

Санкт-Петербургский Политехнический Университет
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчёт о лабораторной работе 8

Дисциплина: Телекоммуникационные технологии

Тема: Модель телекоммуникационного канала.

Работу выполнил:
гр. 33501/3 Кнорре А.В.
Преподаватель
Богач Н.В.

Санкт-Петербург
2018

1 Цель работы

Пакетный сигнал длительностью 200 мкс состоит из 64 бит полезной информации и 8 нулевых tail-бит. В нулевом 16-битном слове пакета передается ID, в первом - период излучения в мс, во втором – сквозной номер пакета, в третьем - контрольная сумма (CRC-16). На передающей стороне пакет сформированный таким образом проходит следующие этапы обработки:

1. Помехоустойчивое кодирование сверточным кодом с образующими полиномами 753, 561(octal) и кодовым ограничением 9. На выходе кодера количество бит становится равным 144.

2. Перемежевание бит. Количество бит на этом этапе остается неизменным.

3. Модуляция символов. На этом этапе пакет из 144 полученных с выхода перемежителя бит разбивается на 24 символа из 6 бит. Генерируется таблица функций Уолша длиной 64 бита. Каждый 6- битный символ заменяется последовательностью Уолша, номер которой равен значению данных 6-ти бит. Т.о. на выходе модулятора получается $24 * 64 = 1536$ знаковых символов.

4. Прямое расширение спектра. Полученная последовательность из 1536 символов периодически умножается с учетом знака на ПСП длиной 511 символов. Далее к началу сформированного символьного пакета прикрепляется немодулированная ПСП. Т.о. символьная длина становится равной 1747. Далее полученные символы модулируются методом BPSK . Задача: по имеющейся записи сигнала из эфира и коду модели передатчика создать модель приемника, в которой найти позицию начала пакета и, выполнив операции демодуляции, демультиплексирования и декодирования, получить передаваемые параметры: ID, период, и номер пакета. Известно, что ID = 4, период 100 мс, номер пакета 373. Запись сделана с передискретизацией 2, т.е. одному BPSK символу соответствуют 2 лежащих друг за другом отсчета в файле. Запись сделана на нулевой частоте и представляет из себя последовательность 32-х битных комплексных отсчетов, где младшие 16 бит вещественная часть, старшие 16 бит – мнимая часть. Ниже приведена таблица перемежения и последовательность ПСП.

2 Ход работы

2.1 Прием и передача сигналов

Приемник и передающее устройство - транслятор - позволяют выполнять последовательность обратимых операций над пакетом обмена данными. В канале передачи информации могут возникнуть шумы, искажающие сигнал. При наличии неизвестного шума на приемнике осуществляется синхронизация записи сигнала по известной опорной псевдослучайной последовательности. При демодуляции и одновременном сужении спектра принятого сигнала также используется корреляционный метод - обратное быстрое преобразование Уолша-Адамара. Таким образом определяется максимальный по абсолютному значению элемент строки матрицы результатов, который укажет на начало пакета (синхронизация) или на бинарный номер строки матрицы Уолша (демодуляция).

2.2 Matlab

Зададим псевдослучайную последовательность:

```
5 %Последовательность ПСП
6 PRS=[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;
7 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;
8 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;
9 -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;
10 -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;
11 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;
12 -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;
13 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;
14 -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;
15 -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;
16 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;
17 -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;
18 -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;
19 -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;
20 -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;
21 -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;
22 ];
```

Рис. 2.1: PSP

Зададим последовательность перемежения:

```
23 %Последовательность перемежения
24 interleaver = [0; 133; 122; 111; 100; 89; 78; 67; 56; 45; 34; 23; 12; 1; 134; 123;
25 112; 101; 90; 79; 68; 57; 46; 35; 24; 13; 2; 135; 124; 113; 102; 91;
26 80; 69; 58; 47; 36; 25; 14; 3; 136; 125; 114; 103; 92; 81; 70; 59;
27 48; 37; 26; 15; 4; 137; 126; 115; 104; 93; 82; 71; 60; 49; 38; 27;
28 16; 5; 138; 127; 116; 105; 94; 83; 72; 61; 50; 39; 28; 17; 6; 139;
29 128; 117; 106; 95; 84; 73; 62; 51; 40; 29; 18; 7; 140; 129; 118; 107;
30 96; 85; 74; 63; 52; 41; 30; 19; 8; 141; 130; 119; 108; 97; 86; 75;
31 64; 53; 42; 31; 20; 9; 142; 131; 120; 109; 98; 87; 76; 65; 54; 43;
32 32; 21; 10; 143; 132; 121; 110; 99; 88; 77; 66; 55; 44; 33; 22; 11];
33
34 %Возвращаем сигнал
```

Рис. 2.2: interleaver

Считаем сгенерированный выходной сигнал из файла test.sig:

```

34 %Получаем сигнал
35 - file=fopen('F:\Users\SashaBoss\Desktop\telecom\knorre\telecom-1\lab8\test.sig', 'r');
36 - IQ_record = fread(file, 'int16');
37 - fclose(file);
38
39 - if(size(IQ_record,1)>8268)%возвращаем к комплексной форме

```

Рис. 2.3: Reading signal

Согласно заданию выделяем реальную и мнимую части сигнала:

```

44 % Передискретизация равна 2, т.е. отсчеты дублируются подряд
45 % Поэтому вещественную часть берем по нечетным числам, а
46 % Комплексную - по четным
47 - real = IQ_record(1:2:end);
48 - imag = IQ_record(2:2:end);
49
50 %возвращаемся к комплексной форме
51 - IQ_record = complex(real,imag);
52
53 - IQ_record = IQ_record(1:2:end);

```

Рис. 2.4: Real and imaginary parts

Демодулируем и строим матрицу Уолша:

```

55 % Демодулируем сигнал
56 - IQ=pskdemod(IQ_record,2);
57
58 % Преобразуем униполярную форму в биполярную
59 - for u=1:length(IQ)
60     if (IQ(u)==0)
61         IQ(u)=-1;
62     else IQ(u)=1;
63     end;
64 - end;
65
66 - signal_to_demodulate2=IQ(length(PRS)+1:end);
67 - signal_to_demodulate1=signal_to_demodulate2./[PRS' PRS' PRS' PRS(1:3)'];
68
69 % Walsh matrix generation by Hadamard matrix index rearrangement
70 % http://www.mathworks.com/help/signal/examples/discrete-walsh-hadamard-transform.html
71 - N=64;
72 - hadamardMatrix=hadamard(N);
73
74 - HadIdx = 0:N-1; % Hadamard index
75 - M = log2(N)+1;
76
77 - binHadIdx = fliplr(dec2bin(HadIdx,M))-'0'; % Bit reversing of the binary index
78 - binSeqIdx = zeros(N,M-1); % Pre-allocate memory
79 - for k = M:-1:2
80     % Binary sequency index
81     binSeqIdx(:,k) = xor(binHadIdx(:,k),binHadIdx(:,k-1));
82 - end
83 - SeqIdx = binSeqIdx*pow2((M-1:-1:0)'); % Binary to integer sequency index
84 - walshMatrix = hadamardMatrix(SeqIdx+1,:); % 1-based indexing
85 % *****

```

Рис. 2.5: Walsh matrix computation

Переходим к двоичному виду:

```

87 - signal2=reshape(signal_to_demodulate1,[64 24]);
88 - Walsh_row_number = zeros(size(signal2,1), 1);
89
90 - for j=1:1:size(signal2,1)
91 -     for i=1:1:length(walshMatrix)
92 -         if(walshMatrix(i,:)==signal2(j,:))
93 -             Walsh_row_number(j)=i;
94 -         end
95 -     end
96 - end
97
98 % Из 10-го числа в бинарный код
99 - for i=1:1:24
100 -     line(i,1:6)= de2bi( Walsh_row_number(i)-1,6);
101 -     line(i,1:6)=line(i,end:-1:1);
102 - end;
103
104 - sig=reshape(line',[1 144]);

```

Рис. 2.6: Conversion to binary

Производим декодирование с учётом перемежения:

```

107 -         for i=1:1:144
108 -             sig2(interleaver(i)+1)=sig(i);
109 -         end
110 -
111 -     trl=poly2trellis(9,[753 561]);
112 -     tblen=9;
113 -     decode_packet=vitdec(sig2,trl,9,'cont','hard');
114 -     msg=[0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
115 -     decoded =decode_packet(tblen+1:end)
116 -     [n1,r1] = biterr(decoded,msg(1:end-tblen))

```

Рис. 2.7: Decoding

Отсутствие ошибок в декодированном пакете сигнализирует о верном решении задачи.

3 Выводы.

В данной работе мы создали модель приемника, выполняющую операции демодуляции, деперемежения и декодирования. Единичный опыт показал пригодность данной модели.