

# Разработка модели физического уровня OFDM модема часть 2



Минаев Илья Геннадьевич, 4 апреля 2011г.

OFDM технология предполагает как можно более эффективное использование спектра, за счёт передачи данных параллельно по множеству близко расположенных друг к другу узкополосных частотных подканалов. При этом на части частот передаётся информация, часть несущих используется для передачи пилот-сигналов, а часть частот не используются. OFDM сигналы успешно противостоят многолучёвости, за счёт введения циклического префикса перед каждым символом- повторяющейся последовательность не несущей информации, на которую приходится всё влияние многолучёвости. В результате на выходе передатчика мы должны получить сигнал такой структуры:

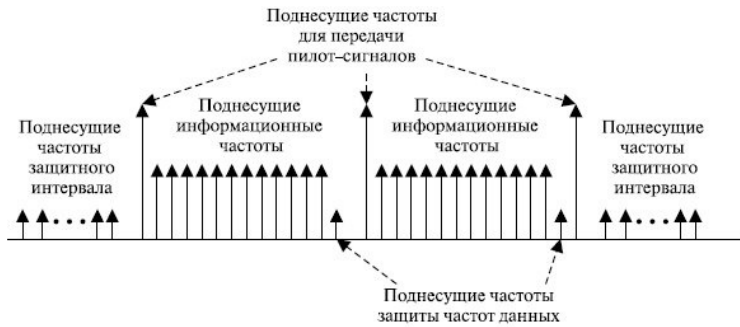


Рис 1. Структура OFDM сигнала

Теперь подробно опишу манипуляции с сигналом от ИС до ПС, по ходу буду приводить фрагменты программы моделирующей OFDM систему.

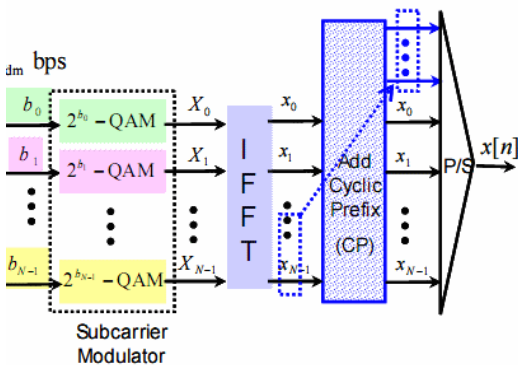


Рис 2. Модулятор OFDM

Цифровой сигнал, преобразованный в системе кодеров, поступает на полосовой модулятор, где собственно моделируется чаще всего какой-то PSK или QAM.

```
% A: Параметры системы
% -----
M = 4; % созвездие QPSK
no_of_data_points = 128; % информационные значения
block_size = 8; % размер каждого блока ofdm
cp_len = ceil(0.1*block_size); % длина циклического префикса
no_of_ifft_points = block_size; % точки для FFT/IFFT
no_of_fft_points = block_size;
% -----
% B: % +++++ ПЕРЕДАТЧИК +++++
% -----
% 1. Генерация случайных бит
data_source = randsrc(1, no_of_data_points, 0:M-1);
```

```

figure(1)
stem(data_source); grid on; xlabel('Data Points'); ylabel('transmitted data phase representation')
title('Transmitted Data "0"')
% 2. QPSK модуляция
qpsk_modulated_data = pskmod(data_source, M);
scatterplot(qpsk_modulated_data);title('MODULATED TRANSMITTED DATA');

```

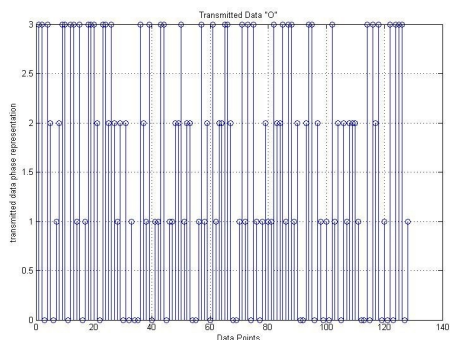


Рис 3 исходный сигнал

Далее необходимо с помощью мультиплексора разбить один поток данных на множество потоков, которые будут поступать на блок обратного БПФ. Сразу возник вопрос: как реализовать мультиплексирование потока данных по частоте? Эту операцию выполнят следующим образом: последовательный поток преобразуют в матрицу, каждый столбец которой далее будет преобразован в OFDM сигнал. После ОБПФ нужно создать циклический префикс (ЦП). Выбор его длины - это компромисс. С одной стороны, чем длиннее он будет, тем устойчивее сигнал к многолучёвости. С другой стороны скорость в подканалах и так не высока, а добавление длительного ЦП сделает её ещё ниже. На практике длительность ЦП равна наибольшей задержке в канале, что может достигать четверти длины символа.

```

% 3. Создание блока ОБПФ
% Задаёмся числом столбцов матрицы
num_cols=length(qpsk_modulated_data)/block_size;
data_matrix = reshape(qpsk_modulated_data, block_size, num_cols);
% Далее формируем пустую матрицу, в которую перейдут данные с IFFT
cp_start = block_size-cp_len;
cp_end = block_size;
% Порядок операций над столбцами и создание ЦП
for i=1:num_cols,
    ifft_data_matrix(:,i) = ifft((data_matrix(:,i)),no_of_ifft_points);
    % Вычисление и добавление ЦП
    for j=1:cp_len,
        actual_cp(j,i) = ifft_data_matrix(j+cp_start,i);
    end
    % Добавление ЦП в существующий информационный блок сигнала
    ifft_data(:,i) = vertcat(actual_cp(:,i),ifft_data_matrix(:,i));
end

```

Далее необходимо произвести демультимплексирование, дабы перевести полученные временные отсчёты в последовательный поток данных.

```

% 4. Преобразование данных в последовательный поток
[rows_ifft_data cols_ifft_data]=size(ifft_data);
len_ofdm_data = rows_ifft_data*cols_ifft_data;
% Вид передаваемого OFDM сигнала
ofdm_signal = reshape(ifft_data, 1, len_ofdm_data);
figure(3)

```

```
plot(real(ofdm_signal)); xlabel('Time'); ylabel('Amplitude');
title('OFDM Signal');grid on;
```

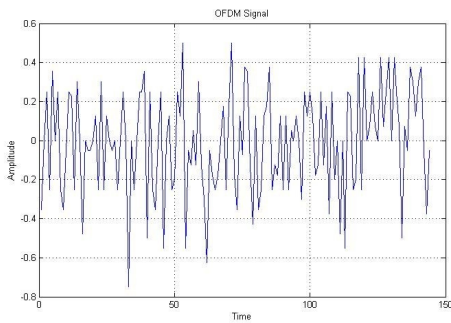


Рис 4. OFDM сигнал

Но передавать такой сигнал нельзя- его нужно усилить. Те же КАМ предъявляют высокие требования к линейности усилителей, что делает невозможным использование мощных усилителей. Одним из свойств, присущих только системам с OFDM, является отношение пиковой мощности к средней мощности (Peak to Average Power Ratio (**PAPR**)), которые отчетливо видны на рис 4. Величина PAPR складывается из двух составляющих. Первой составляющей, определяющей PAPR, является количество точек созвездия для каждой из несущих. При использовании квадратурно-амплитудной модуляции с количеством точек на созвездии большим, чем у 4-QAM, эти точки не будут соответствовать постоянной мощности. Для 256-QAM PAPR равен 4,2дБ (в предположении, что точки созвездия распределены равномерно). Однако эта величина очень мала по отношению к тому слагаемому PAPR, которое является результатом использования множества поднесущих. При каждом удвоении количества поднесущих пиковая мощность возрастает на 6 дБ, в то время как средняя мощность – только на 3 дБ. Для 128-ми поднесущих эта составляющая PAPR равна 21 дБ, а общая величина PAPR равна 25,2 дБ. Из-за проблемы с высоким пиком среднего коэффициента в OFDM, при использовании нелинейных усилителей нарушается спектральная сетка OFDM сигнала, что приводит к множеству ошибок.

Особенно остро проблема встала в области спутникового цифрового телевидения, где нужна высокая мощность, и используется 256-QAM. Но и отказываться от использования OFDM не стали, ввиду её хорошей защиты от многолучёвости. Было предложено много методов для смягчения действия усилителя. Это простейшие обрезаящие и фильтрующие методы, но они приводят к увеличению битовой ошибки. Альтернативные методы основаны на кодировании, другие на *Многократном Представлении Сигнала* (MSR), но при их использовании дополнительно нужно передавать информацию об изменении с передатчика на приёмник, что ухудшает спектральную эффективность. Достаточно недавно был придуман метод, который не требует дополнительной передачи. Метод *Резервирование Тона* (Tone Reservation (**TR**)) предполагает выделение дополнительных поднесущих, которые несут дополнительную информацию, которая минимизирует PAPR (прошу прощения за тофтологию).

В TR основная идея состоит в том, чтобы зарезервировать некоторые поднесущие OFDM, называемые *Пиковые Тоны Сокращения* (PRT) для снижения PAPR. Эти поднесущие не несут информации о данных, и используются только для того, чтобы уменьшить PAPR. Этот метод ограничивает вектор данных, и сокращает пиковый вектор, при этом не имеет связи с сеткой частот. Из-за этого условия в сигнале не возникают нелинейные искажения. Значения, использованные для преобразования сигнала, просто кодируются и передаются.

Схема TR.

Суть состоит в том, что бы уменьшить PAPR сигнала  $x = \text{IFFT}(X)$  таким образом, что бы  $\text{PAPR}(X+C) < \text{PAPR}(X)$ , где  $C$ - добавленный пиковый уменьшающий сигнал.

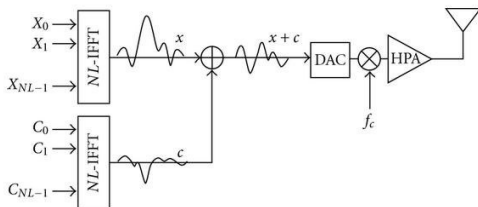


Рис 5. Схема TR

Предполагается, что объединённый сигнал  $(x+c)$  будет иметь почти постоянную мощность. В действительности же добавление сигнала  $c$  приведёт к увеличению мощности на величину. Встаёт вопрос: как получить корректирующий сигнал? существует несколько методов, приведу один из них:

#### TR-Clipping-Based Technique

Этот метод состоит в резком усечении входного OFDM сигнала.

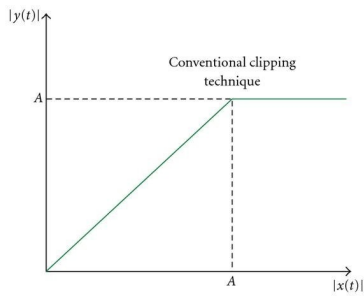
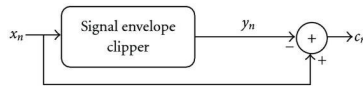


Рис 6 Метод усеечения

Тогда корректирующий сигнал получается вычитанием из входного сигнала урезанного. После чего он поступает к некоему фильтру и преобразуется согласно принципу TR.



Далее представлен блок генератора, уменьшающего пик сигнала.

Рис 7 Схема генерации корректирующего сигнала

Сохраняются только те значения корректирующего сигнала, которые относятся к PRT, остальные обнуляются:

$$C_k = \text{FFT}(c_n)$$

$$\hat{C}_k = C_k \text{ if } k \in \text{PRT} ; 0 \text{ if } k \notin \text{PRT}$$

При каждом повторении алгоритм обновляет вектор  $X_k$  ( $X_k = \text{FFT}(x_n)$ ), добавляя к этому вектор  $\hat{C}_k$ .

$$X_{k+1} = X_k + \mu \hat{C}_k$$

где  $\mu$  - шаг метода градиента. На схеме показан принцип уменьшение PAPR градиентным методом с усеечением в частотной области. Блок IB фильтр гарантирует совпадение пиков сигнала с компонентами корректирующего сигнала. Поскольку изменение выполняется в частотной области, в этот алгоритм нужно просто внести спектральные ограничения, таким образом снизим мощность тонов.

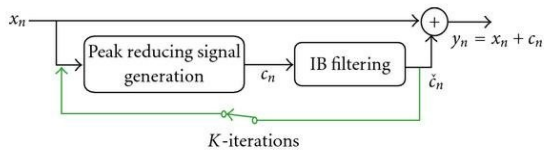


Рис 8. принцип уменьшение PAPR

```

% -----
% C: % +++++ метод уменьшения PAPR усеечением +++++
% -----

avg=0.4;
clipped=ofdm_signal;
for i=1:length(clipped)
    if clipped(i) > avg
        clipped(i) = avg;
    end
    if clipped(i) < -avg
        clipped(i) = -avg;
    end
end

figure(4)
plot(real(clipped)); xlabel('Time'); ylabel('Amplitude');
title('clipped Signal');grid on;

```

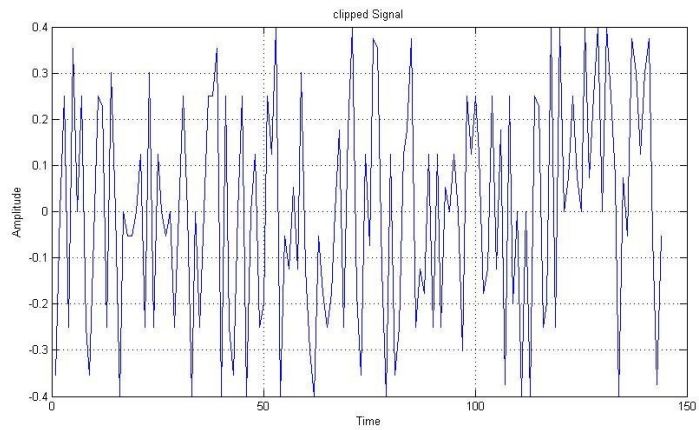


Рис 9. Усечённый сигнал

далее подаём сигнал на нелинейный усилитель

```

% -----
% D: % +++++ HPA +++++
% -----

% Чтобы показать влияние усилителя просто добавим случайный комплекс шума
% когда мощность превышает среднее значение, в противном случае ничего не добавляем.

% 1. Генератор случайного шума
noise = randn(1,len_ofdm_data) + sqrt(-1)*randn(1,len_ofdm_data);

% 2. Передача сигнала прошедшего через усилитель
Рассмотрим передачу 2 методами: с усечением и без
%без усечения
for i=1:length(ofdm_signal)
    if ofdm_signal(i) > avg
        ofdm_signal(i) = ofdm_signal(i)+noise(i);
    end
    if ofdm_signal(i) < -avg
        ofdm_signal(i) = ofdm_signal(i)+noise(i);
    end
end
end
figure(5)
plot(real(ofdm_signal)); xlabel('Time'); ylabel('Amplitude');
title('OFDM Signal after HPA');grid on;

```

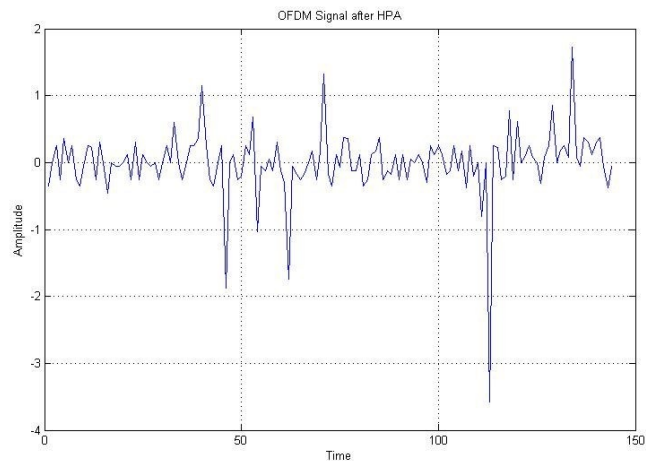


Рис 10. Усиленный не усечённый сигнал

```

%c усечением
avg=0.4;
for i=1:length(clipped)
    if clipped(i) > avg
        clipped(i) = clipped(i)+noise(i);
    end
    if clipped(i) < -avg
        clipped(i) = clipped(i)+noise(i);
    end
end
figure(6)
plot(real(clipped)); xlabel('Time'); ylabel('Amplitude');
title('clipped Signal after HPA');grid on;

```

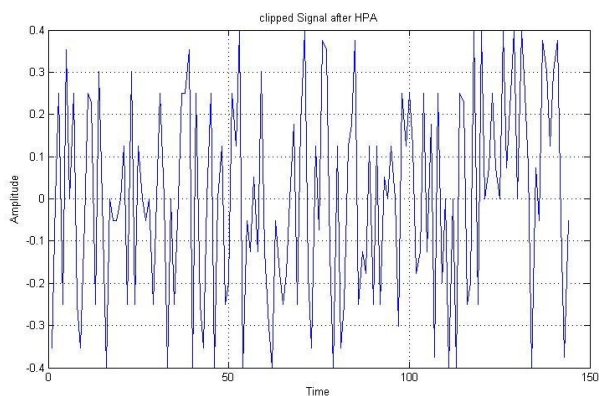


Рис 11. Усиленный усечённый сигнал

Далее опишем канал:

```

% -----
% E: %   +++++   КАНАЛ   +++++
% -----
% Создание многолучевого канала
channel = randn(1,block_size) + sqrt(-1)*randn(1,block_size);

```

Далее в приёмнике мы должны выполнить обратные действия: преобразовать поток данных в матрицу, отбросить циклический префикс, сделать БФП, и сформировать итоговую информационную последовательность.

```

% -----
% F: %   +++++   ПРИЁМНИК   +++++
% -----
% 1. ofdm сигнал, прошедший через канал
after_channel = filter(channel, 1, ofdm_signal);
% 2. шум
awgn_noise = awgn(zeros(1,length(after_channel)),0);
% 3. добавление шума в сигнал
recvd_signal = awgn_noise+after_channel;
% 4. Преобразование последовательного потока данных в параллельный
recvd_signal_matrix = reshape(recvd_signal,rows_ifft_data, cols_ifft_data);
% 5. Отбрасывание ЦП
recvd_signal_matrix(1:cp_len,:)=[];
% 6. Выполнение БПФ

```

```

for i=1:cols_ifft_data,
    % БПФ
    fft_data_matrix(:,i) = fft(recvd_signal_matrix(:,i),no_of_fft_points);
end
% 7. Преобразование в последовательный поток
recvd_serial_data = reshape(fft_data_matrix, 1,(block_size*num_cols));
% 8. Демодулятор
qpsk_demodulated_data = pskdemod(recvd_serial_data,M);
figure(7)
stem(qpsk_demodulated_data,'rx');
grid on;xlabel('Data Points');ylabel('received data phase representation');title('Received Data "X"')

```

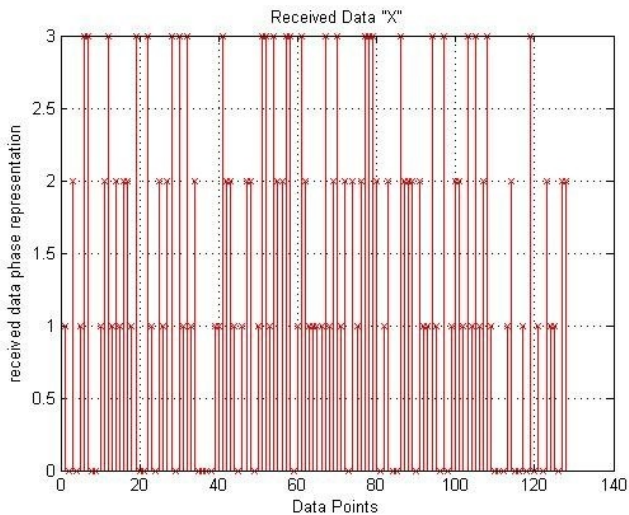


Рис 12. Итоговый сигнал без усечения

```

% -----
% F: %   +++++ ПРИЁМНИК сигналов с усечением +++++
% -----
% 1. ofdm сигнал, прошедший через канал
after_channel = filter(channel, 1, clipped);
% 2. шум
awgn_noise = awgn(zeros(1,length(after_channel)),0);
% 3. добавление шума в сигнал
recvd_signal = awgn_noise+after_channel;
% 4. Преобразование последовательного потока данных в параллельный
recvd_signal_matrix = reshape(recvd_signal,rows_ifft_data, cols_ifft_data);
% 5. Отбрасывание ЦП
recvd_signal_matrix(1:cp_len,:)=[];
% 6. Выполнение БПФ
for i=1:cols_ifft_data,
    % БПФ
    fft_data_matrix(:,i) = fft(recvd_signal_matrix(:,i),no_of_fft_points);
end
% 7. Convert to serial stream
recvd_serial_data = reshape(fft_data_matrix, 1,(block_size*num_cols));

```

```

% 8. Demodulate the data
qpsk_demodulated_data = pskdemod(recvd_serial_data,M);
figure(8)
stem(qpsk_demodulated_data,'rx');
grid on;xlabel('Data Points');ylabel('received data phase representation');title('Received Data clipped "X"')

```

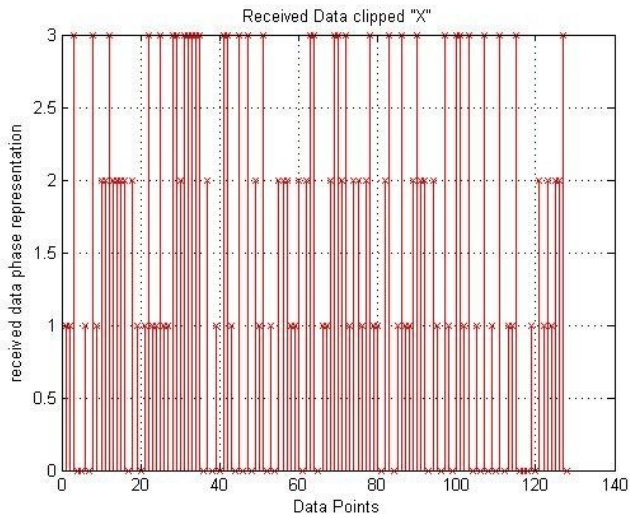


Рис 13. Итоговый сигнал с усечением

Данные полученные при исследовании подобных моделей показали примерно следующие результаты:

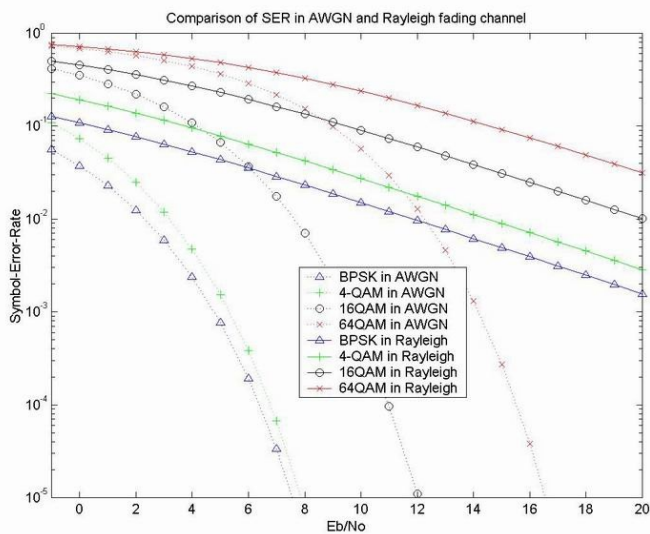


Рис 14. Вероятность символьной ошибки

При увеличении точек кам вероятность ошибки возрастает, что вообще-то логично. Зато возрастает и скорость передачи. Так же исследования показали, что в случаях, когда максимальная задержка в многолучевом канале не превышала длины ЦП, вероятность ошибки не увеличивалась.

Все используемые мной модели, Вы можете взять здесь: <http://narod.ru/disk/9208333001/OFDM.rar.html>

