Cognitive Radio Communication Laboratory

Lab. 1

Pseudo Noise Sequence and Gold Code

I. Purpose

From this lab., we will learn how to generate pseudo noise sequence (PN-sequence) and Gold code. Besides, we will realize the autocorrelation property of noise and the cross-correlation property of difference Gold code sequences.

II. Principle

Linear feedback shift register (LFSR) can generate pseudo random sequence. Its block diagram is shown in Figure 1. It consists of r 1-bit shift registers. The addition is performed in Galois Field with 2 elements, and thus, can be calculated by modulo-2 operation, namely XOR gate. When the generated sequence of an r-stage LFSR has a period of $2^r - 1$, we call this sequence as maximum length sequence or M-sequence. The criterion of an LFSR to generate an M-sequence is that its coefficients, $c_0, c_1, \ldots\ c_r$, must satisfy the primitive polynomial.

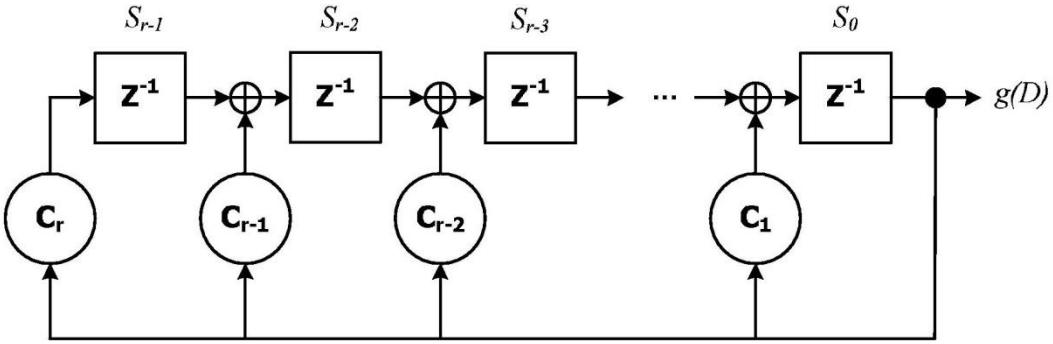
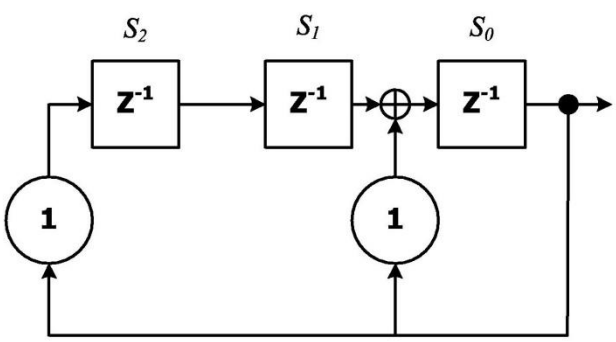


Figure 1. Linear feedback shift register.

A polynomial of degree r is said to be primitive if polynomial $g(D) = c_0 + c_1 D + c_2 D^2 + \cdots + c_r D^r$ is irreducible, which means that it is not the product of any two polynomial, and if the smallest integer n for which $g(D)$ divides $D^n + 1$ is $n = 2^r - 1$. For degree r, we can have more than one primitive polynomials. The theory about the coefficients of primitive polynomials is well developed. Primitive

polynomials of any degree r are shown to exist and can be found on internet or books related with spread spectrum. Hence, we only need to know how to use them.



Cycle	S_2	S_1	S_0
0	1	1	1
1	1	1	0
2	0	1	1
3	1	0	0
4	0	1	0
5	0	0	1
6	1	0	1
7	1	1	1

Figure 2. Example of a primitive polynomial of degree 3.

For example of degree 3, the coefficients of the primitive polynomial can be expressed as 13_{oct}, which is in octal, and thus $c_3 = 1$, $c_2 = 0$, $c_1 = 1$, $c_0 = 1$. Its implementation by LFSR is given in Figure 2 together with the possible 7 states and the generated sequence.

A. PN sequence

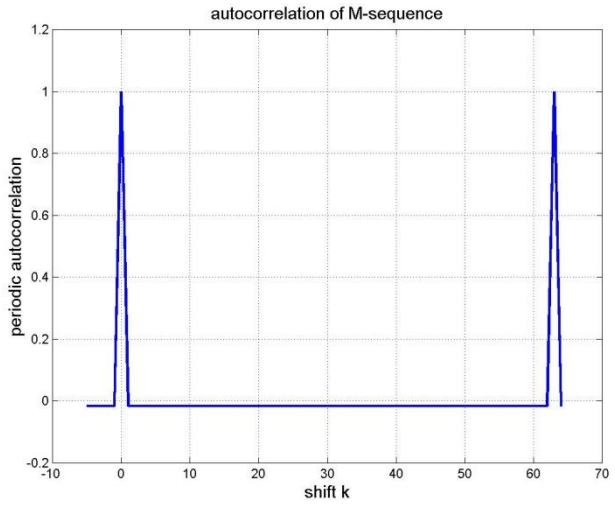


Figure 3. Periodic autocorrelation of the m-sequence with period N equal to 63.

The m-sequence generated by LFSR has many properties. Here, we observe its periodic autocorrelation property first. For an m-sequence with a period of N, define periodic autocorrelation as

$$\Theta_N(k) = \frac{1}{N}\sum_{i=0}^{N-1} a_i a_{(i+k)} \quad (1)$$

where $a_i = (-1)^{b_i}$, which maps the sequence with element $b_i \in \{0, +1\}$ to a sequence with element $a_i \in \{+1, -1\}$. In addition, $(\cdot)$ denotes the modulo operation, i.e. $(i+k)$ denotes $(i+k)\ mod\ N$, which generate the sequence with offset k. We then have

$$\Theta_N(k) = \begin{cases} 1 & k = lN \\ -1/N & k \neq lN \end{cases}, \quad (2)$$

where l is an arbitrary integer. Figure 3 shows the results of $\Theta_N(k)$ for $N = 63$.

B. Gold Code

Gold code, made by PN-sequence, has an excellent cross-correlation property. Define full-period cross-correlation of two sequences $\boldsymbol{b} = [b_1 \quad b_2 \quad b_3 \quad \dots \quad b_N]$ and $\boldsymbol{b'} = [b'_1 \quad b'_2 \quad b'_3 \quad \dots \quad b'_N]$ containing element b_i and b'_i as

$$\Phi_{bb'}(k) = \frac{1}{N}\sum_{i=0}^{N-1} a_i a'_{(i+k)} \quad (3)$$

where $a_i = (-1)^{b_i}$ and $a'_i = (-1)^{b'_i}$, similarly. The full-period cross-correlation has only three possible values,

$$\Phi_{bb'}(k) = \begin{cases} -\frac{1}{N}t(r) \\ -\frac{1}{N} \\ \frac{1}{N}(t(r)-2) \end{cases} \quad (4)$$

where

$$t(r) = \begin{cases} 1 + 2^{0.5(r+1)} & for\ odd\ r \\ 1 + 2^{0.5(r+2)} & for\ even\ r \end{cases}. \quad (5)$$

and $N = 2^r - 1$. We call two sequences $\boldsymbol{b}$ and $\boldsymbol{b'}$ as preferred pairs.

Finding preferred pairs $\boldsymbol{b}$ and $\boldsymbol{b'}$ of m-sequence to generate the sets of Gold codes, the following conditions must be satisfied.

1. $r\ mod\ 4 \neq 0$.
2. $\boldsymbol{b'} = \boldsymbol{b}[q]$ where q is odd and either $q = 2^k + 1$ or $q = 2^{2k} - 2^k + 1$. $\boldsymbol{b}[q]$ indicates that $\boldsymbol{b'}$ is obtained by sampling every qth element of $\boldsymbol{b}$.
3. $\gcd(r,k) = \begin{cases} 1 & for\ odd\ r \\ 2 & for\ \ r\ mod\ 4 = 2 \end{cases}$.

When a preferred pair, $\boldsymbol{b}$ and $\boldsymbol{b'}$, is found, the family of Gold codes is defined by $\{\boldsymbol{b}, \boldsymbol{b'}, \boldsymbol{b} + \boldsymbol{b'}, \boldsymbol{b} + D\boldsymbol{b'}, \boldsymbol{b} + D^2\boldsymbol{b'}, \dots, \boldsymbol{b} + D^{N-1}\boldsymbol{b'}\}$, where $D^i\boldsymbol{b'}$ represents to shift the m-sequence $\boldsymbol{b'}$ for i units.

Based on m-sequence, a typical configuration to generate the complete family of Gold codes with length N equal to 31 is shown in the following.

1. Choose a non-zero initial state $(s_0, s_1, \ldots\ s_{r-1})$. Load the state into the upper LFSR in Figure 4. Set the zero state $(s'_0, s'_1, \ldots\ s'_{r-1}) = (0,0,\ldots,0)$ into the lower LFSR. Generate sequence $\boldsymbol{b}$.
2. Similarly, load non-zero initial state $(s'_0, s'_1, \ldots\ s'_{r-1})$ into the lower LFSR and set the zero state $(s_0, s_1, \ldots\ s_{r-1}) = (0,0,\ldots,0)$ into the upper LFSR. As a result, we can generate sequence $\boldsymbol{b}'$.
3. Load the non-zero initial state $(s_0, s_1, \ldots\ s_{r-1})$ that generates sequence $\boldsymbol{b}$ into the upper LFSR and fill the 31 non-zero states into the lower LFSR sequentially. For each state, a corresponding Gold code of length 31 can be produced. As a result, we can generate 31 Gold code sequences. In total, the code family of 33 Gold code sequences is obtained.

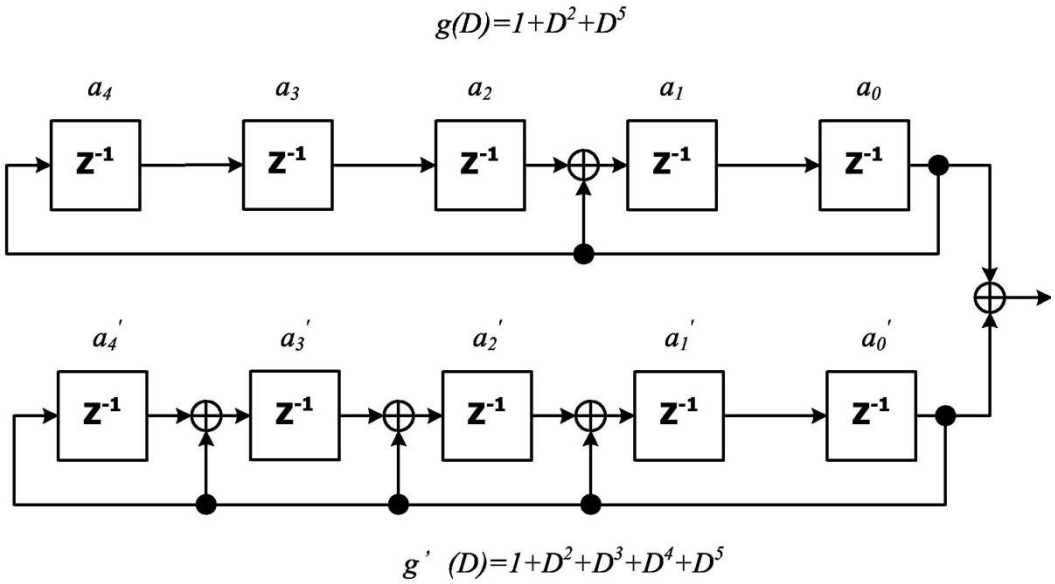


Figure 4. Gold code generator.

III. Procedures

1. According to [1], the coefficients of the primitive polynomial of degree 4 can be denoted as 23_{oct}, please generate the m-sequence by LFSR in Figure 1 with Matlab.
2. Check the periodic autocorrelation of the m-sequence that you generate.
3. The coefficients of the primitive polynomial of degree 5 can be represented as 45_{oct}, 75_{oct}, 67_{oct}. The even-numbered students use 45_{oct} and the odd-numbered students use 75_{oct} to generate sequence $\boldsymbol{b}$. Use condition 2 on page 3 with $q = 9$ to generate sequence $\boldsymbol{b}'$. Please use Matlab to check the full-period cross-correlation property of sequence $\boldsymbol{b}$ and $\boldsymbol{b}'$.

4. Now, for even-numbered students, please use 75_{oct} to generate another m-sequence $\boldsymbol{b}''$. For odd-numbered students, please use 67_{oct} to generate another m-sequence $\boldsymbol{b}''$. Check if sequence $\boldsymbol{b}'$ in Q3 is the same as sequence $\boldsymbol{b}''$ with certain offset.
5. Generate the family of 33 Gold code sequences with length 31 based on $\boldsymbol{b}$ and $\boldsymbol{b}'$.
6. Now, transfer the elements of all binary sequences in Q5 into {+1,-1}. Denote the first sequence as the base sequence $\boldsymbol{s_1}$. Check the full-period cross-correlation of this base sequence with the remaining 32 sequences in Q5, i.e. $\boldsymbol{s_j}$ for $j = 2, \ldots 33$, by calculating

$$\Phi_{s_1 s_j}(0) = \frac{1}{N}\sum_{i=0}^{N-1} s_{1,i} s_{j,i}$$

7. Use the $(l+2)$th sequence and the $(l+19)$th sequence in Q6, where l is your last digit of student ID. Check

$$\Phi_{s_{l+2} s_{l+19}}(k) = \frac{1}{N}\sum_{i=0}^{N-1} s_{l+2,i} s_{l+19,(i+k)}$$

==========

8. Please use Verilog to implement LFSR and generate the m-sequence of 23_{oct}. Use the same initial state as that during Matlab simulation. Compare the results.
9. For even-numbered students, please use 45_{oct} and 75_{oct} for the coefficients of your upper and lower LFSRs in Figure 4. For odd-numbered students, please use 75_{oct} and 67_{oct} for the coefficients of your upper and lower LFSRs in Figure 4. Please use Verilog to implement the Gold code generator. Generate three code sequences and see if they are contained in the results that you generate in Q5. Note that two sequences are regarded the same if they only differ with certain offset

IV. Results

1. Write down the initial state that you use. Print out the m-sequence that you generate. Use command "stem" to depict them.
2. Draw the autocorrelation result in the same way as that in Fig. 3.
3. Write down the initial state that you use to generate sequence $\boldsymbol{b}$. Print out the sequence $\boldsymbol{b}$ and $\boldsymbol{b}'$ according to procedure 3. Calculate their full-period cross-correlation.
4. Write down the initial state that you use to generate sequence $\boldsymbol{b}''$. Print out the sequence $\boldsymbol{b}''$. Please write down how you check sequence $\boldsymbol{b}'$ and $\boldsymbol{b}''$ to see if they are the same sequence with different offset. (Hint: for m-sequence with length $2^r - 1$, it has only one run of r continual 1's.)
5. Print out 4 sequences among 33 Gold code sequences according to procedure 5.

(Try to save papers.)

6. Write down the base sequence. Draw the results of full-period cross-correlation in a figure with x-axis indicating index j and y-axis indicating the cross-correlation result $\Phi_{s_1 s_j}(0)$.
7. Write down the base sequence. Draw the results of full-period cross-correlation of s_{l+2} and s_{l+19} in a figure with x-axis indicating offset k and y-axis indicating the cross-correlation result $\Phi_{s_{l+2}s_{l+19}}(k)$ for $-31 \leq k \leq 31$.

==========

8. Draw the block diagram of the LFSR with coefficients 23_{oct}. (10%) Please implement your block diagram by Verilog to see if the RTL simulation results are the same as the Matlab results. Print out the Verilog codes (20%) Print out the timing diagram of RTL simulation (10%). Try to make the numeric expressions clear in the timing diagram that you print.
9. Draw the block diagram of your Gold code generator. (10%) Print out your Verilog codes including test bench (25%). Print out the timing diagram of three code sequences (10%). Show that three code sequences generated by Verilog are part of the sequences in Q5. (10%)
10. Why do Q9 and Q5 generate the same Gold code sequences? (5%)

V. Appendix

For m-sequence, it has 1 run of one's of length r,

1 run of zero's of length $r-1$,

1 run of one's and 1 run of zero's of length $r-2$,

2 run of one's and 1 run of zero's of length $r-3$,

…

2^{r-3} runs of one's and 2^{r-3} runs of zero's of length 1.

VI. Reference

[1] R. L. Peterson, *Introduction to Spread Spectrum Communications,* Prentice Hall, 1995.