The system RT141 that will be described in part of a project named RT41. RT41 software is dedicated to the digital sound processor 41 realized at ENCM (Paris) by Giuseppe Di Giugno. The aim of RT41 is to manage in real time a flow of data, described by a MUSIC 5 type of language, in a score that al lows the possibility of controlling it during the execution with real time gestures.

1. INTRODUCTION

RT41 control system (P. Besançon, 1983-85) is part of a general software named RT41 the aim of which is to give the musician-composer a series of tools for the use of the 41 processor (G. Di Giugno, 1984 - B. Sapir, 1984).

The software that will be described in this paper concerns the real time control of a score that is to be performed on the 41 system. This system is conceived for the study of sounds. It provides for the possibility of using a symbolic language, MUSIC 5 type, to write input sources. The aim of this work was not to implement MUSIC 5 in its standard form on the 41, but rather to define a language that could either appear as a substructure or as an extension of MUSIC 5, depending on the points of view.

This language is organized to enable data real time control through gestures. It also provides for the digital recording of such gestural events and thus allows the editing of a gestural score that results from the intervention which occurs during the execution. In order to take the greatest possible advantage of the processor's possibilities and to achieve a system that is not too much to much in the operative modalities of the processor itself, we had to make some choices that allow both to exploit all of the 41 processor's effectively available resources and to meet the expectations of a user on a quality software (sympathy of the processor, symbolic input language, different kinds of musical operations etc....)

2. RT41 SYSTEM GENERAL CHARACTERISTICS

The fundamental function of the RT41 system is to send in execution the score divided into sections, with the possibility of checking the symbolic in good time within the response limits of the system by "session" we mean a period in which the synthesis algorithm, the memory functions and the control modalities remain unchanged. The section therefore defines the reference frame in which it is possible to intervene with input devices during
realization in this application see M. Martin, 1977-F. Amza, O. Herman, Z.P. Perjus, 1973. Here we want to explain how the execution control and the score modifications were organized from a "musical" point of view.

Figure 2 shows the parallelism between the two parts: background or PART CONTROL and foreground or M1 CONTROL.

The different paths indicate the different ways of controlling the parameters of the score. The first path represents the "normal" execution of the score, that is to say with pre-established values and times. The second path represents the total M1 control which allows to modify the time and the value of the parameters. The third path represents the M1 control upon the values that will be assigned to the parameters but at the time indicated in the score. The fourth path represents the M1 control on the time of the parameter modifications, but with pre-defined values in the score.

The particular attention given to the words "time" and "value" is explicitly connected with the definition of the two fundamental concepts of micro- and macro-events.

By score we mean the description of sound events, but also the definition of the controls that are up to the performer during the execution of the synthesis. This system is not yet complete, and therefore a description is given on its present state of development. For the M1 system, a full score includes four parts: an instrument reference concerning the Data Memory (DM) locations that are the input parameters of the synthesis algorithm, with their particular types of conversion (frequency in Hertz, amplitude in decibels or other); a reference to the M1 control that regards the link between the input devices and the controlled parameters but also the way by which such control is to be made; a reference to the printing format of the score on the screen during the performance; and the score itself. This score is made, for each one of the instruments, of a list of notes andpausesthat define the sound fragments and their temporal articulation. Before we examine the structures of data and the details of the performers controls, we want to classify what we need by "instrument".

When the event "end of note" happens before the event "beginning of the successive note", there is a pause between the two notes. On a programmer such as the 41, when turning off an "instrument" independently from the other ones is impossible, this pause needs to be explicitly produced. We have therefore considered the instrument as being the result of the association of the modules that realize the synthesis algorithm, with a timer that manages the event's
duration and with a modulus named ADDU. ADDU generates the segmented envelopes to be multiplied by the output signal of the moduli that built the algoritms. In this way, the instrument is not disac-
tivated, but its amplitude is simply set to zero.

On the contrary, when an event is to be executed, the amplitude varies in time following a simple sini-

gal envelope.

The horizontal reading above the parameters when they are synchronized at the action time of the

evelope; a vertical reading corresponds to an in-
dependent time evolution for each parameter, each one with its own action time and duration.

The system contains parameters that particular instrument has at its disposal, and must end with the END statement. According to the previous paragraph, a score will contain a list of the hierarchical events, that is, a list of notes that are respectively defined by the note and end statements. The notes are then subdivided into microevents, that is, a list of notes to be synchronously generated to the input parameters of the instrument, at the beginning of the micro-

event. As can be seen in figure 3, a score to be read either horizontally (a) or vertically (b).

As we have said before, the MM41 system is still at a working stage and we actually only have an inter-

mediate language called MM41 that enables us to write musical scores with a syntax of the MM41 5 kind. This is the only reason we have to choose the correct functional language of the MM41 system. All the following examples will be described with this lan-

guage.

The score is divided into sections, and each section is constituted by all of the single voices that are defined in a sequential way. A maximum of 16 voices are allowed in parallel. The definition of each voice must begin with the BIOS statement, that will fix the number of the instrument and the number of in-

put parameters that particular instrument has at its disposal, and must end with the END statement.

The reading above the parameters when they are synchronized at the action time of the envelope; a vertical reading corresponds to an independent time evolution for each parameter, each one with its own action time and duration.

This leads to the possibility of having a total synchronization between the different parameters or their independency. In this way, a traditional score can be executed with all the predefined and syn-

cronized parameters and, at the same time, a to-

tal or partial synchronization can be left among the parameters determined in real time. This led us to organize the score of an instrument according to the diagram described in figure 4, and in the
case of our data structure. The left part describes the "vertical" score of an instrument as a succession of macroevents (notes and pauses). We wish to point out that the microevents refer to a location of the control table and not to the real value which is to be assigned to the parameter. This allows us to manage data and time values that come from the score itself and also from the gestural input. Thus both vertical and horizontal readings are easily implemented.

During the execution of a score, when some parameters follow the succession of macroevents and others follow, in an asynchronous way, the order determined by gestural operations, it is often difficult to know the current state of the score. In order to help the performer, we are realizing a printing process on the terminal screen to show the evolution of the score in real time and to enable live printing of comments specified in the score by the TYP statement.

4. ENVELOPER

We have already defined the instrument as the association of a synthesis algorithm, a timer and a LIMIT. These last two modules of the processor are used to generate an amplitude envelope that has to be multiplied by the output signal of the synthesis.

fig. 3. Diagram of a score for the HZF41 system

fig. 4. Data structure of scores
Such an envelope is considered as being "normalized" in the way that the global duration (the total duration of all segments) is considered to be the minimum one that this envelope can assume for any note of the score. The "normalized" envelope can be divided into three fundamental parts: attack, sustain and decay (see in figure 5). Each one of them can be defined with more than one segment. While the sustain and decay parts are optional, the attack part must always be defined. Thus, for each note of the score, three coefficients will be calculated: one for the segments of the attack part, one for those of the sustain part and another one for the decay part. These coefficients allow the time expansion of the envelope, so that the total duration of the envelope will coincide with the duration of the note. In the same way, a silence is realized through the timer (tm0 and the LMI); they create a single segment the duration of which equals the duration of the pause, but with an amplitude set at zero. We will point out that during the calculation of these coefficients, there can be some error, or imprecision, due to the fact that the minimal duration of one segment cannot be inferior to one millisecond (the precision of timers). There are two statements in the language MUPPY to describe an envelope: DSV indicates the three locations of the Data Memory (DM), where the LMI is to be connected; and the number of segments that create the envelope; DSS defines each segment with two values (time, amplitude).

5. REAL TIME CONTROLS

From the previous discussion, it is clear that these different kinds of real time controls on the parameters through gestural input devices. After having analyzed the different possibilities, we kept those combinations that seemed to us more interesting from a musical point of view. In figures 6, 7, 8 we have schematized the most significant combinations and interactions between the time evolution of one parameter (microevent) or more parameters (macroevent) given by the score, and manual commands in real time. In these figures we have considered both evolutions as two independent functional samples in time. Each significant point is represented as a couple value (a) - time.

The figure 6 shows the combinations relative to a vertical reading that is to a macr0event control (we recall the two possible ways of reading a score). We will try to give you details about these figures:

6a: SN (score): F(i) represents the combination time-value for a parameter that is defined in the score.

6b: AD (Direct Access): G(a) represents the combination time-value relative to a gesture

6c: SN (Sample and Hold): the times defined in the score are associated to the values given in real time by the input command.

Fig. 6 - Score - Gesture relationship (Microevents)
7.a: MN (motion): represents the articulation between the notes as it is defined in the score.
7.b: RT (Real Time Input): represents the time at which the gesture occurs. These times can be considered individually and can also be used to indicate the action time of a macroevent, or two at a time, to define both starting and ending times (and therefore also the duration) of the events.
7.c: TR (Trigger): The input command determines the action time of the note, while its duration remains that defined in the score. In this case, we shall have some problems in designing a superposition of two macroevents. What it is impossible to do on the 41 with a single instrument, and we must be able to "kill" the first one with an arbitrary decay time (in order to save time) before starting the next one for execution.
7.d: TG (Trigger and Gate): two gestural actions will determine the starting time and the duration of a macroevent. (It can be for example, the attack and release of a key). In this case the articulation is entirely defined by the performer.

Figure 8 shows the combinations of another type
of horizontal reading that allows one to vary the
"metronome", that is the duration of the macroevents
which, in this case, means both notes and pauses.
To help in the comprehension of this figure we re-
presented differently the two types of macroevents:
the small arrows represent the pauses, while the
big ones represent the notes. We suppose, besides,
that notes and pauses are alternated equally
spaced in time. This particular case is only
used to give a clearer illustration of what
happens to the durations when the im-
put devices are used to modify the metronome:

S.a: MM (Score): represents the successive macro-
events with the duration defined in the score.

S.b: RTI (Real Time Input as metronome factor): re-
presents the evolution of the metronome as it is
defined in real time by the performer/con-
ductor.

S.c: MTC (Total Metronome Control): the durations of
both notes and silences are modified ac-
cording to the metronome factor (for rallen-
tandi and sostenendi).

S.d: MNC (Note Metronome control): only the durations
of the notes are modified in real time.

the durations of the pauses remain as defined in
the score.

S.e: ME ( Silence Metronome control): in this case
only the durations of pauses are modified propor-
tionally to the metronome factor, and the
durations remain as defined in the score.

For the moment we only achieved the first type of
controls (described in figure 6), but the other
ones are almost completed. We fixed the data struc-
ture, as shown in figure 9, to facilitate the macro-
event controls. This structure comes from the
fact that in a gestural intervention, there are
two types of extra temporal information: the iden-
tifier of the input device used, and the data trans-
mitted by the gesture itself.

Essentially, the identifier allows one to extract
from the RTI control table (VARRTI), the type of
control associated to the input device (MM, AD, SQ
...), and from the pointer table (POINTERI), a point-
ter to the list of parameters that are controlled
by the previous input device. The list of param-
ters will give the addresses of the locations, whe-
re the data are to be loaded after having been

Fig. 9 - RTI data structure
suitably converted, as specified in the instrument manual. The type of control indicates which of these locations belong to the Data Memory (DM) or to the STOP. The STOP circuit is a zero of the FDM memory where the input data wait before being treated, in the case of a sample and hold (SH) control.

6. EXAMPLE OF A SCONE DEFINED WITH THE LANGUAGE NOTES

We will comment on an example of a score (figure 10) to be executed by an FM instrument represented in figure 11.

SET 1:
define the section, that is to say, the period in which neither the configuration of the 4i nor that of the control system will change. The section fixes the limits of the possible real time interventions. The value 1 of this statement establishes that in this section there will only be one instr

EXT 3:
the instrument which will then be defined in the number one and has three input parameters.

ENV 0 10 13 5 3 4
This statement defines the envelope associated with the FM algorithm. The envelope generator (ENV) that creates the segments of the envelope is connected with the locations 9 and 10 of the Data Memory, the locations where the data relative to the segments (current, increment and final values) will be loaded. The envelope will consist of five segments: segment 3 initiates the attack and segment 4 initiates the decay part of the envelope.

fig. 10 - Example of a score for the FM instrument

fig. 11 - Diagram of a FM instrument

SET 0 10 0000 5 8000 55700 10 750 20 7
We have here the description of all the five seg-
ments as shown in figure 5. The envelope is consi-
dered to be normalized at its minimal possible du-
eration, in this case 100 milliseconds [10(1)x5x+10] +20. Then for each rate, the duration will be calcu-
lated on the base of this envelope. An envelope, therefore, can only be expanded and never reduced in time.

LOC 6 4:
This statement fixes the address of the Data Mem-
ory locations that need to be updated. The carrier parameter corresponds to location number 6, the modulation parameter to location number 4, and the deviation parameter to location number 4.

At this point, the score can be read with vertical-
ly and horizontally. The columns define the evolu-
tion of the single parameters (microwaves), while the lines characterize the global sound events (macroevents).
This statement determines which type of conversion is to be used on the data before sending it to the locations listed above. In our case, all the data will be converted from frequencies in hertz into sampling increments. The instrumentality header of the score being achieved, we will now see the practical one:

PI: PI T1;

This statement defines which input devices are to be activated to control the synthesized parameters. PI indicates that the potentiometer input one will pilot the carrier frequency. PI or potentiometer number two will do the same on the modulation frequency while the key number one, need T1 will control the deviation.

AD at D2;

This statement determines the different control types respectively assigned to the devices listed in the previous statement. The first potentiometer, will then act in Direct Access, the second one in Sample and Hold, while the key T1 will be used as a sequencer to sort the list of values of the deviation.

SNG 40,000 /10,000 =

This statement is used to limit the input device range of action. PI for example, will send data to the carrier frequency at location number six in a range of 40 to 6000 Hertz. And here ends the gestural header of the score. We will now consider the effective description of the musical score.

The NO statement is divided into two parts. The first one (three data) is related to the envelope: the global duration (in seconds) of the note, and the duration (in percent) of the attack and sustain parts of the envelope. This enables us to adjust and randomize the envelope for each note according to the duration of the note. The second part of the NO statement concerns the values that will be assigned to the parameters in the synthesis algorithm. For example, 80 as the first NO is concerned, the duration of the note will be four seconds, with twenty percent of this duration (80 sec.) reserved for the attack, forty percent of it for the sustain (16 sec.) and the remaining forty percent of the total duration is therefore left for the decay part of the envelope.

The carrier and modulation frequencies are fixed at 1000 Hz, the spatial in 6. When the symbol "*" occurs, it means that the data are defined as previously.

In case of an AD control, the list of values relative to the parameter is not memorized in the score. For our example, there will be 20 list foresee for the parameter of location six (carrier frequency), because the data will be given in real time through PI. In the same way, there will not be any memorized list for a SK Control. But the data sent from the input device (in our example PI), will be assigned to the parameter at the time pre-determined by the score. This preserves the continuity of sound without distorting a note by a sudden change of the modulation parameter, which is not the case for example when PI is moved.

Finally, with the statement RI, we define in seconds the duration of the pause.

The PM statement indicates the end of the current instrument's score, while the RO statement indicates the end of the section.

A score defined in such a way, can then be compiled and sent for execution.

7. CONCLUSION

The project MITH started with the idea of realizing a flexible and efficient system, to be used by musicians who did not have any particular technical knowledge in the digital area. We also wanted to create a friendly environment that enabled the artist to work with the 41 system, without being linked too much to a specific and restrictive application or to the physical structure of the digital processor. This system could be used for different applications.

Pedagogy: The interaction between scores and gestures allows for a direct experimentation with digital synthesis techniques.

Organization: The synthesis algorithm: the sonority of the synthesized sound, enables one to test and adjust "normally" the input parameters for a better sound definition.

Creation or re-organization of scores: the recording of gustural actions can be used and recombined with the initial score.

Score interpretation: The input devices can also assume limited control: it is therefore possible, during a live performance, to leave some parameters undefined, thus being able to control the system in real time.

The choice of MUSIN 5 as a reference input language constitutes a cultural base that determines a familiar environment for the musician, even though MUSIN 5 was not conceived for real time applications. The possibility of interacting with real time actions, has broadened the possibilities of using MUSIN 5 type of language.

The system is already operative, and enables one to execute a maximum of 30 simultaneous voices and gives the possibility of modifying their own evolution by means of different kinds of musical control. Moreover, it is possible to make different kinds of conversions in parameters values: at present we have implemented the most useful conversions, while we are still working on the creation of more sophi-
As we have previously said, the different modes of control permit one to operate on single parameters of the "instruments", but it may be useful to point out that it is always possible to gather more parameters in a single manual command. In such way, it is possible to operate at a higher level than with the single parameter: this can guarantee a greater correlation in a group of preselected parameters, just like it occurs in acoustical phenomena.

The operating system that supports all of these features, and replaces, during the execution, the MIDI P571's operating system, also allows a real time printing process of both MIDI and gestural scores; recording process of the gestures, and other processes such as loading a section, or short section processors, all activated at the terminal by the performer.

In conclusion, the use of this system by some musicians, for the realization of projects, confirmed its validity and provided useful suggestions for its future development. With regard to this, we wish to underline the modular construction of the program, that gives a greater flexibility for the maintenance and the expansion of the software, and the use of a methodology based on the P571 Beta (O. De Puli, G. Haas, 1982) that facilitated its conception and development.

REFERENCES


Rapporti istituto C.S.C., Università di Padova.


ca musicale, pp. 24-35, Ancona.
