(12) UK Patent Application (19) GB (11) 2 443 968 (13) A

(43) Date of A Publication 21.05.2008

(21) Application No: 0722700.2

(22) Date of Filing: 20.11.2007

(30) Priority Data:
(31) 0623097 (32) 20.11.2006 (33) GB

(51) INT CL:
H04N 7/15 (2006.01)

(56) Documents Cited:
WO 1996/023388 A1
US 6963353 B1
US 20050108328 A1
US 20020064149 A1

(58) Field of Search:
INT CL H04N
Other: WPI, EPODOC, Full text database cluster TXTE

(71) Applicant(s):
Codian Ltd
(Incorporated in the United Kingdom)
14 Waterside Drive, Langley, SLOUGH,
SL3 6EZ, United Kingdom

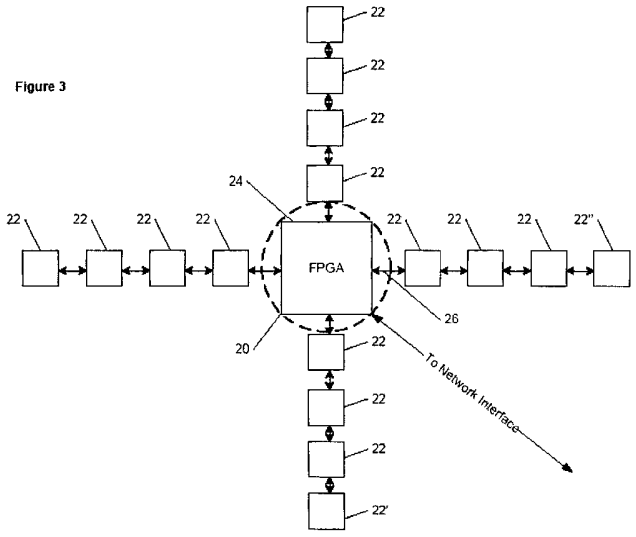
(72) Inventor(s):
Simon James Winn Evans

(74) Agent and/or Address for Service:
Scott & York Intellectual Property
45 Grosvenor Road, ST ALBANS,
Hertfordshire, AL1 3AW, United Kingdom

(54) Abstract Title: **Video conference multipoint control unit (MCU) allowing unicast and multicast transmission**

(57) A video processing architecture for a multipoint control unit (MCU) comprises plural video image signal processing means being interconnected via switches operable to choose between unicast and multicast onward transmission of (received) image data. Also independently claimed is a method of routing data in such a multipoint control unit (MCU), comprising storing a network map holding data representing a network topology, the unicast or multicast mode switching being dependant on the network map. Information may be transmitted in unicast mode until a routing branch is reached, when multicast mode may be used. By using high bandwidth internal links and an architecture with few bottlenecks, prior problems in providing high definition endpoint display, with different video compositions for each conference participant, are alleviated. Also, processing image data internally using intelligent multicasting, scaling and compression decisions allows processing and bandwidth requirements to be met.

Figure 3



GB 2 443 968 A

PRIOR ART

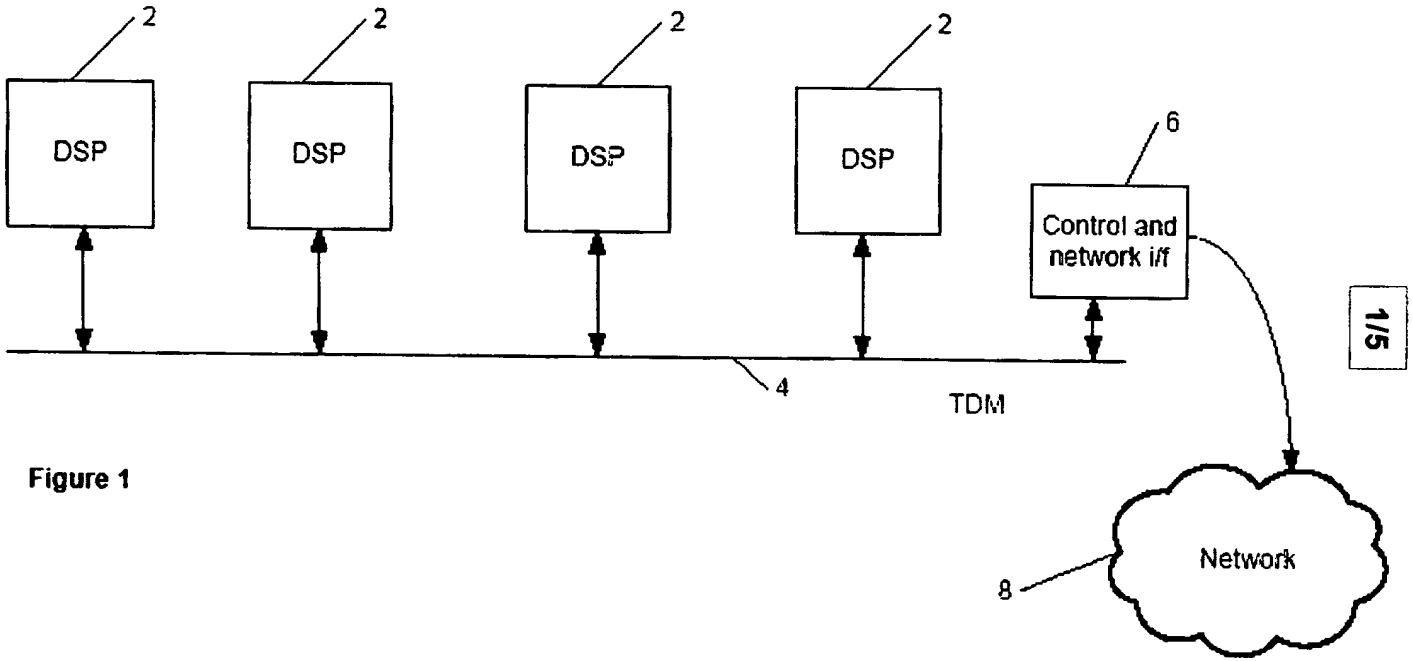


Figure 1

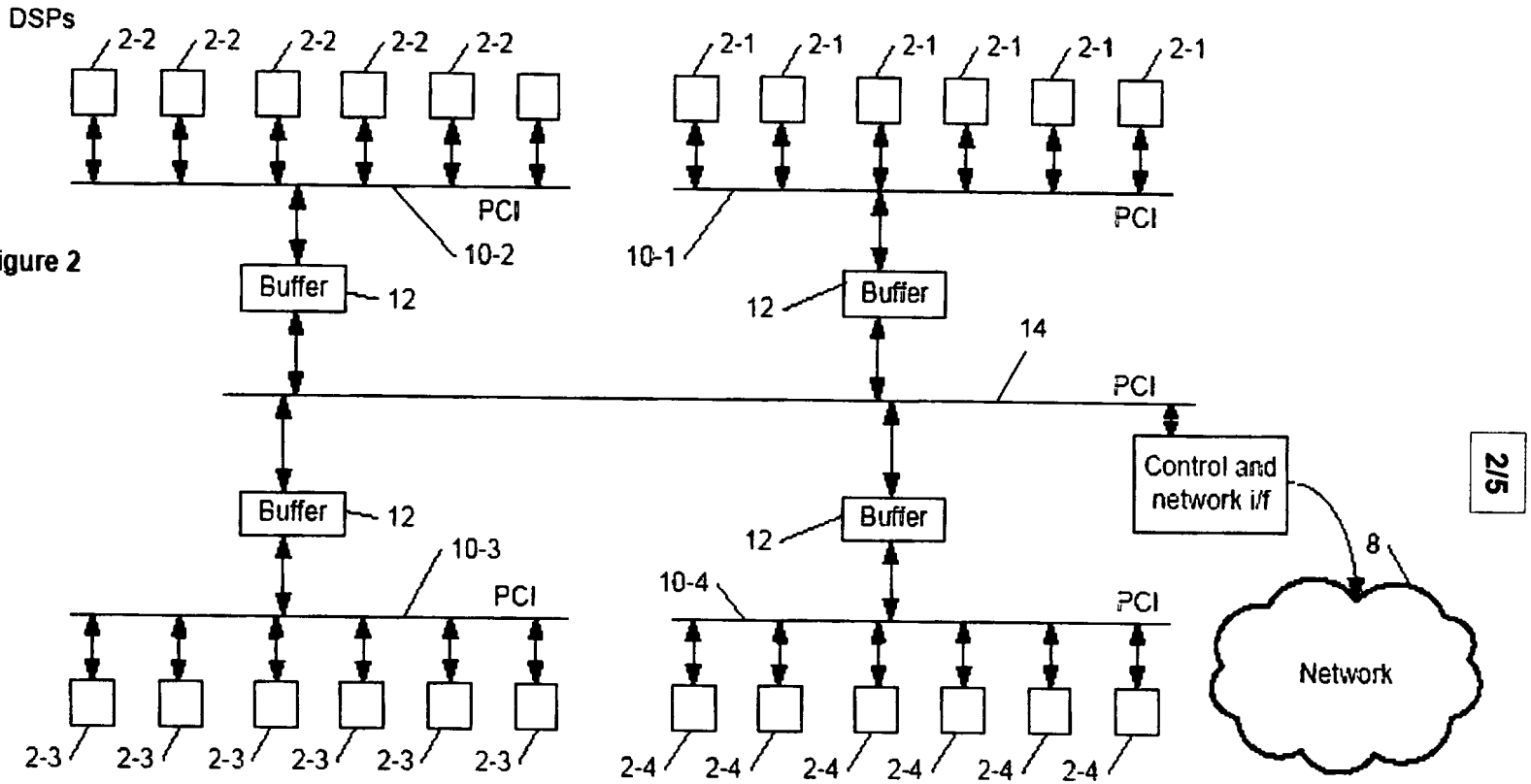


Figure 2

PRIOR ART

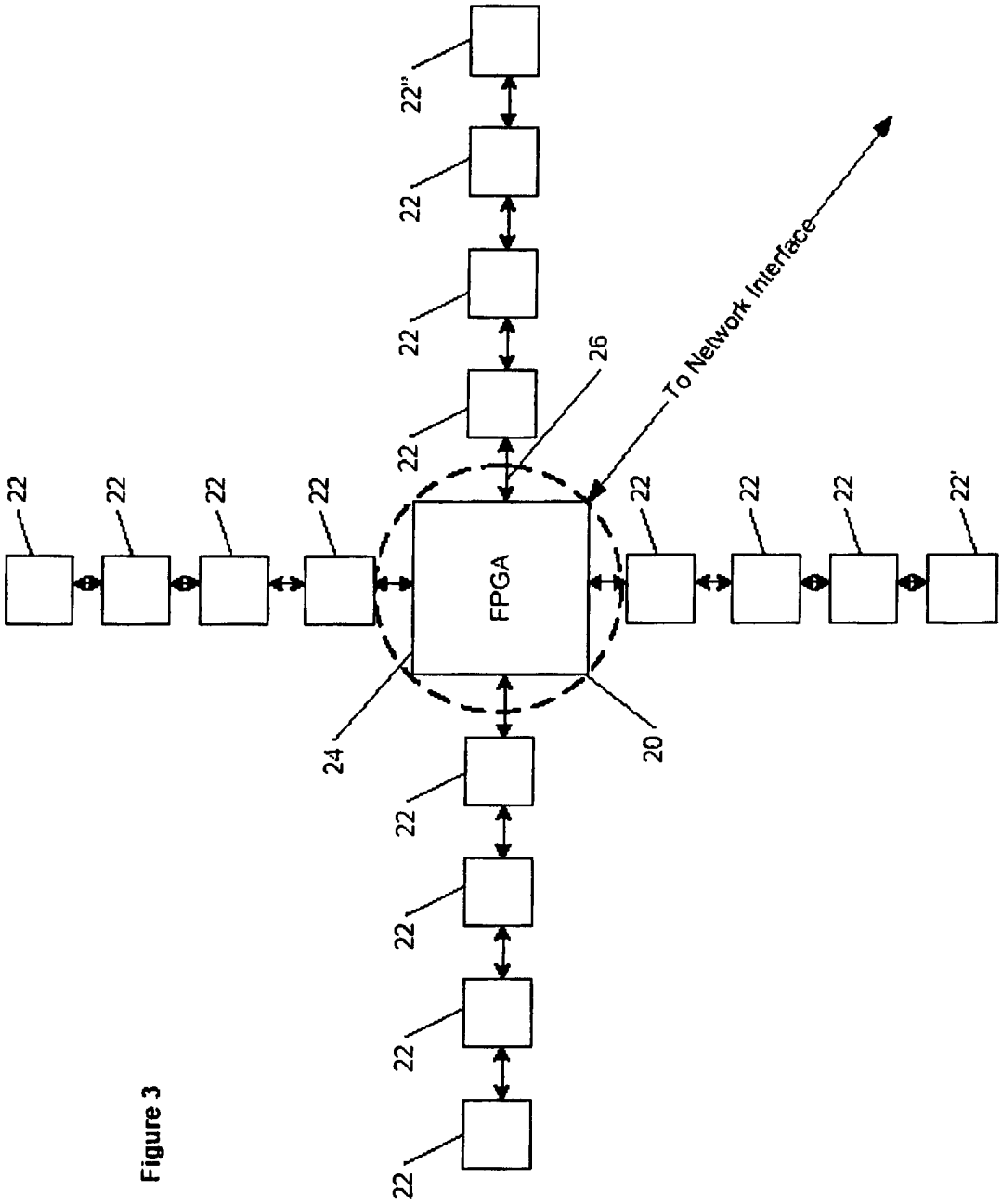

22''
22
22
22
26
To Network Interface
22
22
22
22
FPGA
22
22
22
22'
24
20
22
22
22
22

Figure 3

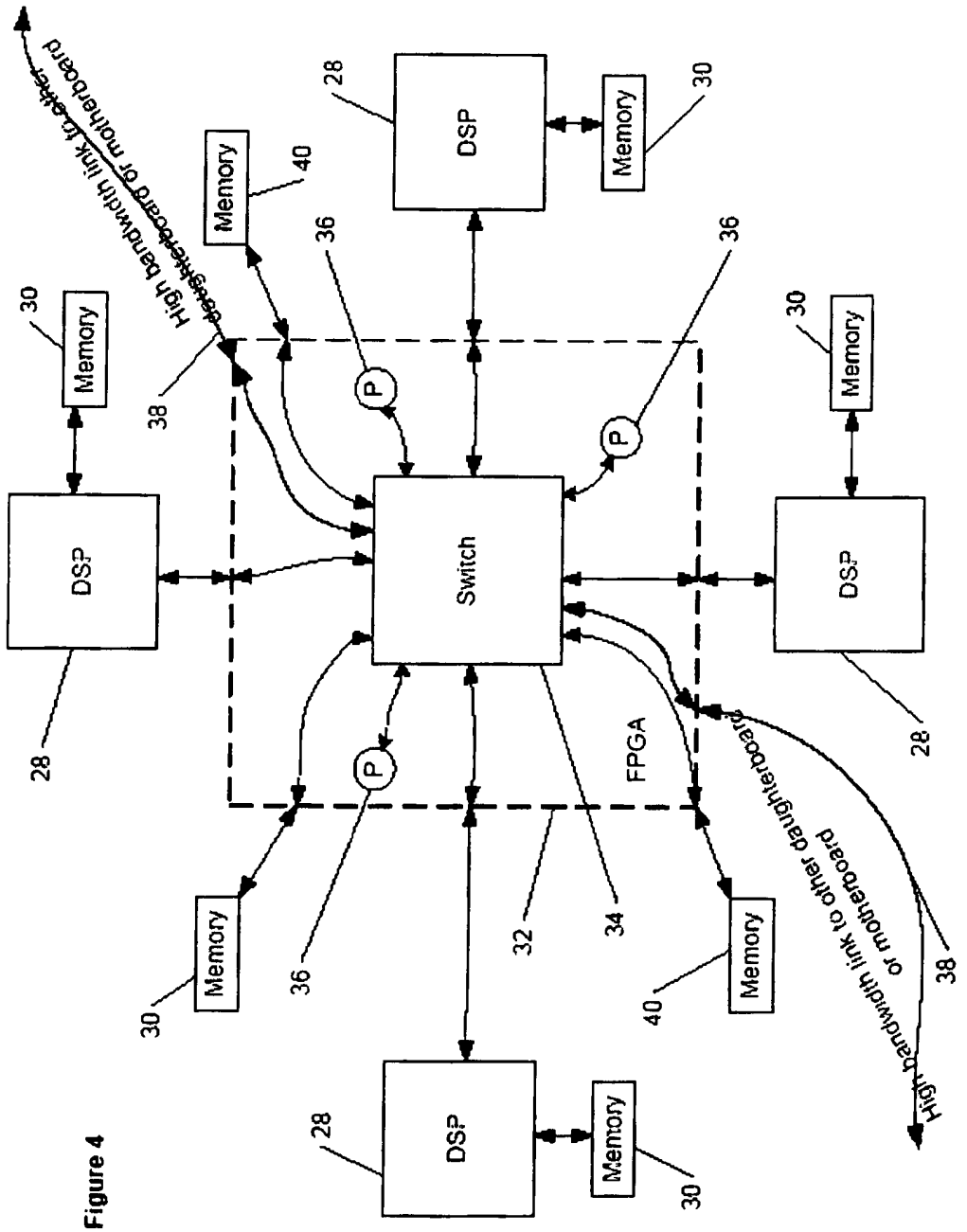


Figure 4

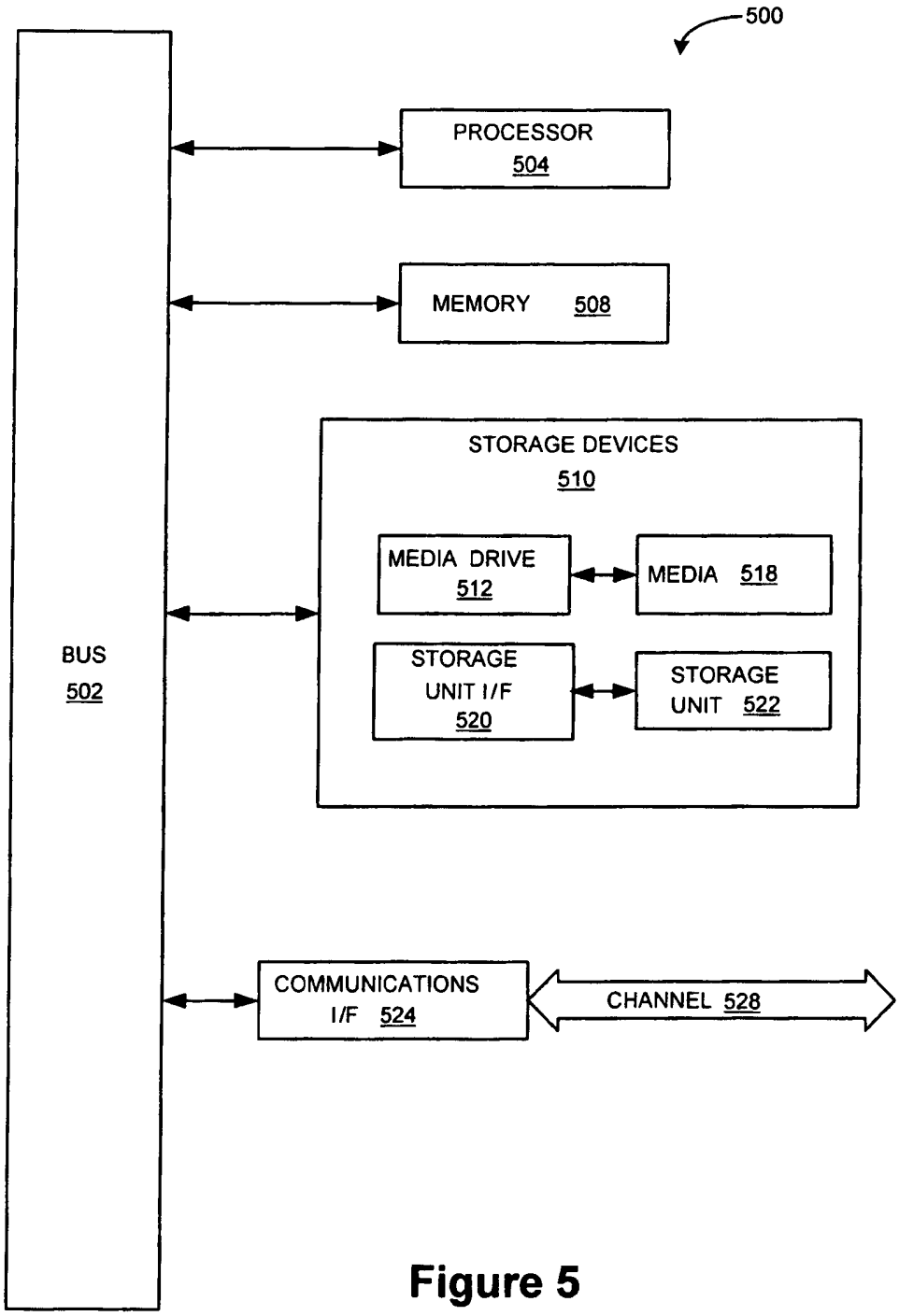

500
PROCESSOR
504
MEMORY 508
STORAGE DEVICES
510
MEDIA DRIVE
512
MEDIA 518
STORAGE
UNIT I/F
520
STORAGE
UNIT 522
BUS
502
COMMUNICATIONS
I/F 524
CHANNEL 528

Figure 5

HARDWARE ARCHITECTURE FOR VIDEO CONFERENCING

This invention relates to a hardware architecture for a multipoint control unit.

Video conferencing and the associated hardware, falls broadly into two camps. In the first camp, "conferencing" occurs between only two participants and the participants are connected directly to one another through some form of data network. In this form of network, only two endpoints are involved and true conferencing only occurs if multiple participants are present at one of the two endpoint sites. Examples of this type of conferencing are, at the low technology end, PC enabled endpoints interconnecting using software such as Netmeeting or Skype and at the higher end equipment using dedicated endpoint hardware interconnected, for example, via ISDN links.

In the second camp, video conferencing allows more than two endpoints to interact with one another. This is achieved by providing at least one centralised co-ordinating point; a so-called "multipoint control unit (MCU)", which receives video and audio streams from the endpoints, combines these in a desired way and re-transmits the combined composite video/audio stream to the participants. Typically the conference view transmitted to the endpoints is the same for each endpoint. The composition may change over time but is the same for all the participants.

The provision of only a single composition is a significant problem because each participant must therefore receive a conference stream tailored so that it is acceptable to the least capable endpoint in the conference. In this situation therefore many endpoints are not used to their full capacity and may experience degraded images and audio as a result.

More recently, modern MCUs such as the Codian MCU 4200 series have been designed to allow a unique view to be created for each participant. This allows the full capabilities of each endpoint to be utilised and also allows different compositions for different participants so that, for example, the emphasis of a particular participant in the conference may be different for a different user. However, the processing of video data in real time is a highly processor intensive task. It also requires the movement of large

quantities of data. This is particularly so once the data has been decompressed in order to perform high quality processing. Thus processing power and bandwidth constraints are a significant bottleneck in the creation of high quality video conferencing MCUs which allow multiple views of the conference to be produced.

Figure 1 shows a typical prior art MCU architecture.

This architecture has a plurality of digital signal processors 2 such as the Texas Instruments TMS series, which are interconnected via a Time Division Multiplexed (TDM) bus 4. A controller and network interface 6 is also connected to the TDM bus. Each DSP 2 is allocated one or more frames on the TDM bus. It will be appreciated that the TDM bus is a significant bottleneck. Whilst increased processing power for the MCU may be achieved by adding more powerful DSPs or additional DSPs, all the data flowing between DSPs and between the network 8 and the DSPs must fit into a finite number of time slots on the TDM bus 4. Thus, this form of architecture scales poorly and cannot accommodate the processing requirements of per-participant compositions.

Figure 2 shows an alternative prior art configuration. A plurality of DSPs 2-1 are each connected to a PCI bus 10-1. Similarly, a plurality of DSPs 2-2, 2-3 and 2-4 are connected to respective PCI buses 10-2, 10-3 and 10-4. The PCI buses 10-2, 10-3 and 10-4 are in turn connected via buffers 12 to a further PCI bus 14. A significant advantage of this architecture over that shown in Figure 1 is that the DSPs in group 2-1 may communicate amongst one another with the only bottleneck being the PCI bus 10-1. This is true also for the groups 2-2, 2-3 and 2-4. However, should a DSP in group 2-1 wish to communicate with a DSP for example, in group 2-3, the PCI bus 14 must be utilised. Thus although this architecture is a significant improvement on that shown in Figure 1 in terms of scalability and the ability to effectively use a plurality of DSPs, the PCI bus 14 must still be used for certain combinations of intra-DSP communication and thus becomes a performance limiting factor for the MCU.

Attempts have been made to offload processing from DSPs. For example, IDT has recently released details of a so-called "Pre-processing switch (PPS)" under part number IDT 70K2000. The PPS carries out predetermined functions before delivery to

a processor such as a DSP or FPGA. Processing is determined based on the address range on the switch to which packets are sent. The chip is designed for use in 3G mobile telephony and is designed to offload basic tasks from DSPs which would normally be carried out inefficiently by the DSP. US6,883,084 also proposes the use of path processing although in that case it is proposed as an alternative to a Von Neumann type sequential processor. This patent proposes the use of a plurality of path processors carrying out simultaneous processing of alternative data sets so that the processing of unusable data sets does not delay programme flow. It teaches against a hybrid approach including path processors with other types of processors.

According to the invention provides a video processing architecture for a multipoint control unit, comprising a plurality of signal processing means adapted to perform processing of data representative of video images, the signal processing means being interconnected via switches operable to choose between unicast and multicast onward transmission of received data.

In a third aspect, the invention provides a method of routing data in a multipoint control unit having a plurality of signal processing means, comprising storing a network map holding data representative of a network topology which interconnects the signal processing means, providing a plurality of switches to switch data between the signal processing means, and carrying out switching in unicast or multicast mode dependent on the network map.

Other aspects and features of the present invention will become apparent to those ordinarily skilled in the art upon review of the following description of specific embodiments of the invention in conjunction with the accompanying figures.

Figure 1 is a schematic block diagram of a prior art MCU architecture;

Figure 2 is schematic block diagram of an alternative prior art MCU architecture;

Figure 3 is a schematic block diagram showing a motherboard and a plurality of daughterboards in accordance with the present invention; and

Figure 4 is a schematic block diagram of a daughterboard in accordance with the invention.

Figure 5 illustrates an exemplary computing system that may be employed to implement processing functionality in embodiments provided herein.

With reference to Figure 3, a motherboard 20 carries a field programmable gate array (FPGA) and other associated components. The motherboard 20 may include control circuitry which, for example, enables an auto attendant interface to be produced to allow users to configure the MCU and which may also control data flow in the MCU. These components may alternatively be on a separate board as is known in the art.

The motherboard 20 also includes connectors which permit the mounting of one or more daughterboards 22. In the preferred embodiment, four daughterboards may be connected to the motherboard 20. The connection may for example be made using pluggable connectors. By using a plurality of such connectors, in the preferred embodiment the daughterboards are both electrically coupled and mechanically mounted to the motherboard by such connectors.

The motherboard 20 carries an FPGA 24 which carries out routing functions (among other functions). Primarily, the FPGA 24 routes data between the controller (not shown), network interface (not shown) and the plurality of daughterboards 22. The FPGA 24 preferably has four high bandwidth links 26 which may have a bandwidth of 3Gb/sec or higher and which connect the motherboard with a first layer of daughterboards. It is noted that links 26 (and 38 below) may include physical links, a switch fabric, or other suitable structures or system for connecting motherboards, daughterboards, and DSPs. Data flows to the distal daughterboards are routed through the first layer of daughterboards as explained in more detail below.

With reference also to Figure 4, each daughterboard typically has four DSPs 28 each with associated memory 30. Each daughterboard also has an FPGA 32 which incorporates a switch 34. Switch 34 may include structure logic for receiving packets

on an input and sending the packets out in a selectable manner, e.g., similar to a network switch. The FPGA 32 includes stream processors 36 which are described in more detail below, and two high bandwidth links 38.

The daughterboards 22 are each identical and the links 38 may be used to connect to another daughterboard or to the motherboard. In this way, extra processing capability may be added to the architecture simply by adding additional daughterboards. In a minimal configuration, a single daughterboard may be mounted on the motherboard. In a maximal configuration, four daughterboards may be mounted to the motherboard and each daughterboard may have additional daughterboards (three in this example) stacked thereon. As explained above, each daughterboard itself may include four DSPs and thus in this particular example, a configuration including four daughterboards, the architecture may have 64 DSPs. Of course, various numbers of DSPs and/or daughterboards may be used and the maximal configuration is with reference only to this particular example of 16 daughterboards, each including four DSPs.

Several strategies are used to alleviate bandwidth congestion on the interconnects between the DSPs. Firstly, each interconnect between daughterboards operates at a bandwidth of 3 Gb/sec or higher which is a substantially higher bandwidth than in the prior art. Secondly, the daughterboards each have only four DSPs sharing a local interconnect which may communicate amongst one another without using bandwidth on any other interconnect in the architecture. Thus with appropriate resource allocation, the DSPs on any one daughterboard may experience high utilisation without significant bandwidth impact for the architecture as a whole.

Furthermore, data may flow between DSPs in any one of the four branches shown in Figure 3, without using bandwidth available to the other branches.

A further strategy involves the use of stream processors 36 located in each of the daughterboard FPGAs. These stream processors take advantage of an unusual characteristic of video conferencing as explained below.

Typically, data flowing between endpoints in a video conference is highly compressed in view of bandwidth constraints, for example, with Internet connected endpoints However, this compression prevents manipulation of the images. Thus within an MCU, video processing is carried out on uncompressed data. Typically this increases the volume of data by a factor between 10 and 100 and typically by a factor of about 80. Thus a typical video stream may have a bandwidth requirement of 50 Mb/sec, for example. This is a significant problem peculiar to video conferencing since processing is carried out on many simultaneous streams and must be carried out in real time. However, since the end result of the processing will be transmitted in compressed form and also typically over a lossy network, it is acceptable to carry out compression within the MCU. Such compression may be lossless or given the nature of the output network, lossy. Thus Applicants have appreciated that the bandwidth constraints within the MCU may be alleviated by performing compression and decompression within the MCU for data in transit between DSPs. However, this in itself is computationally expensive. Accordingly, the novel architecture of the present invention includes stream processors 36 which are formed in each daughterboard FPGA 32. The media stream processors 36 may act on several pixels when performing compression and thus the FPGA may keep a frame or a portion of a video frame in memory 40 so that the processors 36 in this mode are not strictly stream processors.

The processors 36 may carry out further operations including, but not limited to composition, alpha blending, motion compensation, variable length encoding and decoding, frame comparison, combinations thereof, and the like. By carrying out these steps on the fly as data is passed between DSPs 28, processing load is removed from the DSPs and also bandwidth limitations are mitigated.

As a further enhancement, data destined for several different DSPs may be sent in unicast format until a routing branch is required in which case some data may be sent in multicast form. This avoids having multiple streams of the same data passing along the same link. For example, if the daughterboard 22 at the far left of Figure 3 wishes to communicate with a DSP on the daughterboard 22' at the bottom of the figure and also with the daughterboard 22" at the far right of the figure, the data may be unicast until it reaches the motherboard 20 at which point it may be multicast to each of the two

respective branches of daughterboards radiating out from the motherboard and then unicast along each of the branches. This step may be carried out within the FPGA as part of its routing algorithm. To facilitate this, each switch may maintain a representation of the topology of the entire MCU architecture, for example in tree form, and is operable to manipulate the tree and to determine an appropriate multicast or unicast format for the next hop or hops. Alternatively, the route may be determined at the data source and routing information carried with the data which is interpreted by the switches enroute.

The media stream processors 36 may also use factorised scaling to assist with reducing the bandwidth of communications between DSPs. For example, if different participant compositions require differently scaled versions of the same image such as an image scaled to a half for one participant and a quarter for another participant, the FPGAs may be configured to make sensible scaling decisions. In this example the FPGA may scale the whole image to a half, transmit the thereby reduced data as far as the routing branch which chooses between the DSP which will process the half and the DSP which will process the quarter image and at that point further scale the image down to a quarter for onward transmission to the DSP dealing with the quarter scaled image.

The intelligent routing, multicast and scaling/compression operations are carried out by each daughterboard FPGA and accordingly the processing load for these intelligent routing decisions is distributed amongst each of the daughterboards.

In this way, therefore, the architecture described above maximises the utilisation of the DSPs by ensuring that data is ideally allocated to local DSPs and also where data must be transmitted between more distant DSPs, that the data is transmitted in the most efficient format. Furthermore by employing very high bandwidth links between the DSPs, any bandwidth bottlenecks are largely avoided. Accordingly, the architecture provides a highly scalable and very powerful processing platform for high definition per participant composed multi-conference video conferencing.

Of course, other features and advantages will be apparent to those skilled in the art. The foregoing system overview represents some exemplary implementations, but other implementations will be apparent to those skilled in the art, and all such alternatives are deemed equivalent and within the spirit and scope of the present invention, only as limited by the claims.

Those skilled in the art will further recognize that the operations of the various embodiments may be implemented using hardware, software, firmware, or combinations thereof, as appropriate. For example, some processes can be carried out using processors or other digital circuitry under the control of software, firmware, or hard-wired logic. (The term "logic" herein refers to fixed hardware, programmable logic and/or an appropriate combination thereof, as would be recognized by one skilled in the art to carry out the recited functions.) Software and firmware can be stored on computer-readable media. Some other processes can be implemented using analog circuitry, as is well known to one of ordinary skill in the art. Additionally, memory or other storage, as well as communication components, may be employed in embodiments of the invention.

Figure 5 illustrates a typical computing system 500 that may be employed to implement processing functionality in embodiments of the invention. Computing systems of this type may be used in the any one or more of an MCU, controller, motherboard, daughterboard, or DSP, for example. Those skilled in the relevant art will also recognize how to implement embodiments of the invention using other computer systems or architectures. Computing system 500 can include one or more processors, such as a processor 504. Processor 504 can be implemented using a general or special purpose processing engine such as, for example, a microprocessor, microcontroller or other control logic. In this example, processor 504 is connected to a bus 502 or other communications medium.

Computing system 500 can also include a main memory 508, such as random access memory (RAM) or other dynamic memory, for storing information and instructions to be executed by processor 504. Main memory 508 also may be used for storing temporary variables or other intermediate information during execution of instructions to be

executed by processor 504. Computing system 500 may likewise include a read only memory (ROM) or other static storage device coupled to bus 502 for storing static information and instructions for processor 504.

The computing system 500 may also include information storage system 510, which may include, for example, a media drive 512 and a removable storage interface 520. The media drive 512 may include a drive or other mechanism to support fixed or removable storage media, such as a hard disk drive, a floppy disk drive, a magnetic tape drive, an optical disk drive, a compact disk (CD) or digital versatile disk (DVD) drive (R or RW), or other removable or fixed media drive. Storage media 518, may include, for example, a hard disk, floppy disk, magnetic tape, optical disk, CD or DVD, or other fixed or removable medium that is read by and written to by media drive 514. As these examples illustrate, the storage media 518 may include a computer-readable storage medium having stored therein particular computer software or data.

In alternative embodiments, information storage system 510 may include other similar components for allowing computer programs or other instructions or data to be loaded into computing system 500. Such components may include, for example, a removable storage unit 522 and an interface 520, such as a program cartridge and cartridge interface, a removable memory (for example, a flash memory or other removable memory module) and memory slot, and other removable storage units 522 and interfaces 520 that allow software and data to be transferred from the removable storage unit 518 to computing system 500.

Computing system 500 can also include a communications interface 524. Communications interface 524 can be used to allow software and data to be transferred between computing system 500 and external devices. Examples of communications interface 524 can include a modem, a network interface (such as an Ethernet or other network interface card (NIC)), a communications port (such as for example, a USB port), a PCMCIA slot and card, etc. Software and data transferred via communications interface 524 are in the form of signals which can be electronic, electromagnetic, optical or other signals capable of being received by communications interface 524. These signals are provided to communications interface 524 via a channel 528. This channel

528 may carry signals and may be implemented using a wireless medium, wire or cable, fiber optics, or other communications medium. Some examples of a channel include a phone line, a cellular phone link, an RF link, a network interface, a local or wide area network, and other communications channels.

In this document, the terms "computer program product," "computer-readable medium" and the like may be used generally to refer to media such as, for example, memory 508, storage device 518, or storage unit 522. These and other forms of computer-readable media may store one or more instructions for use by processor 504, to cause the processor to perform specified operations. Such instructions, generally referred to as "computer program code" (which may be grouped in the form of computer programs or other groupings), when executed, enable the computing system 500 to perform functions of embodiments of the invention. Note that the code may directly cause the processor to perform specified operations, be compiled to do so, and/or be combined with other software, hardware, and/or firmware elements (e.g., libraries for performing standard functions) to do so.

In an embodiment where the elements are implemented using software, the software may be stored in a computer-readable medium and loaded into computing system 500 using, for example, removable storage drive 514, drive 512 or communications interface 524. The control logic (in this example, software instructions or computer program code), when executed by the processor 504, causes the processor 504 to perform the functions of embodiments of the invention as described herein.

It will be appreciated that, for clarity purposes, the above description has described embodiments of the invention with reference to different functional units and processors. However, it will be apparent that any suitable distribution of functionality between different functional units, processors or domains may be used without detracting from embodiments of the invention. For example, functionality illustrated to be performed by separate processors or controllers may be performed by the same processor or controller. Hence, references to specific functional units are only to be seen as references to suitable means for providing the described functionality, rather than indicative of a strict logical or physical structure or organization.

Although embodiments of the invention have been described in connection with some embodiments, it is not intended to be limited to the specific form set forth herein. Rather, the scope of embodiments of the invention is limited only by the claims. Additionally, although a feature may appear to be described in connection with particular embodiments, one skilled in the art would recognize that various features of the described embodiments may be combined in accordance with embodiments of the invention.

Furthermore, although individually listed, a plurality of means, elements or method steps may be implemented by, for example, a single unit or processor. Additionally, although individual features may be included in different claims, these may possibly be advantageously combined, and the inclusion in different claims does not imply that a combination of features is not feasible and/or advantageous. Also, the inclusion of a feature in one category of claims does not imply a limitation to this category, but rather the feature may be equally applicable to other claim categories, as appropriate.

Claims

1. A video processing hardware architecture for a multipoint control unit, comprising a plurality of signal processing means adapted to perform processing of data representative of video images, the signal processing means being interconnected via switches operable to choose between unicast and multicast onward transmission of received data.

2. A video processing architecture according to claim 1, including a topology store which contains data representative of the links between the signal processing means and wherein the switches access the topology store to make routing decisions and also to make the choice between unicast or multicast onward transmission of data.

3. A video processing architecture according to claim 1, including a topology store which contains data representative of the links between the signal processing means and control means operable to attach routing information to data related to a desired route across the links before the data is transmitted.

4. A method of routing video data in a multipoint control unit having a plurality of signal processing means, comprising storing a network map holding data representative of a network topology which interconnects the signal processing means, providing a plurality of switches to switch data between the signal processing means, and carrying out switching in unicast or multicast mode dependent on the network map.

5. A method of routing according to claim 4, including transmitting information in unicast mode until a routing branch is reached at which data needs to follow a plurality of links and then transmitting information in multicast mode at that routing branch.

6. A computer readable medium encoded with computer program instructions which when executed on hardware for routing video data in a multipoint control

unit having a plurality of signal processing means, causes the hardware to carry out the steps of claim 4.

Intellectual Property Office

For creativity and innovation

14

Application No: GB0722700.2 **Examiner:** Matthew Males

Claims searched: 1 & 4 **Date of search:** 14 March 2008

Patents Act 1977: Search Report under Section 17

Documents considered to be relevant:

Category	Relevant to claims	Identity of document and passage or figure of particular relevance
X	1-4 & 6	US 2002/0064149 A1 ELLIOTT et al - see, for example, paragraphs [2066] [2072] [2090] [2205] [2207] [2209] [2257]-[2305] etc; Figs 19c, 22.
X	1-4 & 6	US 2005/0108328 A1 BERKELAND et al - see abstract; paragraphs [0002]-[0008], [0012]-[0016], [0031] etc.
X	1	US 6963353 B1 FIRESTONE - see abstract; col 2, line 64 - col 3, line 7; col 4, lines 21-31; Fig 1.
X	1	WO 96/23388 A1 VIDEO SERVER INC - see abstract.

Categories:

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art
Y	Document indicating lack of inventive step if combined with one or more other documents of same category	P	Document published on or after the declared priority date but before the filing date of this invention
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

Field of Search:

Search of GB, EP, WO & US patent documents classified in the following areas of the UKC^{X} :

Worldwide search of patent documents classified in the following areas of the IPC

H04N

The following online and other databases have been used in the preparation of this search report

WPI, EPODOC, Full text database cluster TXTE

International Classification:

Subclass	Subgroup	Valid From
H04N	0007/15	01/01/2006