



**HAL**  
open science

# A Real-time Workstation for Physical Model of Multi-Sensorial and Gesturally Controlled Instrument

Jean-Loup Florens, Claude Cadoz, Annie Luciani

► **To cite this version:**

Jean-Loup Florens, Claude Cadoz, Annie Luciani. A Real-time Workstation for Physical Model of Multi-Sensorial and Gesturally Controlled Instrument. ICMC 1998 - International Computer Music Conference, Oct 1998, Ann Arbor, United States. pp.n/c. hal-00481948

**HAL Id: hal-00481948**

**<https://hal.science/hal-00481948>**

Submitted on 7 May 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## A REAL-TIME WORKSTATION FOR PHYSICAL MODEL OF MULTI-SENSORIAL AND GESTURALLY CONTROLLED INSTRUMENT.

Jean-Loup FLORENS, Claude CADOZ, Annie LUCIANI  
A.C.R.O.E.  
(Association pour la Création et la Recherche sur les Outils d' Expression)  
I.N.P.G. Institut IMAG  
(Informatique et Mathématiques Appliquées de Grenoble)  
46, Av. Félix Viallet 38031 Grenoble - France

### ABSTRACT.

Among different currents in musical sound synthesis the instrumental approach developed at ACROE with the CORDIS ANIMA project, needs specific computer and low level software architecture for the real-time interactive simulation. Since the very beginning of this project several real-time configuration have been designed in which the basic principles of physical modeling were implemented at a low software level. These researches works and the improvements of computers allow to design a specific CA station that integrates both high level modeling and compositional tools and real-time instrumental simulation.

**Keywords** : Physical model, Real-time, Sound synthesis, Sound and animated images, graphic interface for physical modeling.

### 1. INTRODUCTION.

In the field of physical model for musical sound synthesis, the CORDIS ANIMA system [1] is mainly oriented toward the real time synthesis where an instrumentalist can interact gesturally and acoustically with a simulated instrument. This instrument being designed either from a real or an imaginary instrument reference model.

The main researches works around the CORDIS-ANIMA project have evolved around the following points:

- 1) The physical model formalism definition and conceptualization : the CORDIS language.
- 2) Experimental and theoretical works (modelisation) on physical models.
- 3 ) Computer architecture and software for interactive real-time synthesis (TELLURIS project [2])
- 4) Force feedback transducers for the gestural interaction.

5) User interfaces that include modelisation, compositional and analysis tools (GENESIS project[3]) .

These works have led to the realization of a specialized work-station that provides the main functionality's of the CORDIS musical synthesis approach. This station, although consisting today in specialized hardware is mainly built around a standard general purpose computer. That makes its diffusion and future evolution possible with a limited cost.

We will present the main functionality's and technical choices in the design of this station in relation with the general synthesis context. In a first part we will briefly examine which links and difference exist between the different synthesis process mainly "signal model" process and "physical model", and their respective real time implementation functionality's and constraints. In a second section we will focus on the physical model interactive synthesis principle and present the evolution of the tools developed at ACROE. In the 3rd part we will describe the main component of the today's station : the hardware, the real-time simulation engine and the user interface tools.

### 2. SOUND SYNTHESIS AND REAL-TIME.

Real-time synthesis is today a permanent concern for musician and researchers in the computer music domain. However, the sound synthesis in computer music current has for a large part been founded on non real-time processes whose potential unlimited complexity and generality were ensuing from the ability to operate in non time constrained ways onto recorded signals. This relation between musician and sound that tended toward the direct sound composition without any instrumentalist nor

instrument and constituted a fundamental break from traditional music originates in the "Musique concrete" and "Electroacoustic" (in 1948 and 1950), currents that have preceded the computer music. Since this epoch, many digital signal synthesis techniques have been brought up, whose common features are the use of a signal based formalism ; that means that somewhere the user (composer) has to think in term of basic or sophisticated signal concepts. These are usually called "signal model" methods in opposition to physical model ones [4]. The most known are additive synthesis, soustractive synthesis, non linear distortion etc.... Real-time implementations of such processes appeared when computing power allowed it, first on heavy architectures inside laboratories (4a,4x IRCAM 75-80 ), then on electronic instruments and specialized workstation using advanced LSI technologies (MARS IRIS station [5] , SIM IRCAM station in the 90s [6]). There were no fundamental revolution in the basic sound synthesis principles that would have been a consequence of the real-time computation possibilities. However, the ability to introduce a real time input control in the process has led to define clearly the status of such inputs and to distinguish them from the "structural data" that constitute the initial definition of the process before real-time session. The formalisms founded on data flow like MIDI control signal and treatment boxes seems to be adequate (rather state automata or procedural specifications for example...) to this purpose. The MAX and FTS systems represents such an approach [7].

The Physical model synthesis approaches can be classified in two main categories, the first one that appeared at ACROE (1979) is mainly founded on energetical and structural properties. The second that has been developed in different computer music centers since 1985 is more founded on informational and functional properties [8],[9].

This last approaches have been growing inside data-flow signal modeling context. It leads the user to use physical signal like force and position, velocity or pressure. The treatment boxes associated to these flow represent different functional part of the physical model or more abstract operators. These approaches are similar to the automaticians one that takes in charge general dynamic systems as input/output boxes. Besides a state variable based formalism have been proposed [10]. However, the signal connexion when used to bring a physical variable signal (force speed, position)/from a physical model component to another always introduces an implicit source of energy as being a controlled generator.

In such an approach the physical model may appears only as a particular mean to synthesize sound, at the same level than other abstract dynamic systems or classical signal processing synthesis processes. One of the interest of these formalisms is that they can easily mix both physical and signal models.

The ACROE physical model approach that was introduced at the end of the 70s with the CORDIS formalism and concepts [11] mainly distinguishes two different macro components of the sound generating process, the instrument one hand, basically as a passive and stable physical system, and on the other hand, the gesture that may supply the activation energy toward the instrument and consequently behaves as an active system closely linked to the previous.

Besides, one of the main goals of the approach was to establish in the context of computer synthesis the conditions of the instrumental playing that were existing in traditional music and had disappeared or have been deeply modified with direct composition on the recorded sound and various signal approaches. Afterwards this instrumental approach was generalized to the concepts of instrumental communication , that extend beyond the computer music domain [13].

In this physical model formalism whose component are provided with such energetical properties the basic means to interconnect the components must preserve these properties and consequently leads to energy conservative links : if component A is linked to component B the energy flow outgoing from A at each time through the connection inputs into B at the same time. This last *energy flow* conservation principle can be opposed to the *data flow* conservation of previous signal models.

Furthermore this interaction linking principle defined and implemented inside the modeled objects world may be extended to connect the modeled object to a real instrumentalist gesture, since both can be seen as mechanical systems. That gave the basic principle to establish the instrumental relation with a physical model synthesis process. This concept of instrumental interaction with the sound synthesis can be seen as closely tied up to the physical model synthesis. These mutual dependencies can be schematically expressed as: 1) Introducing gesture control into any synthesis process requires at least a piece of physical modelisation in order to specify the mechanical reaction of the manipulated device or gestural force feedback generation process. 2) In the physical model world, whereas simplest an most pertinent object are passive, the simplest mean (conceptually) to energize them is the natural gesture or real physical system. These last consideration shows that in contrast with the signal model approaches

that historically issues from recording techniques, the basic and more natural mode for physical model is the interactive real-time.

### 3. THE CORDIS ANIMA TOOLS EVOLUTION.

The ACROE research works were mainly concerned by the design of a computer based tool for the musical creation using physical model synthesis and including instrumental interaction with gestural force feedback interface (FFGT) [2]. The main components of such a tool are:

1) The simulator that executes synthesis computation as PM simulation processes and whose hardware part is made of a computer provided with specific peripheral for the instrumental interaction : the FFGT, the audio and visual output.

2) The model designer that implements the modelisation formalism and allow the user to design the objects to be simulated. The compositional [14] and analysis software tools [15].

In the present section we describe more precisely the basic principle of the approach and the specific problems they induce concerning the design of such a tool.

#### Cordis-Anima principles.

The CORDIS objects are designed as physical system whose internal evolution law can be computed in an explicit and deterministic way. The ability of these objects to be composed with each others is obtained by providing them of the connection points that are dual physical signal input-output pairs. Two kind of connection points are possible : the M point that is a force input/position output and the dual L point is a position input/force output. The basic operation in the object composition consists in linking an L point to a M point. This operation represents the elementary connection that is *energy conservative*. The main difference with other PM approaches is that, at the modelisation level, the user is not concerned by the force/position signals manipulation but only by structural operations as defined.

The minimal elementary components from which general connection network and consequently general physical model can be built are the <MAT> element (provided with 1 M point) and the <LIA> element (2 L points) also called "atoms".

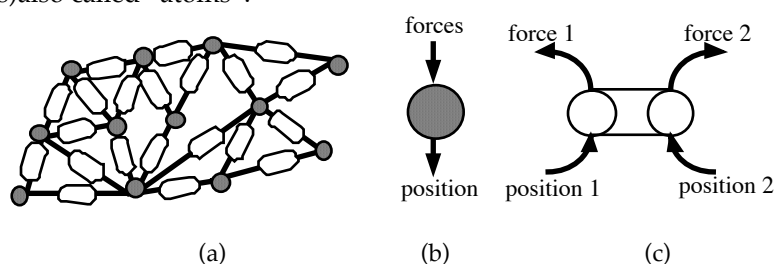


figure 1 - A CORDIS network made of connected <MAT> and <LIA> components (a) - The <MAT> and <LIA> elements and their underlying data-flow (b)(c).

One of the typical <MAT> atom is the free mass which is characterized by the classical dynamic law. The usual <LIA> elements produce two opposite forces according to the action/reaction principle. The <LIA> elements are used in modeling various physical interaction laws. One of the basic <LIA> is the visco-elastic element (or damped spring).

Special components that are also provided with L and M points are used to represent the links with the gestural and audio output devices. One of these gestural components that is used in conjunction with the 16 DDL modular keyboard [16] supplies 16 M points.

#### Basic algorithmic structures.

The implementation of the basic connection link may be realized by the means of two opposite data flows one force flow and one position flow. The whole constitute a modular synchronous data-flow system in which each component computation process communicates with the others through these bi-directional force/position pairs.

The external links are realized by the means of special interface devices the FFGT (Force feedback transducer) that are provided with mechanical sensors and with controlled actuators. The outgoing and inputting data flow of an external interaction link constitute the input control signal of the actuator (typically a force controller actuator) and the output signal of the sensor (typically a position sensor).

July 16, 1998

The CA formalism induces some simplifying properties concerning its data-flow implementation, compared to a general data-flow system : 1) - It is synchronous and flow conservative. That means that each component computation consists in a regular repetitive process that produces one sample data on its output as it consumes one sample data on its input. The inter module data connection are such that there is neither any sample loss nor duplication. 2) - Because of the mechanical interaction representation principle, all the data flow connections are grouped in two dual force/ position data links. 3) - The components of a model whose all connection points belong to one category (L or M) cannot be connected each other. It is true in particular for all the <MAT> and all the <LIA> components. 4) - As a consequence the global computation step of a CORDIS model is divided in two main phases the L phase and the M phase. These phases contain respectively the one step processing of all the <LIA> (<MAT>) elements of the model. Since there are no data dependency inside each category, all the processing in a same phase may be done in an indifferent order. The algorithm generation process (interpretation or compilation of a model) is for that reason a trivial and quasi instantaneous operation whatever complex is the model.

#### **The evolution in the design of the physical model computer.**

The physical model synthesis requires a priori more power computation than signal synthesis. The iterative computation of the motion of every part of the instrument model, and the global state signal contains a priori much more data than the sound signal which is only its 1 or 2 d projection. In addition the gestural coupling implies high reactivity constraint. On the opposite the real-time PM algorithms are highly repetitive and present specific features.

For that reasons the first orientations in designing the simulation machine leded us to built a specific computer. The CTR [17] ("Cordis temps réel") was a 32 bits fixed point machine clocked at 13MHz and made with discrete TTL technology. The physical model program was in a declarative form using the CORDIS language thanks to the implementation of the basic CORDIS components algorithms as micro programmed routines. The CTR could simulate simple models as a 30 points discrete string at 25.6 kHz. Another specific configuration was designed around a floating point 38 bits 12 Mflops processor FPS AP120. This machine that was basically conceived for linear treatments such filtering or FFT revealed a remarkable adequacy for real-time physical modeling thanks to the following features : 1) - The existence of 3 physically distinct and dedicated memories for program, parameter and data. This structure was well fitting to the 3 types of data that can be distinguished in a physical system according to their evolving quickness, that are structural data, parameters and state variable. 2) - The high-performance and programmable interconnection path system was allowing to configure, at each clock cycle, any coherent link between the operators registers or memories even in broadcast mode. 3) - The direct control of the data path from the program, without any intermediate treatment, was providing easy means to predict and control the execution duration of the algorithm. Considering its modest clock frequency this machine allowed to simulate in real-time interactive mode interesting instrument models [18] [19] such as plucked and bowed string, or maracas.

The improvement of general purpose computers and compilers during these last year and in particular with the recent multiprocessor architecture's equipped with RISC processors and powerful shared memory and I/O systems led to abandon more specialized processor like DSP. However these new architecture's as they are designed for general purpose, do not present the same determinism than synchronous signal processors and consequently the performance improvement in the case of the interactive physical model application is not in the ratio of their theoretical performances compared to the previous.

#### **4. THE REAL-TIME PM WORKSTATION.**

It consists in a multiprocessor SGI Power Challenge station equipped with a graphic card and specific hardware. On the opposite to other approaches [7] we could not turn toward a pure software design although it would be attractive for future. The main obstacle is today due to the gestural interaction constraints and reactivity that can only be assumed using a specific hardware configuration. However, the main software parts that are the simulation engine and the modelisation user interface also support non real-time mode. In these cases these can run on more standard hardware. The following table shows the general modes and the corresponding architecture required. The general hardware configuration is represented figure (2) . Besides the SGI computer the specific hardware consists in 3 parts : The gestural interface system, the audio interface system, and the hardware synchronizer. The graphic output function is assumed by a standard Silicon Graphic board (type "Extreme")

Mode	available R.T. channels	Hardware architecture required
Interactive real-time with gestural, audio and visual I/O	audio gestural visual	Complete real-time station
Interactive real-time without audio	gestural visual	Complete real-time station except special audio board
real-time with audio output	audio (standard SGI audio) visual	Standard SGI
	audio (special audio output) visual	Complete real-time station
real-time with only visual output	visual	Standard SGI or UNIX system with OpenGL
non real-time mode	none	Standard SGI or UNIX system

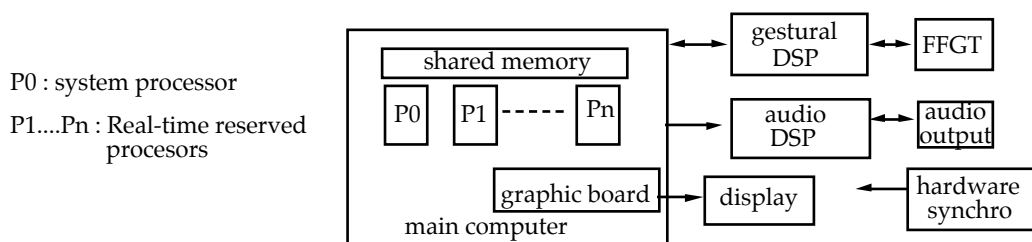


figure 2 - The global hardware configuration.

**The main computer.**

The SGI Power Challenge station, at least equipped with 2 R8000 RISC processors and a VME i/o bus, presents interesting features, allowing both high power computation and reactivity in particular thanks to its inter processor communication system (ref. SGI ) and modular and extensible hardware. The R8000 is a 64-bit super scalar microprocessor chip, which combines a large fast-access, high-throughput cache subsystem with high-performance floating-point capabilities. Its bandwidth satisfies applications with large working sets of data. The chip set delivers peak performance of 300 double-precision Megaflops and 300 MIPS at a clock frequency of 75 MHz.

In this case, the global system of cache is efficient in various types of computations, as real-time simulations which need a high number of floating point data. Indeed, the first level of cache of the floating point unit is 4 Mbytes wide, which is enough for our largest models. The address calculations are performed by the integer unit working on an 16 Mbytes one-cycle-access cache. The integer unit is considered as the address processing unit.

**The hardware synchronization system.**

The main synchronization systems provides different coherent clocks signals toward the samplers that synchronize the corresponding input and output flows. These clocks signal are generated from a main master clock according to a frequency map. This frequency map must be consistent with the flow dependencies that are induced by the algorithm. For example if for each gestural sample treated 40 audio samples are generated the gestural data and audio sampling frequencies ratio must be 40.

Another function of the hardware synchronizer is to control the initial delay between the different clocks starts when initiating a real time session. The mean data queues filling level (in the audio output system) and consequently the audio delay, directly depends on this initial clock start delay. Controlling precisely these delay is a necessary condition to ensure a determinist behaving of the simulator. The hardware synchronizer. is consequently in charge to initiate and stop the real time sessions. Typical frequencies are 44100 Hz for the audio output, 1050 Hz for the gestural channels and 50/72 Hz for the video output.

The hardware synchronizer is programmable and can provide a wide range of frequencies as well as different master clock sources.

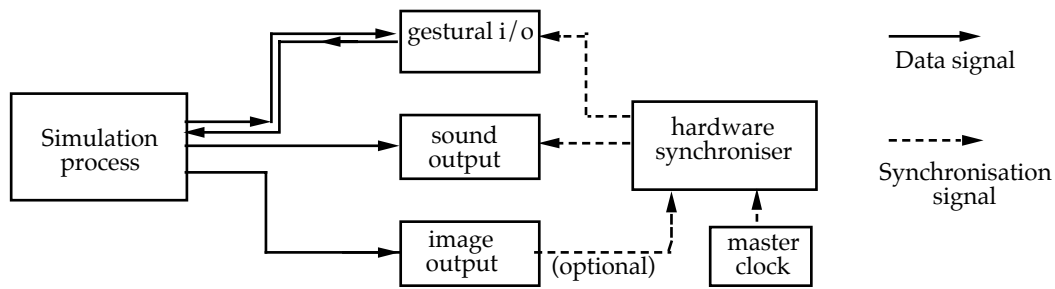


figure 3 - The hardware synchronization structure. The simulation processes are only synchronized by the data flow conservation mechanism.

**Gestural interface system.**

The communication hardware between FFGT and computer is a difficult problem. Because of reactivity data must be transmitted in regular and very high flow pulsed mode. Indeed the transfer phases duration must be the shorter to leave a maximum time for the model computation.

In the past analogic paths were used between the FFGT (the force-feedback. Keyboard) and the simulator where the conversion system was driven by two microprocessors (M68040). This system is still available on the present station but another interface has been designed and will work with digital paths for future FFGT. This last system may provide intermediate treatments that are made on a DSP (today one TI C40 ) board. These treatments consist in geometrical and dynamic transformations. The geometrical transformations allow to obtain well referenced gestural signal when using special mechanical accessories that transform the basic cinematic of the FFGT . Indeed, this concept of versatile morphology that still exist on the FF. keyboard, will be developed on future FFGT. The dynamic compensations treatments are used in conjunction with local over sampling (up to 8 kHz) and allow to enhance properties like stiffness and damping factor limits of the FFGT coupling.

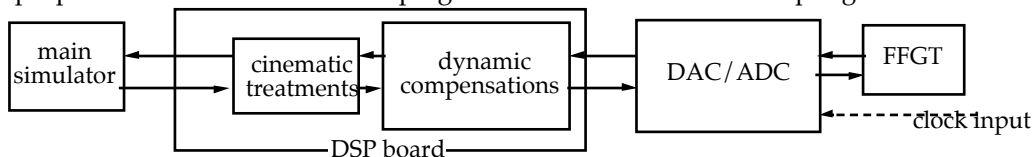


figure 4 - the functional parts of the gestural interface system

**The audio interface system.**

The audio i/o board is installed on the i/o VME bus of the SGI and provides a standard AES EBU audio output (optionally input) . The main function of this board is to run a sampling interpolator designed to adapt the sampling rates of outgoing or inputting audio flows to standard AES sampling rates which are usually 44.1 kHz or 48 kHz.

The computed audio flow and consequently the computing rates are then adjustable in a wide range from 5.5 kHz to 44,1 kHz or 6 kHz to 48 kHz that allows to adapt them to the bandwidth or time constants of the model and take the best advantage of available computing resources. The interpolator is a classical FIR SinC pondered by a Blakman window. The group delay is about 3.4 ms that must be increased with a "protocol delay" giving a total of 5ms. The effective delay is precisely controlled by the synchronization system.

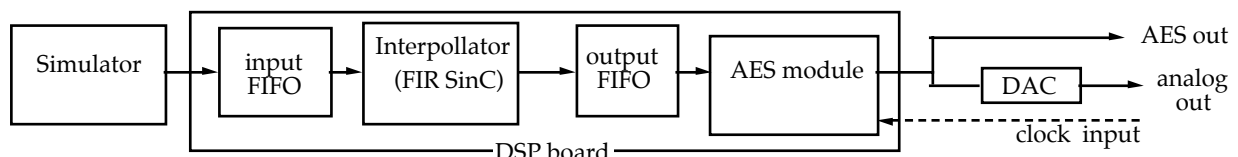


figure 5 - The functional parts of the audio interface system.

**The physical model simulation engine.**

We managed to implement the CORDIS-ANIMA algorithms efficiently by optimizing the code at its lowest accessible hardware level. Using the special real-time devices, one processor is dedicated to system management, whereas all of the others perform the computation without the possibility of being delayed by interrupts. Thus, in the multiprocessor environment, the high computation algorithms use

their cache memory as a local memory and with locking techniques, inter-processor communications are minimized, to avoid the occurrence of costly bottlenecks on the global bus.

The real time simulation engine mainly consists in an open *library of physical modules* and a *kernel* that assumes all real-time synchronization and communication functions.

Physical modules efficiency is obtain by using the regularity that exists in many physical models and that result in multiplicity at the algorithmic level. An optimization then consists in regrouping identical algorithmic structures in pipelined loops. This technique widely used on previous dedicated machines are still available on today's RISC processors and easier to implement thanks to new compilers efficiency. Another means to increase efficiency is to implement in a compact form macro components that are frequently used. The more basic is the mechanical cell (mass spring damper) ; usual ones are homogeneous vibrating structures like strings, plates [15].

The simulation kernel execution consists in different "threads" ( processes) that run on the real-time reserved processors. Each of these real-time threads consists in a repetitive execution of the standard <LIA> and <MAT> phases. The different real-time threads executions are implicitly synchronized by communication with Real-time devices (FFGT, audio output, eventually video output).

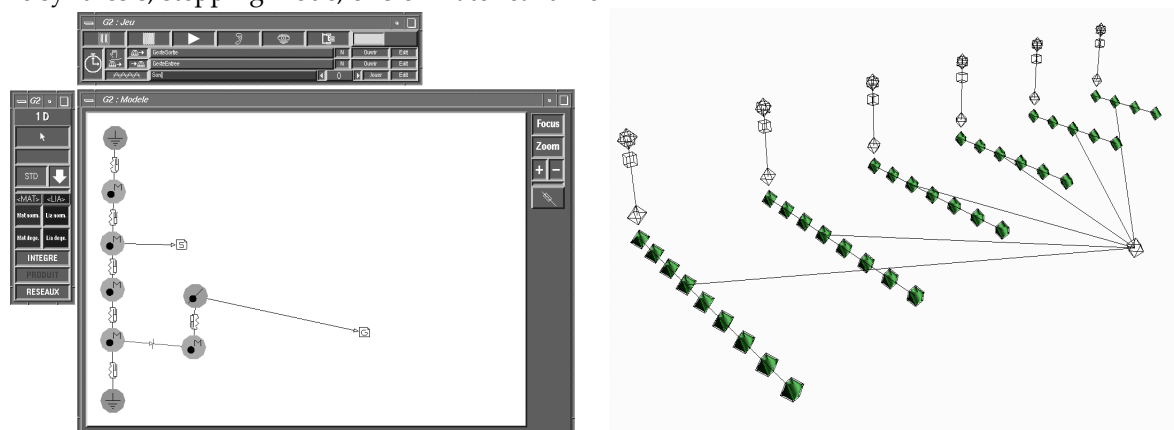
The ability to simulate at different rates several parts of a model is possible. In this goal the <LIA> and <MAT> phases are organized in several level of nested loops in each thread. Usually only 2 level are used : One correspond to the FFGT rate that is basically 1050 Hz. The other corresponds to the audio rate that is an integer multiple of the previous (usually 44.1 kHz or 22,05 kHz) . We notice that the inter-thread communication are possible at any rate level allowing to distribute ( if several real-time processors are available) in easy ways any instrument model.

The basic flow conservation algorithm that is implemented in the different special communication modules is based on a time stamping mechanism : Each independent real time thread or external process such those running on peripheral DSP possesses a local step counter whose value is used as a time stamp for emitted data samples. The receiving process compares the stamp value to its local counter and either goes on, or enters in a wait state or in an error state. The stamps and local counters are a modulo N representation of the temporal indexes that appears in the discrete time equations of the model. This mechanism allows easy debugging and also precise and determinist control of the initial state.

The usual acknowledge protocol may be avoided since for each flow of a physical connection an implicit acknowledge is provided by its symmetric feedback flow ; that shows that the physical connection presents quite the same communication cost than a simple unidirectional signal connection.

### The user interfaces.

The user interface is a new version of the GENESIS-I interface [3] that integrates several functions : 1) - The model designer provides a graphical representation and graphical tools to build the model according to the CORDIS-ANIMA modelisation principles. This model designer includes specific means for the manipulation of large structures and their parameters and especially the structure product [15] 2)- Analysis functions that allow to observe either in real time or at slow rate the motions of the model. Different simulation control modes are supported : Real-time interactive with a FFGT, off-line synthesis, stepping mode, or slow rate real time.



**figure 6** - The working board in GENESIS 2 model designer.(a) The Real Time control window showing an elementary polyphonic instrument that is simulated at 44.1 kHz / 1050 Hz with the FFGT.(b).



## 5. RESULTS AND EXAMPLES OF MODEL SYNTHESIS.

Compared to specific implementations that have been made on the same configuration the real time engine introduces few degradation. Compare to the previous CORDIS-ANIMA engine that was not optimized for real time it provides a speed improvement ratio in a range from 4 to 12 on the same computer. The basic models that can be used as benchmarks are pure vibrating structures such as string, plates or other more complex topologies that can be generated by the GENESIS model designer. A 1d connectivity regular structure (chaplet, ring) of about 350 masses and links can be simulated at 44.1 kHz.

The figure (6b) represents an example of polyphonic elementary percussion instrument that is made of up to 16 short strings of variable lengths . The main parts of this model are computed at 44.1kHz. The gestural i/o frequency and low rate part of the model run at 1050 or 2100 Hz.

## 6. CONCLUSION.

The real-time station for interactive physical model simulation appears as an interesting tool for musicians since it combines the previous GENESIS-1 /CORDIS modelisation building tools, to powerful instrument playing possibilities. In addition such a configuration constitutes a new interesting simulation tool for further researches on physical models.

The today's implementation on SGI power challenge can be adapted to other computers within two main orientations : 1)- High power multi-processors (as extension of the today's computer like onyx or origin) will provide means to improve both complexity and bandwidth of virtual instruments. The bandwidth improvement being important in particular concerning sustained oscillation models. 2)- A large diffusion on more familiar computer will be possible with the new low cost multi-processors architecture provided with a PCI i/o paths. In this same goal the specific hardware that mainly concerns the gestural interaction will have to evolve towards more simple and standardized configuration.

## 7. REFERENCES.

- [1] CADOZ (C), LUCIANI (A) & FLORENS (JL), 1990- "CORDIS-ANIMA : système de modélisation et de simulation d'objets physiques" - Colloque International sur les Modèles Physiques. Grenoble 19/20/21 Septembre 1990.
- [2] UHL (C) 1995 " Hardware Architecture of a Real-Time Simulator for the Cordis-Anima system : Physical Models, Images, Gestures, Sounds"- Computr Graphics International 1995.
- [2b] UHL(C) 1993- "Architecture de machine pour la Simulation d'objets physiques", ACROE internal report, 1993
- [3] CADOZ (C), FLORENS (JL),& LUCIANI (A), 1995- "Musical Sounds, Animated images with CORDIS-ANIMA and its multimodal control interfaces"- ICMC Banff 1995.- pp. 531-532.
- [4] RODET(X), DEPALLE(P),GARCIA(G).-1995 "New Possibilities in Sound Analysis and Synthesis" International Symposium of Musical Acoustics,1995,SFA Paris - pp.422-432.
- [5] CAVALIERE(S), DI GIUNO(P) - "MARS - The X20 device and the SM1000 board" - ICMC 92 pp348-351
- [6] LINDEMANN(E), STARKIER(M), DECHELLE(F) - "The IRCAM Musical Workstation : Hardware Overview andSignal Processing Features" - ICMC 90 - pp. 132-135.
- [7] DECHELLE(F), DE CECCO (M) - "The Ircam Real-Time Platform and applications". ICMC95 -pp.77-83
- [8] KEEFE(D),1992-"Physical Modeling of Wind Instruments", Computer Music Journal, MIT Press, Vol 16 No. 4, pp. 57-73, Winter 1992.
- [9] COOK (P) 1992 - "A meta-wind-instrument physical model", Proc. International Computer Music Conference, San Jose, pp. 273-276.
- [10] DEPALLE(P), TASSART(S), 1995- "State Space Sound Synthesis and a State Space Synthesiser Builder", International Computer Music Conference , 1995. pp 88-95.
- [11] CADOZ(C),FLORENS(JL) 1978- "Fondements d'une démarche de recherche en informatique musicale" - Revue d'Acoustique N°45.
- [12] CADOZ(C), FLORENS(JL),LUCIANI (A), "Responsive input devices and sound synthesis by simulation of instrumental mechanisms : the Cordis system", Computer Music Journal, MIT Press Vol. 8 N°3, 1984.
- [13] CADOZ(C), "Le geste, canal de communication Homme / Machine : La communication instrumentale", Special Issue of "Interfaces Homme/Machine", Hermès Ed, Vol. 13 N°1, 1994.
- [14] CADOZ(C), RAMSTEIN (C), 1990- "Capture, Representation and Composition of the Instrumental Gesture"- ICMC Glasgow 1990.pp 53-56.
- [15] INCERTI(E) "Synthesis and Analysis tools with Physical Modelling : An Environment for Musical Sounds Production" - ICMC 96.

July 16, 1998

- [16] CADOZ(C), LISOWSKI(L), FLORENS(JL) - "Modular Feedback Keyboard", Computer Music Journal, MIT Press Vol. 14 N°2, 1990.
- [17] DARS-BERBERYAN (T), 1982 - Etude et réalisation d'un calculateur spécialisé pour la synthèse sonore en temps réel par simulation de mécanismes instrumentaux -Thèse de Docteur Ingénieur - I.N.P.G. - Grenoble 1982.
- [18] FLORENS (JL), CADOZ(C) 1990- "Modèle et simulation en temps réel de corde frottée". 1er Congrès Français d'Acoustique. Editions de Physique Paris 1990. C2-1990 pp C2-873-876