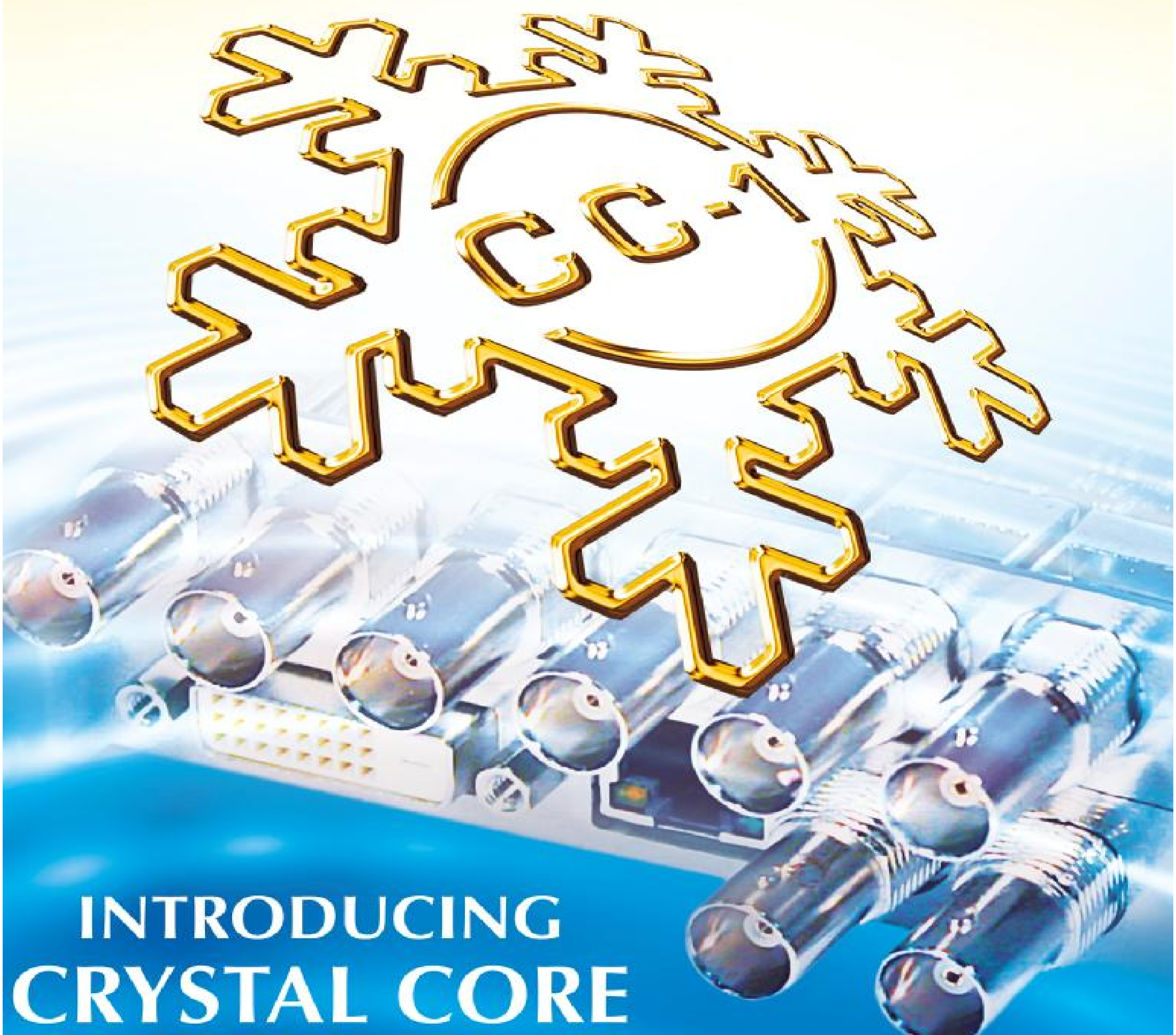
CC-1 Whitepaper

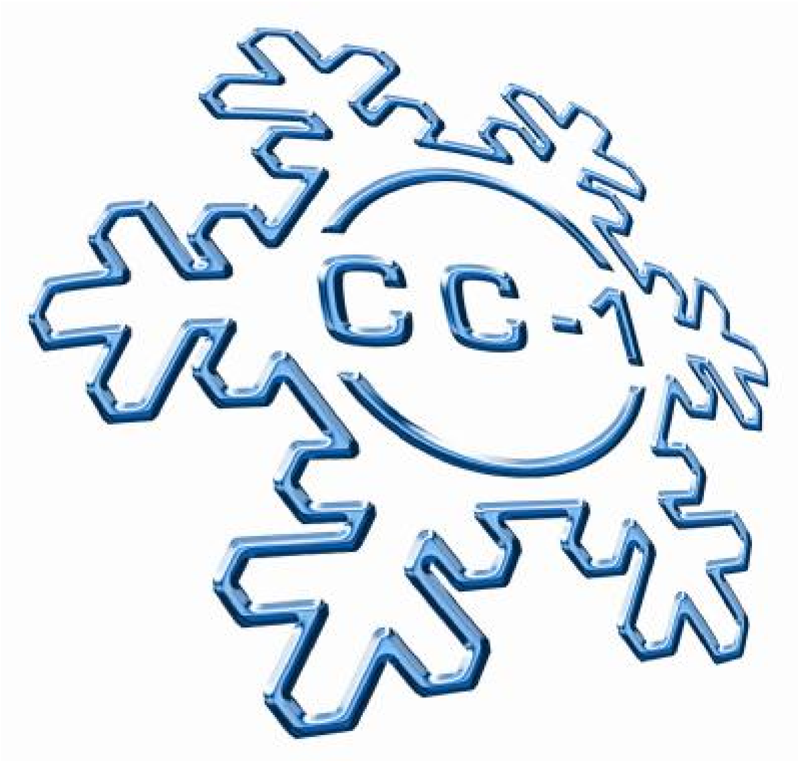


INTRODUCING
CRYSTAL CORE
TECHNOLOGY

The Media Platform
for the 21st Century

FAIRLIGHT

Introducing a new processing platform to support audio production using 21st Century FPGA technology



U.S Patent Pending Serial #: 11/379278
PCT Patent Pending Serial #: PCT/AU2006/0005

1. Introduction

While the demand for larger systems, better quality and lower cost grows exponentially, the professional media industry has continued to rely on two decade-old technology.

Today Fairlight announces a breakthrough – a new stream of audio and video products incorporating its CC-1 (Crystal Core Technology patent pending). This fresh paradigm processes data in a massive Field Programmable Gate Array (FPGA), architected into what amounts to a purpose-built media processing chip.

FPGAs deliver power at price points that will ultimately obsolete the established CPU and DSP/Time-Slice-Bus architectures. With very low processing latency and enough speed to provide smooth analogue-feel tactile response, this technology is fast becoming the bench mark platform for the 21st century. Fairlight's CC-1 is a media-optimised FPGA architecture that harnesses the step change in performance to support improved quality, faster job turnarounds and the development of new creative opportunities for many years to come.

Crystal Core is an aggregation of IP cores, the results of 20+ man-years of development combined with Fairlight's 150+ man-years of experience as a digital audio pioneer.

In recent years FPGAs have emerged as the front-running computer engine. They can be programmed so flexibly that, aside from a host PC and some standard memory and "glue", one chip forms an entire system. And a powerful one – single-chip applications include a 200-plus channel audio recorder/editing/mixer with full I/O and plug-ins; colour-grading for uncompressed HD video; integrated audio/video editing systems; DXD, Super HiDef and emerging 3D audio standards. For larger systems CC-1 uses a wide, fast data highway to interconnect across chips, between computers, or from room to room.

CC-1 introduces a disruptive new technology by delivering improved quality, unparalleled flexibility, scalability, enlarged system scope, and a quantum leap in affordability. And the future looks even brighter – by employing FPGAs, Crystal catches the next upsweep in computer hardware development, transparently inheriting the power increases of each successive generation of silicon, continuing the promise of Moore's Law long after its exhaustion in the microprocessor and DSP technology streams.

2. The Invention

The Crystal Core technology platform CC-1, gets its name from the crystalline interconnect scheme used at every level of system architecture. Processing blocks connect in three dimensions to other blocks, forming an extensible lattice that scales as required to meet the volume processing demands of media applications.

The processing blocks themselves are FPGAs, each programmed with the embedded intelligence required to fulfil the demands of the particular product or system being addressed. The extreme flexibility in programming and operation of these chips is crucial in enabling the creation of a universal hardware solution to a great number of media applications. Fairlight's invention (patent pending) defines an architectural arrangement of IP cores supporting real-time audio and video processing systems.

The first products to be released as a result of this invention will support fully featured audio production systems capable of delivering 230 Channel paths; each with eight bands of EQ, 3 stage dynamics processing, Floating insert point with return, on board HD video, 12 Auxiliary sends and up-to 72 user definable mix busses: all from a **single** CC-1 card.

3. CC-1 Architecture

3.1 Processor Chip

The Crystal architecture is based on the use of high-density FPGAs. Each of these chips contain millions of simple logic parts, that can either statically or dynamically upload a program that configures those parts into complex arrangements for high-speed computer applications. These applications will be used to replace media signal processing hardware that was previously composed of dedicated components or partially-programmable digital hardware.

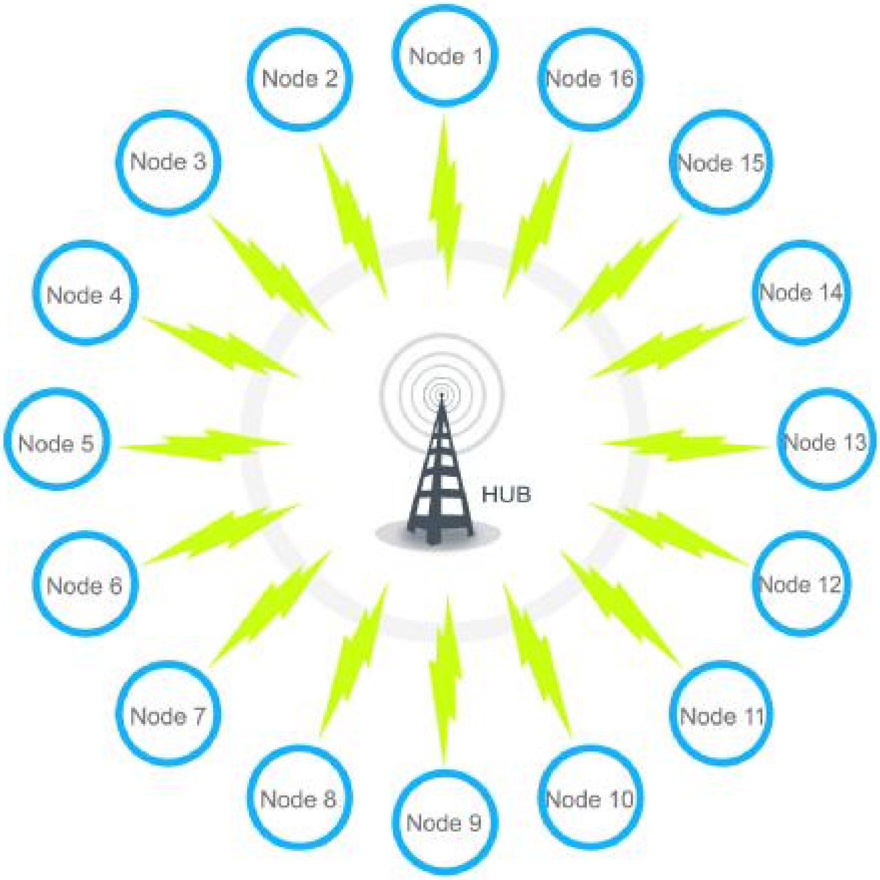
FPGAs can be programmed to form a great variety of standard logic components and I/O ports that can be built into complex sub-systems. Larger programs can even embed whole processors exactly emulating commonly-used DSPs or CPUs into a part of the chip while other parts function differently. It is thus possible, with the addition of a few extra components such as RAM and some standard computer bus interface chips, to build a complete system using one programmable device.

A chip can upload partial algorithms while in operation, allowing continuous reconfiguration while operating. This is particularly useful in the context of a user controlled system, where tasks change during a session. It is even possible to take a system down and completely reprogram it to change from, say, a large scale audio mixer into a video colour correction device, in a matter of seconds.

3.2 Hub and Nodes

Each processor chip is programmed to form a hub-node structure, wherein a single hub connects a number of nodes. The hub and each of the nodes are individual IP cores, programmed in hardware to perform a specific task within a media system.

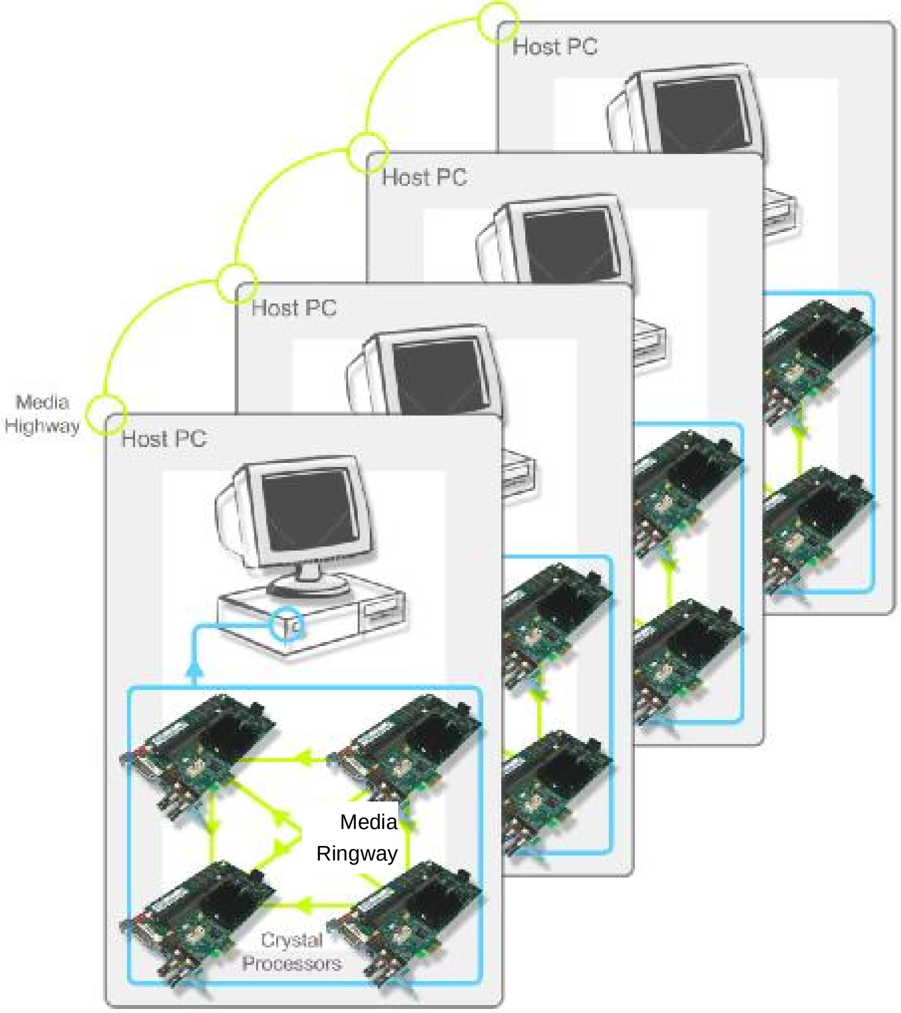
A hub is a signal routing core, effectively a broadcast source transmitting data to all receptor nodes and receiving data back from them. A typical large audio mixing system may use hubs broadcasting 4,000 audio signals of 36-bit depth at 48000 samples per second, or its data equivalent in other media or at other sample rates. Larger hubs may be programmed as needed.



Signal traffic between Hub and Nodes

3.3 Scalability (Audio Applications)

The Crystal Core architecture has been implemented on a PCI Express card (CC-1), which is connected to a suitable host computer. The smallest configuration capable of delivering a 230 channels into 72 mix busses is delivered from a single CC-1 card. Each channel in the system is equipped with eight bands of EQ and 3 stages of Dynamics. A medium system providing an increase in the capability (920 channels into 288 busses) is accomplished in a host computer with four CC-1 boards. Larger systems may be assembled by using multiple hosts with each additional CC-1 card delivering a linear increase in the channel capacity. There is no theoretical limit to the number of processors which can be linked.



Large system comprising multiple hosts connected by Media Highway

The configuration of four circuit boards per host is not a hard limit of the architecture. By changing the number of Media Highway connections on each circuit board, the number of circuit boards per host could be increased or decreased. The configuration being discussed here is seen as a good balance between the resource requirements of connectivity and processing, and one that will suit the range of applications currently being targeted.

The fundamental operations of a media engine are processing and connectivity. Processing means mathematical and logical operations. Processing is carried out in mathematical sub-units called “nodes” which are formed within the chip by loading processing algorithms into it. Once a program is loaded, that part of the chip is temporarily dedicated to its processing function, which may be a series of operations designed to achieve any required media processing outcome.

Connectivity means getting the right signals to the right processing node. For example, in an audio system with 200 source channels, there may be more than 1000 actual channels being processed and routed to different destinations, since each channel must be handled before and after each of its multiple stages of processing. This routing requirement consumes a considerable amount of computer resources, so the processing and connectivity requirements must be balanced against each other when designing a media engine.

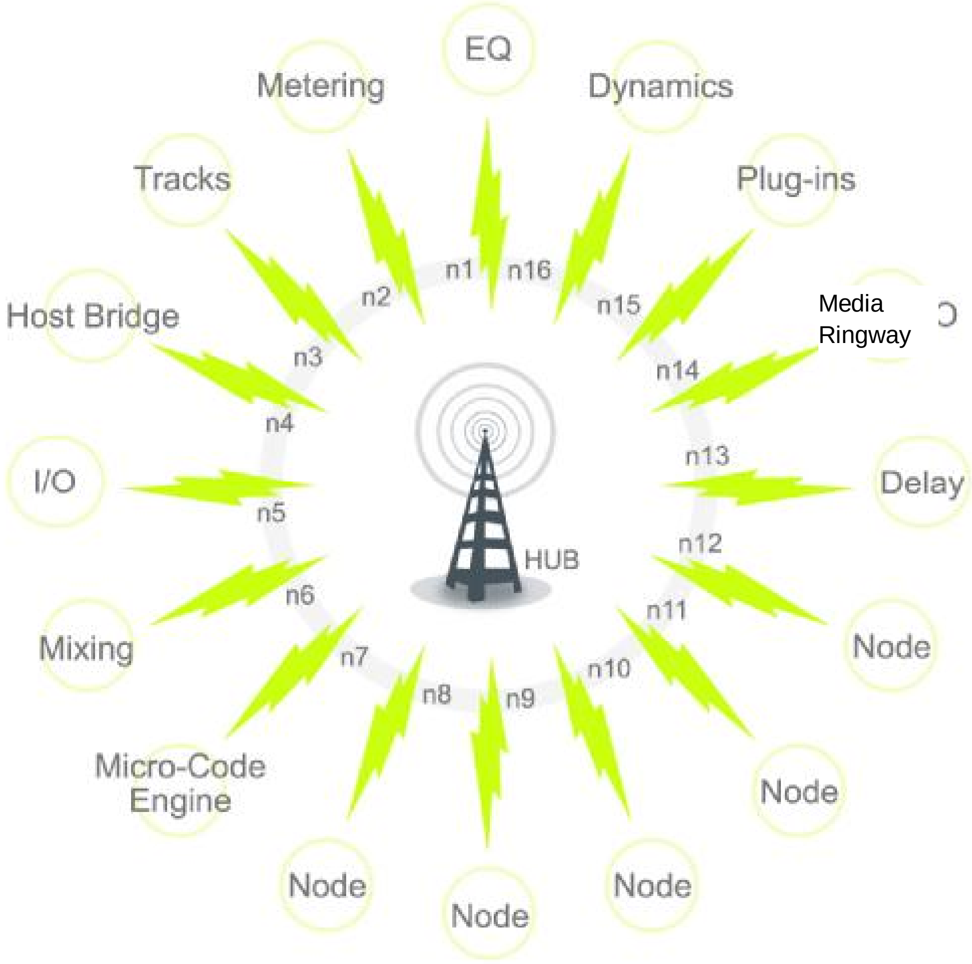
The Crystal Media Engine challenges other systems using acceleration hardware because it can allocate both processing and routing resources flexibly to different processing nodes as needed, allowing all its power to be distributed effectively. Each processing node is individually programmed for its specialized function, and the “size” of the node, or its cost in hardware resources is completely flexible. A simple node performing input and output functions may use a fraction of the resources of a complex node like a multi-channel equalizer, and the system will allow precisely the right amount of resources to be allocated to each task. This means that resource usage is optimized, so the maximum possible number of signals can be brought to their appropriate processors and mathematically transformed. By contrast, the architecture of previous systems is locked into standard configurations, where resources are hard-wired to signals that may not need them. This causes waste of resources and thereby increases system cost and reduces flexibility.

Previous systems have achieved expansion through a linear connection bus which allocates a specific number of channels to each discrete processor. The speed of the bus also limited the total system size. In general these systems were hampered by having limited processing power per block, which dictates and constrains the algorithms that can be run, and a fixed architecture, which commits resources to processes even when they are not needed. The Crystal architecture connects in three dimensions, allowing almost limitless expansion. It can connect multiple circuit boards within a single computer, or multiple computers, fusing them into a single system of immense power. In addition, its reprogrammable allocation of channel connections to processing nodes ensures the flexibility to make full use of the processing power.

Unlike the technology it replaces, which is based on interconnected circuit boards using dedicated Digital Signal Processors (DSPs), the Crystal Media Engine will be a complete architecture on a single chip, with the option of seamlessly linking other chips to the first when extra capacity is required.

3.4 Example – Audio Mixing System

The following diagram shows the application of the Hub/Node architecture to an audio editing/mixing system such as Fairlight's Anthem.



Audio Mixer Node Map

The mixing system above is implemented using the Hub-Node architecture. As explained earlier, the hub is a signal routing IP core, effectively a broadcast source with intelligence and scalable bandwidth. Its function in this system is to pass audio samples between the processing Nodes. Each node is in turn connected to an IP core responsible for media processing or I/O.

Each node is an IP core, programmable on the fly to perform any floating or fixed point computational or I/O task required by the user's current application. The nodes used in the mixing application are: Mixing, Metering, I./O, Linking, ASIO Interface, Track Interface, Delay Compensation, Dynamics, Equalization (EQ), Miscellaneous 1 (Oscillator and Monitoring) and Miscellaneous 2 (Plug-ins/3rd Party).

Each node has access to the hub's full signal bandwidth, in this case 4,000+ channels, which are delivered on the hub's Broadcast Bus. All nodes have this access simultaneously, but are provisioned as part of their programming with the intelligence to receive the exact subset of channels needed to perform the node's current task. The ability to precisely control the bandwidth to each node liberates power within the chip for other needs, thereby making efficient use of its capacity. This aspect will be discussed later in section 4. Dynamic Resolution Optimization.

3.5 CC-1 IP Core Implemtation

As previously discussed, CC-1 is a Crystal processor board containing a single FPGA loaded with an aggregation of IP cores.

The table below shows Fairlight's current portfolio of IP cores and the number that can be hosted on a single CC-1 card. Note that CC-1 is capable of all the processing capabilities listed below, simultaneously and continuously.

System latency is 2 sample periods, for each hub to node transaction, and all output channels are in phase.

IP Core #	Name	Description	CC-1 Module Specification	Comments & DRO
1	Broadcast Hub	n x n broadcast router	2000 x 2000 channels 36-bit FP x 48 kHz	
2	EQ node	N channel N-bit equalizer	230 channels, 8 bands each	Uses 72-bit floating point data format. Hi and low pass filter switchable between 12 and 24dB per octave.
3	Dynamics node	N channel dynamics	230 channels, 3 stages each	Variable knee compressor, Limiter and Expander/Gate
4	Mixing Node	N channel N Buss Mixer	230 channels, 72 busses	
5	MADI node	N channel MADI Xmit/Recv	336 channel MADI I/O	Equivalent to 7 standard MADI ports
6	SD-SDI Node	N streams of SDI Xmit/Recv	1 SD-SDI input and 1 output.	
7	AES3/SPDIF Node	N channel AES Xmit/Recv	2 stereo input, 4 stereo outputs.	Used in the SX-20
8	Float/Int	Full conversion		

	conversion module	to/from FP		
9	Tracks Node	Provides N playback and M record tracks	192 playing tracks; 64 record tracks. 1x Video track.	
10	Transport Control System	Position and time sync	44K1 to 192K.	Including LTC/MTC and Video support.
11	Monitoring	N x M monitoring node	32 in, 8 out monitoring system.	The monitoring matrix is also connected to a secondary switching router providing connection to up to 128 speakers. (16 x7.1 speaker sets)
12	Metering	N channel Signal monitoring node	255 channels of hi-res metering.	Includes formats such as VU, DIN-PEAK etc.
13	Audio Bridge node.	N + M real-time host audio bridge.	96 channels from host; 64 channels to host.	Initial release will support ReWire and VST plugins.
14	Delay Node.	N channels of delay.	256 channels	
15	Micro Code Node	SIMD co-processor	Sine/Noise Osc, Cue Sends, Graphic EQs	
16	Mega MAC	Multi-port memory access controller	5 port, 2.8GB/Sec, 1GB DDR memory space.	
17	Host/Client DMA controller	Bi-directional, multi tiered.	190 MB/Sec host transfer bandwidth.	
18	Reserved for third party 1		TBD	
19	Reserved for third party 2		TBD	
20	Reserved for third party 3		TBD	
21				

4. Dynamic Resolution Optimization

The World's first combination fixed/floating point professional media platform -- with variable precision.

There are two established signal processing paradigms that are used in professional digital media systems: fixed point and floating point. Whether they are using fixed or floating point, current systems also have a fixed precision from end to end. For example, they are either 32 bit fixed point **OR** 32 bit floating point. Each paradigm has advantages as discussed further below.

A unique feature in the design of the CC-1 architecture allows Fairlight to implement **both** paradigms in the Crystal Core.

Fairlight's revolutionary Dynamic Resolution Optimization (DRO) architecture enables the optimal precision needed for a specific task to be used within each of its Nodes. This uncompromising design means ultra-precise 72-bit fixed point can be used in CC-1's EQ Node, while optimal 36-bit floating point can be used in the Mixing node. In areas where extreme precision is not required, CC-1 adjusts the precision accordingly. For example, audio metering is more than adequately specified at 16-bit fixed point. DRO is unique, and is patented by Fairlight worldwide.

Dynamic Resolution Optimization (DRO) provides unsurpassed quality by allowing the best processing for the task at hand. This not only improves quality, but exponentially increases efficiency, providing greater performance at a lower cost.

Traditional systems must maintain the highest precision required from end to end, using either fixed point or floating point paradigms (not both). These aging and inflexible architectures are made to look like costly compromises, and dinosaurs with inherent disadvantages for some tasks.

DRO supports both sides of the resolution debate as referenced below by delivering a solution and without the need for any compromise.

"The double precision 48-bit processing is used when long time constants are required. This occurs when low frequency filters are on the job and when compressors, expanders and limiters are used with their relatively slow attack and release times. If 24 bits are all that are available when more precision is required, the results are a problem. The function misbehaves and the least damaging result is poor sound quality. The worst result is amplifier or loudspeaker damage due to a misbehaving DSP crossover, making double precision a must-have for superior audio."

(Greg Duckett and Terry Pennington of Rane Audio, "Superior Audio Requires Floating Point", published on the Rane Audio website http://www.rane.com/note153.html)

Another example:

"The first observation is that digital filtering when we allow the user to select high-Q, very low-frequency filters is difficult at the best of times. Even 64-bit floating point can produce significant error energy if the best filter forms are not used. Even for floating point, it is important to use forms that have normalized state variables so that imbalances in the state values do not lead to further degradation of the precision of the result. Clearly, the performance of 32-bit floating point and 24-bit integer will be considerably inferior to that of 64-bit floating point, so we might conclude that *it is not possible to achieve high-quality results for these extreme filter settings*. Furthermore it is shown that sweeping the settings of a filter with time excites some aberrant behavior when the state variables are not normalized, even with 64-bit floating-point arithmetic. 48-bit integer is proposed as a compromise between economic realizability and ultimate precision. The increased headroom and guard bits allowed by the format provide enough precision to allow some extreme filter settings and still preserve a 24-bit result after several stages of processing."

(Andy Moorer - pioneering digital audio engineer and currently head of computer science at Adobe. 48-BIT INTEGER PROCESSING BEATS 32-BIT FLOATING POINT FOR PROFESSIONAL AUDIO APPLICATIONS, available at http://www.jamminpower.com/PDF/48-bit%20Audio.htm)

DRO serves as yet another example of the disruptive nature of Crystal Core technology. DRO's dramatically improved performance, greater efficiency and lower cost signals the end of the legacy media processing era.

For the planned implementation of the DRO architecture refer to the comments inserted on the table attached to section 3.5.

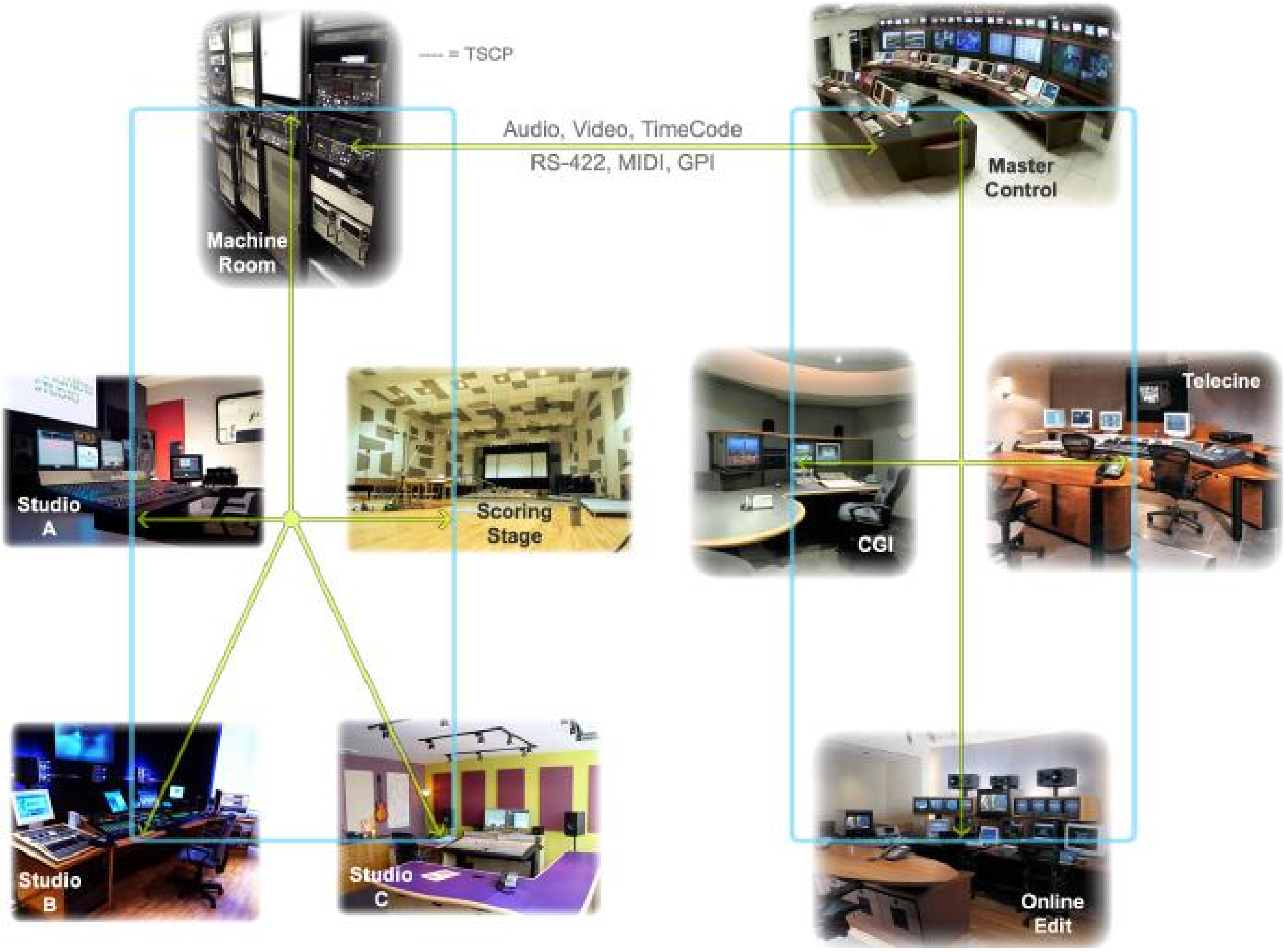
5. Total Studio Connectivity Protocol (TSCP)

The basis for TSCP (Total Studio Connectivity Protocol) a new Fairlight protocol called Media Highway, which can transmit multiple data streams of different formats, sample rates and bit depths simultaneously, with clock transmission and recovery from and to any point on a network. The Media Highway protocol connects a variety of systems and assets facility-wide using existing cabling infrastructure.

The extension of Media Highway is TSCP (Total Studio Connectivity Protocol) which redefines the concept of a media system by allowing connected processing units to act in parallel or independently, and to communicate as much or as little as required. So, for example, a single connection can carry the control signals and video content of a machine-room device into any studio, while another connection joins two computers in different rooms to handle a particularly large mixing session.

The Total Studio Connectivity Protocol is a means of connecting different physical locations, providing data in many formats simultaneously, and providing a timing system that allows them all to work together on the same (or different) material. A typical application is a video-audio post production house, in which different rooms are used for different tasks: recording, editing, mixing, switching, colour correction etc.

At any one time, several rooms may be working on different aspects of the same material – for example, a recording studio may be used for an orchestral scoring session on a film, while at the same time a person in a voice booth may be adding commentary. Both recording rooms will be controlled from one point, and all the audio channels, plus video camera feeds to compensate for lack of direct sightlines, plus headphone mixes and video program material will be exchanged between these three sites. The source of video playback may be in a central machine room, and must be controlled using RS-422 from the main control room, as well as providing its video and audio channels to all three rooms.



Post production house during film scoring and voice-over session

The TSCP system is a network of Media Highway connections, plus a control system encompassing all of them. In the diagram above the Media Highway connections are shown as heavy black arrows, with the type of data carried by each connection listed nearby.

5.1 TSCP Physical Connection

Media Highway is a unidirectional point to point connection which may use optical cable, coaxial cable, standard Ethernet (CAT6), or SATA links. The type of cable used determines the bandwidth of individual links, and a system may contain a mixture of cable types designed to provide exactly the required bandwidth at the most affordable cost.

Each TSCP site is a Crystal sub-system, consisting of at least a host computer with a Crystal processor circuit board. A site may be as small as one I/O box with a minimal Crystal subsystem, and the capacity for just two MH connections, or as large as a control room site containing a fully-blown mixer or video switcher, with as many Media Highway connections as needed. The MH cables are connected to ports on the Crystal sub-system, and from there via nodes to the Hub of the system (see Connection Between Media Highways.)

5.2 Media Highway Frame Structure

Data on Media Highway is carried in frames, which have the structure shown below.

<table>
<tr><td rowspan="2">Timing</td><td>Video1 (SDI-HD)</td><td>Audio (48KHz)</td><td>RS-422</td><td>GPI</td><td>MIDI</td></tr>
<tr><td>Video2 (SDI-HD)</td><td>Audio (44.1KHz)</td><td colspan="2">Timecode</td><td>KVM</td></tr>
<tr><td>Header</td><td colspan="5">Payload</td></tr>
</table>

Media Highway frame showing example data types

The frame is divided into two sections, header and payload. The header contains synchronization information as described below.

The payload contains a number of data sections, each of which can carry any type of signal required. Typical signals used in media include multiple video channels of various formats, multiple audio channels of various formats, timecode, RS-422 control signals, MIDI (Musical Instrument Digital Interface), GPIs (contact closure signals) and KVM (keyboard-video-mouse data from computers.

5.3 Units

A unit is a collection of data signals, which can be switched into different Media Highways as a single entity. For example, a Betacam video player uses a number of signals: one SDI video, 4 audio and RS-422 for control. All these signals, with their corresponding ports on an I/O box, can be controlled and switched as one entity by the TSCP software controller.

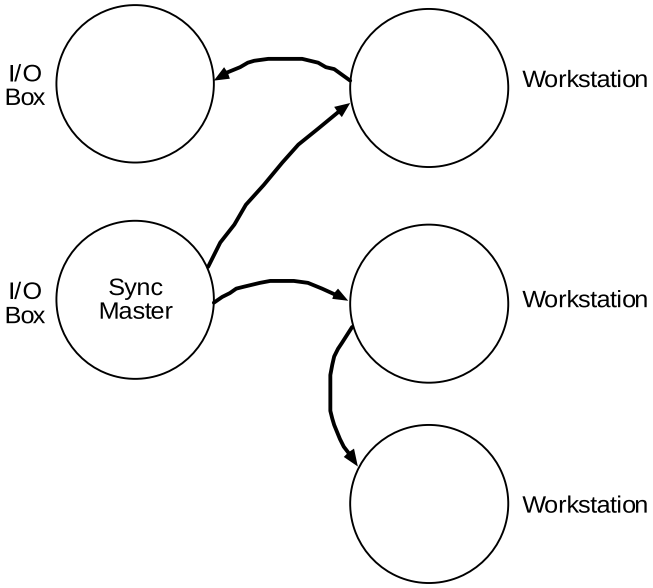
5.4 TSCP Clock Synchronization

The timing system can be divided into two areas, data rates and system clocks. The data rate is the frequency at which data frames are transmitted in the system, and this rate may be different from the sample rate of any or all transmitted signal types. The data rate does not even need to be particularly regular, but it does need to be at least as fast as the highest sampling frequency of any transmitted data.

Data from constant-frequency formats, such as video, audio and timecode, are transmitted at the data rate frequency, although in general the latter may be different from the native frequency of the signal. For example, the master clock frequency may be 48000 Hz, while one of the data formats could be audio at 44100 Hz. This audio data will be written into a buffer at its destination at the master clock frequency, and read from the buffer at its native frequency of 41000. A delay of one sample will ensure that the sample data has changed before each read from the buffer, but for this to be true, the local 44100 Hz clock must be derived from the system master clock This is accomplished by first recovering the master clock, then converting it to all the clock frequencies needed at the destination.

The system clock must be very regular, and must be recoverable at any site on the network. That is to say, a low-jitter clock of the same frequency and at a fixed phase offset from the master clock must be able to be generated at each site. The source of the system clock can be injected at any site, and may be the output of an SPG, a video or audio machine or any other source convenient to the facility. Its frequency is not important, but generally it is chosen to be a multiple of all data sample frequencies, so that clocks at each of their frequencies can be generated by simple division of the system clock. Typical frequencies would be in the range 25 to 30 MHz.

5.5 TSCP Clock Propagation



TSCP Sync Connection

In a TSCP system, a number of sites are connected by Media Highway. One of these sites is the Sync Master, whose System Clock is generated locally, usually referenced to a high-quality House Clock. Its timing signal is carried to other sites, and can be transmitted from them to further sites. Each additional site in the chain is called a “tier” from a synchronization point of view. The system can support any number of tiers, with the proviso that sync signal quality may be compromised as clock frequencies decrease and the number of tiers increases.

6. New Products

CC-1 powered products are being introduced at the AES show to be held in San Francisco October 6th 2006. The first products will be the DREAM 2 product family. These products form a range of audio production tools for recording, editing and mixing applications. All initial product outcomes will be delivered from a single CC-1 card which will connect with any of the four available surface technologies (Satellite, Station, Constellation, Anthem). The end user's physical I/O requirement is supported with either of the SX-48 or SX-20 modular I/O products.

Anthem 240 channel large format mixing powered by a single CC-1 card.

DREAM II October 2006	Channels	Max Physical I/Os	Mix Busses	EQ	DYN	3rd Party Plug-ins	Max Disk Tracks		Integrated on board Video	I/O Types supported
							Play	Rec		
CC-1-96	96	116	32	8 Bands Every Channel	Three Stages Every Channel	Up to 64 channels	96	64	HD/SD Genlocked at any standard frame rate SDI I/O	Analogue, AES/EBU, MADI, SPDIF, DXD.
CC-1-144	144	164	56				144	64		
CC-1-230	230	212	72.				192	64		

CC-1 Card fitted with MADI breakout card CMI-1 provides support for up to 212 physical I/Os. The I/Os are connected via the 8 BNCs and DVI connector to up-to four SX-48s and one SX-20.

6.1 I/O options

Each CC-1 card can support up-to 7 x Multiplexed data connections. These may be utilized in the form of 64 channel BNC MADI connections or as SDI video data streams. If MADI is utilized this equates to 448 physical inputs and outputs per CC-1 card. These MADI ports can be connected to Fairlight's SX-48 modular remote I/O or any other 3rd Party MADI equipped I/O platform.

The first products released will provide the capability to connect up to 4 x SX-48s delivering up to 192 physical inputs and out puts. In addition the CC-1 provides support for a single SX-20 system which, in addition to providing analogue and digital I/Os, also provides machine control and sync capabilities for the system.

SX-48 Signal Exchange 48 Channels

Fairlight's SX-48 Signal Exchange unit provides flexible I/O management of both analogue and digital audio inputs and outputs and a packing density of up to 48 channels in a single 2U frame. Each standard SX-48 frame ships with a MADI I/O card to provide a multiplexed bi-directional signal flow for each 48 channel unit. Digital signals can be referenced to external AES, Word Clock or Video at any standard frame rate, including tri-level references when used in a Hi Definition production environment. If AES inputs are present, any AES input can be used as a reference for the system. The front panel LED's provide a clear indication as to the reference currently in use.

TSCP

SX-48 introduces Fairlight's "Total Studio Connection Protocol" (TSCP). When enabled through a simple PC management system via an Ethernet connection, up to 254 SX-48 systems can be managed from one central location, effectively allowing routing and patching of a combination of over Twelve Thousand analogueue and digital audio channels. SX-48 makes redundant Audio I/O a thing of the past.

Mixed I/O formats

Central to the SX-48 design is the ability to flexibly configure units with any combination of up to six audio I/O cards plus one MADI I/O card in each 2U frame. These cards can provide 8 x channels of AES EBU I/O with Sample rate conversion, 8 channels of high quality analogue inputs and 8 x channels of Analogueue outputs. Optional relay protected analogue outputs are also available if these channels are required to connect to analogue speakers.

SX-20 Compact remote I/O

Supporting the fundamental I/O requirements of smaller DAWs with mixed output monitoring capabilities, the SX20 provides a comprehensive and cost effective solution. Combining the basic requirements of analogue Mic and line inputs, Analogue outputs, Digital I/Os and Machine control, the SX20 delivers the essentials for most Post Pro applications.

The system is delivered with the following features-

- 12 Analogue outputs, 2 line level inputs.
- 2 Mic/Instrument Inputs with 48V
- 4 SPDIF inputs, 8 SPDIF outputs
- Sample rates up to 192 Khz
- 9 Pin master and slave
- Video sync input
- LTC

7. Third Party Opportunities

Crystal Core has been engineered from the outset to allow the architectural and creative elements required to incorporate third party business opportunities within its product outcome spectrum. A significant amount of CC-1 bandwidth is dedicated to powering plug-ins when used in Fairlight products, and the board is also available for OEM products, as well as those developed in partnership between Fairlight and other manufacturers.

A number of technical partnering models are envisaged, as outlined below.

7.1 Industry Standard Hardware Interfaces

Within the CC-1 environment, Fairlight has created hardware interfaces in a number of standard formats, as IP cores in the fabric of the chip. Current and planned examples include MADI, SMPTE 292 (SDI), Ethernet and AES-3.

Serial interfaces can be further enhanced by providing on-board software control layers such as CobraNet (audio over Ethernet) and M-LAN.

Third parties wishing make use of the immense mathematical power of CC-1 can pump data into and out of it easily using any industry standard hardware interface.

7.2 Industry Standard “Virtual” Interfaces

FairlightAU has created IP cores implementing many of the software standards by which devices exchange data within the PC environment. These include ReWire and ASIO, the latter interfacing to a standard VST environment running in the FPGA. This can self-instantiate VST plug-ins from hard disk, organize channel I/O in ASIO format, and run a huge number of calculation-intensive programs.

Using any these interfaces opens the way for two-way traffic between processes in the PC environment, with third-party products either using CC-1 as an off-board accelerator, or providing host-based processes which CC-1-based products can access.

7.3 Third Party Products using CC-1 Engines

FairlightAU supplies CC-1 engines as a drop-in OEM replacement for audio processing engines in third-party products, or as an acceleration platform for host-based systems. The intention is for third-party products to preserve their unique character and features, but improve performance and/or cost efficiency using the new engine for basic number-crunching.

Interfacing to CC-1 is achieved easily using a Software Development Kit (SDK) that provides a real-time command interface to the engine, worked examples of code, and extensive technical support from Fairlight engineering. Details can be obtained from FairlightAU.

For manufacturers of products not needing the full power of CC-1, a sliding scale of license fees can be applied to ensure cost effectiveness at any size of system. Licenses can be upgraded in the field via software download if further performance is required.

7.4 Third Party IP Cores

Third party manufacturers can develop code to run on Fairlight’s CC-1 processor board in a number of ways, and at a number of levels.

7.4.1 Micro-Code Engine

An exciting avenue for plug-in development is FairlightAU's microcode engine. This is a ready-made calculation engine consisting of adders, multipliers and registers that can be programmed at a high level to implement key signal processing algorithms, without the need to first "fit" them into DSP architecture. FairlightAU provides the interface to route audio streams to the engine inputs and outputs, as well as a high-level programming environment used to implement algorithms at a schematic level, with worked examples and technical support.

7.4.2 Processor Emulation

The development environment for plug-ins is enhanced by the emulation libraries available for FPGAs, which enable insertion of specific hardware device emulators, including popular DSP chips, into the gate array fabric, ready to run existing applications.

In this case FairlightAU provides an interface whereby audio streams are routed to the plug-in inputs and outputs, as well as a mechanism for instantiating the third-party DSP algorithms.

7.4.3 VHDL Development

Manufacturers wanting direct access to the FPGA at gate level can gain significant leverage by cobbling their processes with existing Fairlight IP cores, which can handle much of the housekeeping functions associated with signal processing. Hardware constructs at VHDL level can be instantiated (correct term?) into an FPGA that is also running applications created with the micro-code engine, by emulation, or by using Fairlight's IP cores.

7.5 Security

The FPGA provides its own hardware security, preventing plug-ins being copied illegally from one system to another, thereby avoiding the cracks which have plagued all sectors of the software industry.

8. Background information Why FPGA?

Programmable logic devices have been available for many years. Their uses at first were to replace fixed logic devices – their programmability was a godsend for hardware engineers, who could afford to experiment with circuit design without needing a new board layout for each iteration.

As silicon manufacturing improved, parts became faster and more complex, and in situ programming was developed. This meant that hardware could be updated in the field without opening the box, and it paved the way for dynamic repurposing of hardware resources. Many started to foresee FPGAs becoming the premier devices for application acceleration, relieving host CPUs of repetitive, math-intensive tasks. In the field of signal processing DSPs and microprocessors had shared this territory for some decades, but rapid FPGA development saw them move forward as a viable alternative. FPGA has a much faster real time performance when compares to DSP IC. FPGA can serves as either a standard alone DSP system or coexists as a co-processor to offload the heavy duty processing with a traditional DSP chip.

Dr. Nick Tredennick is Editor of the Gilder Technology Report. The title of his document Death of the DSP[1], written in August 2000, exemplifies the industry's shifting viewpoint towards this emergent technology. He wrote: "High growth in the leading-edge wedge is the foothold for dynamic logic designs. From there they will proliferate to the rest of the embedded systems segments. It is the beginning of the long downhill for the microprocessor and for the DSP." His predictions were highly prescient, and today we see FPGAs proliferating rapidly in telecoms, portable devices, video processing, home appliances, and many other fields.

Cray Inc used FPGAs in their XD-1 super-computer for application acceleration, in preference to other processors. The XD-1 brochure[2]: "Six Xilinx Virtex-4 Field Programmable Gate Arrays (FPGAS) per chassis attach to the RapidArray fabric for massively parallel execution of critical algorithm components, promising orders of magnitude performance improvement for target applications." And in the same brochure: "Well suited to functions such as all reduce operations, searching, sorting and signal processing, the application acceleration subsystem acts as a coprocessor to the AMD Opteron processors, handling the computationally intensive and highly repetitive algorithms that can be significantly accelerated through parallel execution."

The main reasons to choose FPGAs for hardware acceleration today:

- Flexibility
- Integration
- Parallel processing
- Low cost / high performance
- Fast growth path and short development cycle

8.1 Flexibility

The FPGA is like a raw piece of marble, out of which any shape can be chiselled. The engineer can place blocks of math processing, memory registers and communication paths wherever they are most effective, with no imposed structural restrictions. This allows clever designers to utilise all of the chip's power to deliver functionality to the system.

Nick Tredennick wrote[3]: "Unlike the microprocessor or DSP, which implement fixed resources and rely on dynamic algorithms, dynamic logic implements *dynamic algorithms and dynamic resources*"

Fairlight's Dynamic Range Optimization (see later section) is a great example – data word lengths are chosen for each particular context, using neither more nor less processing power than needed. The result yields the best possible quality, and also the greatest number of channels.

Nick Tredennick again[4]: "Because the microprocessor and its cousin the DSP are general-purpose devices, they cover a broad range of applications. The cost of covering a broad range of applications is that the microprocessor or the DSP is not likely to be a perfect solution for any of them." The FPGA can, with skillful engineering, become the perfect solution. At the same time it is a general solution, because its function can be changed in an instant by loading a new program. Literally millions of logic elements can be reprogrammed in a few 10s of nanoseconds. It is the "perfect general solution" for application acceleration.

From Ray Andraka, Andraka Consulting Group[5], a hardware design consultant: “Like microprocessors, many FPGAs can be infinitely reprogrammed in-circuit in only a fraction of a second. Design revisions, even for a fielded product, can be implemented quickly and painlessly. Hardware can also be reduced by taking advantage of reconfiguration.”

From Cray XD-1 brochure[6]: “A well-balanced system matches processing power with memory, interprocessor and I/O bandwidth, ensuring that applications are free to communicate results without experiencing delays.” Because FPGA resources can be freely allocated to required hardware functions, a well-balanced system can be crafted matching each required process, and loaded into the chip fabric just when needed.

Nick Tredennick again[7]: “DSPs generally work on data in multiples of a byte. Dynamic logic implementations can work on any data width (the width can even vary with time to suit the needs of the problem.)” Fairlight have made use of this attribute of FPGAs to implement Dynamic Resolution Optimization (see section below).

8.2 Integration

FPGAs can be programmed to include most of the standard parts in a computer system, including memory, processing engines, communication pathways, IO voltage level controls and switching logic. By keeping many functions on the chip instead of farming them off to peripheral hardware, and by placing them exactly where they are needed in the logic flow, the engineer achieves a smaller, faster system at a lower cost, and generates less heat.

From Ray Andraka[8] “The programmable logic in an FPGA can absorb much of the interface and ‘glue’ logic associated with microprocessors. The tighter integration can make a product smaller, lighter, cheaper and lower power.”

The sheer size of modern FPGAs, measured in numbers of gates or in instructions per second, is now much larger than the largest DSP chips, and this is significant in improving the speed of processing for large systems. Today’s large audio systems use multiple acceleration boards, each containing multiple DSPs. By contrast, Fairlight’s new CC-1 audio engine uses a single FPGA to achieve the same performance as its Constellation system, based on 64 Analogue Devices SHARC DSP chips.

From Allan Cantle, president of Nallatech, an FPGA development company[9]: “In an ideal world, multiple processors would achieve linear performance increase with no additional overhead, and no performance would be lost due to inter-device communication. Unfortunately, the reality is somewhat different. This is because there is overhead introduced by the need for physical communication, and additional cycles are lost trying to coordinate the application partitioned among multiple processors. Although it’s the job of the programmers to work that out when implementing the application, the reality is that no matter how good a job they do, the result will most likely be less than optimal. Due to the physical communications overhead and difficulty of getting multiple devices to work well together, we don’t see a linear performance increase. The bad news is that the more processors you add, the worse the problem gets.

“The “multi-device communications challenge” outlined previously changes significantly when a number of microprocessors or DSPs can be replaced by a single FPGA. This immediately reduces communications overhead. This improvement also applies to multiple-device FPGA computing systems, where the total number of devices is comparatively lower. The communications overhead in a system with a number of FPGAs is considerably less than that of a system composed of DSPs or microprocessors performing at the same level. As a consequence, the likelihood of data bottlenecks is reduced considerably. This also provides an increase in flexibility for the developer, offering fewer restrictions on the specifics of implementation and an easier development path. (Reduce Development Time by Replacing DSPs with FPGAs.doc – Allan Cantle - Nallatech)

Nuvation Current (Online magazine)[10]: “DSP device-based designs have traditionally included FPGAs on board for glue logic and processor peripherals. Now, primary DSP functionality can also be handled by the FPGA, including the glue logic, in one device. The results are reduced size, system complexity, and cost.”

8.3 Parallel Processing

FPGAs can be programmed to perform multiple operations in parallel, unlike DSPs and microprocessors, which are strictly serial instruction machines. Parallelism increases operational speed, and correspondingly reduces latency.

From EE Times May 2006, by Yankin Tanurhan, Vlad Dinkevich, Manish Dharod, and Shafi Syed, Actel Corp[11]

“Parallel computing vs. Turing machines

“There is a big performance difference between the various DSP platforms based on how the platform performs computations. Both general-purpose and specialized DSP processors belong to the class of Turing machines, which perform instructions one at a time... The FPGA and ASIC are 'deprived' of this limitation. In fact, there are few flexibility and performance limitations a modern FPGA puts on a system developer. The FPGA can run parallel processing (i.e. execute multiple instructions at a time); implement Turing machine(s) as needed, including instantiation of soft microprocessor cores; and carry virtually any practical combination of parallel processors and Turing machines on the same silicon. The parallel processing dramatically improves performance of common DSP functions, such as FIR filter, FFT and correlator.”

Nuvation[12]: “Even high-end DSP and microprocessors offer limited parallelism, memory flexibility, and interface capability. The integration of dozens of processors in a real-time design is non-trivial and provides serious complications including design complexity, form factor, power consumption, development time and cost. At the same time, standards of video quality are increasing, and the development of improved video compression algorithms is progressing. The use of FPGAs encompasses the high-end DSP architecture into one flexible device that can be updated seamlessly.”

“To get an idea, the increase in speed of an average DSP algorithm running on a low-cost FPGA vs. a low-cost DSP can be roughly 1 order of magnitude, or up to 10 times faster in the FPGA.”

“Because FPGAs are essentially custom programmable hardware, the designer is able to trade off area against speed to meet requirements. For example, an FPGA may only need to run at 25MHz to perform four parallel computations for a signal processing function, whereas the same function in a DSP may require clock speeds of up to 1 GHz.”

Massively parallel performance means that hundreds of audio channels can be processed simultaneously with very low latency, at optimum signal quality. Low latency is extremely important in professional audio systems, and host-based systems without acceleration may exhibit related problems, as explained in the following Sound on Sound review by Simon Price[13]:

"The other consideration, latency, can refer to two different limitations within a digital audio workstation — I'll call them throughput delay and controller delay. The first of these comes about because any signal processing imposed on audio in a digital system, whether it is gain changes, signal mixing or effects, is a mathematical function which takes time to process. … In a host-based workstation, the main computer processor has to take care of all the calculations, so the software has to impose a delay on the whole system to give the processor enough time to handle everything….. When you have signals coming into a host-based system from the outside world and being routed back out with the rest of the mix, however, this delay can become significant. Given enough latency, it's difficult for musicians to play in time if they're hearing themselves with a delay… The other issue — controller latency — is the responsiveness and resolution of the system to parameter changes (like fader movements) in the software. Again this is potentially a difficulty for host-based systems where the processor is busy and is expected to handle lots of automation as well. The worst-case scenario would be 'zipper noise' as the processor struggles to keep up with parameter changes smoothly."

8.4 Cost / Performance

In signal processing applications FPGAs offer significant cost improvements over traditional DSP processors, with a number of test-cases quoting improvements from ten times to hundred+ times:

From Altera literature[14]: "Berkeley Design Technology Inc (BDTI) is the leading provider of independent DSP benchmarks and publishes periodic analysis, FPGAs for DSP, comparing the FPGA performance vs. common DSP processors. The latest benchmark based on an orthogonal frequency division multiplexing (OFDM) system shows that Altera's first generation Stratix FPGAs provide over 95% cost reduction per channel compared to other DSP processor. (See Table 2).

"Table 2. BDTI Benchmark Results on OFDM System Comparing Stratix FPGAs & Other DSP Processors.

	DSP A	**DSP B**	**Altera Stratix EP1S20-6**	**Altera Stratix EP1S80-6**
Channels	<0.2	~0.7	~20	~60
Cost (1 ku) (*1*)	~$15	~$210	$120	$600
Cost/channel	~$100	~$300	~$6	~$10

End quote"

Nallatech[15]: "The potential technical performance benefits of FPGAs have long been recognized. As FPGA technology is developed and utilization increases, the commercial benefits of FPGAs are becoming clear. These benefits stem from the clear technical advantage that FPGAs can offer, but it is the commercial benefits that will ultimately justify and drive the continued proliferation of FPGA computing within high-performance real-time and embedded systems."

FPGA performance allows very large systems to be constructed with reasonable levels of engineering complexity.

Andraka[16]: “Properly executed FPGA designs typically outperform a DSP microprocessor by a factor of 100:1 , and by more than 1000:1 in special circumstances”

From FPGA Offload Through a Tightly Coupled Computing Architecture, by Douglas O’Flaherty (AMD), Larry Spector (Celoxica) and Prasanna Sundararajan (Xilinx)[17]:

Results quoted from testing FPGA hardware co-processor vs Opteron or Pentium 4 – 3 GHz.

Howe and Inverse Howe Processing	370 x faster
AES Encryption and decryption	13 x faster
Smith-Waterman ssearch34 from FASTA	64 x faster
Multi-dimensional hypercube search	113 x faster
Mersenne twister random number generator	3 x faster

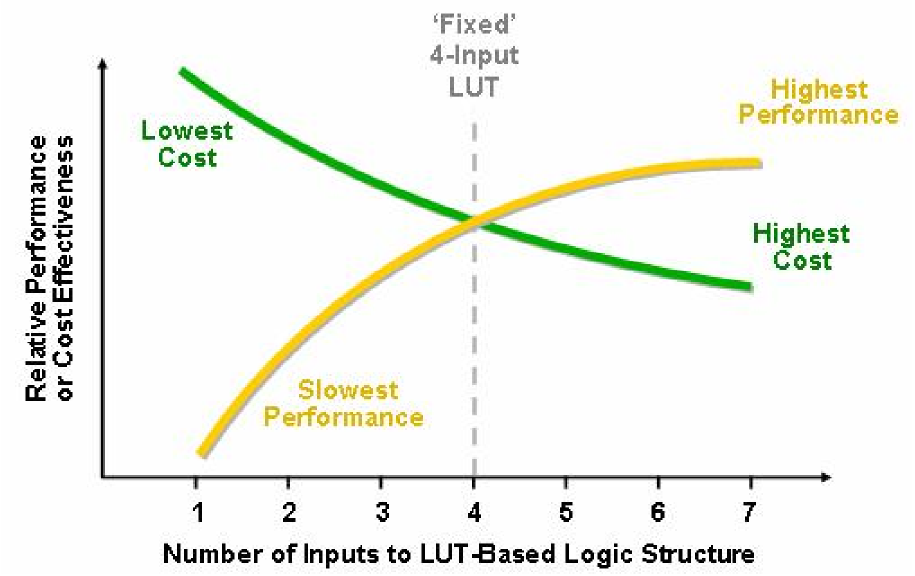
A secondary saving, and related increase in reliability, is achieved thru the drastic reduction in individual hardware components that are required to implement a given function.

8.5 Fast Growth Path

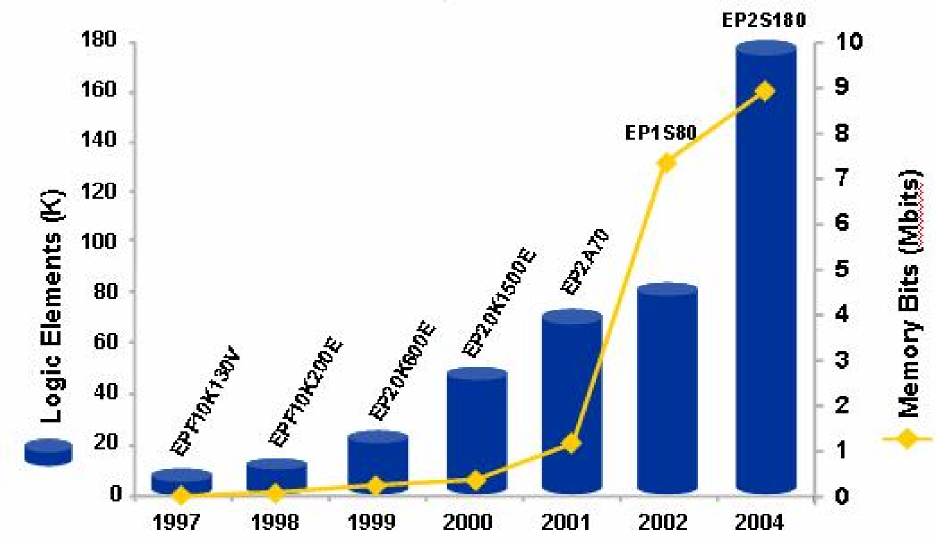
The combination of flexibility and significant processing power makes FPGAs suitable for a large number of applications, which leads to huge production volumes. The resulting increased affordability will entice more application developers to switch from DSP to FPGA-based designs, further increasing production, reducing cost, and promoting competition amongst device manufacturers. This is the ideal climate for growth in the technology. By contrast, both DSPs and microprocessors have seen a slowing in development speeds, while saturation and replacement by FPGAs slow volumes in the markets they traditionally serve.

Craig Petrie, Nallatech[18]: “Over the last decade, the performance capabilities of FPGAs have increased exponentially. Leading vendors such as Altera and Xilinx and have improved the functionality of their reconfigurable devices through the inclusion of memory, processors, multi-gigabit transceivers, and multipliers to the basic FPGA architecture. The result is a flexible, high performance processing device able to perform low latency, parallel processing tasks with low power consumption.”

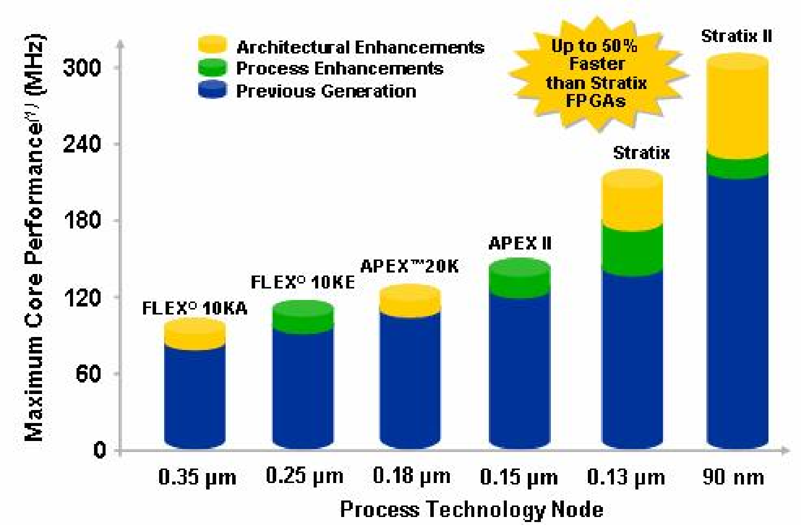
From Allan Cantle, president of Nallatech[19]: “It is becoming clear to users of DSP and microprocessor technologies that while gate counts have increased in line with Moore’s Law, the actual device capability has not. Chip vendors have struggled to successfully translate more gates into greater performance. This contrasts with field programmable gate array (FPGA) technology where device performance is inherently linked to device size and interconnect fabric bandwidth. FPGA vendors have been able to fully capitalize on the continual increase in chip densities predicted by Moore’s Law, resulting in more capable products.”



High performance with lower cost



More logic and memories



Speed performance vs technology

The above charts from Altera shows development of their FPGA product lines. Over the last five years, it shows increases in capacity and speed of 250%, with price reduction of 70%. Power dissipation has also reduced by around 90%, leading to reduced system design costs.

8.6 Programming Environment

FPGA development is still relatively immature, meaning that high-level tools are not as evolved as they are in the DSP and microprocessor fields. Programming talent is clustered at this time around PC products (using mostly C++) and to a lesser extent embedded applications using microprocessors and DSPs. Expertise in FPGA programming is scarce by comparison. New software development tools are starting to emerge that allows designers to use the popular C programming language as a design entry rather than Verilog or VHDL.

Fairlight's challenge has been to build a team with excellent understanding of FPGA techniques and potential, and at the same time leverage its deep knowledge of digital audio requirements and technologies. Having worked with FPGAs for many years, Fairlight was well-placed to see the potential of this technology, and set about building its R&D team into a centre of excellence in this field. The first result, a large fully-fledged audio system using a PC and a single FPGA, is a handsome payoff for the technical investment, but only the beginning of what will emerge in the coming years.

9. References

[1] Nick Tredennick: Death of the DSP, July 2000. Full text can be found at: http://www.rfengineer.cc/dspdeath.htm

[2] Cray Inc. Brochure for XD-1 Supercomputer, Release 1.3. Information on this system available from http://www.cray.com

[3] Nick Tredennick: Death of the DSP

[4] Nick Tredennick: Death of the DSP

[5] Ray Andraka: Digital Signal Processing with FPGAs.htm http://www.andraka.com/

[6] Cray Inc. Brochure for XD-1 Supercomputer, Release 1.3. Information on this system available from http://www.cray.com

[7] Nick Tredennick: Death of the DSP

[8] Ray Andraka: Digital Signal Processing with FPGAs.htm http://www.andraka.com/

[9] Allan Cantle: Reduce Development Time by Replacing DSPs with FPGAs.doc http://www.nallatech.com

[10] Nuvation Current (newsletter), Summer 2004 http://www.nuvation.com

[11] From EE Times 4 May 2006 – Strategies for designing DSP Applications for FPGAs by Yankin Tanurhan, Vlad Dinkevich, Manish Dharod, and Shafi Syed, Actel Corp http://eetimes.com

[12] Nuvation Current (newsletter), Summer 2004 http://www.nuvation.com

[13] TOOLS OF THE TRADE - Digidesign's Pro Tools Explained: Part 1 by Simon Price, published in Sound on Sound Feb 2001. http://www.soundonsound.com

[14] Altera White Paper: FPGAs for High-Performance DSP Applications, May 2005 http://www.altera.com

[15] Allan Cantle: Reduce Development Time by Replacing DSPs with FPGAs.doc http://www.nallatech.com

[16] Ray Andraka: Digital Signal Processing with FPGAs.htm http://www.andraka.com/

[17] FPGA Offload Through a Tightly Coupled Computing Architecture, by Douglas O'Flaherty (AMD), Larry Spector (Celoxica) and Prasanna Sundararajan (Xilinx) From a conference paper, April 2006

[18] Craig Petrie, Nallatech: High Performance Embedded Computing using Field Programmable Gate Arrays http://www.nallatech.com

[19] Allan Cantle: Reduce Development Time by Replacing DSPs with FPGAs.doc http://www.nallatech.com