International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 4 847 - 851

Implementation of FPGA Based MP3 player using Invers Modified Discrete Cosine Transform

Mr. Vinay Vyas
Universal college of engineering,Kaman
Email-Id: vinay.93dj@gmail.com

Mr. Sanket Shinde
Universal college of engineering, Kaman
Email-Id:sanketsanket01@gmail.com

Mr. Rajesh Giri
Universal College of Engineering, Kaman,
Email-Id: rajeshgiri1921@gmail.com

Mr. Mukund Jadhav
Universal college of engineering, Kaman
Email-Id:mukundjadhav@gmail.com

Mr Omkar Mehata
Universal College of Engineering, Kaman,
Email-Id: omkar.mehata@gmail.com

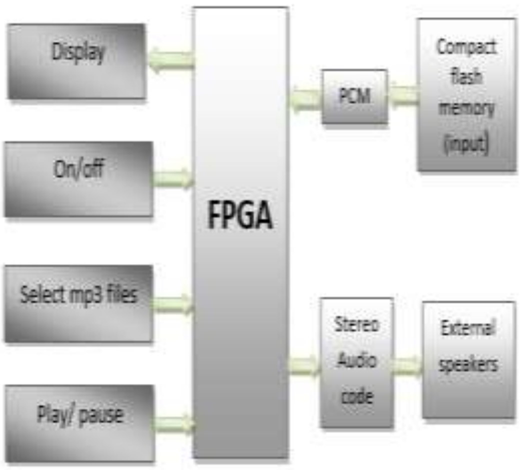
Abstract— Now the world is rapidly evolving with respect to integration and in the field of digital designing Field Programmable Gate Array (FPGA) devices are becoming more popular and proved to be efficient one. The major advantage is that we can design and implement complete embedded system using single FPGA chip in various application area of digital signal processing. Due to increase in packaging density up to such a great extent we can design aMP3player which utilize the platform of "Altera" FPGA board able to read ,decode songs from compact flash memory and able to play it on the external audio systems. This decoding of fetched songs is done by using16-bit pulse code modulated (PCM) outputs utilizes standard MP3 decoding algorithms and VHDL is used to drive external stereo Audio codec which converts the digital PCM outputs into an analog sound wave. The proposed system is designed on the Embedded Development Kit platform and data compression techniques are used for MP3 encoding and decoding which results in an efficient dedicated mp3 decoder in hardware.

*****

I. INTRODUCTION

In recent trend in digital design using a Field Programmable Gate Array (FPGA) is becoming more popular due to higher packaging density and reduced on chip area. Hence a complete embedded system not only built and programed but also successfully implemented using a single FPGA chip providing wide scope of emergence in digital signal processing and its applications. In the proposed system that is FPGA-based MPEG Layer III (MP3) hardware and software co-design has achieved using FPGA also the decoding algorithms are implanted using hardware description language and C language on embedded development kit platform.

II. PROPOSED SYSTEM BLOCK DIAGRAM



III. SYSTEM DESCRIPTION

The inputs to the FPGA MP3 player framework will be a MP3 bit stream that is preloaded onto a compact flash memory (CFM) and any client interface control info. Utilizing pushbuttons, the client will be empowered sweep through the MP3 document rundown, and after that choose, play, delay, and/or stop the tune. What's more, volume control is activated by an adjustment in the on-board rotating encoder dial position.

The yields of the MP3 decoder, that is, 16-bit pulse code modulated (PCM) yields and play the sound records through an outside speaker. The PCM yields should be changed over to simple configuration by means of the on-board stereo sound codec equipment chip before the sound can be heard with an outer speaker that can be appended through the sound jack with a 16 mW intensifier. The framework piece graph is as appeared underneath in fig 1.1.

- **MP3 Decoder**

It decodes the selected MP3 stream runs on the FPGA that will select the sampling frequency specified in the MP3 header whose typical value is 44.1 kHz the further software and hardware designs are integrated on the Xilinx Embedded Development Kit platform.

- **External Peripherals:**

Verilog HDL is used to drive all external peripherals. Most applications utilize devices by means of high-level device-generic commands. Driver software accept these generic high-level commands and break them into a series of low-level device-specific commands.

- **User Interface**

The user interface provides the inputs to control the MP3 player, such as selecting, playing, pausing, and stopping the MP3 files. Whose outputs can be sense by LEDs

Compact Flash Memory Card

The required MP3 files are preloaded in compact flash memory (CFM) having storing capacity of 2GB for the MP3 decoder system in the FPGA. MP3 files are loaded onto to the CFM using a PC and memory card reader.

- **Onboard Stereo Audio Codec**

The Audio codec is used to convert the PCM format signal from the MP3 decoder into an audio signal, which is fed into an external speaker through an audio jack. System functional requirements and performance specifications:

- **Input MP3 bit stream requirements:**

The MP3 player will decode MP3 inputs with various bit rates (from 128 kbps to 320 kbps) and different sampling frequencies (32 kHz, 44.1 kHz or 48 kHz)

- **Decoding speed:**

The ultimate objective of decoding speed is to process MP3 files in real-time. The execution time of the MP3 decoding will be profiled and measured. If the real-time specification can't be met, further optimization will be needed.

- **Background information on the MP3 format:**

The need to reduce the size of audio files without any noticeable quality loss was stated in the 1980ies by the International Organization for Standardization (ISO). A working group within the ISO known as the Moving Pictures Experts Group (MPEG), developed a standard that contained several techniques for both audio and video compression. The audio part of the standard included three modes with increasing complexity and performance, as shown in Figure

complexity

Coding	Ratio	Required bitrate
PCM CD quality	1:1	1.44 Mbps
Layer I	4:1	384 Kbps
Layer II	8:1	192 kbps
Layer III (mp3)	12:1	128 Kbps

Table 1.2: Layer wise Complexity

The third mode, called Layer III, manages to compress music by a factor of 12 with almost no audible degradation. This technique is known as MP3 and has become very popular and widely used in applications today.

IV. OVERVEEW OF MP3 ENCODING PROCESS:

MP3 encoding includes speaking to a tune as a bit stream (a variety of 0's and 1's) that can be recouped by a MP3 decoder (player). The high rate pressure required in MP3 encoding permits tunes to be put away and shared rather effectively and rapidly on PCs and through the web without losing any distinguishable quality. This lossy pressure works by first veiling indiscernible recurrence segments to the human ear, and after that utilizing a few information pressure systems that expel information redundancies.

To start with the simple sound is tested at a particular examining rate, commonly at 44.1 kHz. This is because of Nyquist recurrence's criteria in which the examining rate must be no less than two times more prominent than the biggest conceivable recurrence segment present in the information. Furthermore, since the scope of perceptible frequencies to the human ear is about 20 Hz to 22 kHz, this testing rate is generally picked. The signs are quantized utilizing beat code balance, where every specimen plentifulness is spoken to by 16 bits.

To evacuate redundancies and pack information, recurrence examination procedures are utilized. The PCM tests are separated for 32 meet recurrence ranges, called sub groups utilizing a polyphase engineering that yields in a higher computational proficiency. A discrete cosine change is then connected to expel low vitality signals from high recurrence segments.

Further pressure is accomplished by utilizing a lossless pressure strategy known as Huffman encoding that depends on factual conduct of information. At long last, the bit stream is masterminded into edges that the MP3 decoder will investigate to remake the MP3 sound.

MP3 decoding:

MP3 translating is the converse procedure of MP3 encoding. Luckily, unraveling is not about as perplexing, since it doesn't require a psychoacoustic model (a virtual model of the human ear and how it sees distinctive frequencies). The MP3 decoder's part is to recoup the first sound by investigating certain segments of a casing to pick up data about encoding parameters utilized and afterward utilize reverse methodology to reproduce PCM tests. Every casing comprises of precisely 1152 PCM tests and contains no less than two areas:

- A header section that contains important encoding parameters, such as bit rate and sampling frequency used and an audio data section that holds the encoded bit stream. Some MP3 bit streams contain an optional ID3 tag frame that can be used to store MP3 related information including the title and author of the song.

TAG (optional)	Frame 1	Frame 2	Frame 3	-----	TAG (optional)

The first four bites of each frame is Header and the rest are data. The size of each frame varies according to bitrate. The various fields in a frame of audio data are discussed below.

Header is 4 bytes long and contains sync word to indicate the start of frame. Header contains Layer information (MPEG Layer I, II or III), bitrate information, and sampling frequency and mode information to indicate if the stream is mono or stereo. Error Check this fields contains a 16 bit parity check word for optional error detection with in the encoded stream.

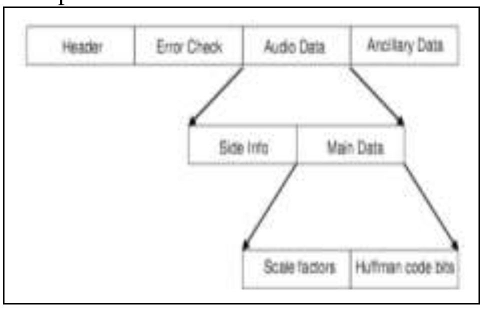


Figure 1.3: MPEG I layer 3 frame format

Side information Contains information to decode Main data. Some of the fields inside information are listed below--

1. It contains scale factor selection information that indicate the number of scale factors transferred per each sub band and each channel. Scale factors indicate the amount by which an audio sample needs to be scaled. Since, human ear response is different for signals at different frequencies, the entire audio spectrum is divided into sub bands. The samples in the more sensitive bands are scaled more than the samples in the lesser sensitive region of the spectrum.
2. It contains global gain which needs to be applied to all the samples in the frame.
3. Information regarding the number of bits used to encode the scale factor. To achieve compression, even the scale factor are encoded to save the bits. This information in the side info will indicate the number of bits to encode a particular scale factor.
4. Information regarding the Huffman table to be selected to decode a set of samples. This information specifies one of the 32 Huffman tables used for Huffman decoding. Main data the main data contains the coded scale factors and the Huffman coded bits.
5. Scale factor are used in the decoder to get division factors for a group of values. These groups are called scale factor bands and the group stretches over several frequency lines. The groups are selected based on the non-uniform response of human ear for various frequencies.
6. The quantized values are encoded using Huffman codes. The Huffman encoding is used to code the most likely values with lesser number of bits and rarely occurring values with larger number of bits. The Huffman codes are decoded to get the quantized values using the table select information in the side info section of the frame.
7. Ancillary data this field is the private data and the encoder can send extra information like ID3 tag containing artist information and name of the song.

The frame size in bytes varies from song to song, and in some cases, even within one song (when using variable bit rates). The general equation for calculating the frame size in bytes is found in Equation 1.4.

Equation 1.4

Frame size (in bytes) = (144* bit rate)/ (sampling rate + padding)

Where 144= (1152 PCM/frame) / (8 bits/byte) and where padding is an integer number to ensure that the frame size is an integer number Bit rate is the rate at which the compressed bit stream is delivered from the storage medium to the input of a decoder while sampling frequency defines the number of samples per second taken from a continuous signal to make a discrete signal. For MP3 encoding, there are several allowed bit rates and sampling frequencies that can be used.

Typically, a sampling rate of 44.1 kHz is used and is known as “CD quality” while 48 kHz is referred to as “DVD quality.”

For MP3 encoding, there are several allowed bit rates and sampling frequencies that can be used, as illustrated in Table 1.5 and 1.6 respectively. These tables are copied directly from the ISO standard document.

Bitrate index	Bitrate specified(Kbps)		
	Layer I	Layer II	Layer III
0000	Free	Free	Free
0001	32	32	32
0010	64	48	40
0011	96	56	48
0100	128	64	60
0101	160	80	64
0110	192	96	80
0111	224	112	96
1000	256	128	112
1001	288	160	128
1010	320	192	160
1011	352	224	192
1100	384	256	224
1101	416	320	256
1110	448	384	320
1111	Forbidden	Forbidden	Forbidden

Table 1.5

Sampling frequency	Frequency specified
00	44.1
01	48
10	32
11	Reserved

Table 1.6

The MP3 decoding process is shown in Figure 1.7. It includes the following stages:

1) Initial reading
2) Huffman decoding
3) Re-quantization and reordering,
4) Stereo decoding, alias reduction,
5) Inverse modified discrete cosine transform (IMDCT) and
6) Synthesis polyphase filter bank

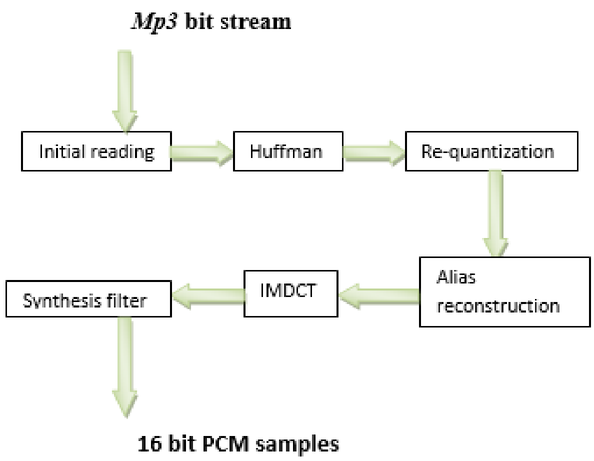


Fig 1.7: Diagram of decoding process

- **Initial reading:**

The incoming data stream is split up into individual frames. The header section of each frame is analyzed to obtain parameters used in the encoding process (i.e. bit rate and sampling frequency). The first action is the synchronization of the decoder to the incoming bit stream by checking if the first 12 bits of the header section are 1's. Scale factor and Huffman table selection bits are also decoded.

- **Huffman decoding:**

The Huffman algorithm is used for lossless data compression. The basic idea of the technique is to assign shorter binary codes to more frequent samples and longer codes to less frequent samples. The Huffman decoding procedure is based on tables that are used to map the Huffman binary codes to the original samples.

- **Re-quantization:**

During the encoding process, the outputs of the MDCT, or frequency domain samples, were pre-quantized in an attempt to use more precision when needed. It turns out that finer frequency resolution is needed for low volume sounds and larger values are coded with less accuracy. Afterwards, the values were scaled, or multiplied by a scale factor, a value that is based on the absolute threshold of the human ear (a frequency dependent function). Larger scale factor are needed if the frequency components are more difficult to hear. So for the decoding process, the values need to be re-quantized. Afterwards, de-scaling is required.

- **Re-ordering:**

In the MP3 encoding process, the use of short windows would generate frequency lines ordered first by subband, then by window and at last by frequency. In order to increase the efficiency of the Huffman coding the frequency lines for the short windows case were reordered into subbands first, then frequency and at last by window, since the samples close in frequency are more likely to have similar values. The reordering block in the MP3 decoding process will re-sort the samples by sub bands, then on windows and then on increasing frequency. For a description on windowing, refer to the IMDCT block.

- **Alias reduction block:**

Aliasing is the overlap of frequency components when energies greater than Nyquist frequency are present. This is the result of the decimation or the reduction of sampling rate in the analysis filter bank process where overlapping of adjacent sub band filters is inevitable. In the encoding process, these aliasing effects are removed to reduce the amount of information that needs to be transmitted. This can be achieved by using a series of butterfly computations that add weighted, mirrored versions of adjacent sub band to each other. In the decoding process, aliasing artifacts must be added to the signal again in order to obtain a correct reconstruction of the audio signal. The alias reconstruction calculation consists of eight butterfly calculations for each sub band.

- **Inverse Modified Discrete Cosine Transform (IMDCT):**

The Inverse Modified Discrete Cosine Transform (IMDCT) is the inverse of the modified discrete cosine transform used in MP3 encoding. The MDCT was used to represents signals as a sum of cosine waves, essentially transforming them to the frequency domain. Compared to the DFT and other well-known transforms, the MDCT has a few properties that make it very suitable for audio compression. First of all, the MDCT has the energy compaction property common to discrete cosine transforms. This means most of the information in the signal is concentrated to a few output samples with high energy. The term modified is used since there is a 50% overlap. The lower 18 values are added with the higher 18 values from the previous frame

And used as output. The higher 18 values are then stored and used the same way when the next frame is being decoded. This overlapping that avoids sharp discontinuities.

Synthesis polyphase filter bank.

The synthesis polyphase filter bank is the final step in the decoding process. It is used to combine the signal energies from all the 32 sub bands. The result output for each frame is 1152 16 bit PCM samples. A polyphase architecture is used since the decimation of the sampling rate allows the use of a lower number of filter coefficients, and thus improves computational efficiency. The method recommended by ISO standard for transforming sub band samples to the Pulse code modulated format involves shifting, matrixing with a 32 point discrete cosine transform that represent band pass filter coefficients, a 512 point window to improve filter quality, and finally a summation for all the sub band. Various algorithms can be implemented that observe symmetry properties and reduce the number of computations. For example, the DCT function can be calculated using a method known as the fast DCT in a similar manner that a DFT function can be more efficiently computed using the FFT. The method recommended by the ISO standard document [5] for transforming the sub band samples into the PCM format is illustrated in Figure

V.FLOWCHART:

The MP3 decoding algorithm described thus far is implemented completed in software using C language. High level flowcharts for the main program, as well as the select song and play song functions are illustrated below.

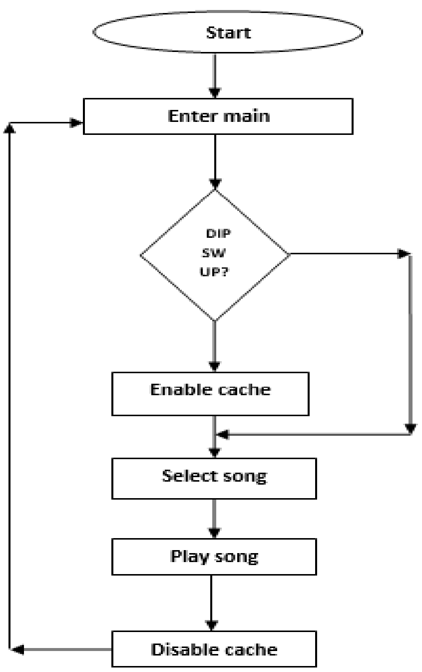


Fig 1.8: Flow Chart for FPGA mp3 player

The Hardware platform

FPGA - Field Programmable Gate Array:

FPGA is a silicon chip with detached rationale doors. It is a coordinated circuit that contains numerous (64 to more than 10,000) indistinguishable rationale cells that can be seen as standard segments. The individual cells are interconnected by a framework of wires and programmable switches. Field Programmable implies that the FPGA's capacity is characterized by a client's project as opposed to by the maker of the gadget. Contingent upon the specific gadget, the system is either "smoldered" in forever or semi-for all time as a major aspect of a load up get together process, or is stacked from an outside memory every time the gadget is controlled up.

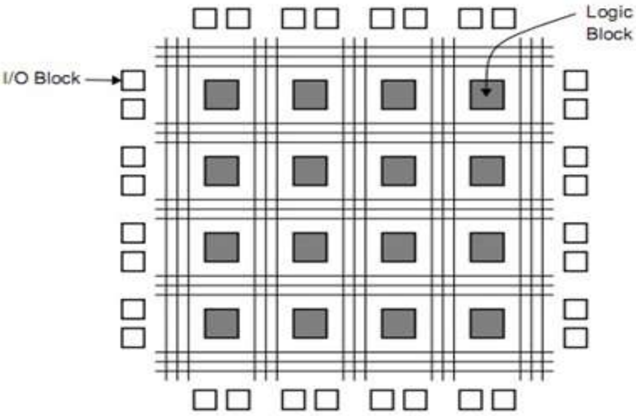


Fig 1.9. The FPGA block

The Field-Programmable Gate Arrays (FPGAs) give the advantages of custom CMOS VLSI, while keeping away from the underlying cost, time delay, and innate danger of a Conventional veiled entryway cluster. The FPGAs are redone by stacking design information into the inner memory cells

Stereo Audio Codec:

The Audio codec is utilized to change over the PCM group signal from the MP3 decoder into a sound sign, which is nourished into an outer speaker through a sound jack.

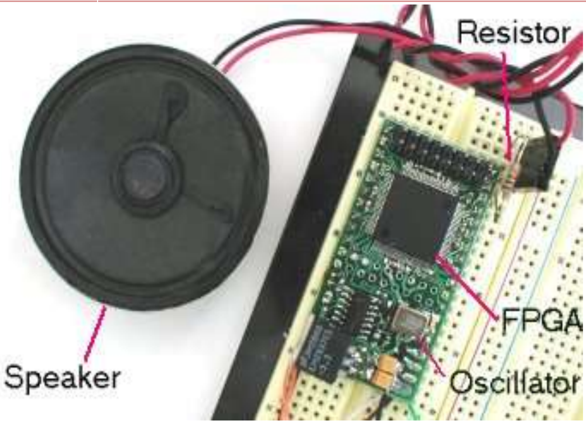


Fig1.10. The functional diagram

Conclusion:

In this paper elaborates complete implementation of music player a using Altera development board which read MP3 files from a compact flash memory device, then decode and play it through the stereo audio codec. Different controls such as song selection pause and stop modes are included. Similarly FPGA can be used in various field of applications.