

MELODY-GENERATING MICROCOMPUTER

The present invention relates to a microcomputer
5 having a melody generating function and more particularly
to its melody play control.

In known microcomputers having a melody generating
function, a melody generator mainly comprises only an
interval generating unit and its control is effected in
10 accordance with a control program for melody generating
purposes. This program is stored in a memory storing
programs for controlling the microcomputer. Thus, the
known microcomputer of this type has the following
problems.

15

(1) Since the generation of melodies requires a great
deal of control programs, the desired controls including
the control of the microcomputer itself requires a memory
having such large capacity as to store a great deal of
20 programs. This leads to an increase in the cost of the
system.

(2) During the playing of a melody, it is difficult to
perform any other control and thus it is impossible to
25 promptly respond to a command applied externally of the
system. In other words, a long melody cannot be played.

The present invention has been made with a view to overcoming the foregoing deficiencies in the prior art and it is an object of the present invention to provide a microcomputer which is inexpensive and capable of promptly responding to any external command even during the playing of a melody.

In accordance with one aspect of the present invention, there is thus provided a microcomputer including a CPU for processing various data, a melody generator for generating a melody signal and a data bus for transferring data which are to be inputted and outputted from the CPU.

The melody generator includes a melody data memory for storing melody data including at least scale data, note duration data and end data indicative of the presence of the end of a melody played, an interval generating unit for generating an interval signal in accordance with the scale data in the melody data, a note duration generating unit for generating a note duration signal in accordance with the note duration data in the melody data, and an addressing unit for receiving an initial address signal from the CPU to supply it as a reading address of the melody data memory and for outputting the addresses succeeding the initial address signal as address signals of the melody data memory at the timing of corresponding

1 note duration signals.

Initially, melody data is read from the melody data memory in accordance with the initial address signal from the CPU so that the interval generating unit generates an
5 interval signal in accordance with the scale data in the melody data thereby performing a playing operation. Also, in accordance with the note duration data in the melody data the note duration generating unit generates a note
10 duration signal so that in accordance with the note duration signal the addressing unit updates the address thus successively reading the melody data from the melody data memory and the interval generating unit generates an interval signal in accordance with the scale data in the melody data read, thereby effecting the playing operation.

15 Also, the melody generator includes, in addition to the above-mentioned construction, an end of melody signal generating unit for generating an end of melody signal in accordance with end data from the melody data memory and a
note duration signal from the note duration generation
20 unit, an interrupt generating unit for applying an end of melody interrupt signal to the CPU in response to the end of melody signal, and a control unit for receiving from the CPU at least a melody control signal for commanding the start or stop of playing by the melody generator and
25 for receiving the end of melody signal thereby starting the operation of the units forming the melody generator in

1 response to the application of a start of play command
signal and stopping the operation of these units in
response to the application of a stop of play command
signal and the end of melody signal.

5 The end of melody signal generating unit generates an
end of melody signal in accordance with the end data from
the melody data memory and the note duration signal from
the note duration generating unit, and the interrupt
generating unit sends an end of melody interrupt signal
10 to the CPU in response to the application of the end of
melody signal. The control unit starts the operation of
the units forming the melody generator in response to the
application of a start of play command signal from the CPU
and stops the operation of these units in response to the
15 application of a stop of play command signal and the end
of melody signal.

Also, the addressing unit includes an address
register into which an initial address signal is set from
the CPU and an address counter into which the initial
20 address signal is set as a preset value which is counted
up (or counted down) each time a signal according to a
note duration signal is applied. The initial address
signal from the CPU is set in the address register and the
initial address signal in the address register is set as a
25 preset value in the address counter whereby each time a
signal according to a note duration signal is applied the

1 preset value is counted up (or counted down) to update
the reading address of the melody data memory.

On the other hand, the control unit includes a melody
control register for receiving and storing at least a
5 melody control signal for commanding the start or stop of
playing by the melody generator and an on/off control
circuit for receiving an end of melody signal to start
the operation of the units forming the melody generator
in response to the application of a start of play command
10 signal and to stop the operation of the units in response
to a stop of play command signal and the end of melody
signal. The melody control register receives for storage
a melody control signal for commanding the start or stop
of playing by the melody generator from the CPU, and the
15 on/off control circuit starts the operation of the units
forming the melody generator in response to the
application of a start of play command signal and stops
the operation of the units in response to the application
of a stop of play command signal.

20 Also, when an interrupt signal is applied, the CPU
sends to the address register an address signal
corresponding to the leading address of a plurality of
preset melodies. When this occurs, the plurality of
melodies are played successively.

25 It is to be noted that with the microcomputer
according to the present invention, the CPU, the data bus

1 and all of the above-mentioned units of the melody generator are formed on one chip.

Also, either one or both of the melody control register and the address storage register may be formed
5 as storages arranged within the same address space as the memories controlled by the CPU.

Also, the address register may be constructed to include an FIFO storage so that a plurality of addresses for the melody data memory, corresponding to the order in
10 which melodies are to be played, are stored in the address register whereby each time an end of melody signal is applied the corresponding address according to the order is set as a preset value in the address counter. In this case, each time an end of melody signal
15 is applied the corresponding address according to the order is set as a preset value in the address counter. Thus, while the plurality of melodies are played successively in sequence, during the interval the CPU is not required to receive the addresses in correspondence to
20 the melodies to be played and the burden of the CPU is reduced.

Further, the microcomputer according to the present invention includes a data selector for receiving the melody data from the melody data memory and the melody
25 data from the data bus so as to select and output one or the other of the two data in accordance with a command

1 from the CPU, and a temporary storage device for
receiving the melody data from the data selector to
temporarily store the data whereby scale data is applied
to the scale generating unit and note duration data is
5 applied to the note duration generating unit.

The data selector receives the melody data from the
melody data memory and the melody data from the data so as
to select and output one or the other of the two in
response to a command from the CPU, and the temporary
10 storage device applies scale data to the interval
generating unit and note duration data to the note
duration generating unit. In this way, the melody data
from the melody data memory and the melody data from the
data bus can be utilized suitably so as to play the
15 melodies having a high degree of freedom.

The above and other objects as well as advantageous
features of the invention will become more clear from the
following description taken in conjunction with the
accompanying drawings.

20 Reference will now be made, by way of example, to the
accompanying drawings, in which:

Fig. 1 is a block diagram showing the hardware
construction of a microcomputer according to an embodiment
of the present invention.

25 Fig. 2 is a block diagram showing details of the
melody generator of Fig. 1.

1 Fig. 3 is a schematic diagram showing a format of melody data.

 Fig. 4 is a block diagram showing the construction of the control circuit.

5 Figs. 5 and 6 are respectively a flow chart and timing chart showing the operation in a single-melody play mode.

 Figs. 7 and 8 are respectively a flow chart and timing chart showing the operation in a continuous play mode.

10 Figs. 9 and 10 are respectively a flow chart and timing chart showing the operation in a forced play mode.

 Fig. 11 is a block diagram showing an exemplary construction of the address register.

15 Fig. 12 is a block diagram showing another embodiment of the melody generator.

 Fig. 1 is a block diagram showing the hardware construction of a microcomputer according to an embodiment of the present invention. A microcomputer 10 includes a CPU 12, an ROM 14, an oscillator circuit 16, a system reset control circuit 18, an RAM 20, a melody generator 22, an interrupt generator 24, an I/O port 26, an O port 28, a data bus 30, an address bus 32, an interrupt control line 34 and a selective control line 36. It is to

1 be noted that the microcomputer 10 includes other
constructions, e.g., LCD drivers, clock timer, stopwatch
timer and analog comparators and their explanation is
omitted since they have no direct bearing on the present
5 invention.

Fig. 2 is a block diagram showing details of the
melody generator 22. A melody address storage register
102 (hereinafter referred to as an address register)
stores a starting address of the melodies stored in a
10 melody data memory 106 (hereinafter referred to as a
melody ROM). In other words, the starting address of the
melodies is written or read in accordance with a command
from the CPU 12 through the data bus 30.

A melody address counter 104 (hereinafter referred to
15 as an address counter) is a counter for designating an
address of the melody ROM 106 through the address bus 105
and it has a function of counting address incremental
signals 117 to advance the address of the melody ROM 106
by one address in response to each counting. Also, the
20 melody starting address set in the address register 102 is
written into the address counter 104 through a melody
address bus 103 as occasion demands. Where the melody ROM
106 is a memory of the type so that the form of data
written into each address contains the address to be
25 executed next, the address counter 104 is not necessary.

Of the melody information to be generated, at least

1 the scale, note duration and end of melody information are stored in the melody ROM 106.

Fig. 3 is a schematic diagram showing the data format of one note stored in the melody ROM 106. As shown in the Figure, the data format includes end data, interval address data, note duration data and attack data. The note duration data is delivered to a note duration data bus 115 and the scale address data is delivered to a scale data bus 121.

10 When an address incremental signal 117 is received from a note duration generating circuit 116 after the end data from the melody ROM 106 has been received, an end of melody signal generating circuit 108 generates an end of melody signal 109 on the assumption that the playing of the corresponding melody has been completed and the signal 109 is supplied to a control circuit 110 and a melody interrupt generating circuit 118.

Fig. 4 is a block diagram showing the construction of the control circuit 110. The control circuit 110 includes a melody control register 140 for storing various data, e.g., melody start and stop information and an on/off control circuit 142 which is subjected to on/off control in response to an end of melody signal and "1" or "0" data of MELC register of the melody central register 140. The melody control register 140 is connected to the data bus 30 and its writing and reading are controlled by

1 commands from the CPU 12. The output from an MELC
register of the melody control register 140 is applied,
along with an end of melody signal, to the on/off control
circuit 142 which in turn generates an on control signal
5 or off control signal. Also, a tempo control signal is
generated from a TEMP register and playing speed control
signals are generated from a CLKC0 register and a CLKC1
register.

A frequency divider circuit 112 receives the playing
10 speed control signals from the control circuit 110 and
the clock signals applied through a gate circuit 120 and
the clock signal frequency is divided in accordance with
the playing speed control signals, thereby generating
frequency divided signals. It is to be noted that an
15 on/off control signal is applied as a gate control signal
to the gate circuit 120 from the control circuit 110 so
that when the gate control signal is off, no clock
signals are supplied to the frequency divider circuit 112
and thus no playing is effected.

20 A tempo generating circuit 114 receives the frequency
divided output from the frequency divider circuit 112 and
the tempo control signal from the control circuit 110 so
that clock signals of the frequency corresponding to the
tempo control signal are generated and applied to the note
25 duration generating circuit 116.

The note duration generating circuit 116 receives the

1 note data from the note duration data bus 115 and clock signals so as to generate a note duration signal.

In other words, each time the time interval corresponding to the note data expires an address incremental signal 117
5 is applied to the address counter 104 and the end of melody signal generating circuit 108. The note duration generating circuit 116 comprises a presetable counter which receives as a set signal the note duration data from the note duration data bus 115 whereby it counts up each
10 time a clock signal is applied from the tempo generating circuit 114 and an address incremental signal 117 is generated upon reaching the preset value.

After it has been detected that the end data of the melody data is set to "1", the end of melody signal
15 generating circuit 108 applies the previous mentioned end of melody signal 109 to a melody interrupt generating circuit 118 and the control circuit 110 at the timing of the application of an address incremental signal 117. The melody interrupt generating circuit 118 receives the end
20 of melody signal 109 generated from the end of melody signal generating circuit 108 at the end of each melody to thereby apply an end of melody interrupt signal 119 to the interrupt generator 24.

The scale address data stored in the melody ROM 106
25 is applied as an address signal to a scale ROM 122 through the scale data bus 121 so that the various scales

1 corresponding to the address signal are read from it and
are applied to an interval generating circuit 126 through
an interval data bus 125. The interval generating circuit
126 comprises a counter presettable to a given count
5 (presettable counter) and it receives as a preset value
the interval data from the interval data bus 125. A
frequency multiplier circuit 124 receives the clock
signals through the gate circuit 120 so that the clock
signals are subjected to frequency multiplication and
10 applied to the interval generating circuit 126. A melody
output control circuit 128 amplifies the output of the
interval generating circuit 126 or the interval output
and generates it as a melody output 129. Then, the melody
output 129 causes a musical instrument (not shown) to
15 perform a playing operation.

The microcomputer according to the present embodiment
is constructed as above described and it is capable of
performing for example the following melody controls with
the reduced load on the CPU 12.

20 (a) A single-melody play mode (a continuing melody is
automatically played once).

(b) A continuous play mode (a continuing melody is
automatically played repeatedly or a plurality of
continuing melodies are combined to thereby play a single
25 melody).

(c) A forced play mode (a forced melody change or

1 stop is effected during the playing).

(a) The single-melody play mode:

Fig. 5 is a flow chart showing the operation of the CPU 12 in this play mode. Here, it is assumed that in order to form a continuing melody, the information (see Fig. 3) of the notes forming the melody have been written in the successive addresses in a part of the melody ROM 106) (it is assumed that the same applies to the play modes which will be described later).

10

(1) In accordance with a command from the CPU 12, the melody starting address of the desired melody to be played in the melody ROM 106 (the address in which is written the information concerning the first note of the continuing melody) is set in the address register 102.

15

(2) A "1" or melody starting information is set in the MELC register of the control circuit 110 in accordance with a command from the CPU 12. This melody starting information is outputted as a play start signal through the on/off control circuit 142. Then, as for example, the gate circuit 120 is opened and clock signals are applied to the frequency divider circuit 112 and the frequency multiplier circuit 124. Also, the melody starting address set in the address register 102 is written into the address counter 104 so that the

20

25

1 address of the melody ROM 106 is set to the starting
address of the desired melody to be played and the
Q playing of the melody is started in accordance with the
melody data written in the starting address.

5 The scale address data of the designated
address is applied to the scale ROM 122 so that the scale
in the address corresponding to the scale address data is
read from the scale ROM 122 and applied to the interval
generating circuit 126 through the interval data bus 125.
10 Then, the output of the interval generating circuit 126
is amplified by the melody output control circuit 128 and
generated as a melody output 129, thereby playing the
melody.

On the other hand, the note duration data of
15 the melody data stored in the designated address is
applied to the note duration generating circuit 116, in
which the note duration data is preset and present data
is counted down in response to the clock signals from the
tempo generating circuit 114, thereby generating an
20 address incremental signal 117 when the note duration data
is reduced to zero (or when the clock signals are counted
up thereby reaching a value corresponding to the note
duration data). The address counter 104 receives the
address incremental signal 117 thereby advancing the
25 address of the melody ROM 106 by one address.

This operation is repeated so that the

1 continuing melody stored in the melody ROM 106 is
automatically played without participation of the
microcomputer.

5 (3) Before the continuing melody is completed, a
"0" or stop play information is set in the MELC register
of the control circuit 110 in accordance with a command
from the CPU 12.

10 Fig. 6 shows the timing chart for such case. At the
time of the setting of this "0", no end of melody signal
has been applied to the on/off control circuit 142 and
thus the playing operation is not stopped.

(4) If a "1" has been set in the end data of the
15 melody data of the melody ROM 106 in the address
designated by the address counter 104, it is assumed that
the data represents the final note address of the
continuing melody and the end of melody signal generating
circuit 108 generates an end of melody signal 109 at the
20 timing of the application of an address incremental signal
117 from the note duration generating circuit 116 or at
the timing that the playing of the final note is
completed.

As a result, the on/off control circuit 142 receives
25 the end of melody signal 109 so that a stop of play
signal for stopping the playing is generated and the

1 operation of the circuits and devices included in the
melody generator 22 is stopped. At the same time, the
melody interrupt generating circuit 118 receives the end
of melody signal 109 and applies to the CPU 12 an end of
5 melody interrupt signal 119 informing the end of the
melody.

The end of melody interrupt signal 119 is applied
to the CPU 12 through the interrupt generator circuit 24
so that if the interrupt is allowed, in the CPU 12 a
10 jump is made to a melody subroutine and an interrupt flag
is read and cleared. Then, an enable condition is
established and a return is made.

As described hereinabove, the desired melody to be
played can be played and ended automatically by simply
15 setting the starting address of the desired melody in the
address register 102 and writing the start or stop
information of the melody in the MELC register of the
control circuit 110 in accordance with the commands from
the CPU 12 and therefore the load of the CPU 12 for
20 playing the melody is reduced greatly.

(b) The continuous play mode :

Fig. 7 is a flow chart showing the operation of
the CPU 12 in this continuous play mode.

25

(b-1) Where a continuing melody is automatically

1 played repeatedly (it is to be noted that in this play
mode the processing of the portions enclosed by the dotted
lines in Fig. 7 is omitted)

5 (1) In accordance with a command from the
CPU 12 the starting address of the desired melody to be
played repeatedly in the melody ROM 106 (the address in
which is written the information concerning the first
note of the melody to be repeated) is written into the
10 address register 102.

(2) A "1" or melody start information is
set in the MELC register of the control circuit 110 in
accordance with a command from the CPU 12. As the result
15 of this operation, the melody start information is
outputted from the on/off control circuit 140. Then, the
circuits and devices of the melody generator 22 are
conditioned to start the playing so that the melody
starting address set in the address register 102 is
20 written in the address counter 104 to set the address of
the melody ROM 106 to the starting address of the desired
melody to be played and the playing of the melody is
started. The operation during the playing of the melody
is the same with the operation in the previously
25 mentioned mode.

1 (3) The address of the melody ROM 106 is
advanced automatically so that when the final address in
which a "1" is set in the end data is designated, in the
like manner as the previously mentioned mode an end of
5 melody signal 109 is generated from the end of melody
signal generating circuit 108 and the signal is applied
to the control circuit 110 and the melody interrupt
generating circuit 118. Due to the fact that at this time
the "1" or melody start information is set in the MELC
10 register of the control circuit 110 and a "0" or melody
stop information is not set, the control circuit 110
generates an on control signal for writing the address set
in the address register 102 into the address counter 104.

15 At this time, the same address as when
setting the melody start information in the MELC
register of the control circuit 110 is set in the
address register 102 and therefore the address data
delivered to the address bus 105 through the address
20 counter 104 again becomes the starting address of the
desired melody to be repeated. In other words, the
desired melody is played repeatedly.

Fig. 8 is a timing chart showing the
operation in such a case.

25 Also, since an end of melody interrupt
signal 119 is generated each time the desired melody to

1 be repeated is played and ended once, an interrupt is
applied to the CPU 12 and the CPU 12 increments the
number of plays n.

5 (5) When the number of plays n reaches the
desired number of plays N-1 in the CPU 12, a "0" or stop
play information is set in the MELC register of the
control circuit 110 in accordance with a command from the
CPU 12. As a result, after the desired melody being
10 played has been played, the melody generator 22 stops
the playing operation. In this case, the same operations
as stated in (3) and (4) of the previously mentioned
single-melody play mode are performed.

15 (b-2) Where a plurality of continuing melodies are
combined to play a single melody:

In the playing operation of the previously
mentioned (b-1) the address which is to be set in the
address register 102 can be changed after the generation
20 of an end of melody interrupt signal 119 to continuously
play a new melody instead of the repeated playing.

In other words, after a "1" has been set in the
MELC register as shown in Fig. 7, the CPU 12 preliminarily
sets the address of the next melody to be played in the
25 address register 102 via the data bus 30. Then, when the
playing of the first melody is completed so that the end

1 of melody signal generating circuit 108 generates an end
of melody signal 109, the address of the next melody to be
played which is then set in the address register 102 is
set in the address counter 104 so that the melody data of
5 the second melody to be played is read from the ROM 106
and the second melody in the repertoire is played.

On the other hand, the melody interrupt
generating circuit 108 receives the end of melody signal
109 to generate an end of melody interrupt signal 119
10 informing the end of the melody.

The end of melody interrupt signal 119 is
applied to the CPU 12 through the interrupt generator
circuit 24 so that when an interrupt is allowed, the CPU
12 causes a jump to the melody subroutine and the number
15 of plays n is incremented. In accordance with the number
of plays n, e.g., $n = 1$ the address of the third melody
to be played is set in the address register 102 through
the data bus 30 and also a tempo control signal is set in
the melody control register 140 of the control circuit
20 110. Then, after the interrupt flag has been read and
cleared, an enable condition is established and a return
is made.

When the number of plays n reaches a given
number $N-1$, a "0" is set in the MELC register so that when
25 next an end of melody signal is generated, an off control
signal is generated from the control circuit 110 and the

1 operation of the melody generator 22 is stopped, thereby
completing the playing.

In other words, during the playing of each
melody the address of a melody to be played next is
5 preliminarily stored in the address register and in this
way the plurality of different melodies are combined to
play a long continuing melody.

(c) Forced play mode:

10 Fig. 9 is a timing chart showing the operation of
the CPU 12 in this play mode. When it becomes necessary
to forcibly change or stop the melody being played by a
request made externally of the system, the request is
fulfilled by the following procedure.

15

(1) At the time of the occurrence of the request
externally of the system, the starting address of a
desired replacing melody whose information is stored in
the melody ROM 106 is set in the address register 102.

20

(2) Immediately following the setting of a "0" or
melody stop information, a "1" or melody start information
is written in the MELC register of the control circuit
110. As a result, in response to the application of the
25 melody start information the address set in the address
register 102 is set in the melody address counter 104 so

1 that the address of the melody ROM 106 jumps to the
starting address of the desired replacing melody and then
the desired replacing melody is played. Fig. 10 is a
timing chart showing the operation in such a case.

5

(3) When it is desired to accomplish forced stop,
the desired replacing melody is set to a silent melody,
e.g., a rest and the melody is stopped forcibly.

Fig. 11 is a block diagram of an address
10 register 102 including a playing order or FIFO storage
for storing a plurality of addresses of the melody ROM
106. The address register 102 includes an FIFO storage
144 connected to the data bus 30 so as to be written and
read in accordance with commands from the CPU 12 and an
15 address counter 146 for designating the addresses of the
FIFO storage 144. The address counter 146 receives an end
of melody signal 109 so as to advance the addresses of
the FIFO Register 144 by one address in response to each
input. If the FIFO storage 144 is a storage of such type
20 that the data written in each address contains the address
which is to be executed next, the address counter 146 is
not needed.

By virtue of this construction, the melody
playing burden of the CPU 12 can be reduced further. In
25 other words, the address of the melody ROM 106, which is
written in the address register 102 in response to the

1 generation of each end of melody interrupt signal 109 as
mentioned in connection with the previously mentioned
"(b-1) the playing of a single melody through the
combination of a plurality of continuing melodies", is
5 written into the FIFO storage 144 before starting the
playing of melody to thereby automatically play a
continuing melody. This operation will now be described.

1) Before starting the playing of a continuing
10 melody, the addresses of the melody ROM 106 which are
necessary for the playing are written in order of playing
in the FIFO storage 144.

2) The playing of the melody is started in accordance
15 with the previously mentioned method.

3) The address which is to be played first is
delivered to the melody address bus 103 and the address
counter 104 reads out the melody information written in
20 this address as well as the succeeding addresses of the
melody ROM 106, thereby playing the melody.

4) At the end of the playing of this continuing
melody, an end of melody signal 109 is generated so that
25 the FIFO storage address counter 146 advances the address
bus of the FIFO storage 142 by one address.

1

5) The address which is to be played second is delivered to the melody address bus 103.

5

6) The above-mentioned (3), (4) and (5) are repeated thereby playing another continuing melody.

10

7) The end of the playing can be detected by counting the number of occurrence of end of melody interrupt signals 119, and the playing can be ended by use of the playing ending method described in connection with the play mode of (b-2).

15

20

25

Fig. 12 is a block diagram showing the construction of another embodiment of the melody generator 22. This embodiment differs from the embodiment of Fig. 2 in that a data selector circuit 130 and a temporary data storage 132 are additionally provided. The data selector circuit 130 is connected to the data bus 30 and a data bus 107 of the melody ROM 106 and it comprises for example a selector having two inputs and one output to select one or the other of the data buses and input the data thereon in accordance with a control command applied through the control line 36 from the CPU 12. The temporary data storage 132 comprises for example a register whereby the data applied through the data

1 selector circuit 130 is retained temporarily and then
outputted to the note duration generating circuit 116 and
the scale ROM 122, respectively. The subsequent
operations are the same as the embodiment of Fig. 2.

5 Therefore, the data stored in either of the ROM 14
and the RAM 20 can be written into the temporary data
storage 132 through the data bus 30 and the data selector
circuit 130 so as to play a melody or melodies. As a
result, the playing can be controlled by both of the
10 melody data from the ROM 14 or the RAM 20 and the melody
data from the melody ROM 106. This means that where the
microcomputer finds its melody capacity provided by the
melody ROM 106 alone deficient in accordance with the
purpose to which it is applied, the melody or melodies
15 making good the deficiency can be stored in the ROM 14 or
the RAM 20 so as to effect the playing.

Also, by so constructing as to output the data in the
temporary data storage 132 to the data bus 30, it is
possible to reduce the test time required at the time of
20 shipping the melody ROM 106.

It is to be noted that while, in the above-described
embodiments, the address counter 104 is adapted to count
up, it is needless to say that there are instances
where it is adapted to count down depending on the storage
25 method of the melody ROM 106. Further, either one or both
of the address register 102 and the memory control

register 140, which are connected to the data bus 30, can be constructed so as to cause for example the RAM 20 to serve the function or functions thereof.

It is noted that, whilst the term "melody" has been used throughout the specification for convenience, this term is herein defined as not only including an agreeable succession of musical sounds, but also imitation sounds of animals and sound effects in games.

CLAIMS

1. A microcomputer comprising:
 - a CPU for processing various data;
 - a melody (as herein before defined) generator for generating a melody signal;and
 - a data bus for transferring input and output data of said CPU,
 - said melody generator including;
 - melody data memory means storing melody data containing at least scale data, note duration data and end data indicative of the presence or absence of end of melody;
 - interval generating means for generating an interval signal in accordance with scale data in said melody data,
 - note duration generating means for generating a note duration signal in accordance with note duration data in said melody data, and
 - addressing means for receiving an initial address signal from said CPU to supply the same as a reading address to said melody data memory means and for applying addresses succeeding said initial address signal as address signals to said melody data memory means at the timing of corresponding note duration signals.
2. A microcomputer according to claim 1, wherein said



1 melody generator further includes:

end of melody signal generating means for generating
an end of melody signal in accordance with said end data
of melody data from said melody data memory means and a
5 note duration signal from said note duration generating
means;

interrupt generating means for receiving said end of
melody signal to send an end of melody interrupt signal
to said CPU; and

10 control means for receiving at least a melody control
signal for commanding the start or stop of playing by said
melody generator from said CPU through said data bus and
for receiving said end of melody signal to start operation
of said various means forming said melody generator in
15 response to the application of a start of play command
signal and to stop the operation of said various means in
response to a stop of play command signal and said end
of melody signal.

20 3. A microcomputer according to claim 2, wherein said
addressing means includes an address register for storing
an initial address signal from said CPU through said data
bus, and an address counter into which said initial
address signal in said address register is set as a preset
25 value to count up or down said preset value each time a
signal according to each said note duration signal is



1 applied.

4. A microcomputer according to claim 3, wherein said
control means includes a melody control register for
5 receiving and storing said melody control signal
indicative of the start or stop of playing by said melody
generator from said CPU through said data bus, and an
on/off control circuit for receiving said end of melody
signal to start operation of said various means forming
10 said melody generator in response to the application of a
start of play command signal and to stop the operation of
said various means in response to a stop of play command
signal and said end of melody signal.

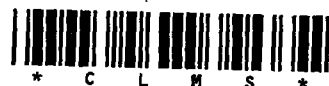
15 5. A micro-computer according to claim 4, wherein said
CPU is responsive to the application of an interrupt
signal to send an address signal corresponding to a
leading address of a plurality of melodies stored
preliminarily to said address register through said data
20 bus.

6. A microcomputer according to claim 5, wherein said
CPU, said data bus and said melody generator are formed
on one chip.

25 7. A microcomputer according to claim 4, wherein said



- 1 melody control register comprises a storage device
arranged within the same address space as memories
controlled by said CPU.
- 5 8. A microcomputer according to claim 3, wherein said
address counter comprises a storage device arranged
within the same address space as memories controlled by
said CPU.
- 10 9. A microcomputer according to claim 1, wherein said
address register includes play order memory means for
storing a plurality of addresses of said melody data
memory means corresponding to an order in which a
plurality of melodies are played whereby each time said
15 end of melody signal is applied, corresponding one of
said stored addresses in said order is set as a preset
value in said address counter.
- 20 10. A microcomputer according to claim 1, wherein said
melody generator further comprises data selector means for
receiving melody data from said melody data memory means
and melody data from said data bus to select and output
one or the other of the same in accordance with a command
from said CPU, and temporary storage means for receiving
25 the melody data from said data selector means to
temporarily store the same and to thereby output scale



1 data to said interval generating means and note duration
data to said note duration generating means.

11. A microcomputer substantially as hereinbefore described
with reference to, and as shown in Figures 1 to 11 or Figure
5 12, of the accompanying drawings.

10

15

20

25

