Design of high-order QAM modem algorithm

Yinghui Chang*, Changlong Liu b, *
The 54th Research Institute of CETC, Shijiazhuang, China

a 17703318067@163.com, 
b, * Corresponding author e-mail: liuchanglong@chinaunicom.com.cn

Abstract. More and more attention has been paid on the issue of bandwidth utilization. Nowadays, with limited frequency spectrum resources, the capacity of traditional communication system has not been able to meet the demands of users. Quadrature Amplitude Modulation (QAM) has become a critical technical solution for broadband wireless access and wireless video communication due to its advantages of high frequency spectrum utilization and high-power spectrum density. In order to further improve the control information transmission performance of high-order QAM, we propose a high-order QAM modulation and demodulation algorithm, which is designed from the aspects of interleave between channels, leading sequence, constellation modulation, frequency domain sampling and (Orthogonal Frequency Division Multiplexing) OFDM generation, etc. The simulation results demonstrate that the algorithm can ensure the reliable reception of control information.

Keywords: High-order QAM, modulation and demodulation, transmission performance, frequency band utilization, broadband wireless access.

1. Introduction
With the development of modern communication technology, especially the rapid development of mobile communication technology, new demands and new services are constantly generated, which lead to the increasing strain of frequency resources [1-3]. Large amounts of multimedia data need to be transmitted in limited bandwidth. Thus, spectrum efficiency has become an important topic. Owing to the advantages of high spectrum efficiency and high-power spectral density, QAM (Quadrature Amplitude Modulation) has turn to an important technology of the broadband wireless access and wireless video communication scheme [4-5]. QAM system has been widely used in many broadband application fields, such as digital TV broadcasting, Internet broadband access. QAM can be used for digital modulation [6]. Specifically, Digital QAM has different types, such as 4QAM, 8QAM, 16QAM, 32QAM, and so on. Among them, 16QAM and 32QAM are widely used in digital cable TV system. Nowadays, satcom modems with 16QAM modulation technology has appeared in the international market, for instance, the CDM-600 produced by the American enterprise COMTECHEFDATA. The satcom modem supports speeds up to 20Mbps. The rapid development of wireless communication technology requires higher data transmission rate, transmission efficiency and bandwidth utilization. That means, the choice of efficient and feasible means of modulation and demodulation plays a crucial
role in improving the effectiveness and reliability of signals [7-8]. In this context, QAM has become an important solution for broadband wireless access and wireless video communication.

2. Algorithm design

2.1. Channel interlacing
Due to the large fading difference of each transmission channel, the transmission bit error rate is relatively high. Therefore, information interleaving between channels is designed to share the decoding pressure of error bit channel and improve the reliability of multi-channel transmission.

2.2. Leading sequence
The leading sequence is composed of short training sequence and long training sequence, which is used to realize synchronization, correction of frequency offset and phase offset at the receiver end. The specific frame structure can be seen in Fig. 1. The sampling frequency of the leading sequence is 200MHz, which is consistent with the sampling rate of the data part in the frame structure. Firstly, the short training sequence is a short sequence of 512 long sampling points that is repeated for 10 times. The receiver utilizes its cross-correlation to carry out rough synchronization. On the normalized cross-correlation result, the frame head position of rough synchronization is found by setting a threshold. Secondly, the long training sequence comprises of two sequences with the same length of 2048 sampling points and truncated 1024 sampling points, which is mainly employed for carrier synchronization, frame head precision synchronization and fine deviation estimation.

![Fig.1](image)

**Fig.1** Short training sequence and long training sequence (the number in the figure is the number of sampling symbols)

In this system, each channel has an independent training sequence because it has the maximum eight-channel data. The procedure of generating long training sequence and short training sequence is divided into two steps. The first step is to generate a specific source code sequence, which is produced by a given generation polynomial consisting of 0 and 1. Then, the source code sequence is modulated by pi/2-BPSK into the corresponding short training sequence and long training sequence. Here, 0 in the source code sequence is modulated to $-1/\sqrt{2}(1+1j)$, and 1 is modulated to $1/\sqrt{2}(1+1j)$. After that, the modulated signal goes through OFDM multi-carrier map and IFFT transform to finally obtain the training sequence.

2.2.1. Short training sequence. In this scheme, since all of the frames sent by the four channels have short training sequences, four groups of independent short training sequences need to be generated. First of all, a 96-long pseudo-random sequence consisting of 0 and 1 is generated through using the generating polynomial $g(x) = 1 + x^4 + x^5 + x^6 + x^8$. The initial state is set as [1 0 1 0 0 1 0 1]. The 96-long 01 random sequence is averagely cut to 4 parts and then successively provided to the four channels as the 12 long source sequences generating short training sequences, respectively, just as the Table 1 shown below:
### Table 1. A source code sequence used to generate short training sequence

| Channel 1 | 1 0 1 0 0 1 0 1 |
| Channel 2 | 1 1 1 0 1 1 1 0 |
| Channel 3 | 1 1 1 1 1 0 1 1 |
| Channel 4 | 1 0 0 1 1 1 0 1 |

Then, the source code sequence of each channel is modulated into complex symbols by pi/2-BPSK, that is, 0 is modulated into \(-1/\sqrt{2}*(1+1j)\), and 1 is modulated into \(1/\sqrt{2}*(1+1j)\). Finally, the modulated 12 symbols are sequentially mapped to positions 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, and 54-th of the 64 subcarriers, and then subjected to IFFT transformation and multiplied by a power normalization factor \(\sqrt{13}/3\).

The above short training sequences are generated in the frequency domain for satisfying the system transmission in the frequency band with a certain bandwidth. Considering that the system is designed for four-channel transmission, the design of short training sequence should meet the orthogonal design of the four channels as far as possible. According to the sub-carrier interval of the system, and only four channels are considered in this system, the relatively simple 64-point FFT is selected as the multi-carrier modulation mode of the short training sequence, and the frequency mapping symbol is 12 long. Additionally, the mapping symbol is 4 times to the sub-carrier, so as to ensure the orthogonality of the sequence after FFT between the four channels. The location of the 64 subcarriers mapped to the limited and bandwidth requirements is listed above. After sampling in the 32x frequency domain and 8x sampling in the time domain, the short training sequence of each channel is 10 repeats of the intercepted 512 points. Therefore, we can conclude that the short training sub-sequence finally used for receiver frame header capture is 512 long, not just 12 long.

Specifically, in the transmitter and receiver, the short training sequence is generated in advance to store in the ROM, and is directly read and used in the corresponding module. At the receiving end, the short training sequence is only utilized in the time domain operation. Hence, the above generation process, although involves the multi-carrier modulation process, the final frame header capture process has no correlation to the frequency domain characteristics of the specific sequence.

#### 2.2.2. Long training sequence

Similar to the generation method of short training sequence, 448 long pseudo-random sequence comprising of 0 and 1 is generated by the same generating polynomial, and the initial state is set to \([1 0 0 0 1 0 0 1]\). The 448-long 01 random sequence is averagely divided into 8 segments, which are provided to eight channels in turn as 56 long source sequences for generating long training sequences. The details are shown in the Table 2 below.

### Table 2. A source code sequence used to generate the growth training sequence

| Channel 1 | 1001001010101001111011010011001101101 |
| Channel 2 | 000110000111010101111100100001001111111 |
| Channel 3 | 00110100000010001110010111000011010011101101 |
| Channel 4 | 0000010110110111101101011000111111101101100100 |

The source code sequence of each channel is modulated by pi/2-BPSK to a complex symbol, namely, 0 is modulated to \(-1/\sqrt{2}*(1+1j)\), 1 is modulated to \(1/\sqrt{2}*(1+1j)\). Finally, the 52 symbols after modulation are mapped to the positions from the 5-th to the 60-th subcarriers in the 64 subcarriers in turn, and the IFFT transformation (sampled in the 32-fold frequency domain here) is performed and then multiplied by the power normalization factor \(\sqrt{13}/4\).

In the concrete implementation, since the short training sequence and the long training sequence are both fixed values, the final generated symbol sequence only needs to be stored in the ROM, and there is no requirement to be generated temporarily again.
2.3. Modulation constellation

In the proposed scheme, orthogonal frequency division multiplexing (OFDM) is employed for transmission. The QPSK modulation is used for control frame and the 1024QAM/2048QAM modulation is utilized for data frame. The mapping relationship between bit and modulation symbol is shown in Fig. 2 and Fig. 3, respectively. Due to the small amount of data and the high reliability of transmission in engineering, the control part adopts small low-order constellation modulation. Particularly, the QPSK constellation commonly used in engineering is chosen here, which maps two bits of information into a complex symbol. In the user data part, taking the requirement of high throughput transmission into account, 1024QAM/2048QAM constellation is adopted to obtain the high-order 1024QAM/2048QAM constellation, which maps the 10-bit/11-bit information into a complex symbol. Note that the symbolic power is normalized after mapping.

The program name: CABLEOFDM_tx_Modulate.vhd
Input: i_clk - input drive clock signal with clock rate of 58.368MHz;
i_rst - Global reset signal, high level reset;
i_frame_st - Input frame header indicator, high level valid;
i_bit_st - Input data frame header indicator, high level valid;
i_bit_en - Input data is a valid indicator. High level valid;
i_bit - input data;
Output:o_frame_st - output frame header indicator, high level valid;
2.4. Frequency domain sampling and OFDM generation

Pilot insertion function: The pilot data is inserted into the position of the corresponding subcarrier.

Implementation: Store 284 pilot data in the ROM. The data sent from the last module is reshaped in the way of ping-pong RAM, leaving the space where the pilot is, then read the pilot data from ROM through the counter, insert it into the data center, and finally normalize the data and output it to the next module.

The program name: CABLEOFDM_tx_AddPilotSyms.vhd
Input: i_clk - input drive clock signal with clock rate of 58.368MHz;
i_rst - Global reset signal, high level reset;
i_frame - Input frame header indicating signal, high level valid;
i_data_frame - Input data frame header indicating signal, high level valid;
i_data_en - Input data valid indication signal, high level valid;
iData_real - Input I-channel data;
iData_imag - Input Q-channel data;
Output:o_frame - Output frame header indicating signal, high level valid;
o_data_frame - Output data frame header indicating signal, high level valid;
o_data_en - Output data valid indication signal, high level valid;
oData_real - Output I-channel data;
oData_imag - Output Q-channel data;

2.4.1. Pilot. OFDM frames are inserted with some reference signals known to the receiving end, which are named pilot. The pilot of the system is distributed in the subcarrier and utilized for phase error correction, channel estimation and equalization at the receiving end. Here, pilot reference sequence is generated by the built-in function of MATLAB. In this system, all pilots to generate hyper-frames are composed of specific 1 and -1 sequences. The generation process produces 0 and 1 bitstreams for the given generation polynomial, and then maps them to the corresponding 1 and -1, in which 0 and 1 map to -1 and 1, respectively. In this scheme, the generated polynomial is \( g(x) = 1 + x + x^6 \).

The initial state is set to [1 0 0 1 0 1]. Since the generated sequence is relatively long, every 4 bits is represented as hexadecimal (right high), then the pilot frequency used for a hyper-frame in this program is specifically expressed as:

\[5E5C94BB95DF02C82F2E4ADDCAE7016497172DE665F3803AC839673BAF14815E5C94BB95DF02C82\]
Store the 284-pilot data in ROM. The data sent from the last module is reshaped in the way of ping-pong RAM, leaving the space where the pilot is, then read the pilot data from ROM from the counter, and insert it into the data center, and finally normalize the data and output it to the next module.

The program name: CABLEOFDM_tx_AddPilotSyms.vhd
Input: i_clk - input drive clock signal with clock rate of 58.368MHz;
i_rst - Global reset signal, high level reset;
i_frame - Input frame header indicating signal, high level valid;
i_data_frame - Input data frame header indicating signal, high level valid;
i_data_en - Input data valid indication signal, high level valid;
i_v_data_real - Input I-channel data;
i_v_data_imag - Input Q-channel data;
Output: o_frame - Output frame header indicating signal, high level valid;
o_data_frame - Output data frame header indicating signal, high level valid;
o_data_en - Output data valid indication signal, high level valid;
o_v_data_real - Output I-channel data;
o_v_data_imag - Output Q-channel data;
The data length of the input pilot module is 112-1590, that is, 284 OFDM blocks, and each OFDM block contains 1590 data corresponding to effective subcarriers. The function of the pilot module is to insert the pilot frequency and the subcarriers corresponding to the invalid data of each OFDM block into the input OFDM block.

2.4.2. Sampling in time domain. In this system, 8 times time-domain sampling is adopted, that means, after 2048 IFFT transformation, 8 times time-domain sampling is conducted.

The program name: CABLEOFDM_tx_UpSample_32x.vhd
Input: i_clk_12x - Input drive clock signal;
i_clk_32x - Input drive clock signal;
i_rst - Global reset signal, reset at high level;
i_frame - Input frame header indication signal, high level valid;
i_data_frame - Input data frame header indication signal, high level valid;
i_data_en - Input data valid indication signal, high level valid;
i_v_data_real - Input I-channel data;
i_v_data_imag - Input Q-channel data;
Output:o_frame - Output frame header indication signal, high level valid;
o_data_frame - Output data frame header indication signal, high level valid;
o_data_en - Output data valid indication signal, high level valid;
o_v_data_real - Output I-channel data;
o_v_data_imag - Output Q-channel data;
The input data length of each frame of the upsampling module is 113*2048, namely 118 OFMD blocks. Each OFDM block contains data corresponding to 2048 subcarriers, and the data of two adjacent subcarriers are separated by 12 low-rate clocks. The data length of each frame of the output module is also 113*2048, and the data of the two adjacent subcarriers are separated by 32 high-rate clocks.

2.5. IFFT module
Module function: the 2048 data are respectively mapped to 2048 subcarriers to complete the discrete Fourier transform. The data output from the upsampling module is first transferred to IFFTShift by means of ping-pong RAM, and then the sent to the IFFT IP core. IFFT modulation is completed by means of IP core, and data normalization is realized by adjusting the scale factor of IFFT.

The program name: CABLEOFDM_tx_UpSample_32x.vhd Input: i_clk - Input drive clock signal with clock rate of 200MHz;
i_rst - Global reset signal, reset at high level;
i_frame - Input frame header indication signal, high level valid;

![Fig.5 IFFT hardware structure block diagram](image)

i_data_frame - Input data frame header indication signal, high level valid;
i_data_en - Input data valid indication signal, high level valid;
iv_data_real - Input I-channel data;
iv_data_imag - Input Q-channel data;
Output:o_frame - Output frame header indication signal, high level valid;
o_data_frame - Output data frame header indication signal, high level valid;
o_data_en - Output data valid indication signal, high level valid;
ov_data_real - Output I-channel data;
ov_data_imag - Output Q-channel data;

The input data length of each frame of IFFT module is 113*2048, that is, 113 OFDM blocks, each of which contains data corresponding to 2048 subcarriers. After IFFT carries out IFFTShift, IFFT transformation and CP operation on the input 113 OFDM blocks, the data length of each frame of output IFFT module is 113*(2048+128).

2.6. Group frame (control frame + user data)
Combine the data of the OFDM block controlling the frame with the other 112 OFDM blocks. The realization method is to store the data of the control frame block in the ROM, and determine when to read the data from the ROM through the frame header signal and the counting signal. The following Fig. 6 shows the hardware implementation block diagram of this module:

![Fig.6 Block diagram of hardware structure](image)

The program name: CABLEOFDM_tx_construct_frame.vhd
Input: i_clk - Input drive clock signal with clock rate of 58.368MHz;
i_rst - Global reset signal, reset at high level;
i_frame - Input frame header indication signal, high level valid;
i_data_frame - Input data frame header indication signal, high level valid;
i_data_en - Input data valid indication signal, high level valid;
iv_data_real - Input I-channel data;
iv_data_imag - input Q-channel data;
Output: o_frame - Output frame header indication signal, high level valid;
o_data_frame - Output data frame header indication signal, high level valid;
o_data_en - Output data valid indication signal, high level valid;
ov_data_real - Output I-channel data;
ov_data_imag - Output Q-channel data;

Input a frame data into the group frame (mac) module, which consists a total of 113 blocks, and each block length is 1601, corresponding to 1601 effective subcarriers. This module inserts the mac block stored in ROM into the frame header, and outputs a frame of data length of 112*1601.

3. Simulation
Taking the end-to-end bit error rate (BER) performance of the overall system, CRC block-divided block error rate BLER and the physical layer whole frame error rate FER as measurements, the system performance simulations are carried out. In details, the performance is shown in Fig. 7.

In Fig. 7, it can be observed that under the condition of BER equaling to 1e-6, the signal to noise ratio (SNR) of the system data information transmission performance is about 34 dB, and the BLER is about 1e-3 and the FER is 4e-2.

The simulation is limited to ideal frequency offset and phase offset conditions under the Gaussian conditions. If each module of the receiver is considered comprehensively, and the BER is supposed to be less than 1e-6, and the SNR should be kept at less than 36dB.
Due to that the control information is modulated through using QPSK and the lower 1/2 convolutional code, thus the performance is significantly superior to that of the data information transmission. From Fig. 8, we can find that the BER of the control information can be reduced to $10^{-6}$ when SNR is about 4 dB. Compared to the data information, there is about 30 dB performance gain, which can fully guarantee the reliability of receiving the control information.

4. Conclusion
In order to promote high-order QAM control information transmission performance, we put forward a high-order QAM modulation and demodulation algorithm. The proposed system is improved from the perspective of the channel interlacing, leading sequence, constellation modulation, frequency sampling and OFDM generation. The simulation results show that the algorithm can guarantee the reliability of receiving the control information, which can support 1024 QAM and 2048 QAM and 4096 high order modulation.

References
[1] Wanjian Jiang. Implementation of QAM Modulation Demodulation Based on Simulink. 2018, 1(1).
[2] Dayong Hu. A Simulation Method of 16QAM Modulation and Demodulation Based on Matlab Platform[A]. IEEE Beijing Section, Beijing Jiaotong University. Proceedings of 2017 7th IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies[C]. IEEE Beijing Section, Beijing Jiaotong University: IEEE BEIJING SECTION, 2017:4.
[3] Ahmed Nasraden Milad, M. Aziz M, Rahmadwati. Neural Network Demodulator for Quadrature Amplitude Modulation (QAM)[J]. International Journal of Advanced Studies in Computer Science and Engineering, 2016, 5(7TB).
[4] Zachaeus Kayode Adeyemo, Ibraheem Folarin Bello, Damilare Oluwole Akande, Abdulahi Abiodun Badrudeen. Effect of Rician Factor in Satellite Communication Signal with MAP-MQAM Demodulation[J]. International Journal of Wireless and Microwave Technologies (IJWMT), 2015, 5(5).
[5] Nagendra Kumar, Praveen Kumar Singya, Vimal Bhatia. On ASER performance of higher order QAM schemes in two-way multiple-relay networks under imperfect CSI[J]. IET Communications, 2020, 14(10).
[6] Ayan Banerjee, Anindya S. Dhar. Novel architecture for QAM modulator - demodulator and its generalization to multicarrier modulation[J]. Microprocessors and Microsystems, 2005, 29(7).
[7] Mobile Communications; Reports on Mobile Communications Findings from Faculty of Sciences Semlalia Provide New Insights (Zf and Mmse Detectors Performances of a Massive Mimo System Combined With Ofdm and M-qam Modulation) [J]. Computers Networks & Communications, 2020.
[8] Takahiro Kodama, Gabriella Cincotti. OCDM systems with PSK and QAM codes[J]. Optics Communications, 2020, 465.