A NOVEL SYSTEM FOR MUSIC SYNTHESIS USING A DIGITAL SOUND PROCESSOR IC

Giancarlo Caironi, Antonio Battaiotto, Paolo Lupi,
Guido Torelli and Ermes Viani
SGS Microelettronica S.p.A.
20041 Agrate Brianza, Milan, Italy

ABSTRACT

A 16–channel, polyphonic and polytimbric, fully digital sound processor IC for music synthesis is described. It is microprocessor–controlled and the coded music information is resident in an external memory.

The analog filters and attenuators, typical of traditional sound generation, can be completely eliminated. The major advantages are drastic reduction of board spacing, no need of any kind of calibration, sound characteristic stability with time and from set to set. Applications include classic electronic organs, keyboard synthesizers and personal computer musical peripherals.

1. INTRODUCTION

Traditional electronic organs and keyboards are based on the principle of subtractive synthesis [1] (see Fig. 1). Following this approach, a series of square or staircase waveforms with frequency related to the note played, are generated by division, starting from a common high–frequency oscillator. As these waveforms contain various amounts of the basic frequency harmonics, it is easy to extract the sound spectra characteristic of a specific instrument by using various types of analog filters.

The other characteristic, typical of each instrument, is the amplitude variation with the time of the sound when the note is played (the so called "ADSR" envelope: Attack, Decay, Sustain, Release).

Various microprocessor–controlled sound generators working on the above described principle have been proposed (e.g. [2]).

The major drawback of this solution is the high number of analog filters, voltage controlled attenuators (VCAs), and linear or exponential voltage controlled filters (VCFs) required to synthesize particular sounds.

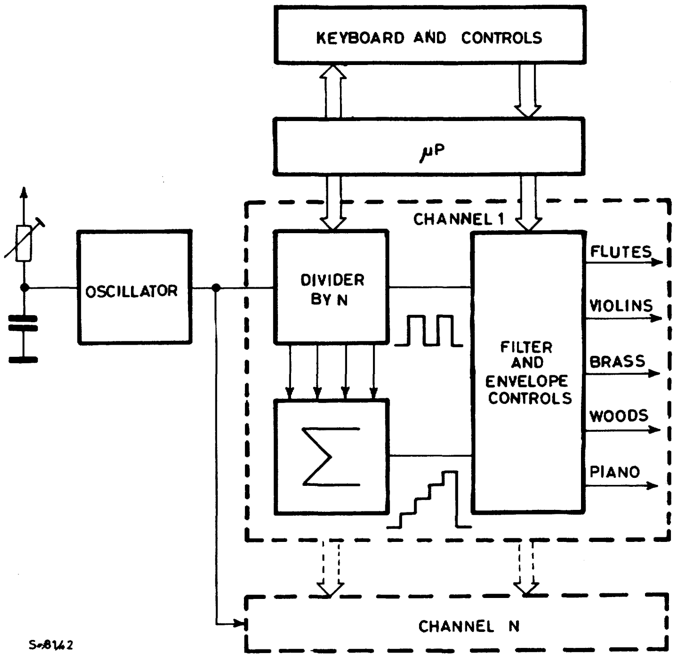


Fig. 1 - *System based on the principle of subtractive synthesis*

0098-3063/85/0800-0505$01.00 © 1985 IEEE

Manuscript received June 10, 1985.

The system presented in this paper operates with the look-up table principle, and is fully digital. It is based on an especially-developed integrated circuit (IC), named Digital Sound Processor (DSP)*, and requires minimum component count to achieve sound generation (see Fig. 2).

The DSP synthesizes the desired sound, starting from a table stored in an external memory (typically a ROM), which contains the digital codes of the sampled values of the analog waveforms to be synthesized.

Different tables are stored in the memory, corresponding to different sound waveforms.

A microprocessor programs the address of the table to be read from the memory, and the frequency at which bytes of the tables have to be read. After each readout, the IC generates a 1.5-μs output current pulse whose amplitude depends on the read value, thus giving rise to a Pulse-Amplitude-Modulated (PAM) output current signal (see Fig. 3). The synthesized analog signal is obtained by means of a simple external low-pass filter or integrator.

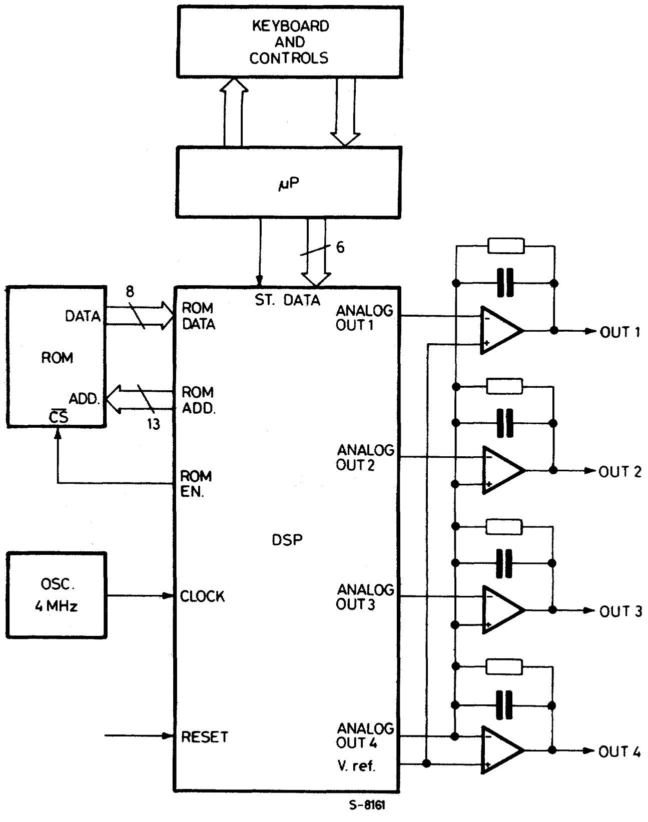


Fig. 2 - *Typical system configuration based on the DSP IC*

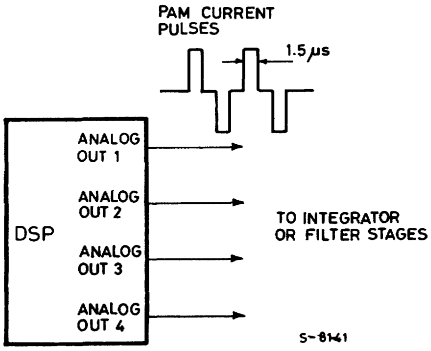


Fig. 3 - *PAM output current signal*

Only one filter or integrator per output is required, as already shown in Fig. 2.

Main characteristics of this device are :

- 16 independent sound channels;
- maximum addressable memory: 8k bytes (64k bits);
- selectable table length and reading mode (table length can range from 16 to 2048 bytes with a typical value from 32 to 64 bytes);
- possibility of mixing between two tables;
- possibility of interpolation between two values of the same table, through multiple readouts;
- integrated 8-bit D/A converter, 1/2-LSB precision;
- integrated 10-bit attenuator (maximum attenuation: -60 dB);
- automatic smoothing of amplitude variation; and
- 4 selectable analogue outputs.

A standard 4-μm N-MOS technology is used for this 5-Volt only, 40-pin DIP device. The die size is 220 × 236 mils^2.

*SGS M114

2. DESCRIPTION OF THE SOUND SYNTHESIS TECHNIQUE

As already mentioned, the sound generation is based on the look–up table principle.
As the actual sound waveforms to be synthesized generally change continuously with time, different tables have to be stored in the external memory and read sequentially by the DSP. To minimize the number of different tables to be memorized and to avoid non–continuous sound transitions when two different tables are read sequentially, a mixing feature between two tables is provided. When two different waveforms have to be synthesized successively, the two corresponding tables are read and an integrated microprocessor–programmable 4–bit interpolator determines the data to be fed to the D/A converter. Thus, the contents of the two tables are mixed, and a smooth passage between the two waveforms is achieved. In this way, minimization of the number of tables to be written in the memory is also achieved, as even relatively different tables can be sequentially addressed.

Two different codings are available to write data into the tables: Absolute Value and Delta Modulation Coding.

With the former, the instantaneous values of the sound waveform samples are stored. To reconstruct the analog signal from the PAM output, a simple low–pass filter is required for each output pin of the DSP.

With the latter coding, the difference between two contiguous sampled values of the waveform is stored. In this case, an integrator stage is required for each analog output to reconstruct the signal.
The current output of the IC makes it easier to implement the external stage.
The Delta Modulation also has the advantage of the possibility of interpolation between two values, as an appropriate circuit is provided in the DSP.
The disadvantage of this coding is the need to take care that the integral of all samples of each period is always zero, otherwise a D.C. component could saturate the external integrator. This means that appropriate sample values have to be stored in the tables. Moreover, when the sound to be synthesized is changed, the current table has to be completely scanned and synthesized before beginning the scanning of the new table.

Depending on the sound characteristics, two different techniques of synthesis and, therefore, of coding are used :

- fully PCM; and
- repetitive readout of the same table.

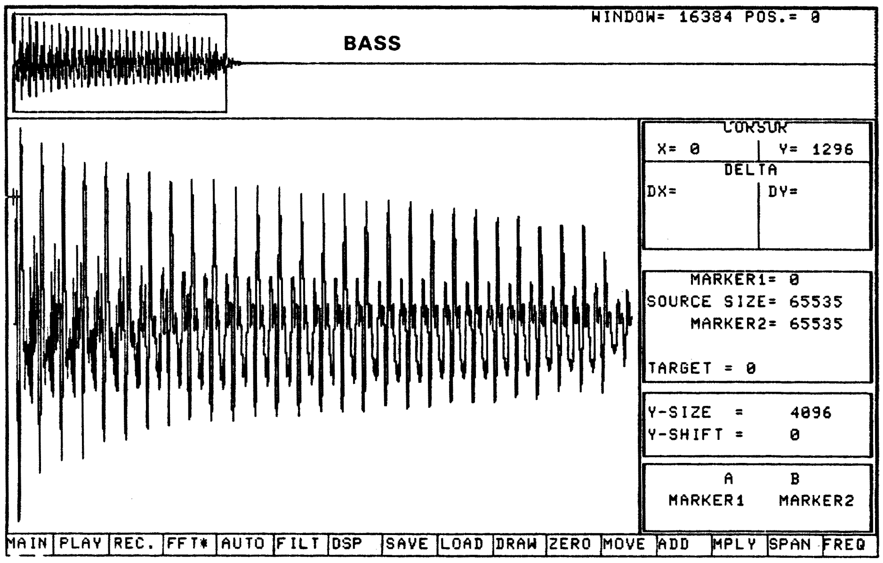
The fully PCM technique is used for the non–repetitive (non–periodic) parts of the sound to be reproduced.
In this case, the table is read only once and, therefore, it can occupy a relatively large part of the external ROM.

The repetitive readout is, instead, used for the generation of the so–called "periodic" parts of the sound.
In this case, only one period of the sound is recorded in the table that will repeatedly read at the right speed in order to achieve the desired sound.
In order to have a dynamic variation of the sound spectrum during the sound evolution in the time, several tables having the same length, but of different content, are sequentially read.

Fig. 4 shows the recording of a "Bass" sound. It can be noted that there is a continuous sound variation from the beginning to the end of the periods considered.
Fig. 5 shows the first period of the same recording, the second period (Table A) and a period at the end of the same recording (Table B).
In practice, only these 3 periods are stored into the memory. The first period is "non periodic" and, therefore, it is stored with PCM technique. The other two periods are, instead, sequentially read and mixed in order to have a smooth variation between them. The mixing characteristic is shown in Fig. 6.

Fig. 7 and Fig. 8 are, instead, related to the sound envelope and the frequency variation with the time.

The tables are read with a frequency equivalent to the frequency of the note to be generated multiplied by the length of the table.
In this way, the sampling frequency is synchronous with that of the signal to be reproduced. The most important result is the complete elimination of intermodulation.



BASS

WINDOW= 16384 POS.= 0

CURSOR
X= 0 | Y= 1296

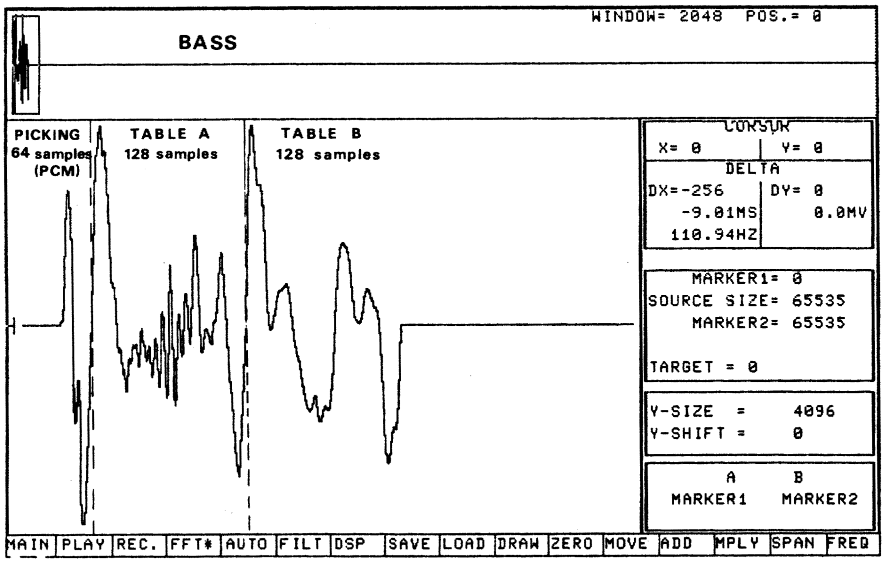
DELTA
DX= | DY=

MARKER1= 0
SOURCE SIZE= 65535
MARKER2= 65535
TARGET = 0

Y-SIZE = 4096
Y-SHIFT = 0

A B
MARKER1 MARKER2

Fig. 4 - *Recording of a "Bass" sound*



BASS

WINDOW= 2048 POS.= 0

CURSOR
X= 0 | Y= 0

DELTA
DX=-256 | DY= 0
-9.01MS | 0.0MV
110.94HZ

MARKER1= 0
SOURCE SIZE= 65535
MARKER2= 65535
TARGET = 0

Y-SIZE = 4096
Y-SHIFT = 0

A B
MARKER1 MARKER2

Fig. 5 - *Bass-sound recording: first, second and final periods*

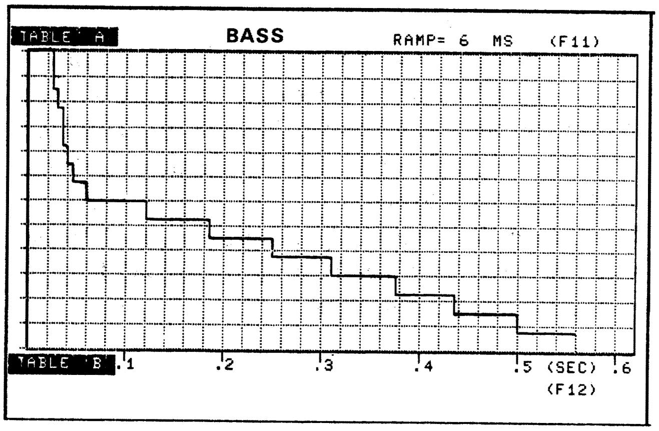


Fig. 6 - *Mixing characteristic between tables A and B*

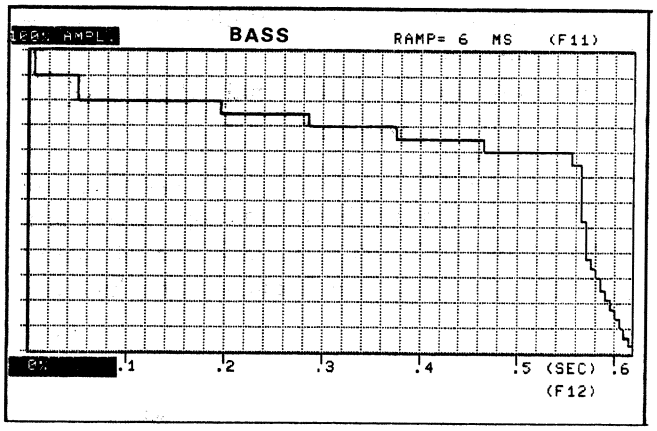


Fig. 7 - *Envelope characteristic of the generated sound*

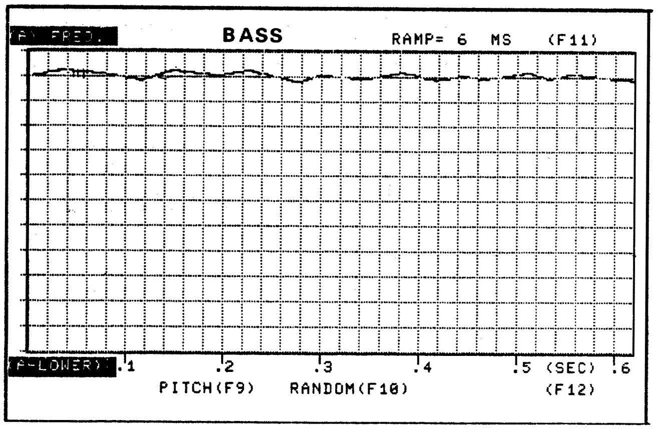


Fig. 8 - *Frequency variation of the generated sound*

The other important element in the sound synthesis is the amplitude variation with the time, i.e., the envelope.

This parameter is controlled, in real time, by the microprocessor through 6 control bits. These bits are used to address an internal ROM with 64 output levels. Attenuation step is 0.75 dB for the attenuation range 0 to -40 dB, and increases in the remaining of the range.

The DSP has an internal circuitry which automatically smooths the step between two different attenuation levels programmed successively, with a time constant that is inversely proportional to the frequency of the note played. This feature of the DSP is shown in Fig. 9.

The continuous line is the ideal theoretical ADSR envelope. The microprocessor controls the discrete values (steps). Of course, the steps will be as small as the interventions of the microprocessor to control the amplitude are more frequent, with only the limitation of the smallest step value.

The dotted line shows, instead, the result of the smoothing circuitry. The same result is shown enlarged in Fig. 10.

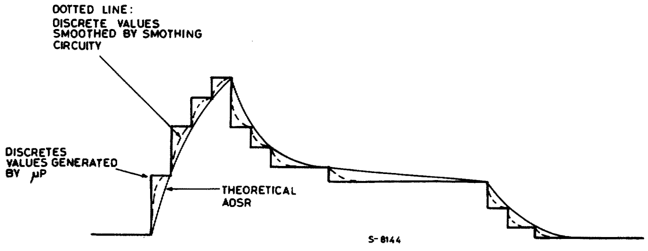

DOTTED LINE:
DISCRETE VALUES SMOOTHED BY SMOTHING CIRCUITY
DISCRETES VALUES GENERATED BY µP
THEORETICAL ADSR
S-8144

Fig. 9 - *Sound envelope generation (ADSR)*

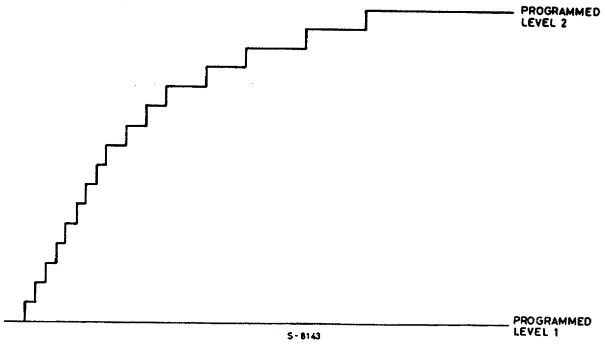


Fig. 10 - *Automatic smoothing characteristic of the generated envelope*

3. SOUND ANALYSIS

The tables to be written in the external memory can be a simple reproduction of the most significant periods of the original sound.
However, the choice of these periods cannot only be based on the sound waveform, but the sound spectrum should be also considered in order to avoid a redundancy of the tables. In fact, it can happen that two periods, with different waveforms, have, instead, the same frequency spectrum, due to different phases of the various harmonics.
The human ear cannot distinguish the phase differences and, therefore, one period is enough to reproduce two apparently different waveforms.

The Fast Fourier Transform (FFT), and its reverse, which allows the reconstruction of the original waveform, is the simplest method to be implemented in equipment for sound analysis.
This can be a computer that, with an appropriate software, can automatically generate the look–up table to be put in the ROM, starting from the digitally converted original information.
Advanced personal computers can easily do this job, providing also interactive video and sound information to the analyst.

An alternative approach has also been developed; it is a board that can be connected to a simple oscilloscope, which provides sound analysis without passing through the Fourier Transform. However, both the original sound and the synthesized one can be seen and heard.
This board automatically programs an EPROM from the tables produced by the sound analysis, in the right form for the DSP.

4. DESCRIPTION OF THE DIGITAL SOUND PROCESSOR IC

The block diagram of the Digital Sound Processor IC is shown in Fig. 11.
As shown by the dashed lines, the device can be divided into five sections, each including one or more functional blocks.
The diagram also shows the main data flow between different blocks.

The device receives the information regarding the sound to be synthesized from a microprocessor or a microcomputer (referred to as μP in the following) via the μP interface. The frequency at which the

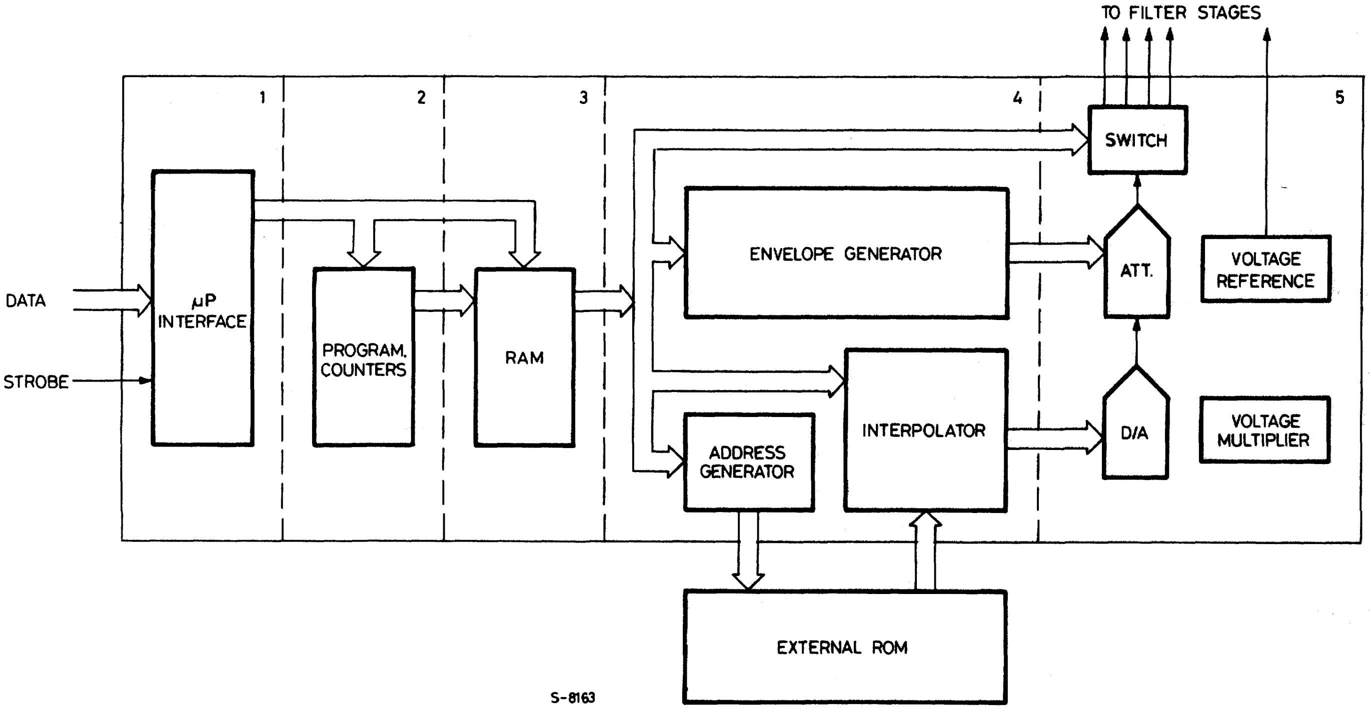


Fig. 11 - *Block diagram of the Digital Sound Processor IC*

tables containing the sampled values of the music waveforms have to be read, is determined by the programmable–counter section. The address of the tables to be read, are stored in the integrated RAM, together with other current data necessary to generate the desired sound. The fourth section performs digital processing of the data read from the external ROM, while the last section converts the processed data to the analog current pulses which are delivered to the selected output pin.
Sixteen independent sound channels and four analog output pins are provided.

Microprocessor interface

The Microprocessor Interface receives data from a μP via a parallel bus (6 data lines and a strobe line).
Data are transferred to the IC asynchronously, using a very simple and flexible communication protocol. Synchronization with the internal circuits is then performed by the integrated interface to allow correct data loading into the memory and into the programmable counters.
The timing of data transfer is shown in Fig. 12.

48 bits are necessary to define the sound to be generated by one of the sixteen integrated sound channels. They are divided by the μP into eight 6–bit groups, which are sent sequentially to the IC. To signal that valid data are present on the parallel bus, the μP causes a level transition (high/low or low/high) on the strobe line.
The strobe line transitions are also counted by the IC: after eight transitions, a synchronization routine is internally started, thus allowing the 48 received bits to be transferred correctly to the internal circuits. Of course, before each data transfer from the μP to the DSP IC, the internal strobe transition counter is reset to provide proper operation.

As mentioned above, 48 bits are transmitted by the μP at each complete data transfer :

- 4 bits address the channel to be allocated to the sound which is being entered;
- 9 bits program the frequency of table scanning, i.e., the frequency at which the bytes of the tables stored in the external ROM have to be read. Thus, they also program the frequency of the generated sound;
- 16 bits program the addresses of the two tables to be read by the IC to generate the desired sound;
- 7 bits program the new attenuation value of the signal delivered to the IC analog output, and select the variation law relative to the passage from the old to the new attenuation value;
- 6 bits select the table lengths and reading modes;
- 4 bits program the interpolation coefficient for the mixing feature between the two tables; and
- 2 bits select the analog output pin where the sound channel output has to be delivered.

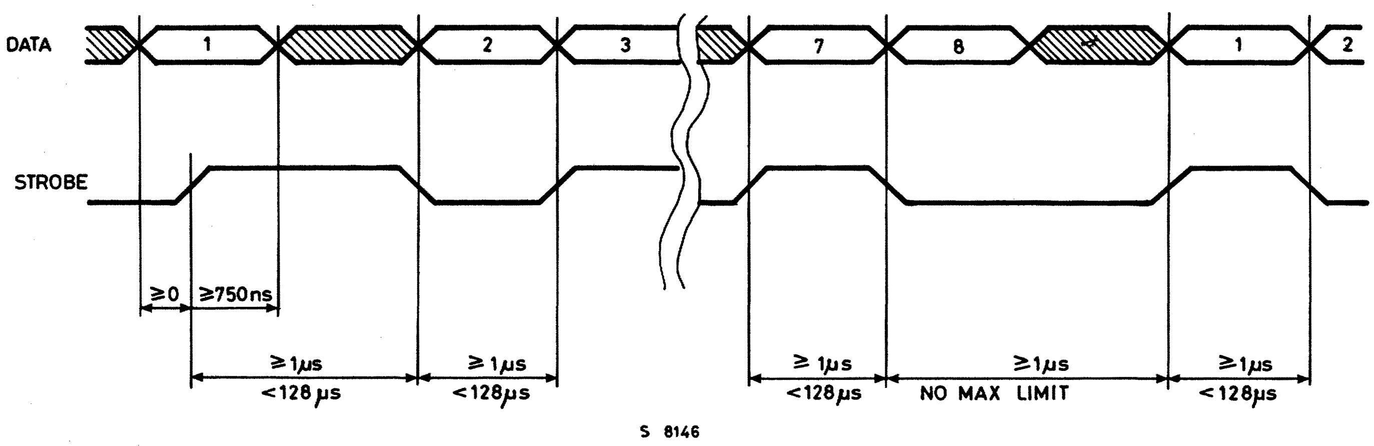


Fig. 12 - *Data–bus timing*

Programmable counters

16 programmable counters (one for each sound channel) are integrated in the device. They are directly controlled by the μP–interface through the 9 readout frequency programming bits, and their main task is to determine the exact time when the variable–amplitude current pulses have to be delivered to the analog–output pins. In this way, the programmable counters control the frequency of the generated sound (the repetition frequency of the output pulses ranges from 8.1 kHz to 38.5 kHz).

Two signals are output from the programmable–counters section: the 4–bit address of the sound channel which requires the current pulse to be output and a start pulse for a "service routine", which is performed by the circuits integrated in the last three sections of the IC. These can serve only one request at a time so, during the execution of the service routine (2 μs), the outputs of the programmable counters are latched.

A service routine request may be issued simultaneously by two or more programmable counters. On the other hand, a request may be issued during the 2–μs time interval needed to execute a previously–requested routine.
Priority logic is integrated, as well as circuitry to provide queue management. Different routine requests are executed sequentially, according to the order of priority. The priority for all 16 channels is internally fixed.
Of course, the delay in the execution of some service routines causes noise in the generated sound (the so–called "collision–noise"). Statistical analysis has demonstrated that, under normal conditions, the amount of collision noise introduces negligible degradation in the sound output by the reconstruction filters.

Integrated RAM

The device integrates a 1568–bit static RAM, where the μP interface loads the remaining data received from the μP relative to each of the 16 sound channels.
This memory also stores, for each channel, a quantity of internally generated data required for correct sound generation, such as the current value of attenuation (necessary to obtain the amplitude envelopes) and the current addresses reached in the external–ROM table scanning.

It should be pointed out that the memory relative to the data transmitted by the μP is duplicated.
In the first memory block, the data received by the μP interface are temporarily loaded. These data are then transferred to the second memory block when new table scanning is started. Data stored in this second block are actually read by the other circuits of the device.
Therefore, a ROM table can be scanned completely before beginng synthesis of a new sound, thus ensuring that the generated analog output signal has zero average value and allowing simple use of the Delta Modulation Coding technique, provided that proper values have been stored in the tables.

Digital sound data processing

When a programmable counter signals that the relevant sound channel needs a service routine, sending a start pulse and transmitting the channel address, all the data relative to that channel are read from the RAM.
These data are then processed by the three functional blocks that form this section.

The first block generates sequentially the two addresses of the bytes of the two ROM tables to be read, thus providing the readout of the normalized–amplitude value of the sample of the sound to be synthesized. 58 reading modes are available, to allow different combinations of table lengths and scanning modes.

The data read from the two tables are then processed by an interpolator, which performs the arithmetic operation of linear combination between the two bytes with a μP–programmable 4–bit coefficient. Thus, a smooth passage between the two tables is allowed, as explained in section 2.

The last block (the envelope generator) programs the value of the attenuation to be performed on the output pulses generated by the channel, thus determining the amplitude of the reconstructed sound signal. The passage between two different attenuation values can either be instantaneous or follow an exponential law according to the principle described in section 2 (see Figs. 9 and 10). The exponential variation ranges automatically over the 8 most significant bits. By programming a sound channel successively with different attenuation values, a smooth change in the amplitude of the generated waveforms can be obtained, as well as various amplitude envelopes.

Analog interface

This section performs the conversion of the digital data output by the digital–processing section of the device to an analog signal, which, after external filtering, will give rise to the actual sound signal.

The conversion is achieved by means of a D/A converter implemented with a R–2R ladder network controlled by MOS transistors. This network converts the 8–bit data obtained by the interpolator into a current signal.

A second 10–bit programmable R–2R ladder network attenuates the current signal generated by the D/A converter, under control of the envelope generator. The attenuation range is 0 to -60 dB.

The generated current signal is then delivered to one of the four analog output pins, by means of an analog–switch network controlled by the two bits sent by the μP. Of course, pulses coming from different channels can be delivered to the same output pin.

Four separate sound output pins are provided to allow the user, e.g., to separate solo and accompaniment, or bass and treble, to generate stereophonic or quadrophonic sound, to obtain sound movement, etc.

The device also generates a buffered reference voltage (2.5 V) to provide bias for the non–inverting input of the operational amplifiers of the external reconstruction filters.

A voltage multiplier is integrated to generate the high voltage level (~12 V) necessary for the correct operation of the analog–interface section.

The last two sections of the chip, i.e., the data processing and the analog interface sections, have a pipeline structure, which provide a constant delay on all the sound channels.

So, although a service routine needs only 2 μs to be performed, the time elapsed between the start of the routine and the end of the 1.5–μs pulse delivered to the analog output pin of the IC is 6.5 μs. The pipeline structure allows non–critical circuitry to be used in these sections. Moreover, external ROMs or EPROMs with 450–ns access time can be used to store the sound coding tables.

5. SYSTEM DESCRIPTION

The minimum system configuration is made up of a microprocessor that controls the Digital Sound Processor IC and of a 64k–bit memory, which contains the coded music information.

In a simple system, the microprocessor can also perform keyboard scanning and command detection (see Fig. 2).

In more sophisticated systems requiring, for instance, dynamic keyboard operation (i.e., sensing of the pressure with which the key is played), an additional microprocessor, or a dedicated circuit, can be devoted to this task, leaving the first microprocessor completely involved in controlling the DSP IC (see Fig. 13).

As a single IC can manage up to 16 sound channels, only one DSP component is usually required. However, in very large organs, where several keyboards and instrumental "registers" are available, more DSPs can be used, each with its own microprocessor and relevant memory.

An alternative approach can be the use of a single DSP IC and the extension of the memory size (see Fig. 14). As the DSP can address a maximum of 64k bits, additional 64k–bit memories can be connected in parallel to the first one and enabled, one at a time, by the μP using the Chip Select inputs of the memories.

Obviously, this system configuration is cheaper, but has a limitation in the number of the timbres that can be played at the same time. Only those timbres belonging to a single 64k–bit block will be addressed at the same time by the DSP.

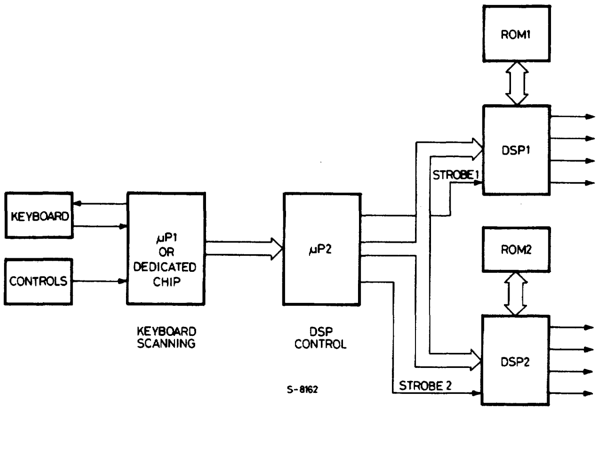


Fig. 13 - *System configuration with several DSPs*

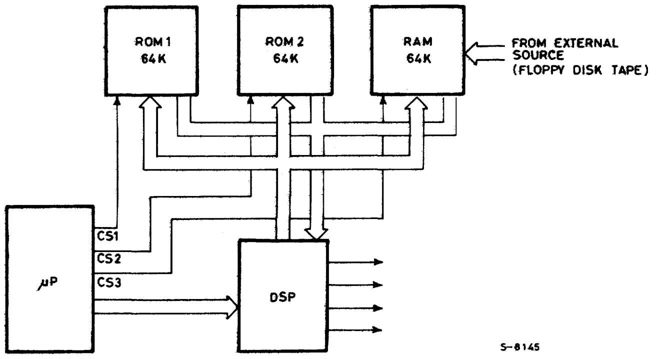


Fig. 14 - *System configuration with a single DSP addressing several 64k- ROMs/RAMs*

Of course, the external memories where the sampled values of music waveforms is stored, are generally ROM or EPROM.

However, static RAM can also be used for this purpose. In this case, typical of the music synthesizer, the instrumentalist has the possibility of creating new sounds that can be put in the memory and recalled on command.

6. CONCLUSIONS

A new system for music generation has been presented. Music is synthesized by means of a fully–digital μP–controlled sound processor IC.

The main advantages of digital sound processing are the elimination of analog filters and attenuators, which results in minimum component count and printed–circuit board area saving, no need of any kind of manual calibration, and excellent sound characteristic stability.

With this system, a polyphonic and polytimbric musical instrument can be easily implemented. Various system configurations are available, containing one or more DSP ICs, to meet the requirements of different–class instruments.

Acknowledgements

The DSP IC was developed on the basis of original specifications by GEM (General Electro Music), Mondaino, Italy. Special thanks are due to Mr. G. Bodini, Technical Director of GEM, for many suggestions given during the development of the device.

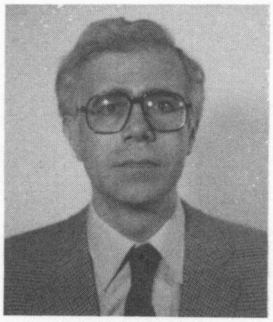
Thanks are also due to V. Daniele for his contribution to the system design, to E. Castaldo for help in logic design and for logic simulation, and to B. Brown, M. Defendi, I. Grisanti, and R. Pollastri for the layout of the chip.

REFERENCES

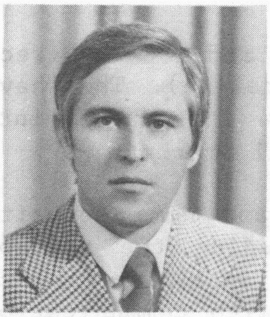
[1] James A. Moorer : “Signal Processing Aspects of Computer Music - A Survey”, *Computer Music Journal*, Box E, Menlo Park CA 94025, p. 4, February 1977

[2] G. Torelli, G. Caironi : “New Polyphonic Sound Generator Chip with Integrated Microprocessor–Programmable ADSR Envelope Shaper”, *IEEE Transactions on Consumer Electronics*, vol. CE-29, n. 3, p. 203, August 1983

BIOGRAPHIES

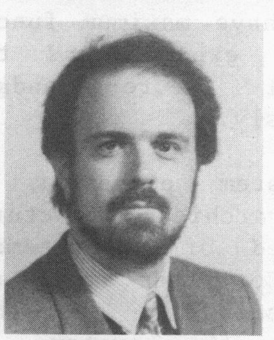


Giancarlo Caironi was born in Milan, Italy, in 1948.
He graduated in Industrial Electronics from the "Gian Giacomo Feltrinelli" Technical Institute of Milan in 1968.
At the end of 1969, he joined SGS, where he was, at first, engaged in the development of digital ICs. In 1975, he moved into the Digital Consumer ICs Application Laboratory.
Since 1978, he has been responsible for Product Marketing of MOS Consumer ICs.
He is a co–author and has presented 5 papers at the IEEE Consumer Conferences.



Antonio Battaiotto was born in S. Donà di Piave (Venice), Italy, in 1939.
He received a degree in Industrial Electronics from the "Beltrami" Technical Institute of Milan in 1965.
At the end of 1967 he joined SGS, where he was, at first, engaged in the Development of digital ICs. In 1972, he moved into the MOS ICs Application Laboratory.

Paolo Lupi was born in Castel di Lama (Ascoli Piceno), Italy, in 1952.
He graduated in Electronic Engineering from the University of Ancona in 1977.
In 1978 he joined SGS, where he is engaged as a design engineer in the field of MOS ICs for Consumer Market.



Guido Torelli was born in Rome, Italy, in 1949.
He graduated in Electronic Engineering in 1973 from the University of Pavia, where after graduating he worked one year as a researcher in the Institute of Electronics.
In 1974 he joined SGS, where he was engaged as a design engineer in MOS ICs Development Department.
He is now a specialist designer and acts as a Project Leader in the field of MOS ICs for Consumer Market.
He is a co-author of several published papers and holds two patents and some patent pendings.

Ermes Viani was born in Sassuolo (Modena), Italy, in 1952.
He graduated in Electronic Engineering from the University of Bologna in 1982.
In 1984 he joined SGS, where he is ingaged es a design engineer of MOS ICs for Consumer Market.