A Multimedia Data Processing System for Real-Time Audio-Visual Data Coding and Decoding

*O. K. Shin**

<目 次>

Abstract

In this paper, we introduce an experimental hardware architecture of a multimedia data processing system, whose application area includes multimedia data authoring, CSCW and video conferencing. The hardware platform comprises most commonly needed multimedia data processing functions such as audio – visual data capture, playback, multistandard compression as well as interleaving of compressed audio visual data. The proposed architecture minimizes the CPU overhead that might be caused by bulky multimedia data processing and assures the fluent data flow among system components. We begin with overal architecture of the whole system, and then implemmentation issues concerning audio – visual data capture/display unit and multistandard commpression unit are discussed.

Key words : Multimedia Processing, Compression, CSCW, Video Conferencing

Ⅰ. Introduction

Recent developments in both software and hardware have brought real time – multimedia data applications possible. In software, models and architectures to deal with multimedia data process-

* 한국해양대학교 이공대학 컴퓨터공학과 전임강사

ing have been studied and developed in operating system[1], [2], [3], in distributed environment [4], [5], [6], [11] in synchronization between media[7], [8], and so on. In hardware, the video compression and decompression silicons, which were so called missing building blocks in multimedia processing system, became recently available. However, many hardware vehicles for multimedia processing are partial in the sense that they are capable of supporting only a limited range of functions among those required by various multimedia applications. For example, some multimedia systems support MPEG decoding for CD - ROM title playback, but not capable of performing video conferencing, or vice versa. Another weak point found in many multimedia systems is that it requires non - negligible portion of the computing power of the main processor of the motherboard(CPU) for interleaving and disinterleaving, and for temporal synchronization between media.

To resolve these drawbacks, and to support the multimedia processing - related functions required by applications, we designed a hardware architecture with the following objectives :

- include audio - visual data capture, playback, encoding and decoding functions in a single unit,
- support multistandard compression while minimizing the number of components,
- include, in the multimedia processing unit, a mechanism for temporal synchronization between media,
- remove bottlenecks in data path between functional units in the whole system.

In the following sections, we begin with the overall architecture of the system, and then the audio - visual data capture/playback unit and compression/decompression unit are discussed.

Ⅱ. Overall Architecture

The hardware platform consists of two subsystems : motherboard subsystem and multimedia subsystem. The motherboard subsystem is based on Intel's Pentium processor and dual bus : PCI(Peripheral Component Interconnect) and EISA. The system, aiming an efficient multimedia processing engine, has important features in its architecture. First, by adopting a high performance and low - medium performance system - bus pair in motherboard subsystem, we can classify and arrange the bus - connected units of the system according to their data traffic. For example, graphics, SCSI and multimedia processing as well as LAN I/F logic are connected to PCI bus while FAX/modem, ISDN I/F logic and other low throughput devices are connected to EISA bus. This approach prevents low speed traffic devices from holding high throughput bus. Second, we have assembled all the audio and video related processing facilities in multimedia subsystem as a single PCI agent. By localizing the multimedia processing functions in a single PCI agent, the multimedia traffic across the PCI bus can be minimized. In this manner, the main processor of the motherboard

is nearly independent from the multimedia data encoding and decoding. Most likely, the mother-board interacts with the multimedia subsystem to initialize and to send and receive the compressed data to and from the multimedia subsystem.

The multimedia subsystem can be divided into two units : AV - Main unit and Codec unit. In AV - Main unit, audio - visual data are captured, pre - processed, played back and delivered to Codec unit. The Codec unit is a piggyback board mounted to the AV - Main unit. It performs all the functions necessary for system level encoding of MPEG and H.221.

We excluded graphics unit from the multimedia subsystem. The main reason is that the third party graphics evolve dramatically and we wanted to reserve the capability of updating our system by simply replacing graphics boards.

In this paper, we focus on audio - visual data capture - playback and codec functions. The overall view of multimedia subsystem is shown in Fig. 1.

Each unit in Fig. 1. is implemented as a separate board where Codec unit is a piggyback board mounted on AV - Main unit(we'll use the term 'board' interchangeably with 'unit'). The AV - Main unit can be used as a stand - alone board(without Codec board) to capture and playback audio and video data.

As shown in Fig. 1, PCI bus interface logic is located in AV - Main unit and interfaces the Pentium - based motherboard subsystem to AV - Main and Codec boards. Traffics on this interface are usually compressed audio - visual data between motherboard and Codec board, commands from

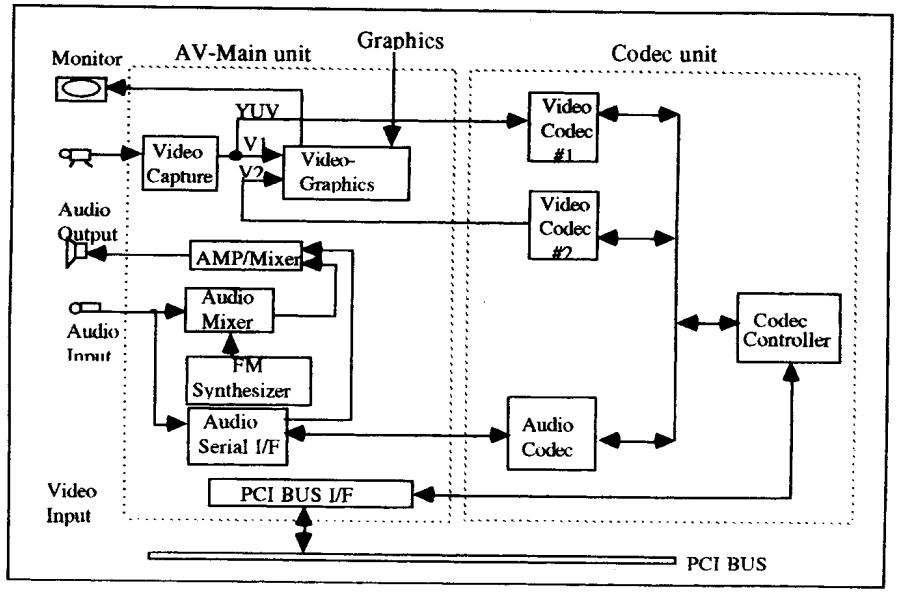

Fig. 1. Overall View of Multimedia Subsystem

motherboard to each board and status information from each board to motherboard, but do not exclude raw audio and/or video data between motherboard and AV - Main board. The traffic between AV - Main and Codec boards are raw audio and/or video data as well as control and status data. The architecture and functions of AV - Main unit and Codec unit are explained in more detail in the following chapters.

Ⅲ. Audio - visual Data Capture - Playback : AV - Main Unit

AV - Main unit can be divided into three subunits : PCI I/F subunit, audio subunit and video subunit. We describe each of these subunits.

1. PCI I/F subunit

Among various commercially available buses, PCI bus is selected for the following reasons :

- An open - architecture bus is wanted.
- From the analysis of mutimedia data processing requirements(e.g., CSCW, video conferencing, high performance graphics support, future extendability to a high speed network such as FDDI, etc.), a high bandwidth local bus is required. And this leads to two competitive local bus structures, PCI and VL bus. Comparing these bus structures, VL bus is limited to only three agents(slots) and the host is locked up when an agent is accessing the bus. On the other hand, in PCI bus system, host operation is independent from the bus operations and up to 10 agents can share the bus.

Meanwhile, the components selected for the multimedia subsystem were not PCI - bus based. As there was no commercially available PCI Bridge at the moment of development, we have implemented this bus interface in FPGA(Xilinx XC3164) with minimum logic so that it could fit into two FPGA chip set. A drawback was found in this approach : the lack of DMA function in audio data capture and playback from the motherboard. We redesigned this part by adding FIFOs between audio capture/playback module and PCI I/F logic(See section Audio subunit). The PCI bus interface logic used is shown in Fig. 2.

2. Audio Subunit

This subunit digitizes analog audio input, delivers digitized data to the motherboard or to the Codec board and plays back data received from the motherboard or the Codec board. This subunit is built around the CS4231 of Crystal Semiconductor which is a single chip solution for audio capture and playback, ADPCM codec and mixing various input audio signals[9]. Input devices connected to

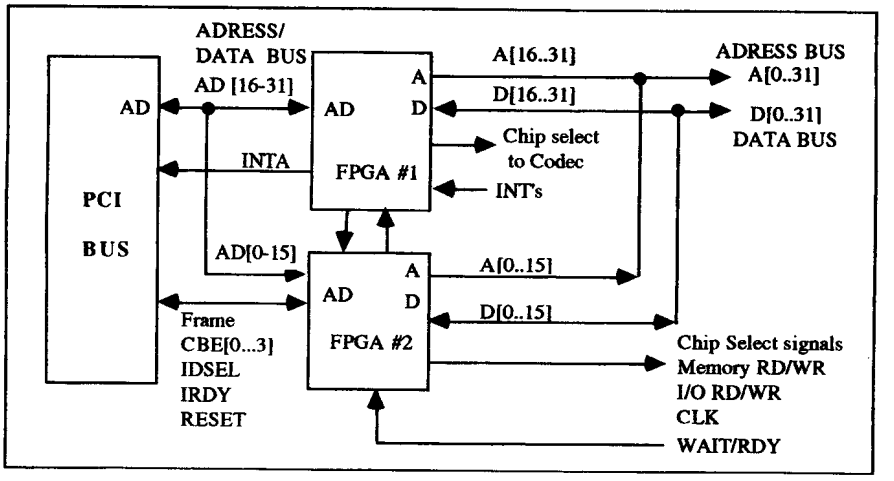

Fig. 2. PCI Bus Interface

this subunit are microphone, Line - In, FM synthesizer and CD - ROM, while analog output signals are fed to speaker and Line - Out. The block diagram of this subunit is shown in Fig. 3. There are two design topics to mention in this subunit : the adoption of FIFO between PCI interface logic and CS4231, and serial audio data path between AV - Main and Codec boards.

1) FIFO between CS4231 and PCI interface logic

In PCI bus, there is no DMA function supported in the strict sense. Instead, a PCI agent can, if the agent is capable of functioning as a bus master, requests the bus and controls it directly once the request is acknowledged. But, in the multimedia subsystem neither this bus mastership is implemented in bus interface logic, nor CS4231 is a PCI bus based component. We could think of

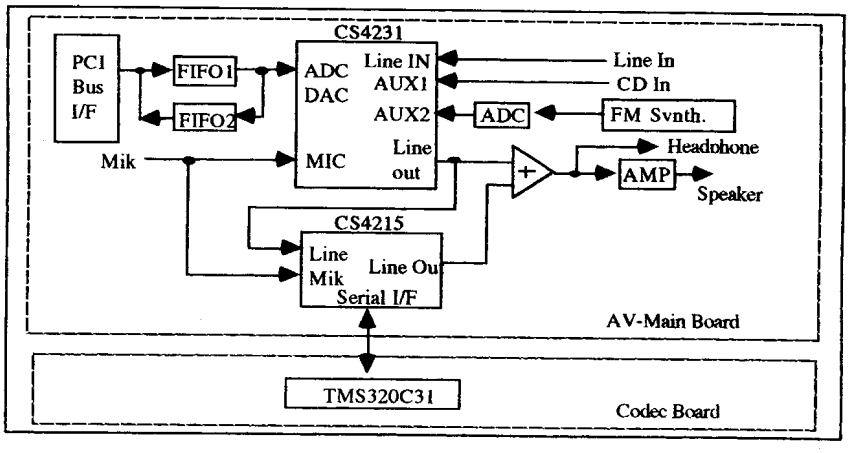

Fig. 3. Audio Subunit in AV-Main Board and Serial I/F to Codec Board

two choices for real time audio capture and playback without H/W intervention : polling and interrupt. Using the pure polling method(without timer interrupt) is out of question in our multi - tasking operating system NT, since the CPU of the motherboard will waste its whole computing power on polling. On the other hand, the interrupt mechanism resulted in ticks in high sampling frequency, the reason being the frequent interrupt requests and latency in NT's interrupt service.

Hence we inserted two FIFOs, one for capture and the other for playback, between the PCI interface logic and CS4231. With these FIFOs and peripheral circuits, CS4231 acts as if there is a DMA controller.

2) Audio data path between AV - Main and Codec board

We need to provide a data path between AV - Main board and Codec board to transfer the captured audio data and decompressed data. The CS4215 audio codec chip [10] was chosen as an interface logic between the two boards as shown in Fig.3. The CS4215 provides a simple serial data communication channel which can be directly connected to the audio DSP(TMS320C31) in Codec board. Besides the simplicity, this interconnection has the following advantages over other possible digital interface logics :

- The data bus of CS4231 is 8 bit, while that of audio DSP in Codec board is 32 bit. Hence, serial interface saves computing power of DSP required for word alignment(TMS320C31 DSP has a serial port operationally independent from other parts of DSP) as well as interconnection pins between the two boards.
- One can mix the output from CS4231 and that from the Codec board.
- One can deliver the mixture of all the input signals of CS4231 to the Codec board for compression.

To avoid the analog audio signal crossing the two boards, we included the serial audio interface in the AV - Main board.

3. Video Subunit

The video subunit captures input video signal from camera, delivers the captured signal to Codec board and receives the decompressed video data from Codec board. This subunit mixes two video input signals with graphics data, thus enables to display multiple video windows on a screen. These functions are provided by video input module, video processor module and video output module. The block diagram of this subunit is shown in Fig. 4.

1) Video input module

This module is capable of decoding NTSC, PAL and SECOM input video. The output of this mod-

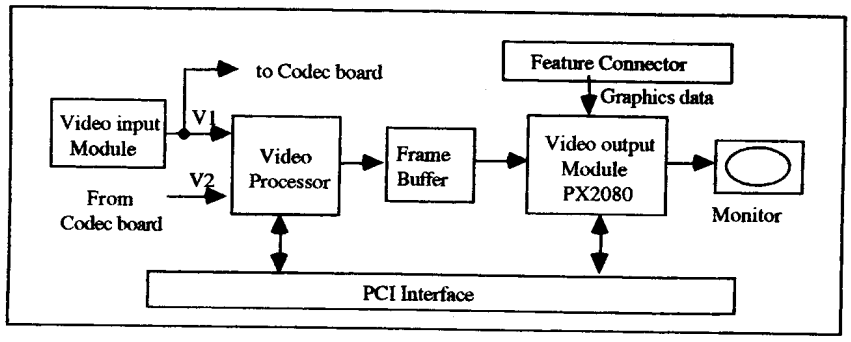

Fig. 4. Blockdiagram of Video subunit

ule is in 4 : 2 : 2 YUV format and is fed to video processor module and Codec unit.

2) Video processor module

The video processor is capable of video data manipulations such as scaling, zooming, windowing, color space conversion and video stream format conversion. The video processor has two input channels, where one is connected to the video input module and the other is connected to the decompressed data output channel of Codec board. The frame buffer keeps the output of the video processor and delivers the stored data into video output module. The processor of motherboard also can access the frame buffer to read image data for further processing or to write back image data for display.

3) Video output module

This module mixes video data input from frame buffer and graphics data from feature connector. The module provides hardware cursor and graphics overlay, and outputs analog RGB signal to drive monitor.

Ⅳ. Audio - visual Data Compression/Decompression : Codec Unit

Composed of general purpose DSPs and the video codec oriented DSPs, the Codec unit is intended to perform all the functions necessary for system level coding and decoding as follows :

- MPEG system level encoding/decoding which includes audio MPEG, video MPEG as well as interleaving and disinterleaving of audio - visual data.
- H.221 encoding/decoding as well as H.261(video) and G.728(audio). The purpose being the video conferencing, the encoding and decoding processes take place concurrently so that the real time video conferencing is possible.

In this tentative approach, there are 3 features to note.

- Flexibility : the Codec board is designed with flexible components so that various encoding/ decoding micro - codes can be downloaded.
- Audio - visual codec in a single board : this can help the lip - synchronization to be exact and easily performed
- Total Codec solution : as this unit performs the system level encoding and decoding, the motherboard can consider this unit as a functional block whose inputs are interleaved MPEG or H.221 data stream(decoding) or separate raw audio and video data(encoding). The outputs of this unit are decoded audio and video data(decoding) or interleaved MPEG or H.221 data stream(encoding). This feature relieves motherboard from system level encoding and decoding.

The Codec board is composed of control subunit, audio codec subunit and video codec subunit. This board has a hierarchical control structure as shown in Fig. 5.

In encoding, raw audio and video data are delivered from AV - Main board to audio codec and video codec subunits for compression. Then the compressed audio - visual data are sent to the control subunit for interleaving. The interleaved data are sent to the motherboard through PCI interface.

In decoding, the interleaved data sent from motherboard are disinterleaved in control subunit and the separate audio and video data are sent to audio codec and video codec subunits for decompression. The decompressed audio and video data are then fed to AV - Main board to be played back. Occasionally, the interleaving or disinterleaving of audio and video data may not be necessary. In this case, the compressed audio and video data can be transferred directly between the motherboard and audio codec or video codec subunits without passing through the control subunit. These paths are shown as dotted lines in Fig. 5.

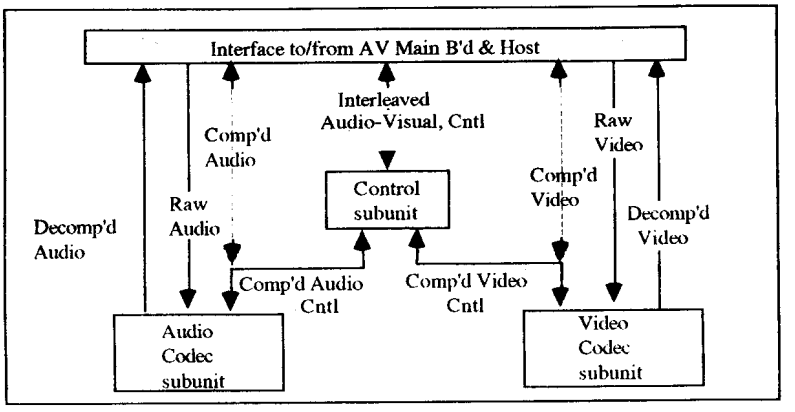

Fig. 5. Control Hierarchy of Codec board

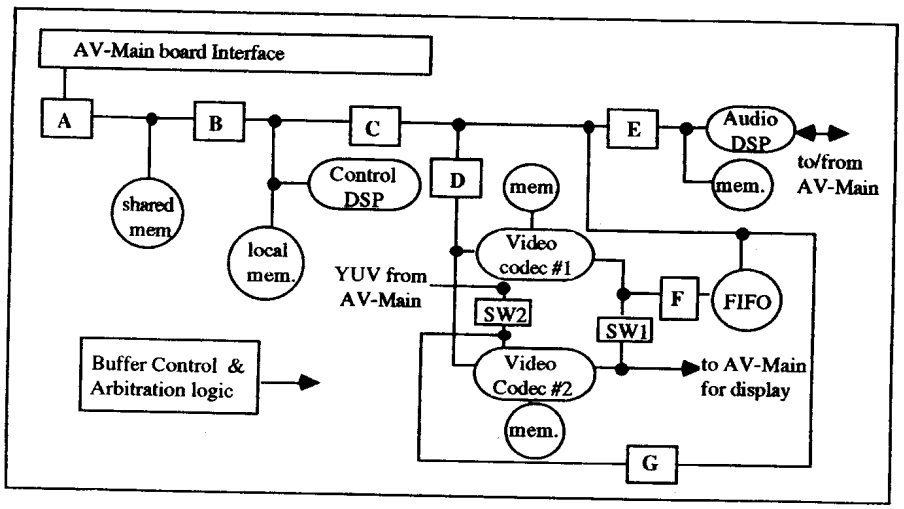

Fig. 6. Architecture of Codec board

The audio and video codec subunits can be controlled by either the processor of motherboard or the control subunit. In normal operation, the control module controls these modules. But for the purpose of ease in development and later diagnostics, paths are provided so that the processor of motherboard also can directly access and control each subunit of the Codec board. In Fig. 6, we depict the Codec board from the point of view of control and data flow among the major components. The Codec board is mapped into the memory space so that the processor of the motherboard can access all the memories and components of the codec board without restriction.

In this figure, the square boxes marked with letters A to G are bus buffers which determine the connection and direction of address and data buses connected to each buffer. Hence the buffer states and the buses provide paths required for control and data transfer. As an example, when the processor of motherboard accesses the FIFO to read out the compressed video data, the buffers A, B and C are ON and D, E and G are OFF, while F is irrelevant. At this moment, the operation of control DSP is momentarily stopped(held) to prevent the conflict between the control signals issued from the processor of the motherboard and those from the control DSP.

1. Operation

At the initialization, the desired coding and decoding microcodes are downloaded to the local memories of audio DSP and video DSPs, while interleaving and disinterleaving microcodes as well as house keeping microcodes are downloaded to the local memory of the control DSP. Then the house keeping codes are initiated and waits for the commands from the motherboard. If there are any commands given in the circular queue of the shared memory, the control DSP interprets them

and generates appropriate commands to audio or video DSPs or, if the command from motherboard is given to control DSP, executes it. At the same time, the control DSP executes interleaving and disinterleaving if it's necessary.

The shared memory is a communication buffer between the motherboard and control DSP. When the processor of motherboard wants to send data or commands, it writes them into the circular queue in the shared memory and interrupts the control DSP to notify it. Upon interrupt, the control DSP checks the shared memory to find the data or commands given. The communication from the control DSP to the motherboard is also based on the circular queue and interrupts. The communications between the control DSP and the audio DSP, and those between control DSP and video codec DSP are also based on interrupts. The informations are exchanged by reads and writes of the control DSP to the corresponding memories or buffers.

2. Control subunit

This subunit is composed of a TMS320C31 DSP, two banks of 512KBytes of SRAM and phase - locked - loop(PLL) circuits for synchronized interleaving of audio - visual data. Here, we describe how the 90KHz PLL is used in the codec board for synchronized interleaving.

In a digital audio - visual compression system, the precise lip synchronization between audio and video data is important since a slight slip in synchronization can be accumulated and can result in a noticeable mismatch in video and audio streams after a while. For instance, in a digital video stream running at 13.5M sample/sec, a slip of one clock period in each NTSC horizontal scan line(525lines/frame*15frames/sec) can result in 2 second's slip in an hour. However, this amount of slip can happen quite often due to the non - accuracy of commercial VCR or video camera. The main cause of this non - accuracy in the picture source is due to the mechanical part(head) of the picture capturing system which is subject to gravitational forces(mainly in camera). Hence, a 90KHz PLL is optionally used in MPEG to lock digital audio and video sources to a fixed frequency. In the codec board, either audio or video sampling rate is locked, but not both at the same time. The block diagram of 90KHz PLL is shown in Fig. 7.

For audio streams, sampling frequencies(32KHz, 44.1KHz, 48KHz) are adjusted to a fixed frequency for PLL input. As an example, the block diagram of the circuits for 32KHz sampling rate is shown in Fig. 8.

3. Audio codec subunit

Composed of a TMS320C31 and 512KBytes of memory, this subunit performs MPEG audio codec as well as G.726 or G.728 codec operations according to microcodes downloaded into the memory. In

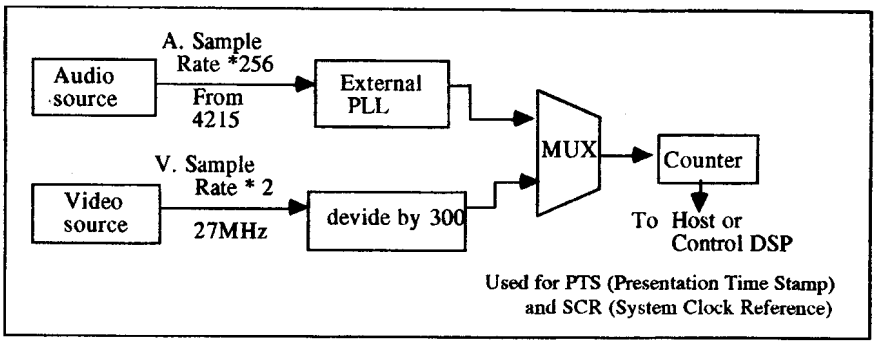

Fig. 7. Blockdiagram of Audio/Video 90KHz PLL

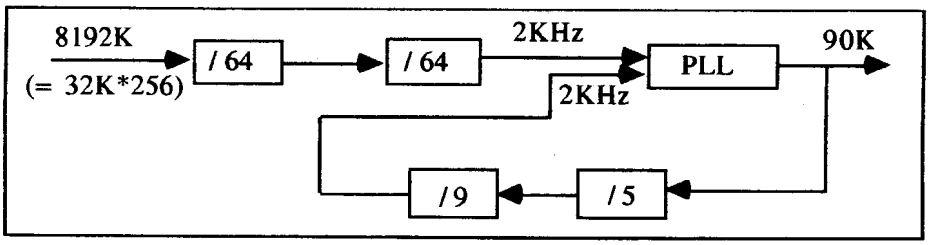

Fig. 8. Blockdiagram of Circuits for 32KHz Audio Sampling Rate

G.728/G.726 mode, this subunit is capable of performing the encoding and decoding processes at the same time to make possible the bidirectional communication required by video conferencing. It's worthwhile to note that the audio subunit can be used also for speech recognition or voice synthesis as we have audio I/O devices connected to this subunit and enough computing power necessary for those operations in the DSP. We have chosen the same DSPs for this subunit and control subunit to simplify the development environment and to reduce the development efforts.

4. Video codec subunit

This subunit is composed of two C - Cube's Video RISC chips, DRAMs and FIFO. The target performance of this subunit is summarized in Table 1.

In MPEG mode of operation, the switches SW1 and SW2 in Fig.6 are ON so that video_codec#1 and video_codec#2 operate in parallel. In encoding, the raw video data comes from AV - Main

Table 1. Target Performance of Video Subunit

	MPEG - I	H. 261
Resolution	352×240	352×288
Frame Rate	30 Frame/sec	30 Frame/sec
Encoding/Decoding	either encoding or decoding	simultaneous encoding & decoding

board and fed into video codec DSPs through the path shown in Fig.6. In decoding, the encoded data coming from the motherboard passes through buffers A, B, C and G. and then fed to the video_codec #1 and #2. Either in encoding or decoding, the first half of a video data is fed intothe video_codec #1 while the second half is fed into the video_codec#2 and the two DSPs operate in parallel. In H.261 mode, both switches SW1 and SW2 are OFF, so that video_codec#1 operates as a encoder while video_codec#2 operates as a decoder. In both MPEG and H.261 modes, the encoded video output is accumulated in FIFO and the decoded video output is directly sent to the AV - Main board for display.

Ⅴ. Conclusion

We described, in this paper, the architecture of the AV - Main and the Codec boards of a multimedia data processing hardware platform, which has audio - visual data capture/playback and system - level encoding/decoding functions. For the fluent flow of massive multimedia data, we have introduced the PCI bus based architecture in the AV - Main board as well as in Graphics board. The architecture of this system also allows the multimedia data related tasks to be independent from the motherboard. Once the video encoding and decoding microcode is fully supported, the audio - visual subsystem can be used for video conferencing, multimedia editing, business presentation, authoring tool, etc. The system architecture can be used with little change for MPEG - 2 level video and audio data copression and decompression.

References

1) Shepard, P and Salmony, M. "Extending OSI to Support Synchronization Required by Multimedia Applications", Computer Communications, Vol. 13, No. 7. pp. 399 - 406, Sept. 1990
2) Song, Dongho et al. "COSMOS : An Extended Operating System for Multimedia Group Presentation", 4th International Workshop on Network and Operating system support for Digital Audio and Video", Lanchaster, UK. Nov. 1993
3) Northcut, J. D. and Kuerman, E. M. "System Support for Time - Critical Applications", 2nd International Workshop on Networking and Operating System Support for Digital Audio and Video", Heidelberg, Germany. Nov. 1991
4) Watanabe et al. "Distributed Multiparty Desktop Conferencing System : MERMAID", Proceedings CSCW '90 Conf. on Computer Supported Cooperative Work, Los Angeles, CA. US. pp. 27 - 38, Oct. 1990
5) Loeb, S. "Delivering Interactive Multimedia Documents over Networks", IEEE Communications Magazine, Vol. 30, No. 5, 1992
6) Baker, Rusti et al. "Multimedia Processing Model for a Distributed Multimedia I/O System", Proceedings of the 3rd International Workshop on Networking and Operating System Support for Digital Audio and Video, San

Diego, CA, Nov. 1992

7) Leydekkers. P. "Synchronization of Multimedia Data Streams in Open Distributed Environments", 2nd International workshop on networking and Operating System Support for Digital Audio and Video", Heidelberg, Germany. Nov. 1991
8) Steinmetz, R. "Synchronization Properties in Multimedia Systems", Journal on Selected Areas of Communications, Vol.8, No.3. Apr. 1990
9) CS4231 Data Sheet, "Parallel Interface, Multimedia Audio Codec", Nov. 1993, Crystal Semiconductor Corp.
10) CS4215 Data Sheet, "16 - bit Multimedia Audio Codec", Sept. 1993, Crystal Semiconductor Corp.
11) D.H. Song, COSMOS, "A Framework for Extended Operating Systems for Multimedia Group Presentation", Proceedings 1995 Pacific Workshop on Distributed Mutimedia Systems, March, Tokai Univ. at Honolulu, USA