A Digital Signal Multiprocessor and its Musical Application

Jean-Baptiste Barrière
31 rue Saint Merri, 75004 Paris, FRANCE

Adrian Freed
CNMAT
Dept. of Music, University of California at Berkeley
1750 Arch Street, Berkeley, CA 94709

Pierre-François Baisée
19 rue Portefoin, 75011 Paris, FRANCE

Marie-Dominique Beaudot
Polyotronics Inc.
1731 Francisco Street, Berkeley, CA 94703

Introduction

We are unifying different sound synthesis and processing techniques in an interactive real-time environment. The availability of increasingly general and cheaper DSP chips and the increased performance of desktop computers is bringing the dream of such an environment within reach. We are using the digital signal multiprocessor system described here to provide the signal processing for research in synthesis and processing algorithms and real-time control software. This research is for musical application, so its use by musicians throughout the development stage is essential.

The digital signal multiprocessor system has a peak performance of 168 million multiply/accumulate/delay operations per second. For audio processing and music synthesis applications, this is roughly 100 times the performance of a medium priced computer workstation and 10 times the speed of single chip DSP coprocessors [Lowe 1989]. We present, in bottom up fashion, the machine, its software, and its musical application in the piece Alton.

The Machine

Physically, the machine is a stack of 8 processor boards interconnected with flat cable busses. Each board contains a Motorola DSP56001 [Motorola 1989], up to 4k x 24 bit words of fast dual-ported static RAM, a DB15 connector for serial I/O, and glue logic for an 8-bit bussed connection to a controlling host processor.

System design, circuit simulation, schematic capture, PCB layout, software design, prototyping, implementation and testing were done by two people on a single computing platform (Macintosh) with a modest budget in a few months. Eight identical double-sided printed circuit boards are used with no backplane. All the parts used are readily available (only the DSP chip is not second sourced). The system is physically small: 9" x 9" x 6", CMOS parts which require little power were used wherever possible.

One of the eight processor boards is customized to be a master "move engine" as shown below. The move engine is the only processor that has access to all the external memory of the remaining seven processors. It is used to route signals between the processors and to monitor and control the activities of each processor.
The move engine does not include a dual port RAM; we use the freed space to include a small "bulk" memory of 32k 24-bit words. There is no provision to add a large dynamic RAM array, as this adds considerably to the cost and complexity of the design [Waveframe 1989]. The most important signal processing task precluded by this decision is reverberation. Sampling "synthesis" and waveguide models [Smith 1985] requiring long delay lines may also be inappropriate for this architecture.

Each processor is booted and controlled over an 8-bit host bus. A standard 8-bit NuBus board is used for connection to Macintosh workstations. It is easy to completely saturate the host processor with mxi and control tasks.

The 56001's two serial ports are available externally through a protection and RFI network on a DB15 connector, using the same pinout as the NeXt computer's DSP port. Audio peripherals, such as A/D and D/A converters, R-DAI recorders and CD players can be connected with this port. It is also possible to use the same connector to attach the NeXt machine's 56001 to the multiprocessor.

One of the two serial ports (SCI) can be used for standard RS232 or RS422 connections or MIDI. The other one (SSI) is designed for high-speed connections to converters. Each port is capable of transmitting and receiving 5 channels of 24-bit digital audio at 48kHz. The total channel capacity of the multiprocessor is therefore 40 inputs and 40 outputs.

The Software

Signal processing code was developed using Motorola's assembler and simulator under Apple's MPW environment. A low-level driver handles communication between the host Macintosh and the digital signal processors. A HyperCard based user interface management system called HyperDSP is used to develop interactive test harnesses for DSP code modules. It provides access to the driver's facilities in a high level interpreted language, HyperTalk.

With its new editing facilities and real-time mxi control of the multiprocessor, MacMix [Freed 1987] has become the primary development environment for the machine. The figure above on the right is a graphic representation of a resonant model of a piano note.

We are extending the Max language [Puckette 1988] to control the multiprocessor. We hope as well that the NeXt music kit [Jaffe & Boyson 1989] can be adapted to the system.

Musical Application of the System to Aion

Jean-Baptiste Barrière's piece "Aion" is the first to make use of the system. It is scored for 2 percussion players, the multiprocessor, a DMP7 digital mixer, an AKAI S1000 sampler (for playing sampled attacks used as excursions, a TX216 (used for textures), and 2 Macintosh II computers (one to control the synthesis parameters, the other to control the whole system).

Central to the application of the multiprocessor to the piece is an analysis/resynthesis technique based on resonance models [Barrière et al 1985; Ponsard et al 1986]. This follows from earlier work on synthesis based on filtered filters to model acoustic resonances of physical systems [Wawrzynek et al 1984, Rodet et al 1984]. The basis of these methods is to extract filter parameters by analyzing recorded sounds and exciting a bank of these filters with synthesized, sampled or live excitations.
This resonance modelling technique was implemented with FORMES [Rodet et al 1984-1] on a Vax-
780 controlling an implementation of CHANT [Rodet et al 1984-2] using an FPS-100 array-processor
for synthesis. Later a real-time implementation of only the synthesis part was implemented on the 4X
digital synthesizer [Asta et al 80], and used by J.B.Barrière in "Epigénèse" [Baïsade et al 86].

Compositional materials for the piece include a database of resonance models of percussions and
samples of their attacks. Processes of the piece are built from these materials with Esquisseas [Baïsade et
al 86].

Attacks were sampled and classified according to their typology: struck, plucked, bowed, strummed,
breath, etc. These attacks are used as excitations for the models and permit unusual couplings: a breath
sound exciting a gong or a tam-tam, and a plucked sound exciting a voice or a wind instrument.

The figure below illustrates how the resonance models are implemented on the digital signal
multiprocessor:

Four of the DSP boards run identical code implementing a pair of two-pole digital resonator banks. The
outputs of each bank pair are crossfaded to create interpolated timbres. The move engine routes and
mixes to spatialize the outputs from the resonator banks into 4 channels of 16-bit D/A conversion. An
excitation engine synthesizes, routes and mixes excitation functions destined for the resonator banks.
Excitations include spectrally and temporally shaped noise, jittered pulse trains and externally generated
sounds from a 16-bit A/D converter. The remaining boards can be used for further excitations (additive
and sampled, for example) and for non-linear processing [Mclayure et al 84]. We are working on other
modules based on generalizing features of physical models of instruments, such as non-linear elements,
waveguides, linear filters, oscillators and FOPs [Rodet et al 84-2].

Alon

Alon is the second principle of time in Greek philosophy, dealing with the internal, psychological time,
as opposed to Chronos, chronological time. The piece is concerned mainly with the relation of complex
timbral structures with non-pitched tone.

Two percussionists play a variety of amplified percussion instruments, which are processed through the
resonance models of other percussive and non-percussive instruments. This is an example of real-time
cross synthesis. Both the trigger and the sound of the live percussions are processed. The trigger is
used to start sequences of events as well as timbral interpolations within a single event, the sound itself
being both amplified and mixed with its processed form and other complementary sources. The
percussionists are often acting as generators: their gestures are provoking processes that in turn either
compete with or compliment one another.

The musical situations evolve through interwoven forms of solo (either by each of the percussionists or
by the computer performing a single complex timbral interpolation), duo (one instrument being filtered
by the models of resonances of others), trio (the computer acting as a more normal soloist playing with
the others), quartet (when each instrument is filtered by the model of an other), and finally a whole ensemble (when the computer is given to play a full polyphony with or without the others). Note that the system described is used in two complementary ways - for processing and orchestration of the percussions sounds.

Conclusion
We are building a small number of these DSP multi processor systems for computer music centers, individual composers and researchers. Though the machine has already proven useful in musical situations, we see it as transitional - a way of learning how to build components of a truly interactive computer music workstation [Wawrzynek and von Eckes 89].

References