1 con código de selección 2, 6.

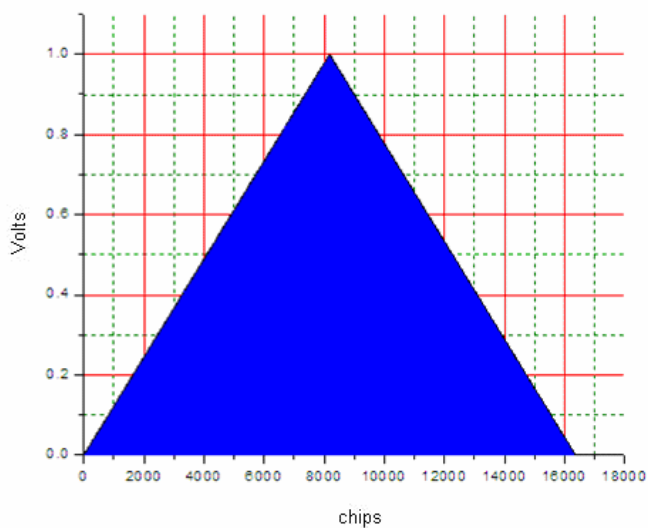


Figura 7.9 Función de autocorrelación del código Gold empleando el satélite 1.

En la gráfica (Figura 7.9) se observa la función de autocorrelación normalizada empleando 16 muestras por chip del satélite 1.

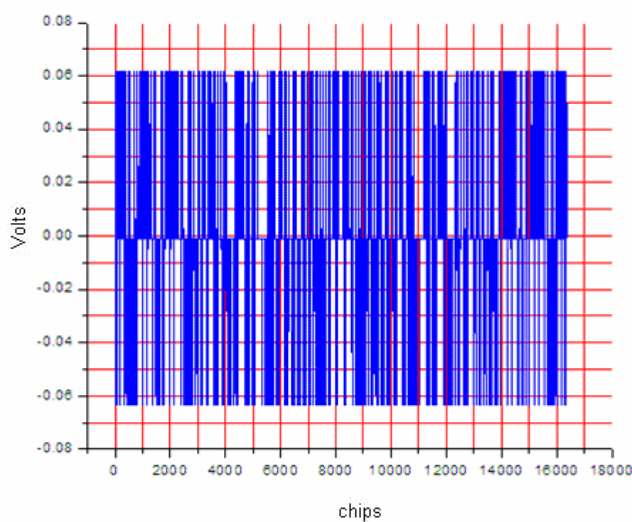


Figura 7.10 Correlación cruzada entre dos satélites diferentes

En esta tesis se calcula la correlación cruzada para la adquisición de los datos, bajo el siguiente esquema de decisión: si no existe correlación entre dos códigos los valores resultantes son casi cero para toda la secuencia, en el caso contrario, si existe una correlación los valores resultantes será un pico (máximo valor) y los demás valores son cercanos a cero.

Se puede apreciar en la Figura 7.10, que la correlación cruzada es de valores cercanos al cero, por lo que se puede aseverar que no existe una correlación entre los códigos Gold de satélites diferentes.

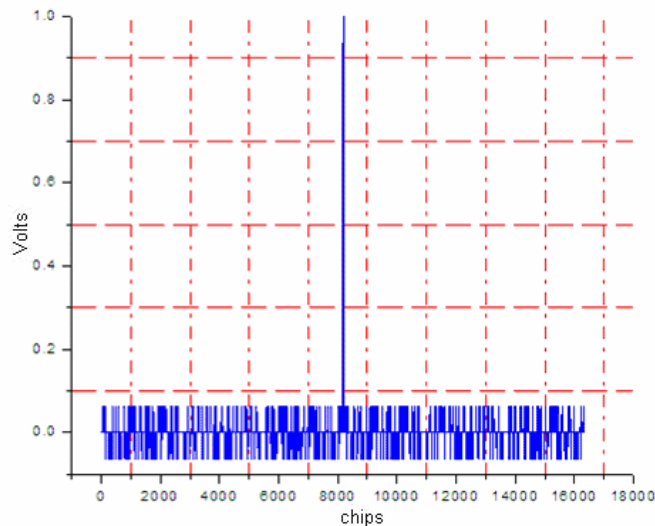


Figura 7.11 Autocorrelación del satélite 1 con 1023 chips a 16 muestras-chip

Caso contrario, si se realiza la correlación cruzada entre un mismo código entonces ahí si se tiene la autocorrelación, como se puede apreciar en la Figura 7.11. Para esta prueba se utilizaron 16 muestras por chip.

Recapitulando, para identificar una señal de códigos Gold se emplea la correlación cruzada. Una autocorrelación nos da un valor máximo, bien definido y más fuerte que la correlación cruzada (cuando dos señales son idénticas la correlación cruzada es la autocorrelación). Si los códigos son ortogonales, la correlación cruzada tiene un valor de cero o casi cero, esto último se cumple

con algunos códigos que no son ortogonales pero son cercanos a la ortogonalidad.

7.4 TIEMPO DE EJECUCIÓN

Se tomaron los tiempos de ejecución para CLL, el DLL y el tiempo de captura (Figura 7.12) el cual es el tiempo que le toma al sistema realizar todas las operación para cada uno de los códigos, en este caso, fueron 32 donde el promedio del tiempo de captura es de 18.2 segundos aproximadamente, esto nos lleva a que para 6 satélites el tiempo de captura es de 1.2 minutos aproximadamente. Con 4 satélites es condición suficiente para obtener la posición de un usuario.

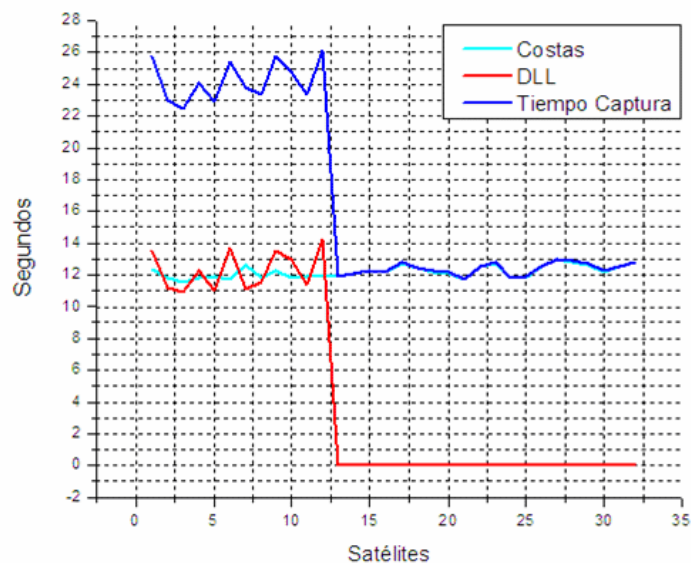


Figura 7.12 Se observa el tiempo de ejecución del DLL, Costas y el tiempo de captura del código

La Tabla 7.1 muestra el desempeño del sistema de adquisición de los códigos GPS

Tabla 7.1 Tiempo de ejecución de los circuitos

Número de satélite	Costas (s)	DLL (s)	Detección (s)
1	12.317	13.47	25.787
2	11.787	11.166	22.953
3	11.527	10.946	22.473
4	11.847	12.268	24.115
5	11.907	11.016	22.923
6	11.717	13.729	25.446
7	12.649	11.146	23.795
8	11.797	11.557	23.354
9	12.278	13.479	25.757
10	11.847	12.958	24.805
11	11.967	11.387	23.354
12	11.897	14.23	26.127
13	11.917	0.03	11.947
14	12.047	0.03	12.077
15	12.268	0.03	12.298
16	12.217	0.02	12.237
17	12.729	0.03	12.759
18	12.417	0.03	12.447
19	12.227	0.02	12.247
20	12.128	0.02	12.148
21	11.726	0.02	11.746
22	12.528	0.03	12.558
23	12.738	0.031	12.769
24	11.827	0.03	11.857
25	11.847	0.03	11.877
26	12.548	0.031	12.579
27	12.998	0.02	13.018
28	12.849	0.02	12.869
29	12.678	0.03	12.708
30	12.228	0.03	12.258
31	12.518	0.03	12.548
32	12.808	0.02	12.828

7.5 RESUMEN

En este capítulo se analizaron los resultados experimentales del generador de códigos Gold, CLL, el acondicionador de señal para el DLL y el tiempo de ejecución del sistema.

CAPÍTULO 8

Conclusiones y Trabajo Futuro

8.1 CONCLUSIONES

Se logró la identificación exitosa de las señales GPS, con una combinación de hasta 12 satélites, a diferentes niveles de recepción con un máximo de un sexto de su amplitud entre ellos.

Para demostrar el buen funcionamiento, se realizó un generador de códigos Gold de satélites GPS reales, modulando estos códigos para su posterior procesamiento.

Se logró el funcionamiento adecuado del Delay Locked Loop modificado para un procesador de señales GPS. La modificación del Delay Locked Loop consiste en emplear la mediana estadística con la cual podemos hacer los cálculos para la detección del desfase de la señal.

La mayoría de los GPS comerciales emplean 12 canales para realizar los cálculos de detección e identificación de los códigos Gold, en cambio se demostró que con este algoritmo solo es necesario un solo canal con lo cual se puede realizar un receptor, en el "back end", lo que resulta un GPS más barato y compacto.

Aparte de lo anterior, se tienen como contribuciones además un generador de códigos Gold y se logró poner en dominio público el funcionamiento propio de un receptor de satélite para la detección y decodificación de señales CDMA

empleando detección coherente al utilizar un Costas Locked Loop y un Delay Locked Loop

8.2 TRABAJOS FUTUROS

- Como trabajo futuro se propone realizar los programas tanto de generación, modulación, adquisición de datos, identificación y calculo de desfaseamiento originados por la tesis en un procesador digital de señales (DSP), por ejemplo TMS320C54x de la empresa Texas Instruments en la versión kit de desarrollo (DSP Starter Kits o DSK) de la misma empresa, ya que, ofrece las prestaciones requeridas de los programas por ejemplo la cantidad de memoria RAM y la velocidad de procesamiento, además tiene un software para programar los DSP's en lenguaje C.
- Evaluar el tiempo de ejecución de los DSP's (emisor y receptor).
- Analizar teóricamente el comportamiento del filtro de 5° orden con respecto al ruido en las señales GPS
- Implementar una interfaz gráfica para el generador de códigos Gold

BIBLIOGRAFÍA

- [1] Babu, R. & Wang, J. "Analysis of INS Derived Doppler Effects on Carrier Tracking Loop" School of Surveying and Spatial Information Systems School of Surveying and Spatial Information Systems.Sydney. 2004.

- [2] Dixon.R.C. "Spread spectrum system", edit. Wiley & Sons. 1976.

- [3] El-Rabdney, A. "Introduction to GPS The Global Positioning System", edit. Artech House. 2002.

- [4] French, G. "Understanding the GPS", edit Geo research. 1996.

- [5] Gold.R "Optimal binary sequences of spread spectrum multiplexing". IEEE Trans. Info. Theory. Páginas 619-621, octubre de 1967.

- [6] Grover, R. "Introduction to random signal analysis and kalman filtering", edit. Wiley & Sons. 1983.

- [7] Lee, B "Scrambling Techniques for CDMA Communications", edit kluwer. 2002.

- [8] Lee, S. Jhong & . Miller, E. Leonard. "CDMA systems engineering handbook". Edit. Artech house, 1998.

- [9] Mertins, A "Signal Analisis: Wavelets, Filter Banks, Time-Frequency transforms and applications", edit Wiley & Sons. 1999.

- [10] Sheriff, R. E & Hu Y. F. "Mobile Satellite Communication Networks". Edit. Wiley & Sons. 2001.

- [11] Spilker, J. J., "GPS signal structure and performance characteristics",
Navegation. Institute of Navegation, vol. 25, no, 2, páginas 121-146,
verano 1978.
- [12] Spilker, J .J."Digital communication by satellite",edit. Prentice hall. 1977.
- [13] Tomasi, W. "Sistemas de comunicaciones electrónicas", edit.
Pearson,2003.
- [14] Torrieri, D. "Principles of spread spectrum communication systems", edit.
Springer. 2005.
- [15] Tsui, J. "Fundamental of global positioning system receivers",edit. Wiley
& Sons. 2000.
- [16] Vaseghi, S. "Advanced digital signal processing and noise reduction",
edit. Wiley & Sons. 2000.
- [17] Xion, F. "Digital modulation techniques", edit. Artech House. 2000.
- [18] Leica Systems, <http://www.leica-geosystems.com>
- [19] Magellan (2006),
<http://www.magellangps.com/products/product.asp?segID=355>
- [20] Texas Instruments (2006), <http://www.ti.com/>
- [21] United States Patent and Trademark Office (2007),
<http://www.uspto.gov/index.html>

Anexo 1

Programas para la realización de la tesis.

PROGRAMA PRINCIPAL

```
%*****
%*****Programa para generar Códigos Gold de satélites GPS*****
%*****lo modula, demodula en BPSK y realiza la adquisición*****
%*****de los datos calculando el desfaseamiento de este*****
%*****
%*****
%***** Análisis, diseño y Programación: *****
%*****Sirgei Garcia Ballinas*****
%*****

clear; clc;
respuesta=zeros(1,2);
num_bits_sat=zeros(1,1);
num_bits_sat(1,1)=1023;
muestras_chip(1,1)=8;
fprintf('*****\n');
fprintf('*Programa que calcula el desfaseamiento de código C/A*\n');
fprintf('*      de satélites GPS      *\n');
fprintf('* Solo satélites del 1 al 12 se pueden calcular *\n');
fprintf('*****\n');
codigos=load('codigos_sat.dat'); %cargamos todos los códigos de los satélites
for i=1:32
fprintf('Satélite: %1.0f   codigo1: %1.0f   codigo2: %1.0f \n',codigos(i,1,1),codigos(i,2,1),
codigos(i,3,1))
end
fprintf('*****\n');
fprintf('Qué código quieres ?\n');
respuesta(1,1)=input(' ejemplo: 2 6 \n Valor 1: ');
respuesta(1,2)=input('Valor2: ');
```

```

fprintf('Cuantas muestras de desfaseamiento');
desfaseamiento=input(' ? : '); %en muestras
%*****
%*****
satelite_ID=genera_pn(num_bits_sat,respuesta); %función que generadora de C/A del satélite
%*****
% Código y modulación BPSK de satélite elegido
%*****
valor_seno=modulado_bpsk2(satelite_ID,muestras_chip,num_bits_sat);
modulado_bpsk=valor_seno(1,:); %señal modulada
portadora_bpsk=valor_seno(2,:); %portadora
onda_referencia=valor_seno(3,:); %valor del tren de pulsos
tiempo=valor_seno(4,:); %tiempo t de la señal
%*****
%Códigos de satélites capturados
satelite_aleatorio0(1,1)=codigos(1,2,1); satelite_aleatorio0(1,2)=codigos(1,3,1); %satélite 1
satelite_aleatorio1(1,1)=codigos(2,2,1); satelite_aleatorio1(1,2)=codigos(2,3,1); %satélite 2
satelite_aleatorio2(1,1)=codigos(3,2,1); satelite_aleatorio2(1,2)=codigos(3,3,1); %satélite 3
satelite_aleatorio3(1,1)=codigos(4,2,1); satelite_aleatorio3(1,2)=codigos(4,3,1); %satélite 4
satelite_aleatorio4(1,1)=codigos(5,2,1); satelite_aleatorio4(1,2)=codigos(5,3,1); %satélite 5
satelite_aleatorio5(1,1)=codigos(6,2,1); satelite_aleatorio5(1,2)=codigos(6,3,1); %satélite 6
satelite_aleatorio6(1,1)=codigos(7,2,1); satelite_aleatorio6(1,2)=codigos(7,3,1); %satélite 7
satelite_aleatorio7(1,1)=codigos(8,2,1); satelite_aleatorio7(1,2)=codigos(8,3,1); %satélite 8
satelite_aleatorio8(1,1)=codigos(9,2,1); satelite_aleatorio8(1,2)=codigos(9,3,1); %satélite 9
satelite_aleatorio9(1,1)=codigos(10,2,1); satelite_aleatorio9(1,2)=codigos(10,3,1); %satélite 10
satelite_aleatorio10(1,1)=codigos(11,2,1); satelite_aleatorio10(1,2)=codigos(11,3,1);%satélite 11
satelite_aleatorio11(1,1)=codigos(12,2,1); satelite_aleatorio11(1,2)=codigos(12,3,1);%satélite 12
%*****
%Códigos Gold de satélites capturados
%*****
satelite_ID_aleatorio0=genera_pn(num_bits_sat,satelite_aleatorio0);
satelite_ID_aleatorio1=genera_pn(num_bits_sat,satelite_aleatorio1);
satelite_ID_aleatorio2=genera_pn(num_bits_sat,satelite_aleatorio2);
satelite_ID_aleatorio3=genera_pn(num_bits_sat,satelite_aleatorio3);
satelite_ID_aleatorio4=genera_pn(num_bits_sat,satelite_aleatorio4);
satelite_ID_aleatorio5=genera_pn(num_bits_sat,satelite_aleatorio5);
satelite_ID_aleatorio6=genera_pn(num_bits_sat,satelite_aleatorio6);
satelite_ID_aleatorio7=genera_pn(num_bits_sat,satelite_aleatorio7);

```

```

satelite_ID_aleatorio8=genera_pn(num_bits_sat,satelite_aleatorio8);
satelite_ID_aleatorio9=genera_pn(num_bits_sat,satelite_aleatorio9);
satelite_ID_aleatorio10=genera_pn(num_bits_sat,satelite_aleatorio10);
satelite_ID_aleatorio11=genera_pn(num_bits_sat,satelite_aleatorio11);
%*****
%*****Inicio suma de satélites*****
%*****
valor_seno_aleatorio0=modulado_bpsk2(satelite_ID_aleatorio0,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio0=valor_seno_aleatorio0(1,:);
valor_seno_aleatorio1=modulado_bpsk2(satelite_ID_aleatorio1,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio1=valor_seno_aleatorio1(1,:);
valor_seno_aleatorio2=modulado_bpsk2(satelite_ID_aleatorio2,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio2=valor_seno_aleatorio2(1,:);
valor_seno_aleatorio3=modulado_bpsk2(satelite_ID_aleatorio3,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio3=valor_seno_aleatorio3(1,:);
valor_seno_aleatorio4=modulado_bpsk2(satelite_ID_aleatorio4,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio4=valor_seno_aleatorio4(1,:);
valor_seno_aleatorio5=modulado_bpsk2(satelite_ID_aleatorio5,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio5=valor_seno_aleatorio5(1,:);
valor_seno_aleatorio6=modulado_bpsk2(satelite_ID_aleatorio6,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio6=valor_seno_aleatorio6(1,:);
valor_seno_aleatorio7=modulado_bpsk2(satelite_ID_aleatorio7,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio7=valor_seno_aleatorio7(1,:);
valor_seno_aleatorio8=modulado_bpsk2(satelite_ID_aleatorio8,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio8=valor_seno_aleatorio8(1,:);
valor_seno_aleatorio9=modulado_bpsk2(satelite_ID_aleatorio9,muestras_chip,num_bits_sat);m
odulado_bpsk_aleatorio9=valor_seno_aleatorio9(1,:);
valor_seno_aleatorio10=modulado_bpsk2(satelite_ID_aleatorio10,muestras_chip,num_bits_sat);
modulado_bpsk_aleatorio10=valor_seno_aleatorio10(1,:);
valor_seno_aleatorio11=modulado_bpsk2(satelite_ID_aleatorio11,muestras_chip,num_bits_sat);
modulado_bpsk_aleatorio11=valor_seno_aleatorio11(1,:);
%Normalizamos la suma para tener 1Vpp
modulado_bpsk_todos=.0833*(modulado_bpsk_aleatorio0+modulado_bpsk_aleatorio1+modula
do_bpsk_aleatorio2+modulado_bpsk_aleatorio3+modulado_bpsk_aleatorio4+modulado_bpsk_a
leatorio5+modulado_bpsk_aleatorio6+modulado_bpsk_aleatorio7+modulado_bpsk_aleatorio8+
modulado_bpsk_aleatorio9+modulado_bpsk_aleatorio10+modulado_bpsk_aleatorio11);
[b,a] = butter(5,2/muestras_chip); %Coeficientes del filtro
%*****

```

```

%*****desfasamos*****
modulado_bpsk_todos=circshift(modulado_bpsk_todos,[0,desfasamiento]);
%*****
modulado_bpsk_todos=awgn(modulado_bpsk_todos,.09,'measured'); %Ruido AWGN
modulado_bpsk_todos=filter(b,a,modulado_bpsk_todos); %Filtramos la señal
% *****
% *****Identificamos*****
%*****
res_costas_dll2=dll_pll_ver_registro(modulado_bpsk_todos,num_bits_sat,muestras_chip,onda_r
eferencia);

```

PROGRAMA PARA GENERAR LOS CÓDIGOS GOLD

```

%función generadora de códigos Gold
%*****
function satelite_ID=genera_pn(num_bits_sat,respuesta)
%generamos G1 y G2 con valor inicial 1 en todos los eventos
%global satélite _id
G1=(ones(1,10));
G2=(ones(1,10));
satelite_ID=(zeros(1,num_bits_sat(1,1))); %creamos un vector para el ID del satélite
G1_final=(zeros(1,10)); %creamos un arreglo inicial con cero para G1 y G2 finales
G2_final=(zeros(1,10));
for i=1:num_bits_sat(1,1), %lo corremos el numero de bits indicados por el usuario
%calcula para G1
G1_final(1,i)=G1(1,10);
modulo_2_G1=xor(G1(1,3),G1(1,10)); %hacemos Xor entre evento 3 y 10
G1=circshift(G1, [0, 1]); %corrimiento a la derecha de G1
G1(1,1)=modulo_2_G1; % ponemos en G1(1,1) feedback el valor del modulo-2
%calcula para G2
G2_final(1,i)=G2(1,10);
modulo_2_G2=xor(G2(1,2),G2(1,3)); %calculamos el Xor de evento 2,3,6,8,9,10
modulo_2_G2=xor(modulo_2_G2,G2(1,6));
modulo_2_G2=xor(modulo_2_G2,G2(1,8));
modulo_2_G2=xor(modulo_2_G2,G2(1,9));
modulo_2_G2=xor(modulo_2_G2,G2(1,10));

```

```

G2_selec=xor(G2(1,respuesta(1,1)),G2(1,respuesta(1,2)));%selección del ID del satélite en G2
satelite_ID(1,i)=xor(G2_selec,G1_final(1,i)); %resultado del ID del satélite
G2=circshift(G2, [0, 1]);          %corrimiento a la derecha de G2
G2(1,1)=modulo_2_G2;              %insertamos el modulo-2 en G2(1,1)
end

```

PROGRAMA PARA MODULAR EN BPSK LOS CÓDIGOS GOLD

```

%realizamos el modulado BPSK del satelite_ID
%*****
function valor_seno=modulado_bpsk(satelite_ID,muestras_chip,num_bits_sat)
val_chip=ones(1,muestras_chip);
val_tabulados=[0];
for n=1:muestras_chip-1,          %generamos el arreglo de valores para las muestras del chip
    val_tabulados_temp=[n*1*2*pi/(muestras_chip-1)];
    val_tabulados=[val_tabulados val_tabulados_temp];
end
tamano_chip=size(val_chip);
tamano_satelite_ID=size(satelite_ID);
%*****
%% Arreglo del tiempo
%*****
tiempo=linspace(0,tamano_chip(2)*tamano_satelite_ID(2)-
1,tamano_chip(2)*tamano_satelite_ID(2)); %generamos vector
tamano_tiempo=size(tiempo);
cont=1;
for ii=1:tamano_tiempo(2)-1,
    cont=cont+1;
    if cont==tamano_chip(2)+1
        tiempo(ii+1)=tiempo(ii);
        cont=1;
    else
        tiempo(ii+1)=tiempo(ii)+1;
    end
end
end
%*****
% Fin Arreglo del tiempo
%*****

```

```

val_tabulados=zeros(1,num_bits_sat*muestras_chip);
val_tabulados_satelite=zeros(1,num_bits_sat*muestras_chip);
n=0;
i=1;
cont_sat=1;
for ii=1:num_bits_sat,
    for iii=1:muestras_chip,
        if satellite_ID(1,ii)==1
            val_tabulados_satelite(1,cont_sat)=1;
            val_tabulados(1,cont_sat)=(n*1*2*pi/(muestras_chip)+0); %modulado
            val_tabulados_portadora(1,cont_sat)=(n*1*2*pi/(muestras_chip)+0);
            cont_sat=cont_sat+1;n=n+1;
        else val_tabulados_satelite(1,cont_sat)=-1;
            val_tabulados(1,cont_sat)=(n*1*2*pi/(muestras_chip-0)+pi); %modulado
            val_tabulados_portadora(1,cont_sat)=(n*1*2*pi/(muestras_chip)+0);
            cont_sat=cont_sat+1;n=n+1;
        end %del if
    end %del for interno
end %del for externo
prueba_modulado=sin(val_tabulados); %señal de RF
prueba_modulado_portadora=sin(val_tabulados_portadora);
%portadora_bpsk=sin(valor_portadora);
valor_seno(1,:)= prueba_modulado; %ponemos el modulado y portadora
valor_seno(2,:)=prueba_modulado_portadora; %valor de la portadora
valor_seno(3,:)=val_tabulados_satelite;
valor_seno(4,:)=tiempo; %el tiempo t de la señal
end

```

PROGRAMA PARA EL COSTAS LOCKED LOOP Y EL DELAY LOCKED LOOP

```

%Costas y el DLL-Med
function
res_costas_dll=dll_pll_sal_ver_registro(modulado_bpsk,num_bits_sat,muestras_chip,onda_referencia)
%inicia costas
[b,a] = butter(5,2/muestras_chip);

```

```

vco_sal(1,1)=0;    % condición inicial;
frecuencia_NCO=1*2*pi/(muestras_chip);
for i=1:num_bits_sat*muestras_chip-1
    multi_I_branch(1,i)=modulado_bpsk(1,i)*vco_sal(1,i);
    multi_Q_branch(1,i)=modulado_bpsk(1,i)*cos(asin(vco_sal(1,i)));
    sal_filtro_I_branch(1,i)=filter(b,a,multi_I_branch(1,i));
    sal_filtro_Q_branch(1,i)=filter(b,a,multi_Q_branch(1,i));
    multi_IQ_branch(1,i)=sal_filtro_I_branch(1,i)*sal_filtro_Q_branch(1,i);
    entrada_vco(1,i)=filter(b,a,multi_IQ_branch(1,i));
    vco_sal(1,i+1)=sin(i*frecuencia_NCO+.5*entrada_vco(1,i)); %salida costas
end
%fin costas
salida_datos=vco_sal.*modulado_bpsk; %demdulacion
% acondicionamos la señal
for i=1:num_bits_sat*muestras_chip-3
    if salida_datos(1,i)>0 & salida_datos(1,i+1)<0 & salida_datos(1,i+2)>0
        salida_datos(1,i+1)= -salida_datos(1,i+1);
    elseif salida_datos(1,i)<0 & salida_datos(1,i+1)>0 & salida_datos(1,i+2)<0
        salida_datos(1,i+1)= -salida_datos(1,i+1);
    end
end
if          salida_datos(1,num_bits_sat*muestras_chip-2)>0          &
salida_datos(1,num_bits_sat*muestras_chip-1)<0          &
salida_datos(1,num_bits_sat*muestras_chip)>0
    salida_datos(1,i+1)= -salida_datos(1,i+1);
elseif          salida_datos(1,num_bits_sat*muestras_chip-2)<0          &
salida_datos(1,num_bits_sat*muestras_chip-1)>0          &
salida_datos(1,num_bits_sat*muestras_chip)<0
    salida_datos(1,i+1)= -salida_datos(1,i+1);
end
%decision de 1 o -1
for i=1:num_bits_sat*muestras_chip
    %datos de salida
    if salida_datos(1,i)>0.0
        onda_para_correlacionar(1,i)=1;
    else
        onda_para_correlacionar(1,i)=-1;
    end
end

```



```

end
%inicio DLL
cont_aux=0; %control del número de chips
cont=1;
salto=0; %incremento de muestras del chip o sea saltamos un chip
dll_temp=zeros(1,num_bits_sat);
[b,a] = butter(2,2/muestras_chip);
%inicia correlación
resul_corr=correlacion(onda_referencia,onda_para_correlacionar,num_bits_sat,muestras_chip);
maxima_corr=max(abs(resul_corr));
if maxima_corr>=0.12
    fprintf('SATELITE ENCONTRADO\n');
    for i=0:num_bits_sat
        senal_entrante=onda_para_correlacionar; %onda que llega
        onda_referencia_central=circshift(onda_referencia, [0,salto]);
        onda_referencia_adelante=circshift(onda_referencia_central, [0,muestras_chip/2]);
        multi_adelante=senal_entrante.*onda_referencia_adelante;
        onda_referencia_atras=circshift(onda_referencia_central, [0,-muestras_chip/2]);
        multi_atras=senal_entrante.*onda_referencia_atras;
        suma=multi_adelante+multi_atras;
        sal_filtro=2*(filter(b,a,suma));
        dll_temp(1,cont)=median(sal_filtro);      %salida
        cont=cont+1;
        salto=salto+muestras_chip;
        cont_aux=cont_aux+1;
    end
    dll_temp=abs(dll_temp);
    maximo_valor=max(dll_temp);
    indice=0;
    cont=0;
    while(indice<1)
        cont=cont+1;
        if dll_temp(1,cont)==maximo_valor
            indice=2;
        end
    end
    fprintf('satelite desfasado %6.2f chips \n',cont-2)
else

```

```

fprintf('NO SE ENCONTRO EL SATELITE \n')
cont=0.0;
end
%buscamos el valor máximo en el DLL
res_costas_dll=dll_temp;
res_costas_dll=[tiempo_costas tiempo_ejecucion_DLL maxima_corr];

```

PROGRAMA PARA LA CORRELACIÓN

```

%función para realizar una correlación
function
correlacion=correlacion(onda_referencia,onda_para_correlacionar,num_bits_sat,muestras_chip)
cont_aux=num_bits_sat*muestras_chip;
correlacion_der=zeros(1,cont_aux);
correlacion_izq=zeros(1,cont_aux);
correlacion=zeros(1,2*cont_aux-2*muestras_chip-1);
cont=1;
correlacion_der=(1/cont_aux)*ifft(fft(onda_referencia,cont_aux).*conj(fft(onda_para_correlaciona
r,cont_aux)));
for i=1:cont_aux
correlacion_izq(1,i)=correlacion_der(cont_aux+1-i);
end
for i=muestras_chip:((cont_aux*2)-muestras_chip)
if i<cont_aux+1
correlacion(1,cont)=correlacion_izq(1,i);
else
correlacion(1,cont)=correlacion_der(1,i-cont_aux+1);
end
cont=cont+1;
end
end
end

```