

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ПЕТРА ВЕЛИКОГО

ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ  
КАФЕДРА КОМПЬЮТЕРНЫХ СИСТЕМ И ПРОГРАММНЫХ ТЕХНОЛОГИЙ

Отчет  
по лабораторной работе №8  
на тему  
"Модель телекоммуникационного канала"

выполнил:  
Кыльчик И.В.  
группа: 33501/1  
преподаватель:  
Богач Н.В.

Санкт-Петербург  
2018

## 1. Цель работы

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.

## 2. Постановка задачи

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 отсчетов 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.

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.

## 3. Теоретическая часть

Физическая среда, по которой передаются данные, не может быть абсолютно надёжной. Более того, уровень шума бывает очень высоким, например в беспроводных системах связи и телефонных системах. Ошибки при передаче — это реальность, которую надо обязательно учитывать. Для надёжной передачи нужно применять и комбинировать разные методы кодирования.

**Свёрточные коды**, в отличие от блочных, не делят информацию на фрагменты и работают с ней как со сплошным потоком данных.

Свёрточные коды, как правило, порождаются дискретной линейной инвариантной во времени системой. Поэтому, в отличие от большинства блочных кодов, свёрточное кодирование — очень простая операция, чего нельзя сказать о декодировании.

Кодирование свёрточным кодом производится с помощью регистра сдвига, отводы от которого суммируются по модулю два. Таких сумм может быть две (чаще всего) или больше.

Декодирование свёрточных кодов, как правило, производится по алгоритму Витерби, который пытается восстановить переданную последовательность согласно критерию максимального правдоподобия.

**Перемежитель** (Интерливер от англ. Interleaver) — блок, реализующий перемежение - один из способов борьбы с ошибками. Предназначен для борьбы с пакетированием ошибок путём их разнесения во времени. Использует перемешивание (перемежение) символов передаваемой последовательности на передаче и восстановление её исходной структуры на приёме. Может использоваться как самостоятельно, так и вместе с помехоустойчивым кодом, являясь в таком случае его составным компонентом.

Благодаря перемежению на входе декодера ошибки равномерно распределяются во времени, в идеале образуя поток независимых ошибок.

**Функциями Уолша** называется семейство функций, образующих ортогональную систему, принимающих значения только +1 и -1 на всей области определения.

В принципе, функции Уолша могут быть представлены в непрерывной форме, но чаще их определяют как дискретные последовательности из  $2^n$  элементов. Группа из  $2^n$  функций Уолша образует матрицу Адамара.

Функции Уолша получили широкое распространение в радиосвязи, где с их помощью осуществляется кодовое разделение каналов (CDMA), например, в таких стандартах сотовой связи, как IS-95, CDMA2000 или UMTS.

**Двоичная фазовая манипуляция** — самая простая форма фазовой манипуляции. Работа схемы двоичной ФМн заключается в смещении фазы несущего колебания на одно из двух значений, нуль или  $\pi$  ( $180^\circ$ ). Данный метод является самым помехоустойчивым.

## 4. Ход работы

### 1. Чтение данных

```
fid=fopen('test.sig', 'rb');
A = fread(fid, '2*int16=>int16', 4);
fclose(fid);
```

Данные передатчика у нас хранятся в файле test.sig. Нам известно, что запись сделана с передискретизацией 2, следовательно нам нужен каждый второй результат. Данные помещаются в массив A для дальнейшей обработки.

```
im_part = zeros(1, length(A)/2);
re_part = zeros(1, length(A)/2);
k = 1;
for i = 1:2:length(A)
    re_part(k) = A(i);
    im_part(k) = A(i+1);
    k = k + 1;
end
```

На данном этапе мы разделяем мнимую и реальную части по отдельным массивам.

### 2. Демодуляция

```

IQ = zeros(1, length(im_part));
for i = 1:length(im_part)
    IQ(i) = re_part(i) + 1j*im_part(i);
end
IQdemod = pskdemod(IQ, 2);

```

Производим демодуляцию сигнала и записываем результат в массив IQ.

### 3. PRS

```

N = 2^nextpow2(length(IQdemod));
signalSpectr = fft(IQdemod, N);
syncSpectr = fft(PRS', N);

mult = signalSpectr.*conj(syncSpectr);
result = ifft(mult, N);

n = find(result >= 200);

signal_to_modulate2 = IQdemod((length(PRS)+n(1)):length(IQdemod));
signal_to_modulate1 = signal_to_modulate2.*[PRS' PRS' PRS' PRS'];

signal1 = reshape(signal_to_modulate1, [64 24])';

```

В данном случае синхропосылкой является массив PRS, для нахождения начала посылки применяем метод быстрой корреляции. Затем мы должны восстановить последовательность которая была до умножения на PRS.

### 4. Таблица функций Уолша

```

N=64;
hadamardMatrix=hadamard(N);

walsh_row = zeros(1,24);
for i = 1:24
    ind = ismember(hadamardMatrix, signal(i,:), 'rows');
    walsh_row(i) = find(ind == 1);
end
walsh_row = (walsh_row - 1)';
sig_matrix = de2bi(walsh_row, 6, 'left-msb');

```

На данном шаге нам надо получить последовательность до применения функций Уолша. В вектор walsh\_row мы записываем индексы из матрицы Уолша, а затем переписываем номера индексов как двоичную последовательность в матрицу sig\_matrix.

### 5. Перемежение бит.

```

sig = reshape(sig_matrix', [1 144]);
convcode(int16(interleaver+1)) = sig;

```

Выполняем перемежение, тем самым получая последовательность после кодирования сверточных кодом.

### 6. Сверточные коды

```
tr1=poly2trellis(9,[753 561]);
packet=vitdec(convcode,tr1,1,'trunc','hard');

packetR=[packet(1:16); packet(17:32); packet(33:48)];
code=bi2de(packetR,'left-msb');
```

Данный раскодирует нашу последовательность. В результате мы получаем ту последовательность которую и ожидали: ID = 4, период = 100 мс, номер пакета = 373.

Мы можем проверить правильность передачи используя контрольную сумму.

```
hDetect=crc.generator('Polynomial','0x1021');
decode_with_crc=generate(hDetect,packet);
crc_send = decode_with_crc(73:88);
```

Данный код генерирует нам точно такую же контрольную сумму что и передатчик. Следовательно посылка получена верно.

## 5. Вывод

В данной работе мы применили навыки полученные в ходе предыдущих семи работ. Мы смогли произвести демодуляцию сложного сигнала и получить правильный ответ на выходе.

Передатчик генерирует помехозащитный сигнал почти из девяти тысяч бит. Данный сигнал сможет выдержать шум примерно в 10 раз превышающий его амплитуду. На приемной стороне мы все равно получим верную исходную последовательность.