Lab 4
Pseudo Random Sequences
EE 445S

Enoc Balderas Daniel Diamont

April 1, 2019

Date Performed: March 25, 2019
Instructor: Professor Evans

1 Introduction

For this lab we explored pseudo random sequences. The main sequence that we observed is the m-sequence (max length). We created the m-sequence using a simple shift register (SSRG). Finally we used the m-sequence at the transmitter to scramble a bit and then used the same sequence at the reciever to descramble the bit.

2 Methods

We started off by creating a $[5, 2]_s$ SSRG to implement the sequence. We tested the sequence by profiling the output with an oscilloscope and making sure that it was periodic. Next we used the m-sequence to scramble a transmit bit, and subsequently descramble the bit at the reciever. This was achieved by starting the SSRG of the transmitter and reciever in the same initial state, and xoring the input bit with the resulting sequence.

3 Results

Table:

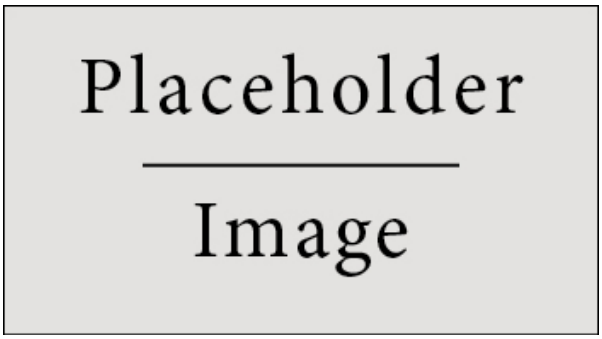

Figure 1: SSRG $[5, 2]_s$.

count	state
[0]	10000
[1]	01000
[2]	10100
[3]	01010
[4]	10101
[5]	11010
[6]	11101
[7]	01110
[8]	10111
[9]	11011
[10]	01101
[11]	00110
[12]	00011
[13]	10001
[14]	11000
[15]	11100
[16]	11110
[17]	11111
[18]	01111
[19]	00111
[20]	10011
[21]	11001
[22]	01100
[23]	10110
[24]	01011
[25]	00101
[26]	10010
[27]	01001
[28]	00100
[29]	00010
[30]	00001
[31]	10000
[32]	01000
[33]	10100
[34]	01010
[35]	10101
[36]	11010
[37]	11101
[38]	01110

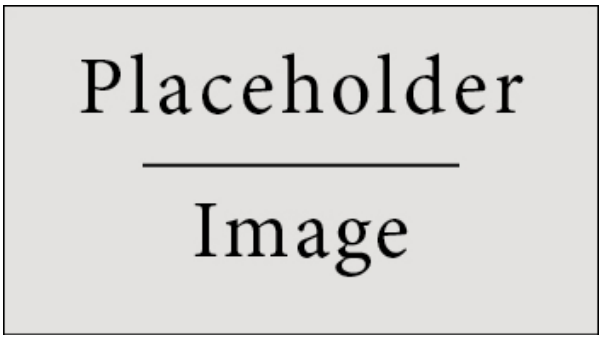

Figure 2: Scrambler and Descrambler.

Table:

count	scrambler output	descrambler
[0]	0	1
[1]	1	1
[2]	1	1
[3]	1	1
[4]	1	1
[5]	0	1
[6]	1	1
[7]	0	1
[8]	1	1
[9]	0	1
[10]	0	1
[11]	0	1
[12]	1	1
[13]	0	1
[14]	0	1
[15]	1	1
[16]	1	1
[17]	1	1
[18]	0	1
[19]	0	1
[20]	0	1
[21]	0	1
[22]	0	1
[23]	1	1
[24]	1	1
[25]	0	1
[26]	0	1
[27]	1	1
[28]	0	1
[29]	1	1
[30]	1	1
[31]	0	1
[32]	1	1
[33]	1	1
[34]	1	1
[35]	1	1
[36]	0	1
[37]	1	1
[38]	0	1
[39]	1	1
[40]	0	1
[41]	0	1
[42]	0	1
[43]	1	1
[44]	0	1
[45]	0 4	1
[46]	1	1
[47]	1	1
[48]	1	1
[49]	0	1
[50]	0	1
[51]	0	1
[52]	0	1

Code:

```
pn = [1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1,
      1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0];

sc = [0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0,
      0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1];

pn(pn == 0) = -1;
sc(sc == 0) = -1;

tmp1 = [pn pn];
tmp2 = [sc sc];

s1 = fft(tmp1);
pn_corr = ifft(s1.*conj(s1))/length(tmp1);

s2 = fft(tmp2);
sc_corr = ifft(s2.*conj(s2)/length(tmp2));
```

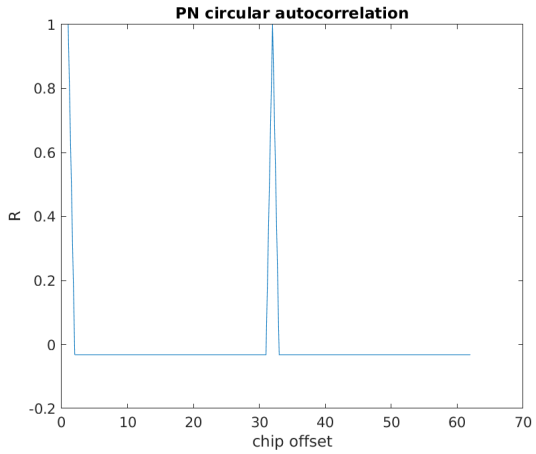

Figure 3: pseudo random sequence autocorrelation.

4 Discussion

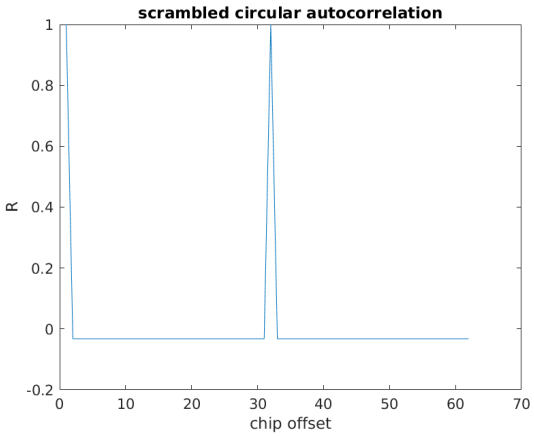

Figure 4: scrambled bit autocorrelation.