FPGA Based Iterative JSC Decoding of Huffman Encoded Data for a Communication System

1. Anjana P M, 2. Ajeesh. A. V.

1. PG student, VLSI & Embedded Systems, ECE Department,TKM Institute of Technology,Karuvelil P.O, Kollam, Kerala-691505,
2. Assistant professor, ECE Department,TKM Institute of Technology,Karuvelil P.O, Kollam, Kerala-691505, India

Abstract— **One of the major problems concerned with the field of communication is the secure and error free transmission of data from transmitter to receiver. Minimizing the power and bandwidth consumption along with providing high quality of service are the competing goals of any communication system. Source coding and channel coding are important components to achieve these goals. Huffman coding, a type of Variable Length Coding (VLC), is widely used for text, image, voice and audio compression schemes. VLC provide reduction in the data rate by exploiting the redundancy in the source-symbol sequence but it is very sensitive to transmission errors. Hence an efficient coding or decoding techniques have to be considered to guarantee a good source-symbol sequence reconstruction. The decoding can be done using a priori knowledge of the length of the source-symbol sequence. This decoder can be concatenated with a channel decoder to perform iterative Joint Source Channel (JSC) decoding. The complete design can be coded using VHDL and simulated using ModelSim 6.2C.**

Keywords— *Variable Length Codes, Joint Source Channel, iterative JSC.*

I. INTRODUCTION

Communication is the field of study concerned with the transmission of information through various means. Communication can be broadly classified as analog communication and digital communication. In analog communication, the message signal is continuous time in nature and in digital communication, the message signal is digital in nature i.e. it has two levels only (0 and 1). The advantages of digital communication over analog communication are noise and distortion immunity, easy for processing, low cost and more privacy and security due to encryption.

The block diagram of basic communication system is Figure 1. The basic communication system consists of information source, source encoder, channel encoder, modulator, channel, demodulator, channel decoder and source decoder. The information source provides a message or sequence of messages to be transmitted. The source encoder, the channel encoder and the modulator forms the transmitter section. Transmitter operates on the message signal to produce a signal suitable for transmission. The channel is merely a medium for transmitting the signal from transmitter to receiver. The receiver performs the inverse operation to reconstruct the original message signal. The source encoder maps the digital signal generated at the source output into another signal in digital form and source decoder performs the inverse mapping to produce original source output.
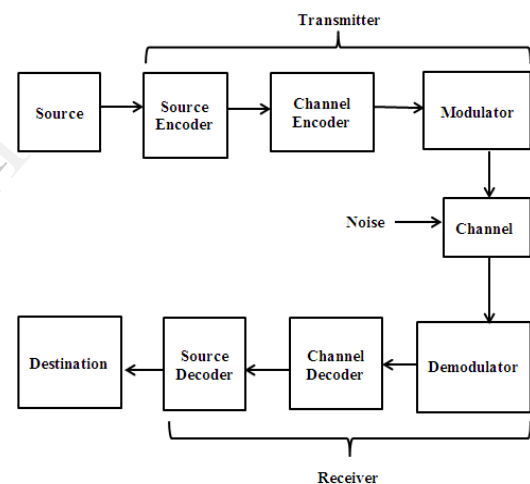

Figure 1. Basic Communication System

Data compression is the art or science of representing information in a compact form. In the last decade we have been witnessing transformation in the way we communicate and the process is still underway. This transformation indicates the ever-present, ever-growing internet, the explosive development of mobile communication, and the ever increasing importance of video communication. Data compression is one of the enabling technologies for each of these aspects of the multimedia resolution [11]. Data compression reduces the number of bits required to represent a text or an image or video. The reason we need compression is that more and more information that we generate and use is in digital form.

This paper is organized as follows. Section 2 gives the details on System architecture. In Section 3, source coding and Huffman coding technique is explained n detail. Section 4 and 5 provides details on the channel coding

technique and modulation scheme used respectively. Conclusions are discussed in the section 6.

II. SYSTEM ARCHITECTURE

In a communication system, source coding, channel coding and modulation are important processes. In source coding, the source encoder maps the information generated by the source into another signal in digital form and the source decoder performs inverse mapping to reconstruct the original information.
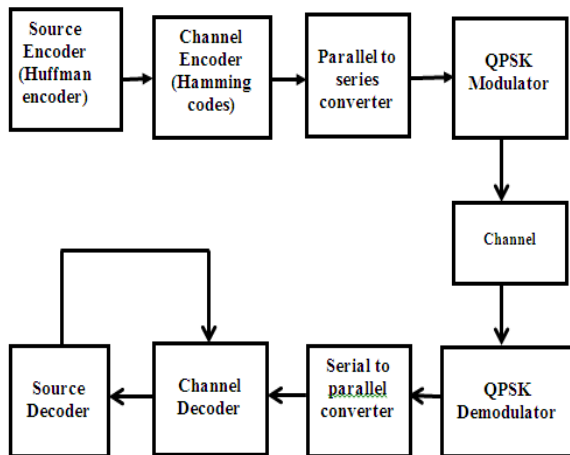

Figure 2. System Architecture

In channel coding, the channel encoder maps the incoming digital signals into channel input and the channel decoder maps the channel output to a digital signal in such a way that the effect of noise is minimized. Here, Huffman coding is used for source coding and channel coding is done using Hamming codes. Modulation is the process of changing the characteristics of carrier according to the data. Here QPSK is used for modulation. An input string is converted to its corresponding Huffman encoded data. This bit stream is then channel encoded using Hamming codes and it is then converted to two bit streams consisting of even and odd numbered bits respectively. These two bit streams are then serialised using parallel to serial convertor and QPSK modulated. The modulated bit stream is then passed through the channel where noise will be added. In the receiver section, the received data is demodulated, channel decoded and source decoded to obtain the original data that is transmitted.

III. SOURCE CODING

An information need to be translated into a stream of bits ('0' and '1') before transmitting to the channel. This process is called Source coding. There are many commonly used ways to translate the information. If there is a way of encoding information such that the alphabets with higher probability of occurrence are assigned with shorter code words, and longer for the lower probability alphabets, then on the whole it may be able to conserve the number of bits

to be sent to the channel while sending the same information. This is what the variable length code can do.

A. Huffman Coding

In computer science and information theory, Huffman coding is an entropy encoding algorithm used for lossless data compression. The term refers to the use of a variable length code table for encoding a source symbol (such as a character in a file) where the variable length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol.

Huffman coding is based on frequency of occurrence of a data item. The principle is to use a lower number of bits to encode the data that occurs more frequently. The average length of a Huffman code depends on the statistical frequency with which the source produces each symbol from its alphabet. A Huffman code dictionary, which associates each data symbol with a code word, has the property that no code word in the dictionary is a prefix of any other code word in the dictionary. The basis for this coding is a code tree according to Huffman, which assigns short code words to symbols frequently used and long code words to symbols rarely used.

Basic Technique

The steps involved in Huffman algorithm is as follows:

1) Source symbols are listed in the order of decreasing probability.
2) The two symbols of lowest probability are assigned 0 and 1.
3) These two source symbols are combined into a new source symbols with probability equal to sum of the two original probabilities.

4) The procedure is repeated until we are left with a final list of source symbols of only two for which a 0 and a 1 are assigned.
5) The code for each source symbol is found by working backward and tracking the sequence of 0s and 1s assigned to that symbol.

Assume you have a source generating 4 different symbols {$a1$, $a2$, $a3$, $a4$} with probability {0.4; 0.35; 0.2; 0.05}. Generate a binary tree from left to right taking the two less probable symbols, putting them together to form another equivalent symbol having a probability that equals the sum of the two symbols. As shown in figure 1.3 repeat the process until one symbol is left. Then read the tree backwards, from right to left, assigning different bits to different branches [13].
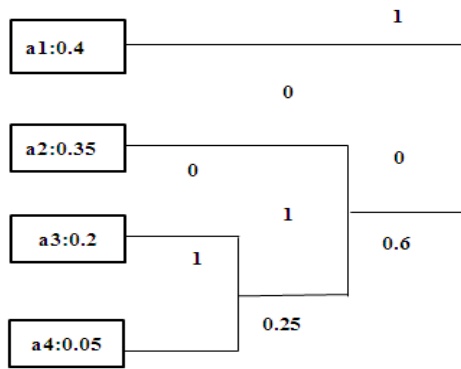
Figure 3. Huffman encoding algorithm

The symbol, its probability and its code is shown in the table 1. It can be seen that the symbol with highest probability is having the shortest code word and the symbol with lowest probability is assigned the longest code word.

Table 1. Code word corresponding to each character

SYMBOL	PROBABILITY	CODE
a1	0.4	1
a2	0.35	00
a3	0.2	010
a4	0.05	011

In the receiver section, the Huffman decoder will have the knowledge about the code table. When the decoder receives a particular string, it will convert a particular code to its corresponding character.

IV.   CHANNEL CODING

In communication systems, the information is represented as a sequence of binary bits. The binary bits are then mapped (modulated) to analog signal waveforms and transmitted over a communication channel. The communication channel introduces noise and interference to corrupt the transmitted signal. At the receiver, the channel corrupted transmitted signal is mapped back to binary bits. The received binary information is an estimate of the transmitted binary information. Bit errors may result due to the transmission and the number of bit errors depends on the amount of noise and interference in the communication channel.

A.   Calculating Hamming Codes

Hamming (n,m) is a linear error-correcting code that encodes m bits of data into n bits by adding (n-m) parity bits. When data is transmitted from one location to another there is always the possibility that an error may occur. There are a number of reliable codes that can be used

to encode data so that the error can be detected and corrected. A Hamming Code can be used to detect and correct one-bit change in an encoded code word.

Create the code word as follows:

1.   Mark all bit positions that are powers of two as parity bits. (Positions 1, 2, 4, 8, 16, 32, 64, etc.).
2.   All other bit positions are for the data to be encoded. (Positions 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 17, etc.).
3.   Each parity bit calculates the parity for some of the bits in the code word. The position of the parity bit determines the sequence of bits that it alternately checks and skips.

    Position 1: check 1 bit, skip 1 bit, check 1 bit, skip 1 bit, etc. (1,3,5,7,9,11,13,15,...).
    Position 2: check 2 bits, skip 2 bits, check 2 bits, skip 2 bits, etc. (2,3, 6, 7, 10, 11, 14,15,...)
    Position 4: check 4 bits, skip 4 bits, check 4 bits, skip 4 bits, etc. (4,5,6,7,12,13,14,15,20,21,22,23,...)
    Position 8: check 8 bits, skip 8 bits, check 8 bits, skip 8 bits, etc. (8-15, 24-31, 40-47,...)
    Position 16: check 16 bits, skip 16 bits, check 16 bits, skip 16 bits, etc.(16-31,48-63,80-95,...)
    Position 32: check 32 bits, skip 32 bits, check 32 bits, skip 32 bits, etc. (32-63,96-127,160-191,...)
4.   Set a parity bit to 1 if the total number of ones in the positions it checks is odd. Set a parity bit to 0 if the total number of ones in the positions it checks is even.

The channel encoding using Hamming codes can detect two errors and correct only one. The channel decoding is done at the receiver section which produces the original source encoded data.

V.   QPSK MODULATION

Quadrature Phase Shift Keying (QPSK) is a form of Phase Shift Keying. In applications that require high transfer rate and low power consumption, QPSK modulation has advantages over other schemes, and double symbol rate with respect to the BPSK over the same spectrum band. Contrast to analogue modulators for generating QPSK signals, where the circuit complexity and power dissipation are unsuitable for certain applications, digital modulator provides digital synthesis and the flexibility to reconfigure and upgrade with two most often used languages (VHDL and Verilog).
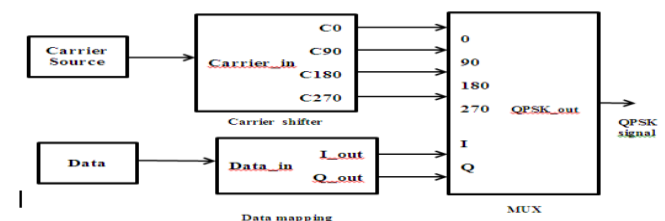

Figure 4. QPSK Modulation

In QPSK, two bits are modulated at once; selecting one of four possible carrier phase shifts (0, 90, 180, or 270 degrees). QPSK allows the signal to carry twice as much as information that ordinary PSK carry using the same bandwidth. The QPSK modulator is programmed to generate a carrier phase which acquires four discrete states (0, 90,180,270) [20]. Two separate streams in-phase 'I', and quadrature phase 'Q' for mapping the data for controlling the four phase different carriers interfaced to multiplexer. The output is selected by multiplexer to provide a digital QPSK signal. The digital QPSK signal of the multiplexer output can be represented as

$$Mux_{out} = \overline{I}\,\overline{Q}C_0 + \overline{I}QC_{90} + I\overline{Q}C_{180} + IQC_{270}$$

The QPSK modulator consists of carrier source to produce a periodic pulse signal(carrier signal) which is fed to a carrier phase shifter; which shift the input carrier into four different phase signals (0°, 90°, 180°, 270°) interfaced to multiplexer as shown in the figure 3.5. While the data is fed to data mapping to generate I and Q signals to influence the four phase different carries. The output is selected by multiplexer which provides digital QPSK signal.

A. QPSK Demodulation

In QPSK demodulation, the demodulated output is obtained based on the modulated input and the four carrier signals.
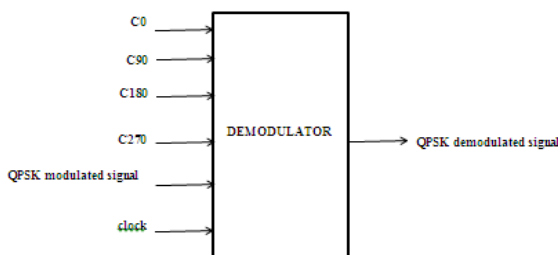

Figure 5. QPSK demodulation

The four carrier shifts i.e. C0, C90, C180, C270 forms the input signal. For every clock edge the value of the modulated signal is checked with any one of the four carrier phase shifts. If it is equal to C0 then the output will be '00'. If it is equal to c90, then the output will be '01'. If the modulated signal is equal to c180 then the output will be equal to '10' and if it is equal to c270 then the output will be '11'.

VI. ITERATIVE JSC DECODING

Iterative Joint Source Channel Decoding (JSC) is the decoding technique where the information transfer takes place between the source decoder and channel decoder. The channel decoder will have the knowledge about the number of bits in the source encoded bit stream. If the number of bits n the channel decoded data is different from the number of bits in the source encoded data, then the channel and source decoding is iterated to obtain the original data that was transmitted.

VII. CONCLUSIONS

Source coding and Channel coding are the important components of communication system in order to achieve high quality of service. Source coding using Huffman encoding algorithm reduces the number of bits in the data when compared with the ASCII representation of the string. The channel encoding using Hamming code is done for error detection and correction. The data is then modulated using QPSK modulation for proper transmission over the channel. In the receiver section, the reverse operation is done to obtain the original transmitted data. The technique can be designed using VHDL and simulated using ModelSim 6.2C design suite from Mentor Graphics.

REFERENCES

[1] Amin Zribi, Ramesh Pyndiah, Sonia Zaibi, Frédéric Guilloud and Ammar Bouallègue,"Low Complexity Soft Decoding of Huffman Codes and Iterative Joint Source Channel Decoding*", IEEE Trans. on commun.*, vol. 60, no. 6, june 2012.

[2] D. M. Kate, "Hardware implementation of the huffman encoder for data compression using Altera DE2 board", *IJAES,* vol. 2, august, 2012.

[3] Tribeni Prasad Banerjee1, Amit Konar2, Swagatam Das2, and Ajith Abraham, "An Intelligent Lossless Data Compressor Implementation using Reconfigurable Hardware", Journal of Information Hiding and Multimedia Signal Processing, vol. 2, Number 1, January 2011.

[4] Kinjal A. Bhavsar, Usha S.Mehta, "Analysis of Test Data Compression Techniques Based on Complementary Huffman Coding", *IJEST,* vol. 3 No. 5 May 2011.

[5] Asadollah Shahbahrami, Ramin Bahrampour, MobinS abbaghi Rostami, Mostafa Ayoubi Mobarhan, "Evaluation of Huffman and Arithmetic Algorithms for Multimedia Compession Standards", *IJCSEA,* vol.1, No.4, August 2011.

[6] Mamta Sharma, "Compression Using Huffman Coding", *IJCSNS,* vol.10 No.5, May 2010.

[7] Gihad Elamary, Graeme Chester, Jeffrey Neasham, "A Simple Digital VHDL QPSK Modulator Designed Using CPLD/FPGAs for Biomedical Devices Applications*", Proc. World Congress on Engineering* , vol 1,WCE 2009, July 13, 2009.

[8] Kanchan H. Wagh, Pravin K. Dakhole, Vinod G. Adhau, " Design and implementation of JPEG 2000 encoder using VHDL", *Proceedings of the World Congress on Engineering,* vol 1 july 2008.

[9] D.A. Huffman, "A method for the construction of minimum-redundancy codes", Proc. of the I.R.E., 1952, pp. 1098-1102.

[10] C. E. Shannon, "A Mathematical Theory of Communication", *The Bell System Technical Journal,* vol. 27, pp. 379–423, 623–656, July, October, 1948.

[11] Khalid Sayood, "Introduction to Data Compression".