CODING AND DRAWING MUSIC SCORES

by

C. APERGHIS-TRAMONI

Institut de programmation

Paris

Abstract

The drawing of scores is a hudge problem if it is to be done in the rules. But a score may be readable even if the perfection has not been reached. The problem is to decide what dificulties will be kept under silence in order to simplify the job, and to explain what are the solutions choosen to solve the remainding problems.

That is what we are presenting in our paper.

We now dispose of enough information to establish a table that gives the number of increments (the ordinate of the note on the stave) acording with the code number of the note and an eventual alteration.

Number of the key	Note	Ordinate	Alteration
12 n	Do_n	7 n	None
12 n + 1	$Do\sharp_n$	7 n	$\sharp$
	$Re\flat_n$	7 n + 1	$\flat$
12 n + 2	Re_n	7 n + 1	None
12 n + 3	$Re\sharp_n$	7 n + 1	$\sharp$
	$Mi\flat_n$	7 n + 2	$\flat$
12 n + 4	Mi_n	7 n + 2	None
12 n + 5	Fa_n	7 n + 3	None
12 n + 6	$Fa\sharp_n$	7 n + 3	$\sharp$
	$Sol\flat_n$	7 n + 4	$\flat$
12 n + 7	Sol_n	7 n + 4	None
12 n + 8	$Sol\sharp_n$	7 n + 4	$\sharp$
	$La\flat_n$	7 n + 5	$\flat$
12 n + 9	La_n	7 n + 5	None
12 n + 10	$La\sharp_n$	7 n + 5	$\sharp$
	$Si\flat_n$	7 n + 6	$\flat$
12 n + 11	Si_n	7 n + 6	None

However, it is well known that in the piano game (the one defined by *Jean Sébastien Bach*) a note altered by a sharp is confused with the one, natural or altered by a flat, that follows. That means:

Do♯ = *Re*♭
Re♯ = *Mi*♭
Mi♯ = *Fa*
Fa♯ = *Sol*♭
Sol♯ = *La*♭
La♯ = *Si*♭
Si♯ = *Do*

That is not to make easier a problem wich is already prety hard. Indeed, if we consider the note number '27' it is represented in our notation by the pair < 2 , 3 > that means that we must consider the third note of the second octavius.

That third note might be writen either *Re*♯ or *Mi*♭ depending of the context of the piece.

Re♯ is an altered note whose ordinate is:

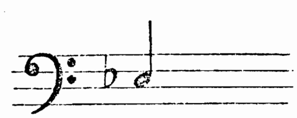
$$(2 \times 7) + 2 = 16 .$$

That means:

Mi♭ is also an altered note but its ordinate is:

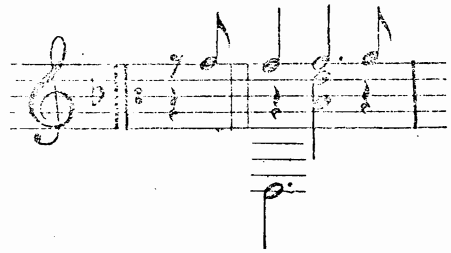
$$(2 \times 7) + 3 = 17 .$$

That means:



Such a choice is guided, as we will see later, by the key signature.

Acording what have been presented up to now, we can give an example of encoding for a given piece of music, wich is the same that has been presented on page one.



Rythm: 3/4 - Keysign: ♭,1 •

< 31 , 0 , 8 > • < 31 , 0 , 16 > < 21 , 0 , 32 > •

< 31 , 0 , 24 > < 29 , 0 , 16 > < 26 , 0 , 16 > •

< 32 , 0 , 8 > •

Each triplet containing the following information:

< 0 , A , L >

O : Ordinate of the note on the stave.

If O = 99 we must draw a rest.

A : Alteration

0 - None.

1 - Sharp ♯ .

2 - Flat ♭ .

3 - Natural ♮ .

L : Lengh of the note acording with the following table:

V	Symbol	Designation	V x 16
6	𝅝.	Doted semi breve	96
4	𝅝	Semi breve	64
3	𝅗𝅥.	Doted half note	48
2	𝅗𝅥	Half note	32
1.5	𝅘𝅥.	Doted crotchet	24
1	𝅘𝅥	Crotchet	16
0.75	𝅘𝅥𝅮.	Doted quaver	12
0.5	𝅘𝅥𝅮	Quaver	8
0.375	𝅘𝅥𝅯.	Doted semi quaver	6
0.25	𝅘𝅥𝅯	Semi quaver	4
0.1875	𝅘𝅥𝅰.	Doted demi semi quaver	3
0.125	𝅘𝅥𝅰	Demi semi quaver	2
0.09375	𝅘𝅥𝅱.	Doted hemi demi semi quaver	1.5
0.0625	𝅘𝅥𝅱	Hemi demi semi quaver	1

We now dispose of two structures, the first one encodes the music itself; the second describes the score. Our problem is then to obtain the score encoding (result) starting with the data given in the music encoding.

To compute that transformation, some data must be given to the program: the rythm and the key-signature.

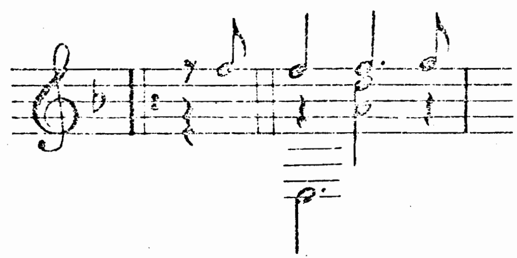
We think that the key-signature might be computed, but such a calculus is a musicological problem that we decided not to solve in our application.

The score will be drawn in two keys (*Fa* and *Sol*)

Algorithm.

The best way to realize the transformation we did present is to handle data using the equivalence table whose content has been given on page four.

An intermediary file is necessary to perform that transformation. It will be computed from the music encoding and will give for each note its duration. Let us consider the same piece as former:



The music encoding is, as seen on the first page:

< 0 >< 53 , 1 > · < 5 >< 53 , 0 >< 53 , 1 >< 26 , 1 > ·
< 15 >< 53 , 0 >< 53 , 1 >< 50 , 1 >< 45 , 1 > ·
< 25 >< 50 , 0 >< 45 , 0 > · < 30 >< 53 , 0 >< 55 , 1 > ·
< 35 >< 26 , 0 >< 55 , 0 > ·

And the new file that we call the duration file is:

< 53 , 5 > · < 53 , 10 >< 26 , 30 > · < 53 , 15 >< 50 , 10 >
< 45 , 10 > · < 55 , 5 > ·

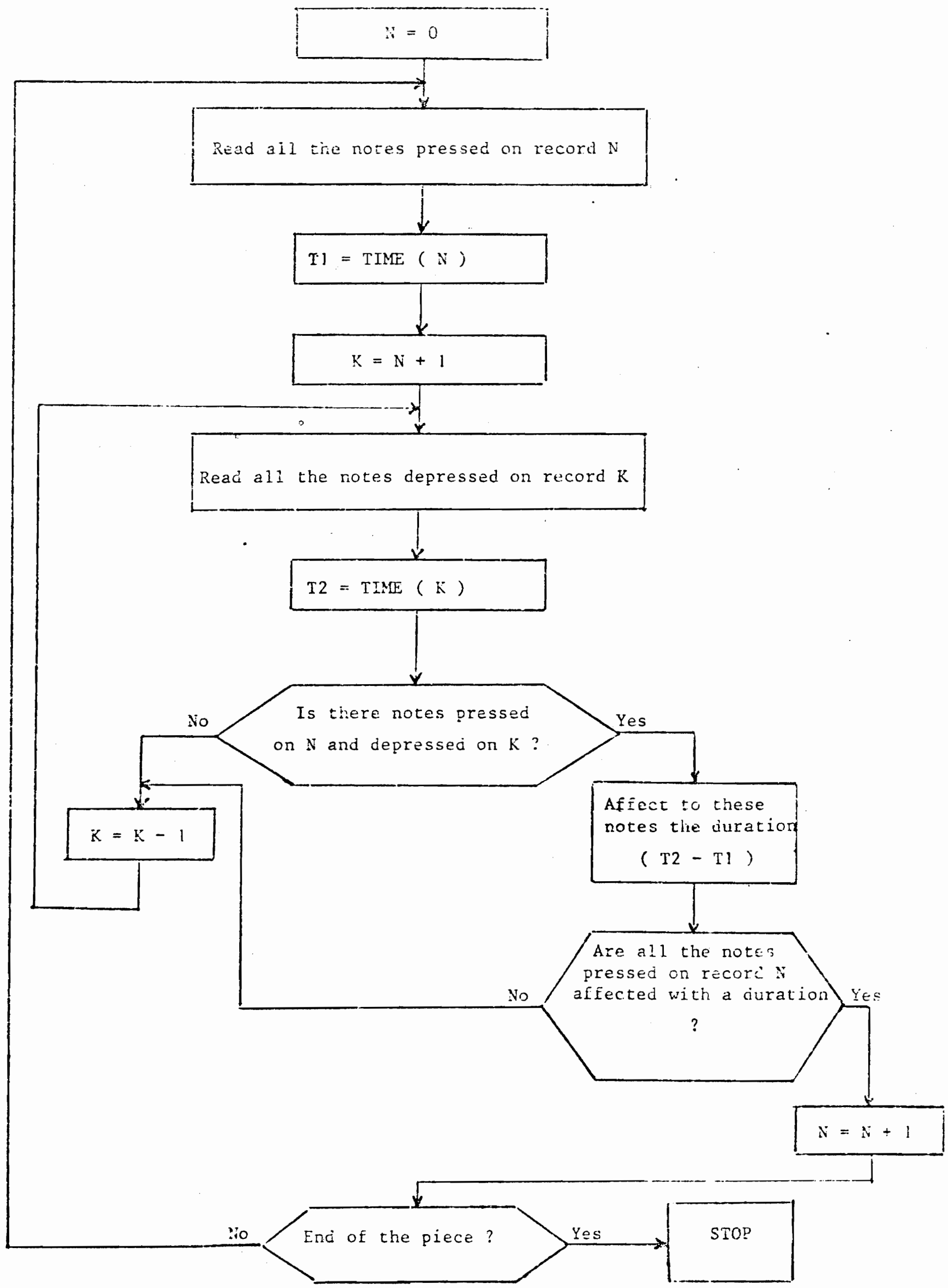
The meaning of each doublet of that new encoding is:

< NN , D >

D — Duration.

NN — Note number.

The algorithm used is the following:



The next job is to compute the lengh of the note using its duration. We must know, to do that calculus, the duration of the unit (the crotchet) wich is in our example " 10 " . The lengh will be then:

$$\frac{\text{DURATION}}{10}$$

And in order to handel integers, we will multiply the result by " 16 " .

$$L = \frac{\text{DURATION}}{10} \times 16$$

In the general case:

$$L = \frac{\text{NOTE DURATION}}{\text{CROTCHET DURATION}} \times 16$$

We so obtain the values given in the table on page seven.

We are now close to the end, we must create the score file using the file we just create. The table given on page four is coded in order to be easilly stored and checked.

i	0	1	2	3	4	5	6	7	8	9	10	11
∅	0	■	1	■	2	3	■	4	■	5	■	6
♯	■	0	■	1	■	■	3	■	4	■	5	■
♭	■	1	■	2	■	■	4	■	5	■	6	■

The black squares represent the value -1 that is easy to test.

The operations that are to be performed are:

- The number of the note is transformed into a doublet $< P , Q >$ using the formula

$$N = 12 \times P + Q \quad 0 \leqslant Q < 12$$

P : Octavius number.

Q : Note number.

- Check the element $K = TAB (\emptyset , Q)$
  - $K > 0 \rightarrow$ The note is natural

  If no key-signature the ordinate is:

  $$ORD = (7 \times P) + K$$

  If key-signature draw if necessary a natural on the ordinate

  $$ORD = (7 \times P) + K$$

  - $K < 0 \rightarrow$ The note is altered and we have to chose between sharp and flat.

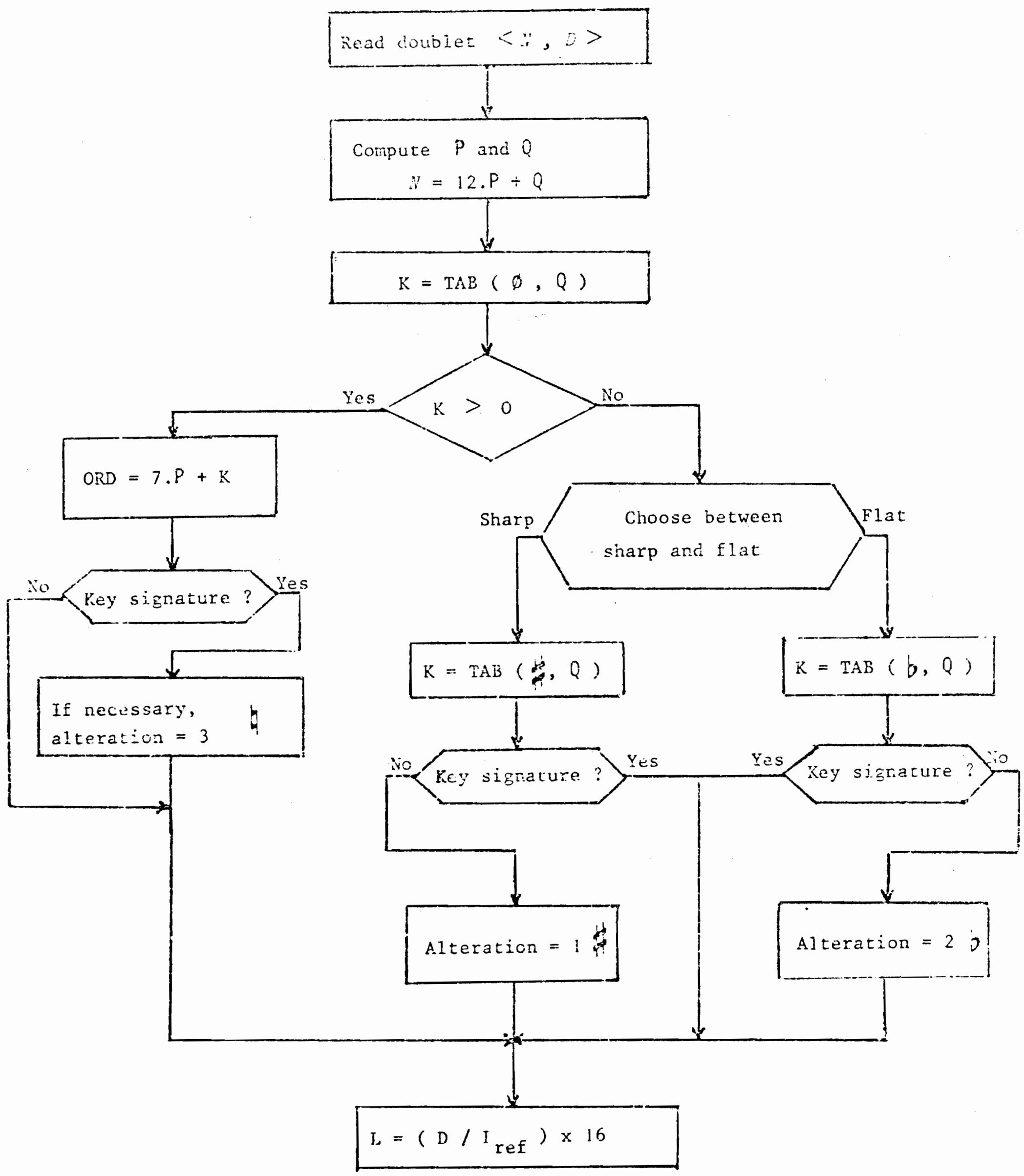
  If sharp $K = TAB (\sharp , Q)$

  If flat $K = TAB (\flat , Q)$

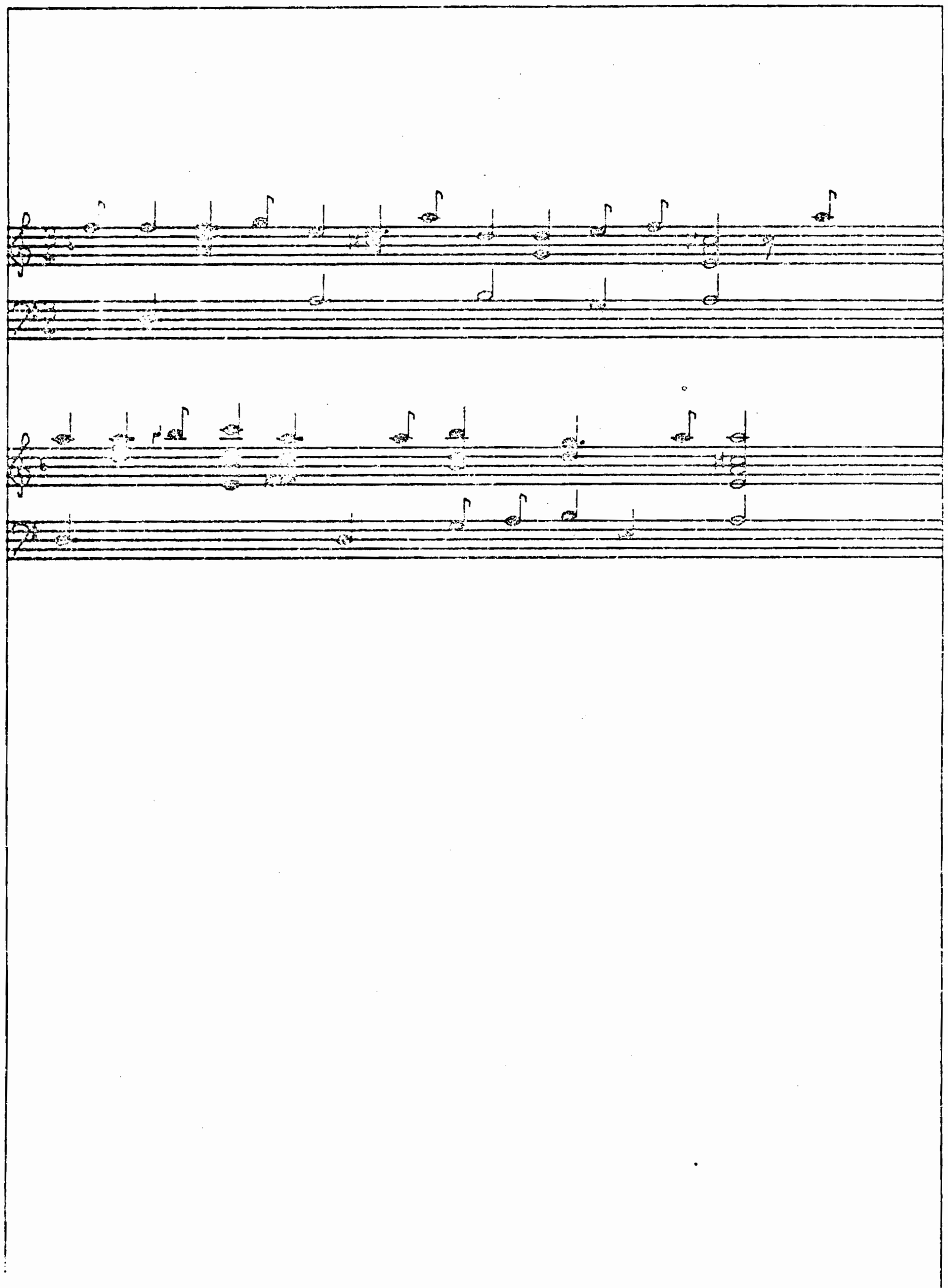
  And the ordinate will be in all the cases:

  $$ORD = (7 \times P) + K$$

The algorithm that describes these operations for a doublet is:


Read doublet < N , D >
Compute P and Q
N = 12.P + Q
K = TAB (Ø , Q)
K > 0
Yes
No
ORD = 7.P + K
Choose between sharp and flat
Sharp
Flat
Key signature ?
No
Yes
If necessary, alteration = 3 ♮
K = TAB (♯, Q)
K = TAB (♭, Q)
Key signature ?
No
Yes
Key signature ?
Yes
No
Alteration = 1 ♯
Alteration = 2 ♭
L = (D / I ref) x 16

Here is finally a score drawn using the processes we did describe:



Conclusion.

We just present in that paper some principles. A bunch of problems are kept under silence because of the hu ge complexity of the subject we tried to expose. Some of them are solved, for instance the detection of the rests; others are under investigation, for instance the direction of the queues of the notes.

Nevertheless, if we look at the result, we can recognize that we dispose of a complete system able to get music from a keyboard and to draw it. We are still far from the perfection, but our research is continuing and we hope, in a near future, to have better results that will make our scores more readable and more complete.

Bibliography.

C. APERGHIS-TRAMONI.

Une structure d'acquisition automatique d'information musicale.
Cycle de conférences ERATTO-CNRS Mars 1977.

C. APERGHIS-TRAMONI.

Système d'acquisition et de restitution de musique.
Thèse de troisième cycle. Institut de programmation. Paris Mai 1977.

A. BERNARDY.

Microprogrammed multiprocessor graphic controler.
Micro 6, Sigmicro ACM. Maryland 1973

H. CHARNASSE - H. DUCASSE.

De l'emploi de l'ordinateur pour la transcription des tablatures.
Revue de musicologie, Paris. Société française de musicologie 1971 N° 2.

R. F. ERIKSON.

The use of computers in music. A state of art repport.
I.B.M. System research institute; New York 1970-71.

P. GREUSSAY.

Modèle de description symbolique en analyse musicale.
Thèse de troisième cycle. Université de Paris VII 1973.

P. H. KNOWLTON.

Capture and display of keyboard music.
Datamation Mai 1972.

M. V. MATTHEWS.

The technology of computer music.
M.I.T. Press 1968.

R. C. PINKERTON.
Information, theory and melody.
Sci. American 194, February 1956.

J. TENNEY.
Sound generation by means of a digital computer.
Journal of music theory, Vol 7 N° 2 Spring 1963.