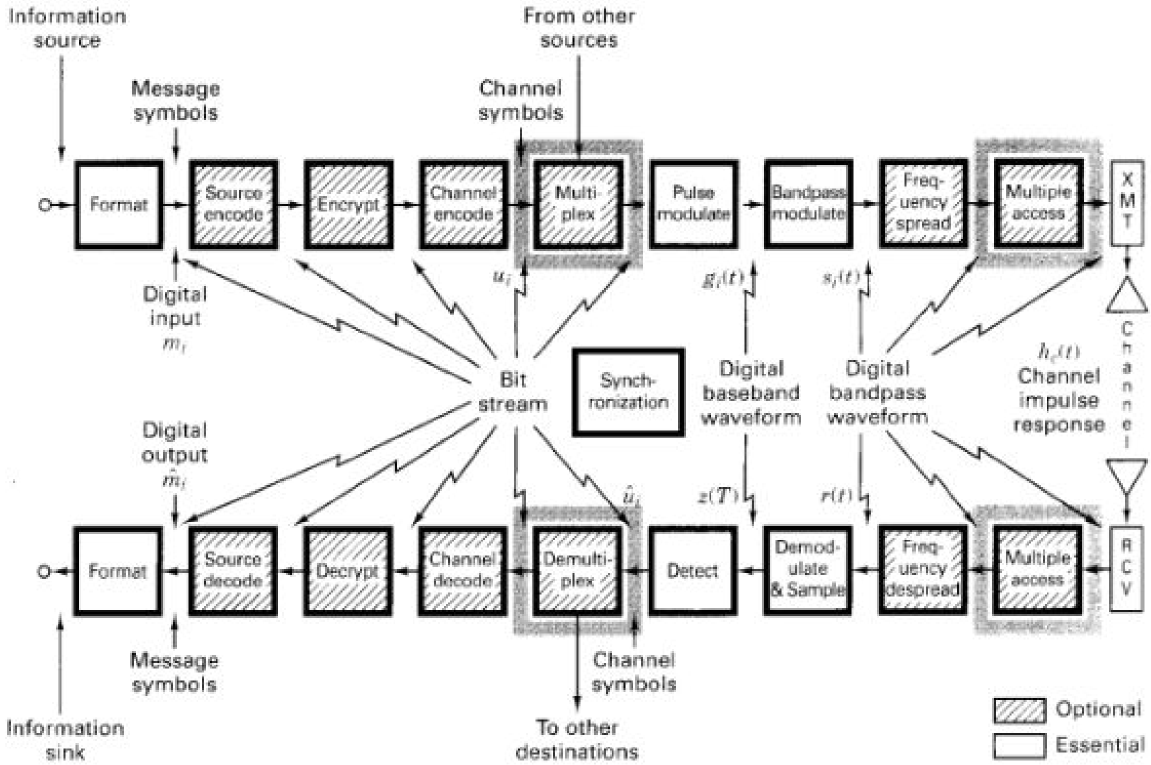
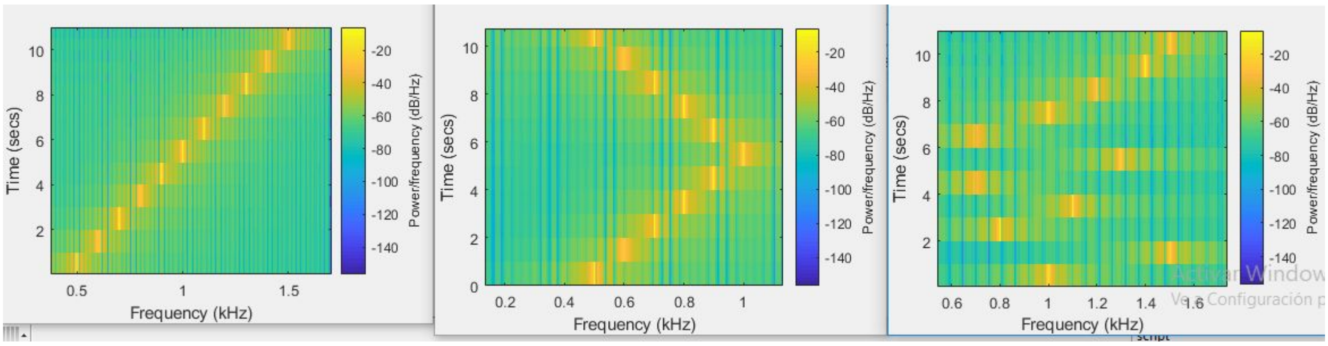
Métodos de acceso

Capítulo 3



Los métodos de acceso por división de código (CDMA) se basan en la utilización de técnicas de espectro ensanchado para que múltiples usuarios puedan hacer uso del canal de comunicaciones en todo momento sin ningún tipo de interferencia. La propuesta básica de estos esquemas de acceso consiste en esconder la información en un ancho de banda mucho más grande del que es necesario para cualquier esquema de comunicación digital convencional con la intención de prevenir interferencias. ¿Cuál es la situación ideal? La utilización de un ancho de banda infinito. Si esto fuera posible, significa que las waveform que se hacen presentes en el canal de comunicaciones luego del proceso de expansión tienen una densidad espectral de potencia igual a la de un ruido blanco, constante en toda la banda de frecuencias y diferente de cero. Entonces, como el proceso de expansión genera estas características sobre las señales puede concluirse que el proceso de expansión es un proceso aleatorio de ruido blanco. ¿Cómo es esto posible? Las técnicas de espectro ensanchado y precisamente las relacionadas a Frecuency hopping permiten entender de forma sencilla el concepto. Para que la información que quiere transmitir la fuente no sea interceptada por usuarios no autorizados, debe cambiar la waveform a transmitir a través del tiempo. Estos cambios están asociados a saltos en frecuencia. Es decir, la información modulada mediante un esquema de MFSK, donde se conoce las frecuencias asignables a las waveform en función de los mensajes digitales ya no estarán presentes en todo momento en la misma ubicación espectral. Saltará a otras frecuencias producto de una modulación extra incorporada al sistema. ¿Qué valores asumirá? Aquí es donde tiene influencia la secuencia aleatoria de ruido blanco planteada como base de las

técnicas de espectro ensanchado. Si fuera posible generar una secuencia totalmente aleatoria de saltos de frecuencia, con una cantidad infinita de frecuencias distintas, se tendría un proceso de ruido blanco producto de frecuency hopping. En esta situación, ¿Podría un jammer arruinar las comunicaciones entre transmisor y receptor? No. La razón fundamental de esta respuesta está dada en que al ser un proceso de ruido blanco el que está detrás de la operación de expansión no permite determinar en tiempos posteriores cuales van a ser los valores asignables en frecuencia, ya que todos los saltos son equiprobables. Esta información puede observarse en la transformada inversa de fourier de la función densidad espectral de potencia de ruido, definida como autocorrelación.
Esta última está conformada por un único impulso para τ=0, lo cual refleja matemáticamente el concepto ilustrado anteriormente. Por lo que al analizar la secuencia de ruido blanco, no puede decirse absolutamente nada de las muestras futuras a partir de las muestras actuales, porque la autocorrelación es prácticamente nula.
Así como el jammer no puede conocer las frecuencias de hopping para insertar potencia en el canal y dificultar la detección del receptor, este último no puede generar el proceso inverso sin ayuda del transmisor. Lo que debería realizar el receptor es el conocido proceso de “despreding”, lo cual consiste en generar localmente todas las secuencias de salto en frecuencia asignadas por el modulador. ¿Significa que en el receptor tiene que generarse nuevamente un proceso aleatorio de ruido blanco? Si y como precisamente se trata de un proceso aleatorio cuya función de distribución de probabilidad es uniforme, no existe ninguna alternativa que el mismo proceso aplicado en transmisor y receptor al mismo tiempo tengan iguales resultados. ¿Cómo es esto? Habría que imaginar, un transmisor y un receptor cada uno con un dado de infinitas caras, donde todas ellas son equiprobables y para ponerse de acuerdo de que frecuencia de hopping van a utilizar para transportar la información, lanzan cada uno su dado. ¿Qué es lo más probable? Que los resultados sean diferentes, existen muy pocas probabilidades que ambos procesos aleatorios arrojen el mismo resultado y mucho menos mantener dicha coincidencia entre resultados de los dados a lo largo del tiempo. Entonces, no puede utilizarse un proceso de ruido blanco para realizar espectro ensanchado. Sin embargo, esta es una de las razones. La otra, resulta estar totalmente relacionada al ancho de banda. Al no tener capacidad de utilizar un ancho de banda infinito, el proceso que se genera es ruido blanco gaussiano, porque si bien resulta plano en su densidad espectral de potencia, no lo es para todas las frecuencias. Está limitado en banda. Producto de esta limitación, no tiene capacidad de cambiar en intervalos pequeños de tiempo con la misma probabilidad entre todos los valores. Estos resultados están presentes matemáticamente en la función de autocorrelación, ya que se trata de una función sinc.
A los procesos aleatorios de ruido blanco gaussiano utilizados como generadores de expansión en frecuencia se los denomina Códigos Pseudoaleatorios. Pero ¿Por qué Pseudoaleatorios? Porque son generados con una ecuación deterministica para tener propiedades similares a las de un proceso aleatorio (Función de correlación y densidad espectral). De esta forma, es posible generar la misma secuencia tanto en transmisor como receptor.
¿Qué función generadora se busca elegir? Aquella que genere en los diferentes intervalos de tiempo valores lo más distintos posibles uno de otros. Entonces al construir la secuencia de saltos utilizados por el sistema parece prácticamente aleatorio.



Cada imagen muestra el espectrograma asociado a un patrón de saltos particular. Las frecuencias están limitadas de 500 a 1500 Hz con separaciones de 100 Hz, es decir, once valores posibles. Independientemente de los valores en frecuencia, en los dos primeros casos parece sencillo darse cuenta cual es la función generadora de los saltos, ya que son funciones lineales. La función generadora para la última gráfica parece ser la correcta para construir la lista de frecuencias.

Como puede observarse, cuando el jammer inspecciona rápidamente las frecuencias seleccionadas para transmitir, no encuentra un patrón legible y simple para intentar predecir cuál será la próxima frecuencia a transmitir a diferencia del resto de gráficas.

Sin embargo ¿Cómo se obtiene dicha función? Debe utilizarse un componente electrónico conformado de un grupo de flip-flop tipo “D” conectados en cascada, conocido como registros de desplazamiento, y con diferentes realimentaciones mediante sumas del tipo modulo 2.

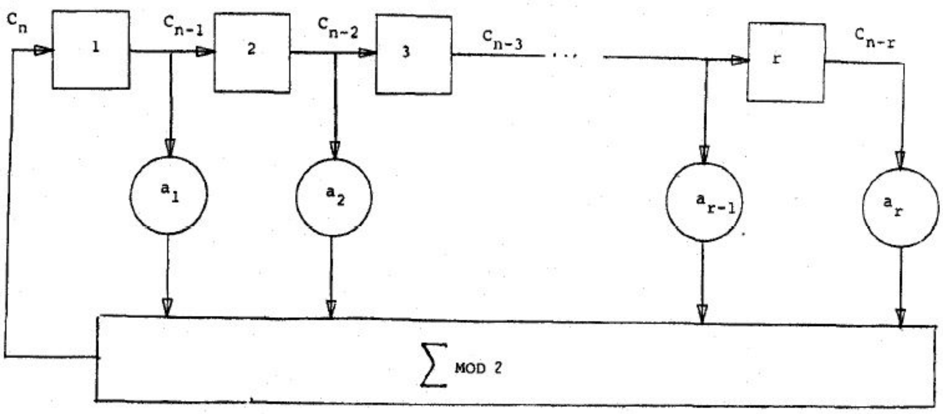


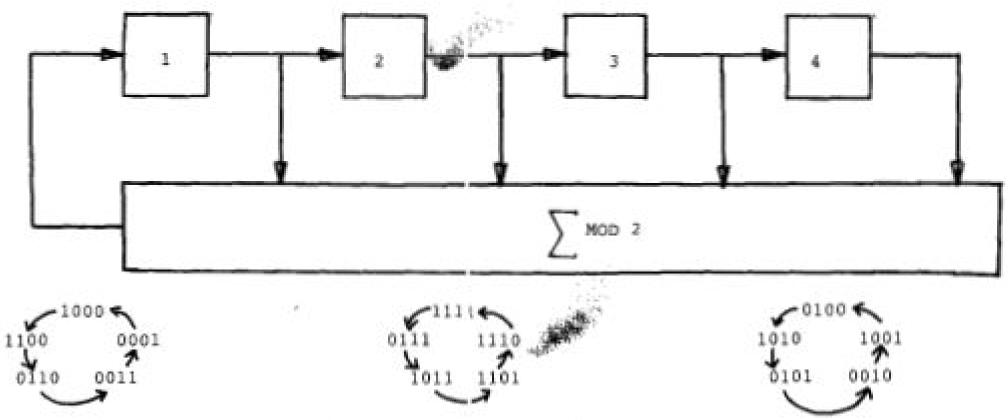
Fig. 4. Simple shift register generator (SSRG).

El registro de desplazamiento consiste en una serie de elementos de memoria binaria que copian a la salida la información de la entrada con cada pulso de clock. El contenido de cada registro (memoria) es combinado a través de coeficientes binarios (1,0) para determinar la entrada a la primera etapa.

Observando la imagen presentada anteriormente, el valor realimentado “C_n” puede ser expresado mediante la siguiente ecuación:

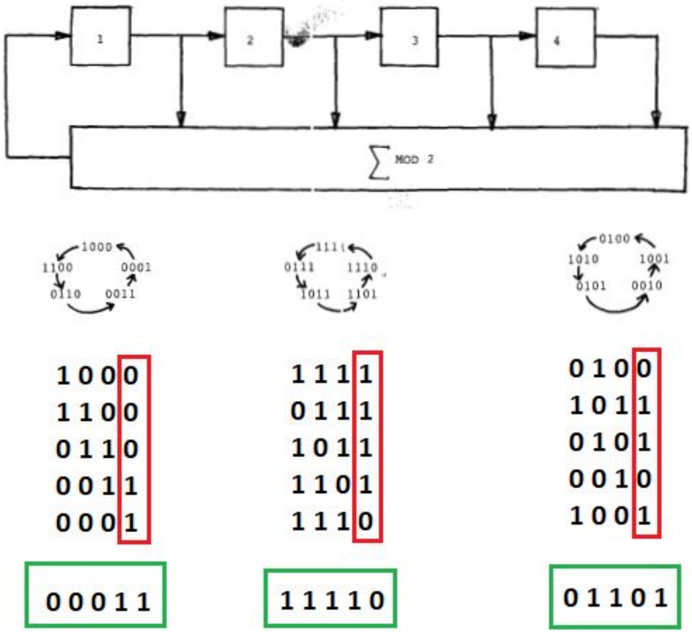
$$C_n = \sum a_k C_{n-k} \pmod 2 \qquad a_r = 1$$

Sin embargo, para poder generar secuencias binarias a la salida de este componente electrónico, es necesario tener en cuenta condiciones iniciales (Valores de los registro de desplazamiento al conectar el dispositivo) y un clock, cuya frecuencia estará definiendo lo que se conoce como tasa de chip "R_c", es decir, la cantidad de variaciones que tendrá la salida del dispositivo en un segundo.
Este factor es fundamental, porque define para el caso de direct secuence el ancho de banda de expansión. Para frecuency hopping se debe determinar la cantidad de chips que serán agrupados para determinar la lista de frecuencias y así poder determinar el ancho de banda de expansión.
En todos los casos, al generar el Código Pseudoaleatorio, debido a la realimentación, es que transcurridas una cierta cantidad de desplazamientos se volverá al estado inicial, determinando de esta manera el periodo para el sistema. La secuencia de chips a la salida del componente electrónico, tiene una determinada forma y luego se repite.
Como se ha comentado, se busca que los patrones del código sean lo menos predecibles o se dificulte encontrarlo. Para ello se debe de tener en cuenta tanto la cantidad de registros de desplazamiento como las realimentaciones con las que cuenta, ya que con ciertas disposiciones se puede alcanzar que la periodicidad del código quede asociado a los L cantidad de registros, definiéndolo con una longitud de "2^L-1". Esta característica vincula a que cuanto mayor sea la cantidad de registros de desplazamiento con la que cuente el sistema, existirá una función de realimentación asociada que permita mayor cantidad de desplazamientos para que la secuencia a la salida se repita.
Se debe de tener presente que cualquier estado inicial que se aplique al sistema, generará una lista de posibles estados, en donde el caso de una lista unitaria o que siempre será igual, es si se establece a todos los registros con "0". En caso de que se inicie con al menos uno de los registros con un valor distinto de "0" se generará una lista de distintos valores de registro y una determinada secuencia de código a la salida.
Por ejemplo, un registro de cuatro etapas tiene cuatro posibles ciclos como se muestra en la figura:

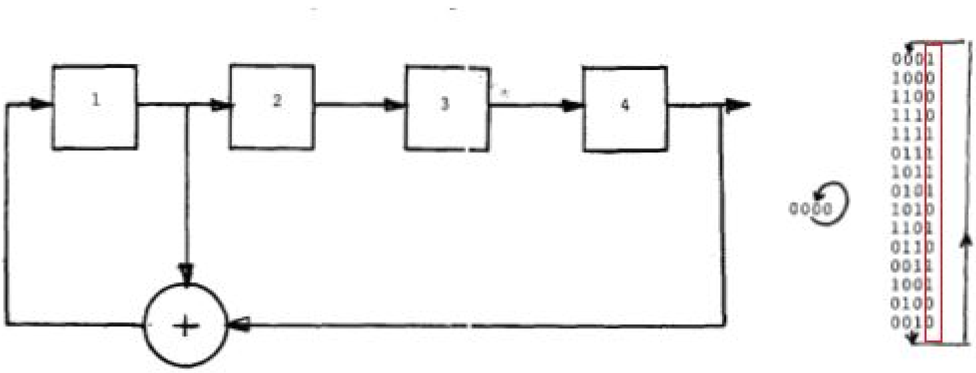


Un ciclo de todos ceros no fue representado, ya que genera siempre la misma salida "0" y los registros de desplazamiento son en todo momento cero "0 0 0 0".
¿Cómo podemos obtener diferentes códigos?
Al cargar en los registros de desplazamiento los valores "1 0 0 0" ó alguno de los que se observa en el primer ciclo se generarán 5 códigos diferentes. Donde todos ellos se diferencian uno del otro mediante desplazamientos temporales. En este caso, la longitud del código no será 2^L-1 , será menor, porque los registros de desplazamientos no asumen todas

las posibles combinaciones que se pueden generar con 4 bits. Para el caso del segundo grupo, los códigos PN generados con sus respectivos registros tendrán el mismo comportamiento que el primer grupo pero serán totalmente diferentes a los del primero. Por último se debe de observar como con las 3 secuencias representadas en conjunto con el registro de todos ceros, asumen todos los posibles valores binarios para un registro de, en este caso, 4 bits. Dejando registro de que no existen otras posibles combinaciones para esa longitud de registro.

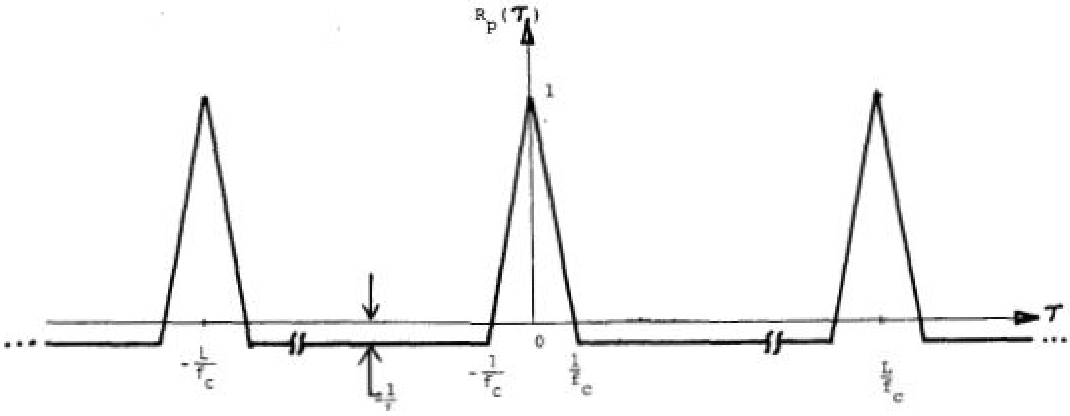


Al analizar propiedades de correlación cruzada dentro de cada grupo es suficiente con obtener la autocorrelación de una de las secuencias, ya que como se detalló anteriormente, los códigos generados para un mismo grupo consisten en desplazamientos temporales de la secuencia en cuestión. Resulta que la función de autocorrelación para cada grupo es diferente y tiene valores de correlación cruzada bastantes distantes de cero, esto perjudica al usuario asignado a dicho código, ya que genera mucha interferencia con otros usuarios si se realizan transmisiones al mismo tiempo. En definitiva se busca que las funciones de autocorrelación y correlación cruzada para el caso de la canalización (asignación de códigos a usuarios) sea lo más próxima a cero. Esto se logra con secuencias de longitud máxima. Una forma de adaptar el dispositivo electrónico para que entregue una secuencia de longitud máxima es modificar las conexiones de realimentación de la siguiente manera:



$$\boxed{1\,0\,0\,0\,1\,1\,1\,1\,0\,1\,0\,1\,1\,0\,0}$$

Este esquema propuesto permite generar para todos los usuarios el mismo código de longitud máxima pero desplazado diferentes tiempos de chips para cada uno. Al analizar la correlación cruzada y autocorrelación se encuentra que puede realizarse solamente mediante autocorrelación porque todos los códigos distintos son versiones desplazadas de un único código. La función permite observar que entre usuarios habrá una mínima interferencia y será uniforme para todos.



Dentro de las propiedades con la que cuentan este tipo de secuencias, se encuentra la propiedad de balance, la cual describe que por cada periodo de la secuencia se encuentran la mitad de componentes ceros y la mitad más uno de componentes de unos. Esta característica indica la presencia de componente de continua distinta de cero que puede ser observada en las características espectrales del código.

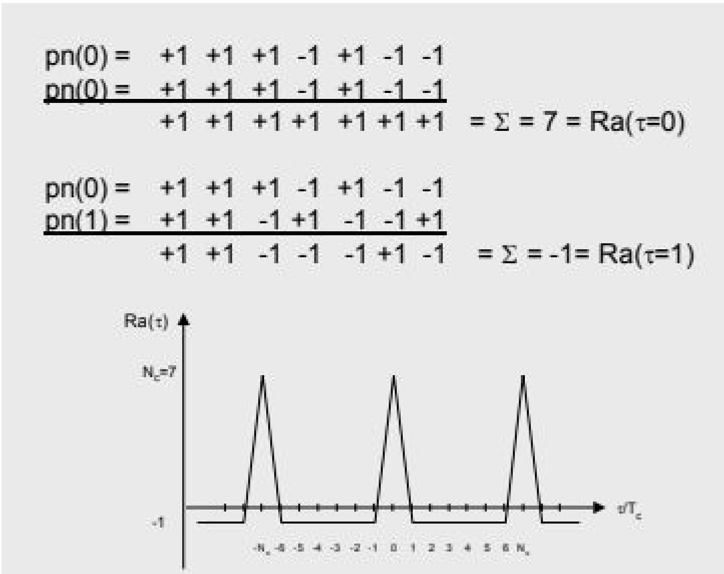
$$pn = +1 \quad +1 \quad +1 \quad -1 \quad +1 \quad -1 \quad -1 \qquad\qquad \sum pn = +1$$

Además, los códigos de longitud máxima tienen una propiedad asociada a la distribución de las corridas. En primer lugar, una corrida es una secuencia continua de un mismo tipo de dígito binario (unos o ceros). Entre la totalidad de corridas que pueden calcularse en un periodo de un código PN siempre puede encontrarse que la mitad es de un elemento, un cuarto de dos elementos, un octavo de tres y así sucesivamente. En códigos cortos esta propiedad tiene dificultades para hacerse presente (se aproxima), es más bien visible en códigos largos. Pero ¿Cómo podría aprovecharse esta propiedad prácticamente? En el proceso de sincronización del código.

Otra propiedad de los códigos es la función de autocorrelación. El origen del nombre “ruido o secuencia Pseudo aleatoria” proviene del hecho de que los códigos de máxima longitud tienen una función de autocorrelación similar a la del ruido blanco gaussiano. Formalmente la autocorrelación puede ser expresada como:

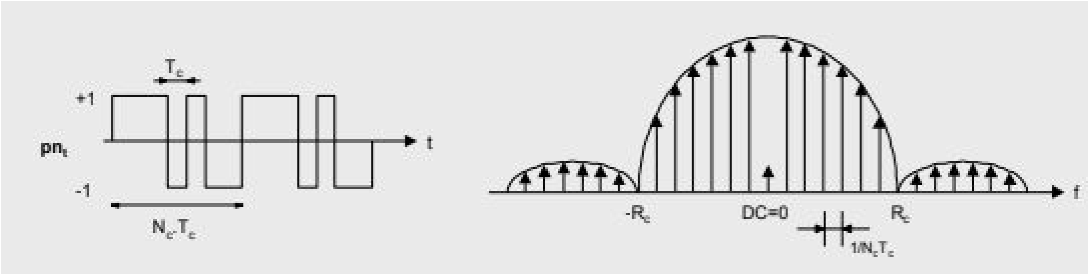
$$Ra() = \int_{-N_c \frac{Tc}{2}}^{N_c \frac{Tc}{2}} P_{ni}(t).P_{ni}(t+\ )$$

De una forma más práctica puede pensarse como la suma de las coincidencias menos la suma de las no coincidencias que surgen de la comparación término a término a lo largo de un periodo para la misma secuencia desplazada τ posiciones.



Como en el receptor se necesita una réplica sincronizada del código presente en el transmisor, se aprovecha esta propiedad de autocorrelación para sincronizar, ya que los picos únicamente están presentes cuando las dos secuencias correlacionadas son las mismas.

Por la naturaleza periódica de la secuencia generada, su espectro estará compuesto por impulsos que estarán separados en el dominio de la frecuencia un valor correspondiente a $\frac{1}{NT_c}$, es decir, que estará en función de la tasa de chips y el largo de la secuencia o periodo. Además, si bien son todos impulsos los que están presentes en el dominio de la frecuencia, se puede observar la envolvente de una sinc al cuadrado, cuyo cruce por cero estará dado en la tasa de chip “Rc”. Por último, se encuentra el efecto de la propiedad de balance, que se hace presente en la componente de continua y con una amplitud de $\frac{1}{N}$ cuando se normaliza la densidad espectral de potencia respecto a su valor máximo.



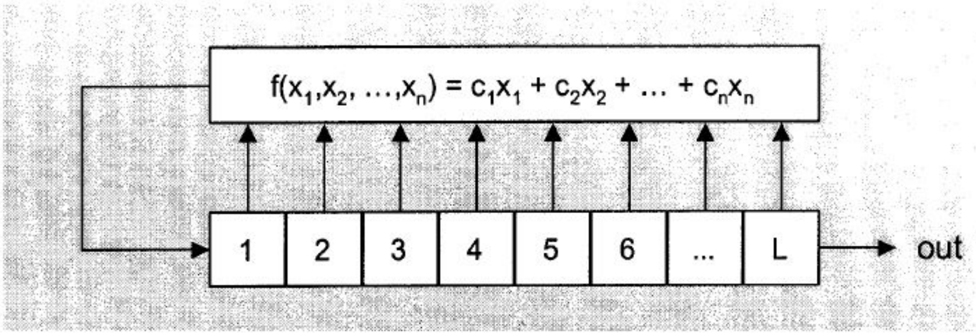
Para terminar, existe otra propiedad igual de importante que la autocorrelación, la correlación cruzada. Esta última describe la interferencia entre códigos P_{ni} y P_{nj} de diferentes usuarios. Para el caso de códigos de longitud máxima se da la condición de que la función de autocorrelación es igual a la de la correlación cruzada, pero desde un punto de vista genérico, la correlación cruzada es definida como:

$$Rc() = \int_{-N_c \frac{Tc}{2}}^{N_c \frac{Tc}{2}} P_{ni}(t).P_{nj}(t+)$$

¿En definitiva se cumplio con las espectativas para generar la secuencia Pseudoaleatoria?
Se creó una función generadora de código bastante sencilla de implementar y con características muy similares a las que posee el ruido AWGN respecto a densidad espectral de potencia y autocorrelación. Sin embargo, los códigos no son de longitud infinita. Al implementar una determinada cantidad de registros de desplazamientos con realimentaciones se hace evidente que en algún momento la secuencia de salida se repite si se aplica el clock durante todo el tiempo. Entonces se busca que la longitud del periodo sea lo más grande posible, para evitar que el jammer rápidamente entienda la periodicidad de la secuencia y se centre en estudiar y replicar un periodo.
Más allá de un jammer puramente intencional, estos mecanismos de generación de código pueden favorecer al rendimiento de un sistema de múltiples usuarios mediante la asignación de una condición inicial distinta a cada usuario de los registros de desplazamiento. De esta forma, con dicho componente electrónico se tendrán 2^L-1 secuencias binarias distintas definidas como códigos PN que pueden ser asignadas una a una a cada usuario.
La particularidad de estas secuencias es que su función de autocorrelación es máxima cuando se compara la secuencia con ella misma y es prácticamente nula para diferentes versiones de la misma desplazada en el tiempo.

Tipos de secuencia
Hasta el momento se ha comentado los códigos PN que se deben implementar para obtener diversas características, pero siempre con la tendencia de obtener secuencias o códigos de longitud máxima que permitan la máxima asignación de usuarios, dando como resultado las mejores propiedades respecto a autocorrelación y correlación cruzada. Es por esta razón que a continuación se detallan distintos códigos que contemplan las propiedades ya enunciadas pero su distinción se basa en la utilidad o finalidad que tienen para su implementación.

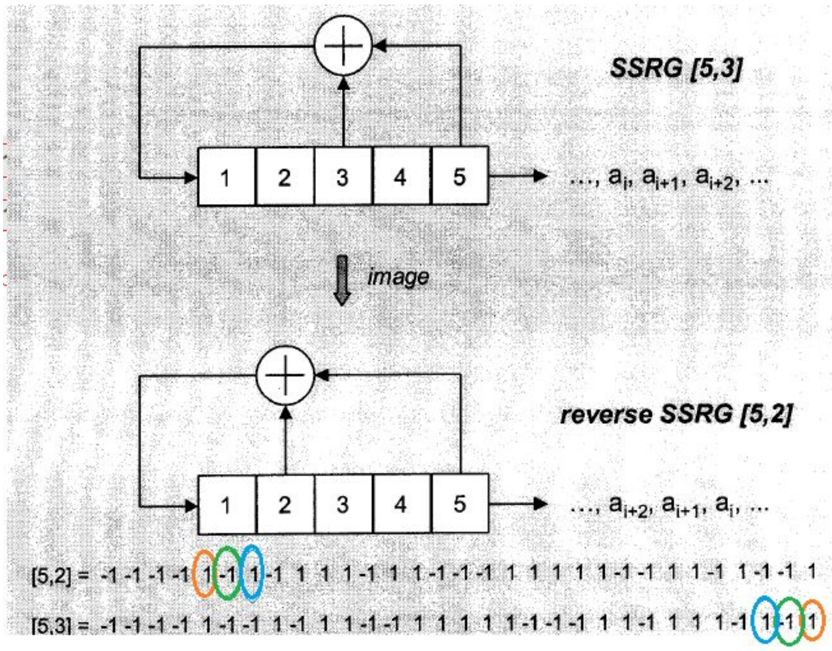
Modelado matemático de códigos PN (Secuencia M)
Primero se debe de considerar como es el esquema básico para la generación de un código, estos cuentan con registros de desplazamiento los cuales se encuentran todos realimentados para formar una única salida. Estos registros son considerados como lineales si la realimentación puede ser expresada como una función de módulo dos.



Para el caso de la imagen anterior, se cuenta con L registros, los cuales están todos vinculados a lo que se conoce como función de realimentación, cada una de estas vías que vinculan al registro con la función de realimentación tendrán asignadas un coeficiente "C_i"que tomará valor 1 en caso de estar conectado o 0 si está desconectado. Jugando con los estados de los coeficientes es que se puede llegar a las "secuencias M" que son aquellas secuencias de máxima longitud, propiedad que ya se ha comentado previamente en qué consiste. Un dato muy importante para la sospecha de estar ante una secuencia M es que la cantidad de coeficientes será par, sin considerar la rama única que se reinyecta en el primer registro. Este dato se puede observar en la siguiente tabla, la cual cuenta con todas las posibles secuencias M para L cantidad de registros que permiten asignarse a 2^{L-1} usuarios distintos, mediante la implementación de una cantidad par de realimentaciones

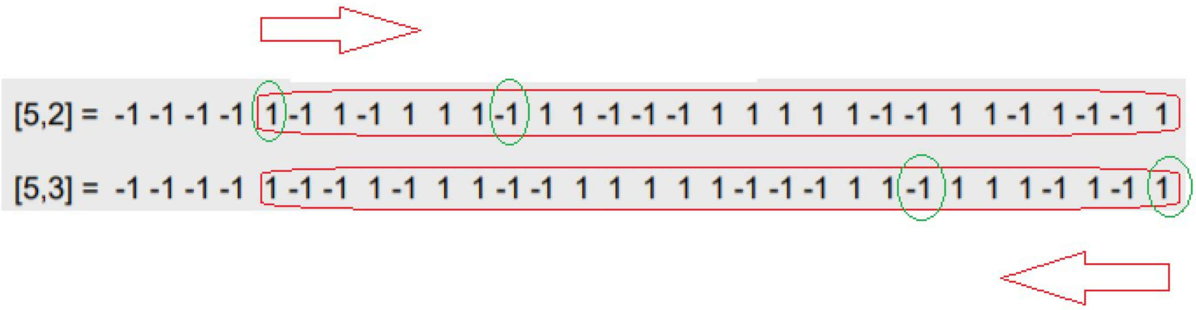
L	$N_c=2^L-1$	Feedback Taps for m-sequences	# m-sequences
2	3	[2,1]	2
3	7	[3,1]	2
4	15	[4,1]	2
5	31	[5,3] [5,4,3,2] [5,4,2,1]	6
6	63	[6,1] [6,5,2,1] [6,5,3,2]	6
7	127	[7,1] [7,3] [7,3,2,1] [7,4,3,2] [7,6,4,2] [7,6,3,1] [7,6,5,2] [7,6,5,4,2,1] [7,5,4,3,2,1]	18
8	255	[8,4,3,2] [8,6,5,3] [8,6,5,2] [8,5,3,1] [8,6,5,1] [8,7,6,1] [8,7,6,5,2,1] [8,6,4,3,2,1]	16
9	511	[9,4] [9,6,4,3] [9,8,5,4] [9,8,4,1] [9,5,3,2] [9,8,6,5] [9,8,7,2] [9,6,5,4,2,1] [9,7,6,4,3,1] [9,8,7,6,5,3]	48
10	1023	[10,3] [10,8,3,2] [10,4,3,1] [10,8,5,1] [10,8,5,4] [10,9,4,1] [10,8,4,3] [10,5,3,2] [10,5,2,1] [10,9,4,2] [10,6,5,3,2,1] [10,9,8,6,3,2] [10,9,7,6,4,1] [10,7,6,4,2,1] [10,9,8,7,6,5,4,3] [10,8,7,6,5,4,3,1]	60
11	2047	[11,2] [11,8,5,2] [11,7,3,2] [11,5,3,2] [11,10,3,2] [11,6,5,1] [11,5,3,1] [11,9,4,1,] [11,8,6,2,] [11,9,8,3] [11,10,9,8,3,1]	176

Existe otra característica para los códigos de secuencia M, la cual describe que si en un registro de desplazamiento de L etapas, cuenta con conexiones de realimentación en las posiciones k y m, se puede obtener un código también de longitud máxima pero cambiando sus realimentaciones a L-k y L-m. Tomando el ejemplo de un registro de L=5 componentes y utilizando como condición inicial “1 0 0 0 0” tendremos que los primeros (L-1) bits coincidiran ya que será la cantidad de desplazamientos hasta que el bit de inicio haya recorrido todos los registros. Una vez transcurridos estos desplazamientos es que comenzará a hacer efecto la función de realimentación, a partir de la salida del L desplazamiento y por lo tanto L bit de la secuencia, es que se visualiza como se refleja el mismo valor de salida pero en el último bit (2^L-1) de la secuencia para el “SSRG[5,3]”



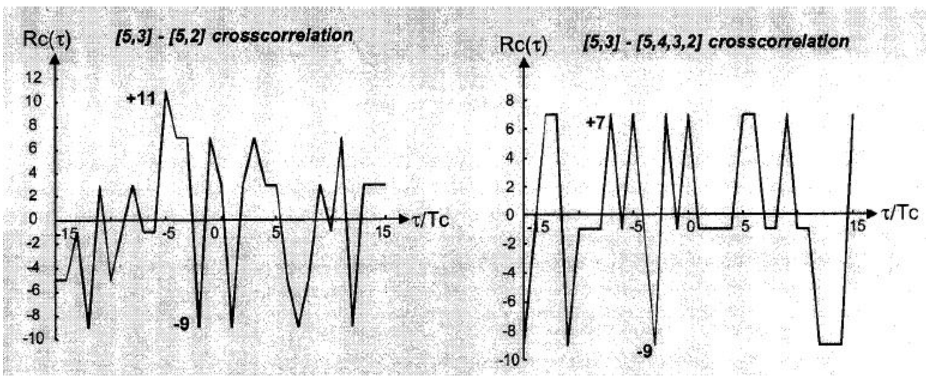
Condicion inicial para ambos registros:
1 0 0 0 0

SSRG [5,3]	Reverse SSRG [5,2]
1 0 0 0 0	1 0 0 0 0
0 1 0 0 0	0 1 0 0 0
0 0 1 0 0	1 0 1 0 0
1 0 0 1 0	0 1 0 1 0
0 1 0 0 1	1 0 1 0 1
1 0 1 0 0	1 1 0 1 0
1 1 0 1 0	1 1 1 0 1
0 1 1 0 1	0 1 1 1 0
0 0 1 1 0	1 0 1 1 1
1 0 0 1 1	1 1 0 1 1
1 1 0 0 1	0 1 1 0 1
:	:
:	:



Con respecto a las propiedades, como ya se comentó, la secuencias M o de longitud máxima cuentan con la propiedad de balance ya que cuentan con 2^{L-1} unos y $2^{L-1}-1$ ceros y también cumple con la propiedad de las corridas, ambas propiedades se pueden observar analizando la secuencia antes presentada.

Respecto a la correlación cruzada su comportamiento no es tan bueno como lo es en la autocorrelación, por lo que en un escenario de múltiples usuarios se debe tener cuidado a la hora de la elección del código para prevenir interferencias. Para ejemplificar esta situación es que se presenta la correlación cruzada entre los dos códigos antes visto [5,3] - reverse [5,2], en donde se puede observar un valor mayor, y entre otro par de secuencias M como lo son [5,3] - [5,4,3,2]



Por último están los aspectos asociados a la seguridad, ya que las secuencias generadas por realimentación lineal, pueden ser determinadas tras (2L+1) chips. Causa por la cual este tipo de secuencias no es utilizado para transmisiones seguras.

Códigos de Barker:

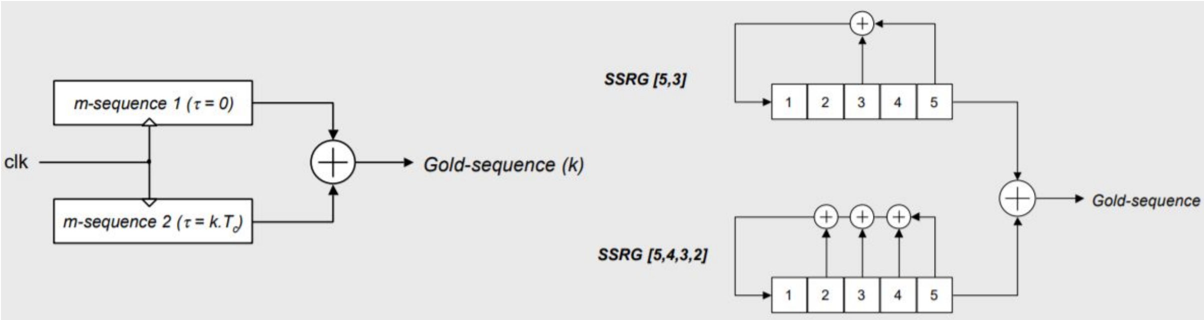
El código Barker es un código binario (código bifásico) desarrollado originalmente para las emisiones de radar que cuenta con la propiedad de que el pico (lóbulo principal) de su función de autocorrelación es igual a la longitud del código. La función de Autocorrelación es impulsiva, lo mismo que en las m-secuencias. Por lo tanto también son ideales para la detección de datos de información.
Los códigos de Barker solo existen para los tamaños de L= 2, 3, 4, 5, 7, 11, 13 por lo que ofrecen un proceso de ganancia limitado. Pueden ser usados como un preámbulo de una secuencia PN larga con un solo propósito, simplificar la sincronización. Las secuencias de Barker son generadas de manera manual de modo que, en el proceso de investigación, produzcan la máxima autocorrelación posible.

CODE LENGTH (N)	BARKER SEQUENCE
1	+
2	+ + or + -
3	+ + -
4	+ + + - or + + - +
5	+ + + - +
7	+ + + - - + -
11	+ + + - - - + - - + -
13	+ + + + + - - + + - + - +

Por último, es conveniente aclarar que con los códigos Barker no se pueden realizar multi emisiones, a menos que se utilicen frecuencias diferentes.

Codigos de Gold:

Estos códigos utilizan como base las secuencias M pero con la intención de realizar una mejora respecto a la correlación cruzada. Para esto, su implementación se basa en la utilización de 2 secuencias “M” en conjunto con una suma de módulo dos, de la siguiente manera:



En los años 60, Gold y Kasami probaron que existían ciertos pares de m-secuencias, que tenían 3 valores de correlación cruzada:

-1,-t(m) y t(m)-2,

donde $t(m)=2^{0.5(m+1)}+1$ para m impar y $2^{0.5(m+2)}+1$ para m par.

A estos pares que cumplen las propiedades de correlación cruzada y que forman parte del mismo grupo de secuencias de longitud "L" se les denomina *secuencias preferidas.* Es necesario aclarar que el factor "m" en las ecuaciones anteriores hace referencia a la longitud de los registros de desplazamiento "L" utilizados para la generación del código PN.
No es casualidad que ambas secuencias "M" utilizadas sean por lo menos de L=5, esto se debe que como Gold necesita de 2 secuencias, si se mira la tabla presentada para las secuencias M, es a partir de L=5 que existen 2 o más secuencias de máxima longitud que cumplen las propiedades mencionadas anteriormente que son conocidas como secuencias preferidas.
Sin embargo, rápidamente podría pensarse que la cantidad de secuencias de código Gold posibles a generar son pocas, porque la cantidad de secuencias M de longitud máxima que cumplen con las propiedades de secuencias preferidas son pequeñas, pero esto no es realmente así, ya que para un par de secuencias de estas características, la familia completa de las secuencias Gold se forma realizando la suma módulo 2 (XOR) de una m-secuencia sin desplazamientos y las respectivas versiones desplazadas de la otra m-secuencia.
Mediante esta implementación se tendrá la misma longitud que las secuencias M utilizadas pero la secuencia resultante no será de longitud máxima y por lo tanto la función de autocorrelación será peor que la de las secuencias M. Con respecto a la cantidad de secuencias que se podrían generar son 2^L-1 y considerando las 2 secuencias M que se están utilizando se llega a 2^L+1. Se podría suponer que estas dos secuencias adicionales se deben a considerar que uno de los polinomios generadores, de alguna de las 2 secuencias M utilizadas, cuenta con su condición inicial configurada con todos ceros. De esta manera se conseguiría una secuencia asociada con la secuencia M que tiene condición inicial distinta de ceros. La otra secuencia que se podría suponer que se añade, mediante la implementación inversa, que consiste en asignar condición inicial de todos ceros a la otra secuencia M.

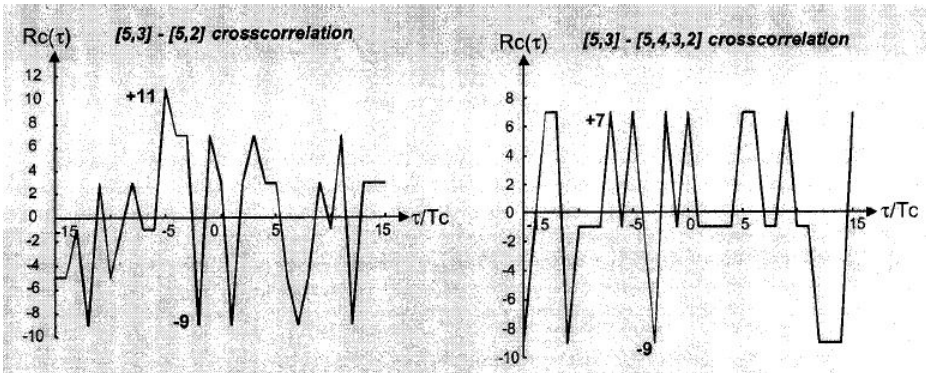
Con respecto a la correlación cruzada, en donde para secuencias M se obtienen múltiples valores, ahora con Gold esto se ve acotado a solo 3 valores dependiendo si el L es impar (odd) o par (even)

L	$N_c=2^L-1$	preferred pairs of m-sequences	3-value crosscorrelations			bound
5	31	[5,3] [5,4,3,2]	7	-1	-9	-29%
6	63	[6,1] [6,5,2,1]	15	-1	-17	-27%
7	127	[7,3] [7,3,2,1] [7,3,2,1] [7,5,4,3,2,1]	15	-1	-17	-13%
8*	255	[8,7,6,5,2,1] [8,7,6,1]	31	-1	-17	+12%
9	511	[9,4] [9,6,4,3] [9,6,4,3] [9,8,4,1]	31	-1	-33	-6%
10	1023	[10,9,8,7,6,5,4,3] [10,9,7,6,4,1] [10,8,7,6,5,4,3,1] [10,9,7,6,4,1] [10,8,5,1] [10,7,6,4,2,1]	63	-1	-65	-6%
11	2047	[11,2] [11,8,5,2] [11,8,5,2] [11,10,3,2]	63	-1	-65	-3%

L	N_c	normalized 3-value crosscorrelation	Frequency of occurence
Odd Impar	$2^L - 1$	$-1/N_c$ $-(2^{(L+1)/2} + 1)/N_c$ $(2^{(L+1)/2} - 1)/N_c$	~ 0.50 ~ 0.25 ~ 0.25
Even (not k.4) Par	$2^L - 1$	$-1/N_c$ $-(2^{(L+2)/2} + 1)/N_c$ $(2^{(L+2)/2} - 1)/N_c$	~ 0.75 ~ 0.125 ~ 0.125

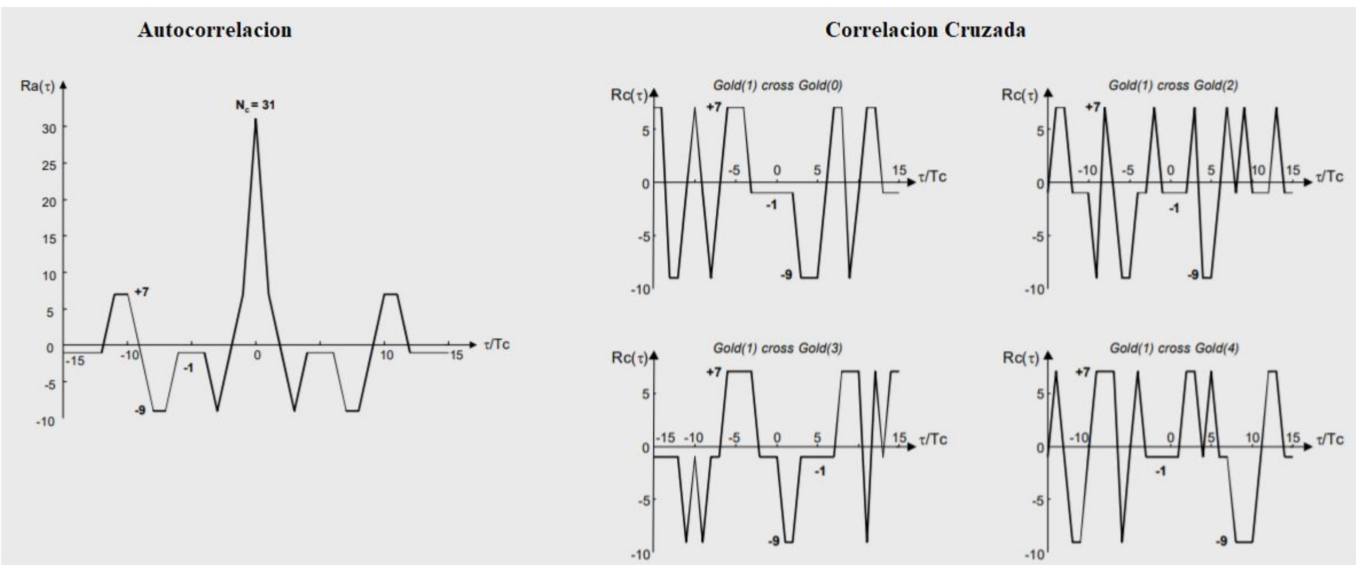
A continuación se puede observar como la autocorrelación ha empeorado pero con respecto a la correlación cruzada se obtienen hasta 3 valores distintos como se ha indicado en la tabla anterior. Situación que para las secuencias M, iba variando según los códigos que se están contemplando para la correlación cruzada.

Los resultados de la correlación cruzada presentados en la gráfica de la izquierda son para dos códigos PN generados por diferentes polinomios de igual orden. En particular, para un polinomio de L=5 y su secuencia imagen. En la imagen de la derecha se muestra la correlación cruzada para el par de secuencias M creadas por polinomios de L=5 definidas como secuencias preferidas.



Analizando particularmente la utilidad de las funciones de autocorrelación y correlación cruzada se plantea la siguiente incógnita ¿Para qué sirven las secuencias Gold? ¿Por qué no utilizar secuencias M? La respuestas a este par de preguntas consiste en describir en primer lugar la finalidad de la función de autocorrelación dentro de un sistema de acceso múltiple por división de código (CDMA). Como se ha detallado anteriormente, la correlación proporciona características valiosas a los códigos de expansión para tareas tanto de sincronismo como de canalización. En particular, una de las propiedades fundamentales para aplicar Spread Spectrum es generar una réplica sincronizada del código en el receptor, sino de nada sirve la tarea de expansión. ¿Las Secuencias M permiten realizar tareas de sincronismo? Si. Debería generarse una secuencia M en el receptor utilizando el mismo polinomio generador que en el transmisor con las mismas condiciones iniciales y realizar la operación de correlación cruzada entre la waveform recibida y el código PN generado de forma local para un tiempo equivalente a la cantidad de chips que contiene un período del código PN. Si los códigos PN son demasiados largos, esta tarea en un principio podría llevar demasiado tiempo y entorpecer el funcionamiento del sistema. ¿Que se espera obtener de la correlación cruzada para secuencias M? Si existe sincronismo, el resultado de la correlación es el máximo posible y si no, lo cual puede implicar desde una pequeña falta de sincronismo hasta una muy grande, el resultado será menor pero constante para cualquier caso. Entonces, deberá realizarse tareas de sincronismo una y otra vez mientras el resultado de la correlación cruzada no sea el máximo, generando cambios en el código PN local producto de desplazamientos temporales. ¿Cuántas correlaciones cruzadas podrían realizarse hasta lograr el sincronismo? Este resultado puede variar, porque si hay un desfasaje muy grande entre ambos códigos, cuando se generen algunos desplazamientos del código local, enganchará enseguida. En el mejor de los casos deberá esperarse en total 2.NTch (una correlación inicial y una corrección del código PN desplazada en un Tch). El peor de los casos, se da cuando ambos códigos están pequeñamente desincronizados. Si esto ocurre, como el resultado de la correlación cruzada no arroja ninguna información adicional más que no hay sincronismo, el receptor debe probar con NTch desplazamientos temporales, porque no sabe si le falta mucho o poco para lograr sincronizar. Entonces ¿Qué aporta Gold? Los codigos de gold al tener una función de autocorrelación más variada (4 valores diferentes, uno para sincronismo y el resto para situaciones de falta de sincronismo) cuando el receptor realiza operaciones de correlación cruzada puede tener referencias de las ubicaciones temporales en el código respecto a los resultados, permitiendo reducir la cantidad de intentos al tratar de sincronizar. La cantidad de usuarios que pueden ser asignados en esta metodología es $(2^L-1) \times (2^L-1)$ ya que para cada polinomio generador, se pueden alternar las condiciones iniciales, generando diferentes códigos Gold de la misma familia. Las imágenes presentadas a continuación utilizan los polinomios de realimentación preferidos [5,3] y [5,4,3,2] donde solo se alternan las condiciones iniciales del primero, dejando siempre al segundo en el estado inicial “0 0 0 0 1”.

Lo que la imagen de la derecha permite visualizar, es la correlación cruzada entre diferentes códigos Gold, generados con desplazamientos lineales de un chip en el código PN del generador [5,3]. Esto no significa que los códigos gold sean desplazamientos temporales de una única secuencia. Las mismas, son todas diferentes lo que conlleva a que las secuencias generadas no sean de longitud máxima. Es decir, exceden a la cantidad esperada para una secuencia de dichas características.



La canalización bajo esta técnica de expansión espectral es posible, porque todos los códigos de Gold asignados tienen correlación cruzada mínima cuando existe sincronismo perfecto, lo cual minimiza la interferencia entre usuarios.
Por último y con respecto a la propiedad de balance, es que dependerá de las dos secuencias M que se tomen para generar el código Gold si cumpliran o no con esta propiedad. Reduciendo la posibilidad de asignación de códigos para que cumplan con la propiedad de balance.

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

[5,3] (0) = -1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 -1 1 1 1 1 1 -1 -1 -1 1 1 -1 1 1 1 -1 1 -1 1

Gold(0) = -1 -1 -1 -1 -1 -1 1 -1 -1 -1 1 1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 -1 -1 1 1 -1

Σ Gold(0) = -7 = not balanced

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

[5,3] (1) = 1 -1 -1 -1 -1 1 -1 -1 1 -1 1 1 -1 -1 1 1 1 1 1 -1 -1 -1 1 1 -1 1 1 1 -1 1 -1

Gold(1) = 1 -1 -1 -1 1 1 1 1 1 1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 -1 1 -1 -1 1

Σ Gold(1) = 1 = balanced

Códigos Hadamard - Walsh
Este tipo de código es el implementado para la canalización en el estándar de telefonía móvil IS-95 con 64 códigos que permiten igual cantidad de canales por estación base. Ya que cuenta con la característica de que todas las secuencias o códigos son absolutamente ortogonales entre sí, a diferencia de lo que sucedía en las secuencias M. Por lo que sí existe perfecto sincronismo, no habrá interferencia entre usuarios.
El código Walsh utiliza matrices del tipo Hadamard, en donde el código es generado mediante grupos de $N = 2^n$ códigos de longitud $N=2^n$, generando de esta manera matrices cuadradas de NxN

$$H_N = \begin{bmatrix} H_{N/2} & H_{N/2} \\ H_{N/2} & -H_{N/2} \end{bmatrix} \qquad \text{con } H_1 = [1]$$

Sin importar la dimensión de la matriz del código a contemplarse, siempre la primera fila estará compuesta por todos unos, mientras que el resto de las filas se compondrán N/2 de unos y N/2 de ceros o "-1". Con esta característica es que la distancia o número de elementos diferentes entre 2 códigos será siempre N/2.

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

$$z_{ij} = \frac{numero \cdot de \cdot digitos \cdot coincidentes - numero \cdot de \cdot digitos \cdot no \cdot coincidentes}{numero \cdot total \cdot de \cdot digitos \cdot en \cdot la \cdot secuencia} = \begin{cases} 1 & \text{Para i=j} \\ 0 & \text{Caso contrario} \end{cases}$$

Es gracias a la característica de distancia con la que cuentan estos códigos, que al analizar 2 códigos distintos, se obtiene que son totalmente ortogonales.

$$\sum_{k=0}^{N-1} h_{ik} h_{jk} = 0$$