Computer Music Department of CNUCE/CNR, Pisa

Proceedings of the

International Workshop on Man-Machine Interaction in Live Performance

Pisa, Italy

June 7-8, 1991

Scuola di Studi Superiori Universitari e di Perfezionamento S.Anna, Pisa

Program Committee
Claude Cadoz (ACROE, Grenoble)
Antonio Camurri (DIST, Genova)
Joel Chadabe (ICMS, New York)
Giovanni De Poli (CSC, Padova)
Giuseppe Di Giugno (IRIS, Frosinone)
Miller Puckette (IRCAM, Paris)
Leonello Tarabella (CNUCE/CNR, Pisa)
Stephen Travis Pope (CCRMA, Stanford)
Michel Waisvicz (STEIM, Amsterdam)

Organizing Committee
Alfonso Belfiore(CNUCE, Pisa)
Graziano Bertini (IEI/CNR, Pisa)
Giorgio Buttazzo (Arts Lab, Pisa)
Paolo Carosi (CNUCE, Pisa)
Leonello Tarabella (CNUCE, Pisa)

Audio equipment
BIA (Banca Intercomunale Audiovisivi), Pontedera (PI)

Staff
Annalisa Caioli
Lara Maria Costanza
Andrea Guerra
Caterina Guzzi
Michele Romboli

Translation and interpreting
Giovanni Giannini

Secretary and Hospitality
Maria Vittoria Masserotti

Papers and Concerts Coordination
Leonello Tarabella

Table of Contents

INTERNATIONAL WORKSHOP ON MAN-MACHINE INTERACTION

IN LIVE MUSICAL PERFORMANCE

PISA 8/6/1991

ENG. ANTONIO PELLECCHIA, CENTRO RICERCHE MUSICALI, ROMA

REAL TIME DSP SYSTEM FLY30

Work realized with the contribution of C.N.R.

In this workshop I speak about the FLY30 system, and in particular about the floating point computing algorithms and about the musician-system interaction.

An important field of the research we are developing in the "Centro Ricerche Musicali" is the test of signal analysis and synthesis algorithms based on floating point computing.

The precision we can obtain with a floating point data structure made by 24 bits exponent field and 8 bits mantissa field is about 1.E-39.

The maximum number achieved with this data structure is about 1.E+38.

The floating point becomes indispensable when we must realize digital signal processing algorithms that require high precision arithmetic.

Initially the traditional synthesis algorithms have been carried out, like oscillators, wave tables and so on, making use of the precision of the floating point data structure.

The floating point precision improves the behaviour of this algorithms and allows the realization of a new set of algorithms requiring considerable precision.

To realize these algorithms we use the floating point digital processor TMS320C30.

This processor is able to perform in 60 nanosec an addition and a multiplication at the same time on 32 bits floating point data structure, by means of a 40 bits precision Arithmetic Logic Unit.

The precision we can achieve with the 32 bits mantissa and 8 bits exponent ALU is on the 10th decimal digit.

Moreover the TMS320C30 instruction set consist of load and store conditioned instruction, integer-floating point conversion *instruction, mantissa and exponent load and store* istruction, three floating point operands arithmetic instruction and so on.

For this reason the numeric processor TMS320C30 is the ideal numeric computing instrument for our scientific and musical purpose.

We have realized, by means of the TMS320C30, digital processing algorithms requiring considerable precision and speed in performing multiplications and additions.

We have put particular care on digital filtering of numerical signals [Fig. 2].

A digital filter is a summation of terms obtained multiplying the previous input and output values by certain coefficients.

The variability field of these coefficients allows us to classify filters on their behaviour.

Therefore, we can classify filters on FIR filters, IIR filters, stable or unstable filters, resonant filters and so on.

Such differences of behaviour don't depend on the algorithm, but they depend only on values of filter's coefficients.

Particurarly, these coefficients determine univocally the transfer function of a filter, and it is always a ratio between polynomial of Z variable quantity.

If we study the position on the complex plane of the numerator polynomial's zeros and the position of the denominator polynomial's zeros called poles, we can easily find out the behaviour of a digital filter.

To know the results we must look at the distances from the zeros and the poles to the unitary radius circle having its centre in the origin of the complex plane.

These distances determine the stability and the behaviour of the digital filter.

We have developed a graphic software, that, once known the coefficients of whatever degree polynomial, it allows us to see all the polynomial's zeros on the complex plane [Fig. 3].

This software is highly effective to study the behaviour of digital filters and to study the filter sensitivity of individual coefficient changing.

As a matter of fact, if you change a coefficient you can see how the zeros on the complex plane move, going away from the unitary radius circle or approaching it.

Our object is to develop a digital computing algorithm that realizes easily digital filters that can be used on various purposes.

One of the possible application obtained from this kind of approach is to realize oscillators without reading value tables.

In fact if a filter works at the limit of its stability, it starts to oscillate producing a wave form that has a harmonic content depending only on filter's coefficients.

When a filter works in condition of limit of its stability, this means that its transfer function must have its poles exactly on unitary radius circle.

The poles positions on the unitary radius circle determine the filter oscillation frequencies.

When the poles are exactly on the unitary radius circle, there is a simmetry in the structure of the filter's coefficients.

The simmetry becomes more evident when the poles are more equally spaced on the unitary radius circle.

In the simplest way you obtain a perfectly harmonic oscillation: the filter oscillates at frequencies multiple of the fundamental frequency.

Through the dinamic modifying of the filter's coefficients, you are able to modulate the oscillation frequencies.

Moreover if a pole goes just a little bit away from the unitary radius circle, the corresponding oscillation goes out from its stability limit and it goes on either unstability or stability condition [Fig. 4].

This allows us to obtain an automatic amplitude envelop of the corresponding oscillation.

In fact a stable oscillation mode decays as a negative exponential, while a unstable oscillation mode grows as a positive exponential.

In this way we can obtain wave forms having harmonic components decaying with different speed.

The dynamic modifying of certain filter's coefficients allows us to modulate in frequency and in amplitude the oscillation modes.

An other application of digital filters is envelopes generation.

To realize envelopes, we use generally a straight segment approximation: we add an increment to the current value of the segment and we verify if we are arrived to the end of the segment.

To realize more natural envelopes, that are generally exponential and non-linear amplitude evolutions, we have to curve the envelope segment [Fig. 5].

We realize that by means of one multiplication and one addition, that is a simple first order digital filter.

In fact, the answer to the step function of a first order digital filter is an exponential function.

According to the filter's coefficient you can obtain a negative or positive exponential function or a straight line.

To realize an envelop curve, you must give the filter's coefficient besides the step value and the final value.

We have forced the constraint that, whatever is the envelop curvature, the time spent by the curve to reach the final value must be the same.

So we obtain a family of exponential functions passing through the starting value and through the final value according to the value of the filter's coefficient [Fig. 6].

Each envelop curve is generated in a simple and automatic way, neither reading values tables nor performing complex algorithms.

So we can obtain more smoothed or more sharpened envelopes, because we can control each curvature of every part of the envelop.

The real time control of the computing parameters of these kinds of algorithms is the more delicate part in all the system.

In fact, if you change just a little some parameters, you obtain considerable effects on the algorithm behaviour.

For example, let's consider a second order resonant filter that produces a constant amplitude sinusoid.

If we change the 10th decimal digit of the filter's coefficient, the filter could become unstable producing a growing amplitude sinusoid.

Moreover the relation between the variation of whatever order filters coefficients and the sound variation obtained is quite sophisticated.

For these reasons the parameters control during the musical performance must be entrusted to a particular figure of performer called "system controller".

The musical performer can play his either traditional or electronical instruments.

He doesn't have to take care of the digital filtering algorithms complexity that concur in his musical performance.

Instead the "system controller" must have an appropriate knowledge both in Music and Informatics: he can so interact with the system control software.

This software has a full graphic interface as main instrument of man-machine interaction, as all the FLY30 system software has.

In this way the algorithms control is not obtained by means of a parameters list, but it's obtained through specific drawings of musical structures.

These structures are obtained connecting basic digital computing graphic patches by means of an enhanched graphic editor.

In this way it's easy to modify a parameter because you have only to modify a drawing.

Let's consider a digital reverber algorithm: we could represent the reverber with a drawing of a room.

We could obtain different kind of reverbers simply changing graphically the size and the shape of this room.

Regarding the physical systems simulation, for example the simulation of an organ pipe or a vibrant string, the best representation is the graphic one.

For example we can modify the organ pipe sound or the vibrant string sound, simply modifying the pipe length or the string length graphically represented on computer.

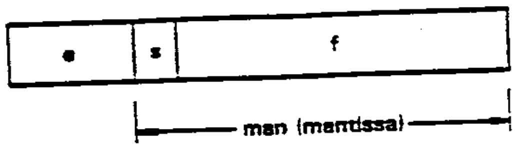
So the user can design musical instruments not developing digital computing algorithms but in a higher level.

This means that the user will be able to draw and realize a musical instrument completely on the FLY30 system.

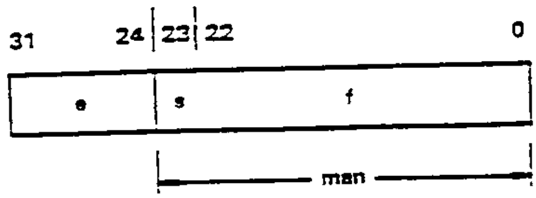
For example, a thin metallic sheet excited by periodic pulses can resound in an organ pipe closed with a membrane, simply drawing it.

In conclusion FLY30 system allows to realize sophisticated floating point digital signal processing algorithms and gives us a particular conception of musical interaction, basically intended as a graphic development of Music-Informatics synergy.

Generic Floating-Point Format



Single-Precision Floating-Point Format



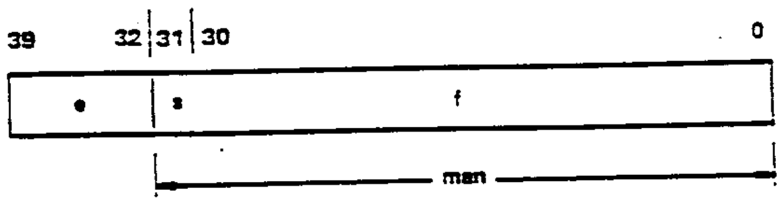
Most Positive: $x = (2 - 2^{-23}) \times 2^{127} = 3.4028234 \times 10^{38}$

Least Positive: $x = 1 \times 2^{-127} = 5.8774717 \times 10^{-39}$

Least Negative: $x = (-1 - 2^{-23}) \times 2^{-127} = -5.8774724 \times 10^{-39}$

Most Negative: $x = -2 \times 2^{127} = -3.4028236 \times 10^{38}$

Extended-Precision Floating-Point Format



Most Positive: $x = (2 - 2^{-31}) \times 2^{127} = 3.4028236683 \times 10^{38}$

Least Positive: $x = 1 \times 2^{-127} = 5.8774717541 \times 10^{-39}$

Least Negative: $x = (-1 - 2^{-31}) \times 2^{-127} = -5.8774717569 \times 10^{-39}$

Most Negative: $x = -2 \times 2^{127} = -3.4028236691 \times 10^{38}$

Fig. 1

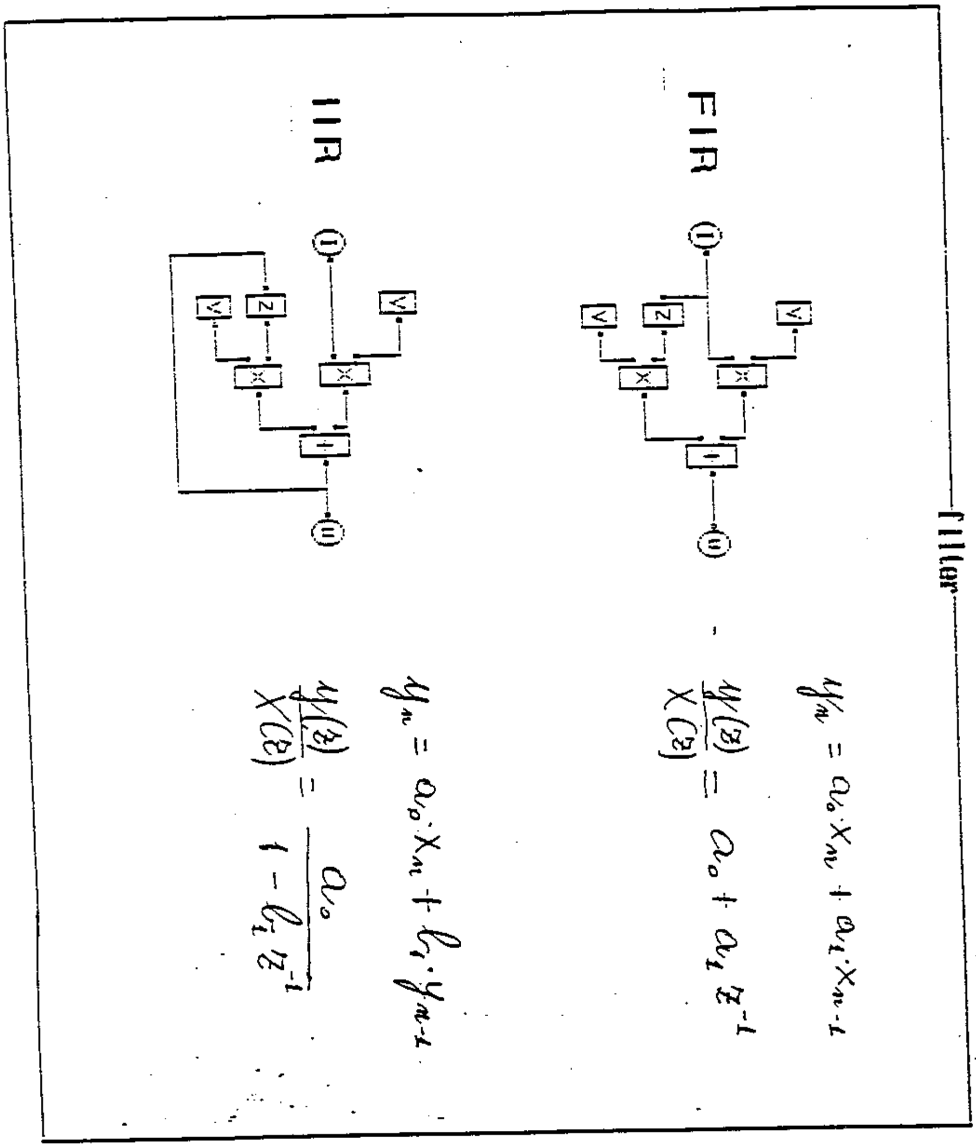
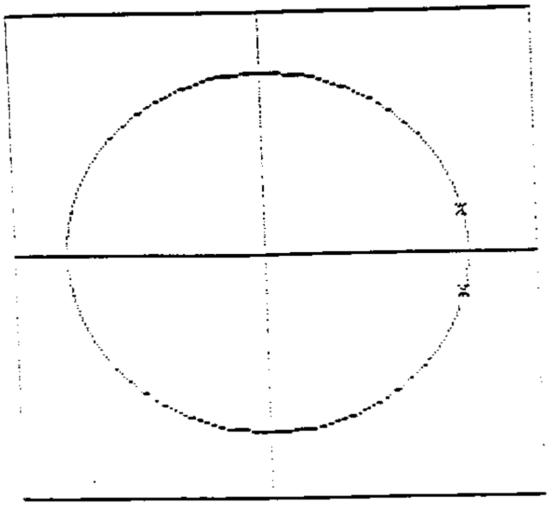

filter
FIR
$y_n = a_0 \cdot x_n + a_1 \cdot x_{n-1}$
$\frac{Y(z)}{X(z)} = a_0 + a_1 z^{-1}$
IIR
$y_n = a_0 \cdot x_n + b_1 \cdot y_{n-1}$
$\frac{Y(z)}{X(z)} = \frac{a_0}{1 - b_1 z^{-1}}$

Fig. 2

ZEROS-POLES SYNTHESIS



polar oscillator

3520 Hz

$y_n = x_n + a_1 y_{n-1} - y_{n-2}$

$a_1 = 1.95128373$

polar multi-oscillator

$y_n = x_n - y_{n-16}$

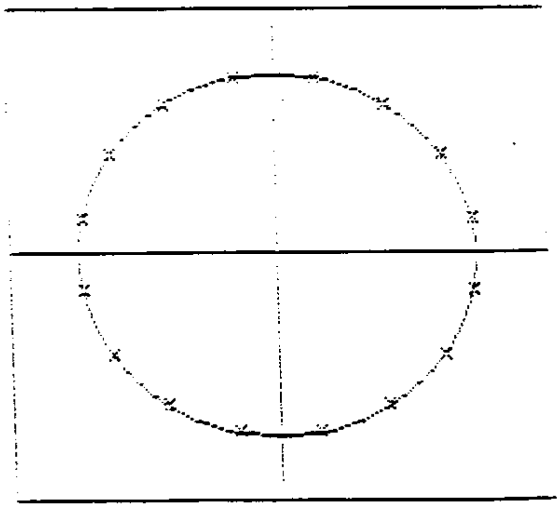
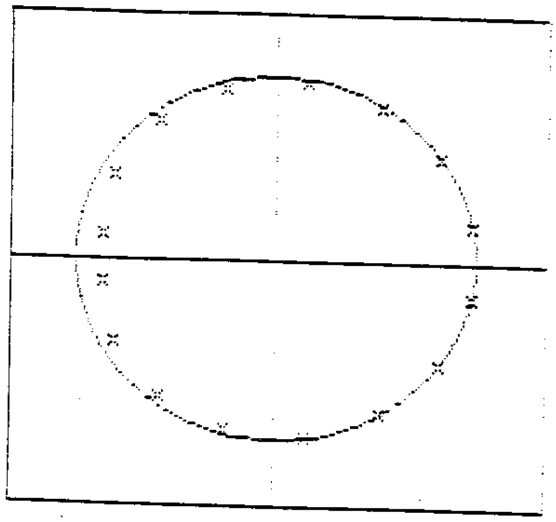


Fig.3

ZEROS-POLES SYNTHESIS

polar oscillator with amplitude decay at high frequencies

$$y_n = x_n - 0.5\,y_{n-15} - 0.5\,y_{n-16}$$



polar oscillator with amplitude decay at low frequencies

$$y_n = x_n + 0.5\,y_{n-15} - 0.5\,y_{n-16}$$

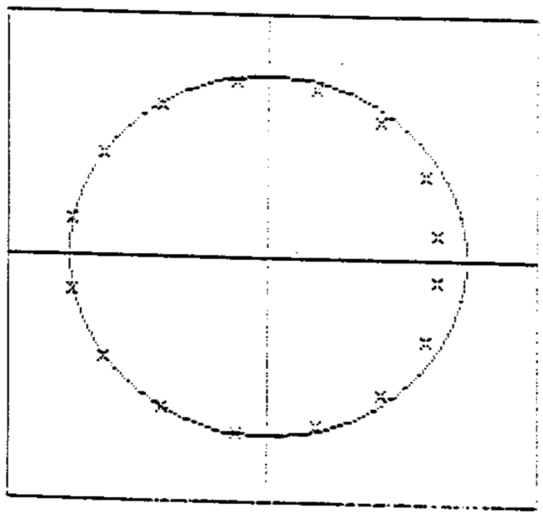
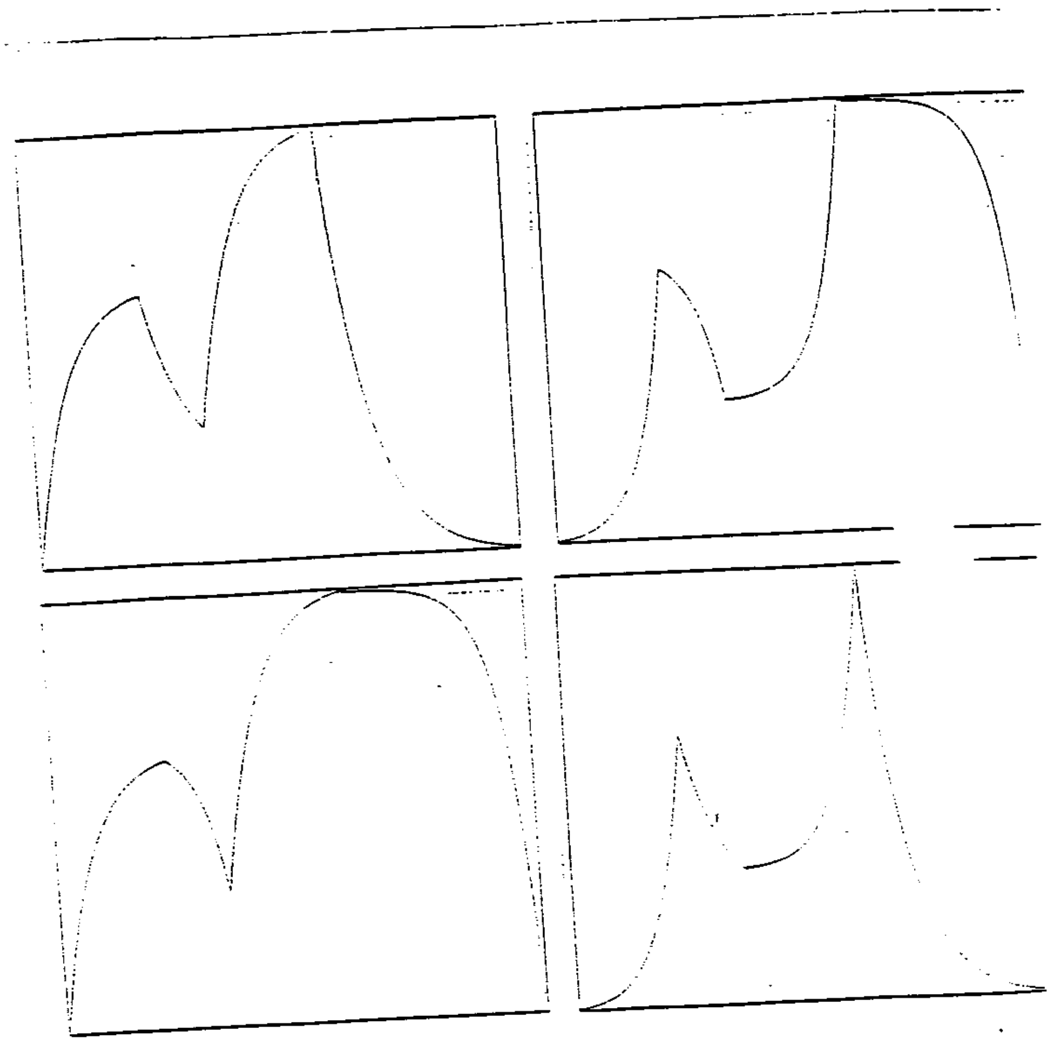


Fig. 4



ENVELOPES

Fig. 5

ENVELOPES

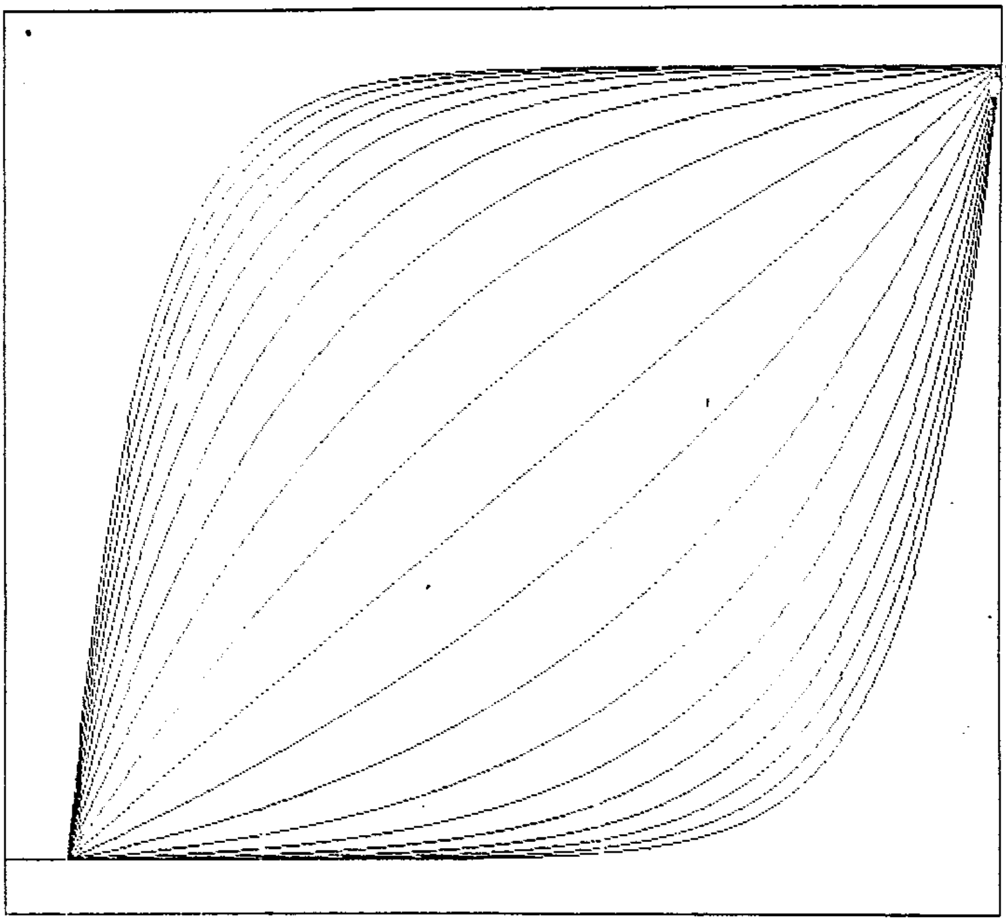


Fig. 6

INTERNATIONAL WORKSHOP ON MAN-MACHINE INTERACTION

IN LIVE MUSICAL PERFORMANCE

PISA 8/6/1991

DR. ALESSANDRA DE VITIS, CENTRO RICERCHE MUSICALI, ROMA

A SYSTEM FOR REAL-TIME SOUND ANALYSIS AND SYNTHESIS

Work realized with the contribution of C.N.R.

In this workshop I speak about a system project developed at "Centro Ricerche Musicali" and it is now being finalized.

This project regards the studies about audio digital signal processing.

The main purpose of our research is to realize a system for acoustic signals analysis and synthesis that is controllable in real time.

Our system is called "FLY30" and it follows the "FLY10" system designed by our artistic director Michelangelo Lupone in 1984.

While we designed the system, we took care in the following features [Fig. 1]:

1] Modular. The system must be expandible to the user's specific requirements.

2] To be interactive in real time. The system must give you the possibility to modify whatever musical parameter in real time.

3] To be user-friendly. The system must interact with the user through an easy-learning software and a full graphic interface.

4] To be a personal system. The system must be installed on a Personal Computer AT with MS-DOS.

5] To have high performance. The system must work on floating-point 32 bits data.

The "FLY30" system carried out in the "Centro Ricerche Musicali" has all these features.

This system is made up by the following subsystems [Fig. 2]:

1] Personal Computer IBM-AT or compatible, used as host computer, with an EGA or VGA graphic adapter, colour monitor, at least 640 Kbytes memory and a 40 Mbytes hard disk.

2] One or more digital signal processing board "SPIRIT30" installed on AT computer.

3] An analog signal input mixer and a quadriphonic output mixer.

4] A "FLY10" system used as acoustic signal generator.

5] Control instruments like frequency meter and oscilloscope.

6] Standard MIDI instrumensts interface.

The system centre is the digital signal processing board "SPIRIT30".

Each board is made up essentially by the following parts [Fig. 3a]:

1] A digital signal processor TMS320C30 by Texas Instruments.

This processor is able to perform parallel instructions on floating-point 32 bits data in 60 nanosec.

Moreover the TMS320C30 contains, internally, all the necessary interfaces to receive and transmit with the external peripherals.

It contains a DMA controller, a serial interface, a timer, 2 data bus, signals to synchronize with other processors.

2] A logic interface with the AT computer and other I/O devices.

This allows a very fast data exchange between the board and the host computer and gives the possibility to connect in parallel optional I/O devices.

3] A fast static RAM with zero wait-state, made up by 5 banks of 32k*32 bits each. We can add other 3 banks of 32k*32 bits, achieving up to 256k*32 bits of fast memory.

4] One or more Analog-to-digital and digital-to-analog conversion board "IC100".

This board is made up of 2 analog input channels and 2 analog output channels and performs a 16 bits conversion.

The analog-to-digital converters work with a maximum sample rate of 6.14 MHz. They perform the average of 64 insuccession samples, achieving a 96 KHz sample rate [Fig. 4].

The synchronization and clocking circuit allows a direct connection with the serial interface of the DSP TMS320C30.

During the approximate 10 microsec sampling time, the 32 receiving data bits are transmitted and the 32 transmitting data bits are received.

Therefore the "SPIRIT30" processing board is right for the use in the field of acoustic signal analysis and synthesis.

A basic part of the "FLY30" system is the applied software.

This software has been fully designed and developed by Antonio Pellecchia, who is the scientific director of the "Centro Ricerche Musicali".

This set of software is made up basically by the following softwares [Fig. 3b]:

1] Graphic software to process digital computing algorithms in real time.

2] Compiler from music language "FLY30" to TMS320C30 assembler.

3] Software allowing a background musical performance.

4] Software to control the musical execution parameters in real time.

The enhanced graphic editor is named "Digital Signal Patcher" [Fig. 5].

This software has been developed under MS-DOS operative system and it has been written in C language.

This software allows you to design and to develop whatever digital signal processing algorithm and to listen to the sound designed through the DSP board.

After you have drawn the digital signal processing algorithm, the software is able to:

1] Interpret the graphic and translate it immediately in the TMS320C30 corresponding assembler code.

2] Compile the obtained assembler source file.

3] Transmit the executable file to the SPIRIT30 DSP board that will execute the digital computing program.

4] Translate the drawing into the music language FLY30 for the following use of the cross-compiler.

The cross-compiler allows you to compile, and then to translate in TMS320C30 machine code, digital signal programs written in FLY30 language [Fig. 3b].

This language has been designed for the purpose of the FLY30 system and the corresponding compiler is now being finalized to be realized by means of the YACC program (Yet Another Compiler of Compilers).

This cross-compiler and the following linker allows you to assign the execution of a certain digital signal processing algorithms to whatever external event.

The FLY30 language allows you to decide what set of parameters must be controlled in real time during the musical performance.

The real time controlling software is a graphic user interface and allows you to modify, simply and quickly, the parameters of the digital signal processing algorithm you have decided to control.

In this way the musical performance is divided up to 2 basic figures [Fig. 2]:

1] The performer of the traditional musical score.

He can play traditional or electronical instruments.

2] The controller of the digital signal algorithms.

He works with the system by means of the controlling software and communicates through the keyboard, the mouse, or other devices.

Finally the system is completed by software that allows to interface the digital signal processing system with standard MIDI instrumentation.

This makes possible the accessibility of the system to the MIDI users, and gives the possibility to use commercial MIDI software like musical score editors.

In conclusion the FLY30 system should answer the requirements of the electronic music producers to have an own personal system for acoustic signal analysis and synthesis at low cost and high performance.

FLY30

1--> MODULAR

2--> INTERACTIVE

3--> USER-FRIENDLY

4--> PERSONAL

5--> HIGH PERFORMANCE

Fig 1

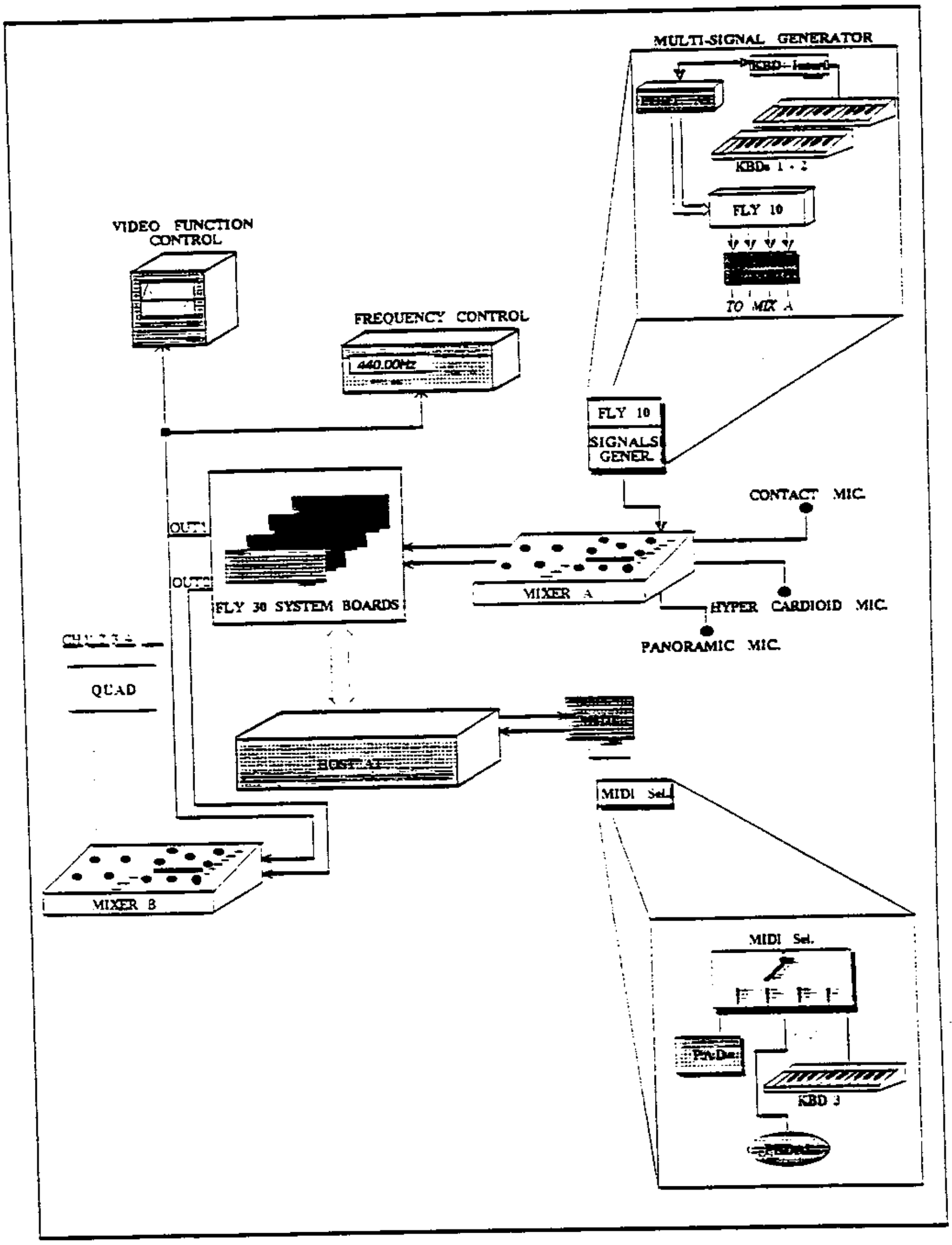


Fig. 2 Fly 30 System: a CRM complete workstation

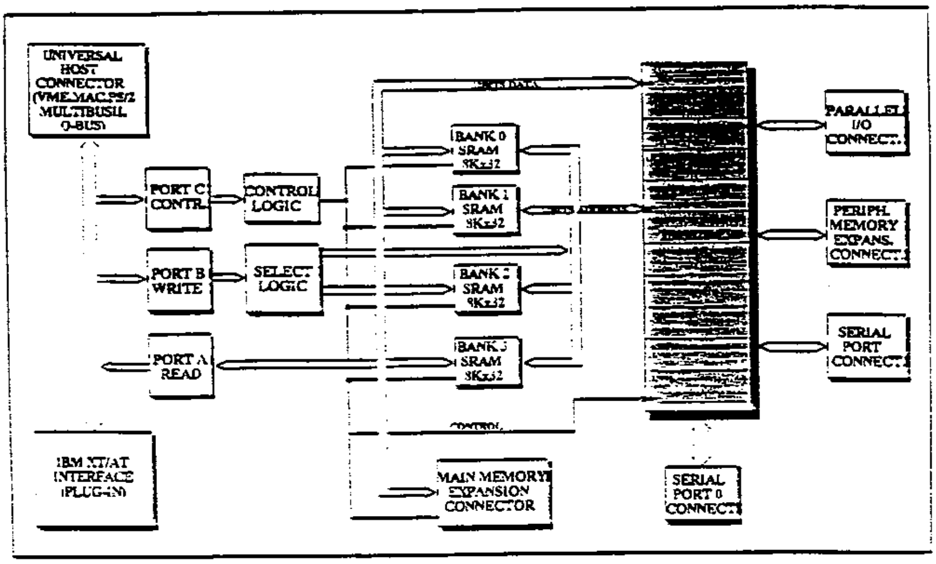


Fig. 3a Spirit 30 architecture

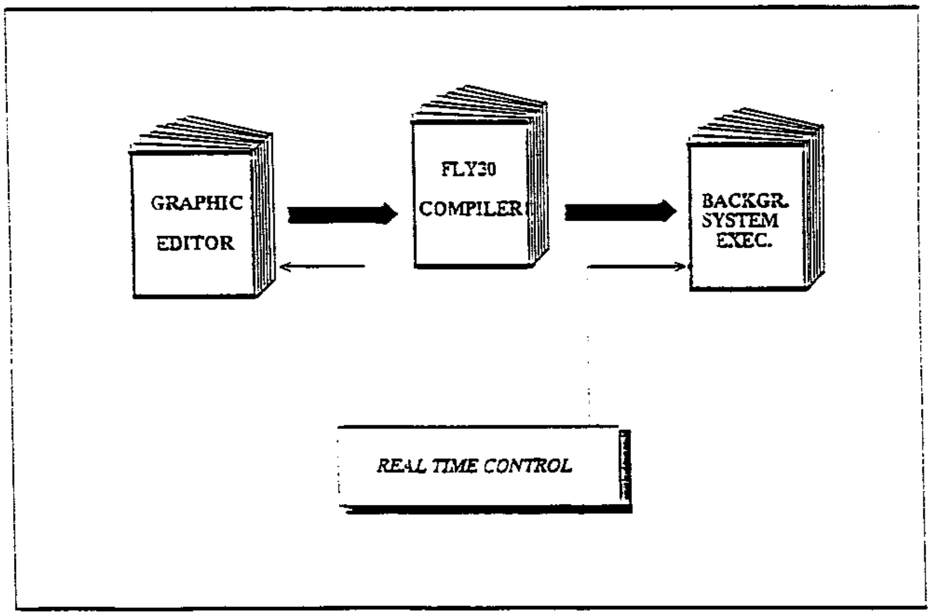


Fig. 3b Fly 30 software system

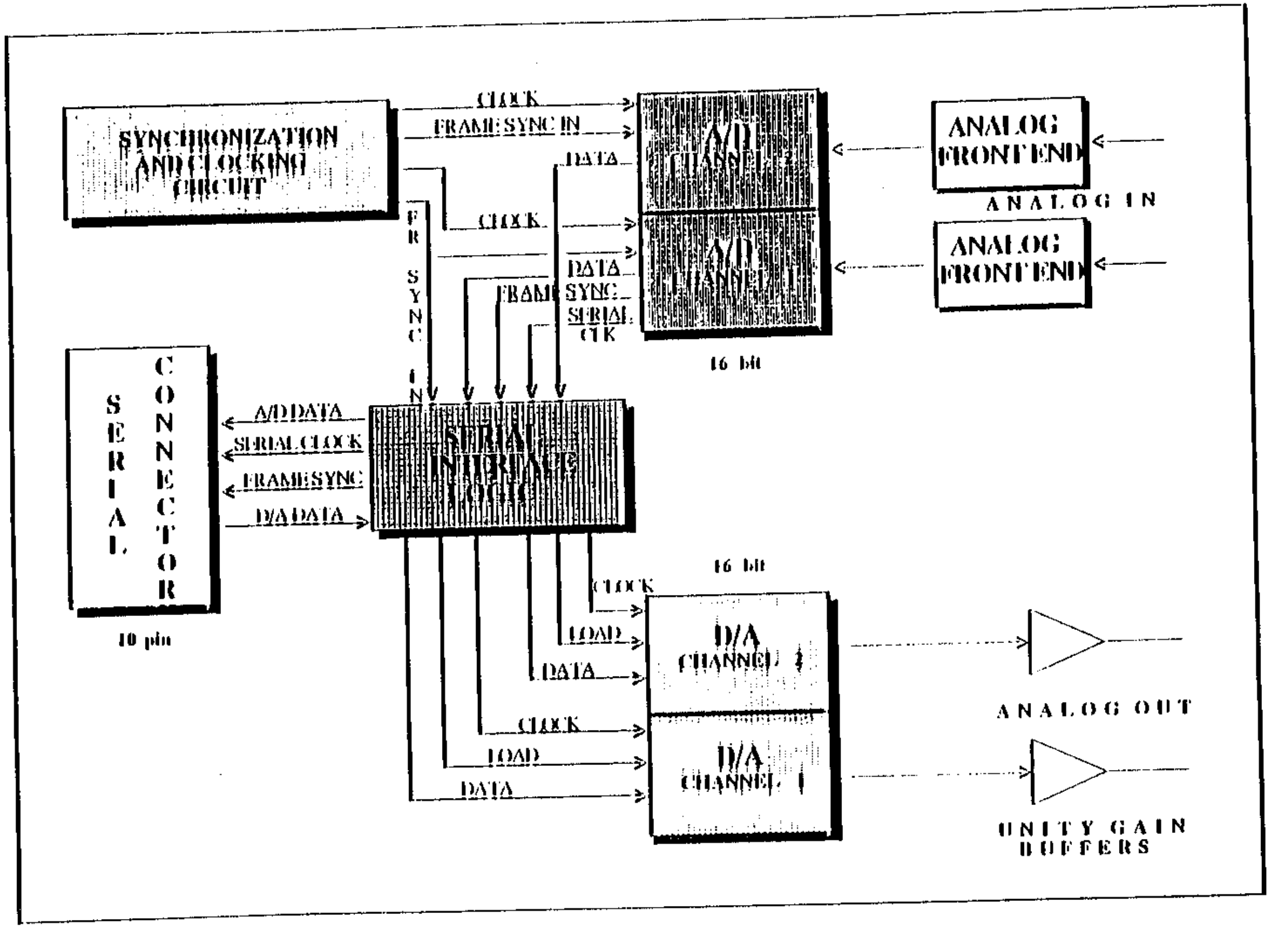


Fig 4 *IC-100 block diagram*

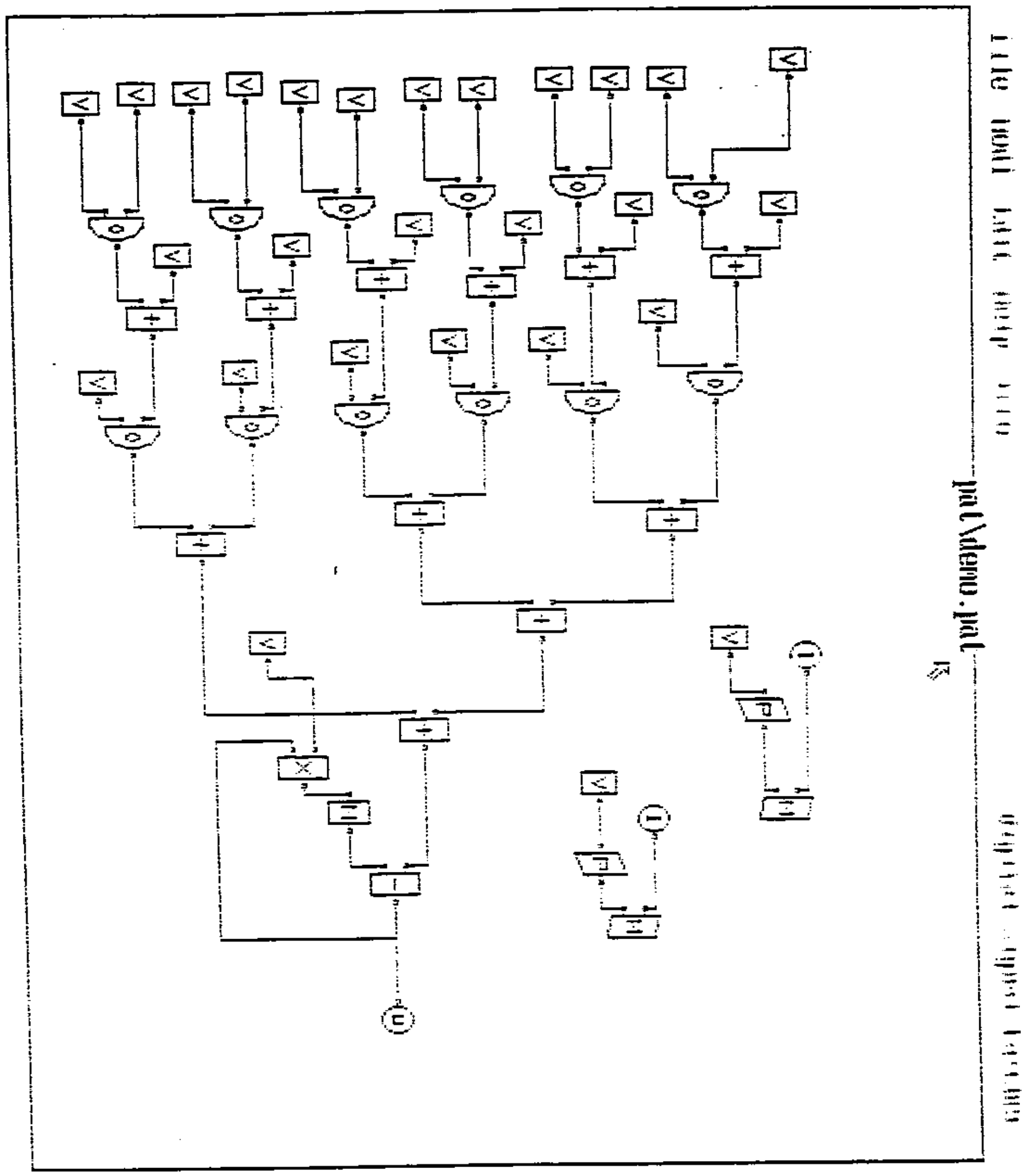


Fig. 5

Bibliografia

ALLES, H. G. "A portable digital sound synthesis system" *Computer Music Journal*, (1977)

ALLES, H.G. / DI GIUGNO, G. "The 4B: a one-card 64 channel digital synthesizer" *Computer Music Journal* (1977)

ALONSO, S. / APPLETON, J. / JONES, C. "A special-purpose digital system for musical instruction, composition, and performance" *Computers and the Humanities* (1976)

ARFIB, D. "Digital synthesis of complex spectra by means of multiplication of non-linear distorted sine waves" *Journal of the Audio Engineering Society* (1979)

ASTA, V. / SCHAUVEAU, A. / DI GIUGNO, G. / KOTT, J. *Il sistema di sintesi digitale in tempo reale 4x Automazione e Strumentazione*, vol. 28 (1981)

BEAUCHAMP, J. W. "A computer system for time-variant harmonic analysis and synthesis of musical tones" in von Foerster/Beauchamp, eds. *Music by computers*, (Wiley, 1969)

BEAUCHAMP, J. W. "Analysis and synthesis of cornet tones using non-linear inter-harmonic relationships" *Journal of the Audio Engineering Society* (1975)

BERANEK, L. L. *Acoustics* (Mcgraw-Hill, 1954)

BUXTON, W. "Introduzione alla computer music" *Musica ed elaboratore*, (La Biennale 1980)

CHOWNING, J. M. "The synthesis of audio spectra by means of frequency modulation" *JAES*, (1973)

DATE, H. / TOZUKA, Y. "An artificial reverberator whose amplitude and reverberation time-frequency characteristics can be controlled independently" *Acoustics* (1966)

DE POLI, G. "Tecniche numeriche di sintesi della musica" *Bollettino del LIMB* (1981)

DI GIUGNO, G. "A 256 oscillator bank" comunicazione a *The 1976 International Computer Music Conference*, (M.I.T., 1976)

ERICKSON, R. "Sound structure in music" *University of California Press* (1975)

GORDON, J. / GREY, J. M. "Perception of spectral modification on orchestral instrument tones" *Computer Music Journal* (1968)

GREY, J. M. / MOORER, J. A. "Perceptual evaluation of synthesized musical instrument tones" *Journal of the Acoustical Society of America* (1977)

HARRIS, C. M. "Absorption of sound in air in the audio-frequency range" *Journal of the Acoustical Society of America* (1963)

KAHRS, M. "A computer language for psychoacoustic study and musical control of timbre" comunicazione a *1977 International Computer Music Conference, University of California - San Diego*

KOENIG, J. M. "Processi compositivi " *Musica ed Elaboratore* (La Biennale 1980)

KUTTRUFF, H. *Room acoustics* (Applied Science Publishers 1973)

LEBRUN, M. "Digital waveshaping synthesis" *Journal of the Audio Engineering Society* (1979)

"Lexicon of analyzed tones" *Computer Music Journal*, vol.1 n.2 1977, vol.1 n.3 1977, vol.2 n.2 1978

LUPONE, M. "Informatica musicale presso la S.I.M." *I Colloquio di Informatica Musicale* (1984)

LUPONE, M. "System Fly" *VI Colloquio di Informatica Musicale, Quaderni di Musica e Realtà* (1985)

LUPONE, M. "Dal pianoforte al computer" *Piano Time* (1987)

LUPONE, M. "Musica Ricerca" *Rassegna di Musica Contemporanea Bari* (1988)

LUPONE, M. "Verso l'esperienza informatica" *Suono Sud* (1988)

LUPONE, M. "La musica elettronica" - intervista - *Professione Musica* (1988)

LUPONE, M. "Ricerca: attitudine musicale - metodo scientifico" *Musica/Scienza - il margine sottile* (1991)

MARQUARDT, D. W. "An algorithm for least-squares estimation of non-linear parameters" *Journal of the Society for Industrial and Applied Control* (1963)

MCADAMS, S. / BREGMAN, A. "Hearing musical streams" *Computer Music Journal* (1979)

MIAN, G. A. / TISATO, G. "Composizione musicale e trattamento numerico della voce" *LIMB* (1984)

MOORER, J. A. "On the loudness of complex, time-variant tones" report *STAN-M-4 Stanford University Department of Music* (1975)

MOORER, G. A. "The heterodyne filter as a tool for analysis of transient waveforms" report *STAN-CS-73-379 Stanford University Department of Musica* (1973)

MOORER, G. A. "The synthesis of complex audio spectra by means of discrete summation formulae" *Journal of the Audio Engineering Society* (1976)

MOORER, G. A. "Signal processing aspects of computer music - A survey" *proceedings of the IEEE* (1977)

MOORER, G. A. / ABBOTT, C. / EASTTY, P. / LAWSON, J. "The 4C machine" *Computer Music Journal* (1979)

OPPENHEIM, A. V. / WEINSTEIN, C. J. "Effects of finite register length in digital filtering and the fast Fourier transform" *Proceedings of the IEEE* (1972)

OPPENHEIM, A. V. / SCHAFER, R. W. *Digital signal processing* (Prentice-Hall 1975)

PREVOT, P. "Controllo in tempo reale di un sistema di trattamento/sintesi del suono" atti del*IV Colloquio d'Informatica Musicale vol. 1*" (1981)

RISSET, J. C. / MATHEWS, M. V. "Analysis of musical instruments tones" *Physics today* (1969)

RISSET, J. C. / WESSEL, D. "Indagine sul timbro mediante analisi e sintesi" *LIMB 2* (1982)

SANTOIEMMA, M. "Rappresentazione formale delle strutture di base degli strumenti per la sintesi del suono" *LIMB 2* (1982)

SAPIR, S. "Computer music in tempo reale" *Strumenti Musicali* (aprile/maggio 1985)

SCHROEDER, M. R. "Natural sounding artificial reverberation" *Journal of the Audio Engineering Society* (1962)

SCHROEDER, M. R. "Digital simulation of sound transmission in reverberal space, part 1" *Journal of the Acoustical Society of America* (1970)

SCHROEDER, M. R. / GOTTLOB, D. / SIEBRASSE, K. F. "Comparative study of European concert halls" *Journal of Acoustical Society of America* (1974)

SONITECH, Int. Inc. *Spirit 30 system: technical reference manual - version 2.0* (1989/90)

SONITECH, Int. Inc. *Spirit 30 system: application notes - version 2.0* (1989/90)

SONITECH, Int. Inc. *Spirit 30 system: EDSP user manual - revision 2.0* (1989/90)

SONITECH, Int. Inc. *Spirit 30 system: IC-100 user manual - version preliminary* (1989/90)

TEXAS INSTR. *TMS320C3x user's guide* (1989)

TEXAS INSTR. *TMS320C10 user's guide* (1983)

TEXAS INSTR. *TMS320C20 user's guide* (1985)

TISATO, G. "Computer music a Padova" *Strumenti Musicali* (1981)

WESSEL, D. L. "Perceptually based controls for additive synthesis" presentato alla *1976 International Conference on Computer Music*

WESSEL, D. L. / GREY, J. M. "Conceptual structures for the representation of musical material" *Technical Report 14 IRCAM - Paris*