

Electronic Musical Instrument Having
Algorithm Selectable Function

Background of the Invention

Field of the Invention

The present invention relates to electronic musical instruments, and more particularly, to electronic musical instruments capable of simulating the sound of conventional non-electronic musical instruments with high fidelity.

Prior Art

There is known a conventional electronic musical instrument comprising a PCM (Pulse Code Modulation) tone generating device (hereafter referred to as a tone generating device (1)) which reads pulse-code-modulated waveform data from a waveform memory based on a clock corresponding to MIDI (Musical Instrument Digital Interface) data generated in response to the operation of, for example, a keyboard by a performer. Such a conventional electronic musical instrument comprises a plurality of sound production channels, for example, 16 sound production channels, and each of these sound production channels independently produces sound by means of timesharing in response to the above MIDI data. For example, one sound production channel produces sound with the tone color of a piano at one timing and another sound production channel produces sound with the tone color of a violin at another timing.

Furthermore, physical model tone generating devices



(hereafter referred to as tone generating devices (2)) are conventionally known which synthesize tones which effectively simulate the sound of a conventional non-electronic musical instrument by simulating the sound production algorithm in the target non-electronic instrument. Such a device is disclosed in US Patent No. 4,984,276.

One example of a liner portion of the above conventional tone generating device (2) is shown in the block diagram of Fig. 9. In this figure, an input terminal 1 is provided, to which an excitation signal waveform data made up of a large number of different high frequency components such as an impulse waveform is supplied. The excitation signal waveform data supplied via the input terminal 1 is supplied to the closed loop circuit via first input terminals of adders 2 and 3. The adder 3 adds the excitation signal waveform data and the output data read from an input memory 5 (MEMORY 2) which delays an input data for the desired time. The output data from the adder 3 is supplied to a multiplier 6 which multiplies it by a multiplicative coefficient C2. The output data from the multiplier 6 is supplied to a first input terminal of an adder 8. The output data from the adder 8 is stored in a temporary memory 9 (TL2) and supplied to a multiplier 11. The temporary memory 9 delays an input data, namely, the output data from the adder 8, for the desired time. The multiplier 11 multiplies an input data, namely, the output data from the adder 8, by a multiplicative coefficient r2. The data read from the temporary memory 9 is supplied to a multiplier 10. The multiplier 10 multiplies an input data, namely, the data read from the temporary memory 9, by a multiplicative coefficient 1-C2. The output data from the multiplier 10 is supplied to a



second input terminal of the adder 8. The adder 8 adds the output data from the multiplier 6 and the output data from the multiplier 10. Each of elements 8 through 10 described above together form a low pass filter (LPF) 12. The output data from the multiplier 11 is stored in an input memory 4 (MEMORY 1) which delays it for the desired time. The data read from the input memory 4 is supplied to a second input terminal of the adder 2.

The adder 2 adds the excitation signal waveform data and the data read from the input memory 4. The output data from the adder 2 is supplied to a multiplier 7 which multiplies it by a multiplicative coefficient C1. The output data from the multiplier 7 is supplied to a first input terminal of an adder 13. The output data from the adder 13 is stored in a temporary memory 14 (TL1) and supplied to a multiplier 16. The temporary memory 14 delays an input data, namely, the output data from the adder 13, for the desired time. The multiplier 16 multiplies an input data, namely, the output data from the adder 13, by a multiplicative coefficient r1. The data read from the temporary memory 14 is supplied to a multiplier 15. The multiplier 10 multiplies an input data, namely, the data read from the temporary memory 14 by a multiplicative coefficient $1-C1$. The output data from the multiplier 15 is supplied to a second input terminal of the adder 13. The adder 13 adds the output data from the multiplier 7 and the output data from the multiplier 15. Each of elements 13 through 15 described above together form a low pass filter (LPF) 17. The output data from the multiplier 16 is stored in the input memory 5. The data read from the input memory 5 is supplied to a second input terminal of the adder 3.

Because the above conventional tone generating device (2)



consists of a digital signal processor (DSP), it can synthesize various tones which effectively simulate the sound of conventional non-electronic musical instruments by simulating the various algorithms of sound production in the target non-electronic instruments by changing the microprogram (for example, see Fig. 10) used in the DSP. The above conventional tone generating device (2) as shown in Fig. 9 is an example of a tone generating device which synthesizes tone which effectively simulate the sound of a stringed instrument by simulating the sound production algorithm in the stringed instrument. An example of another type of tone generating device which synthesizes tones which effectively simulate the sound of another non-electronic musical instruments, for example, wind instruments, by simulating the sound production algorithm in the target non-electronic musical instruments, has been disclosed in Japanese Patent Application Laid-open Publication No. 2-280196.

In the above conventional electronic musical instrument comprising the above conventional tone generating device (1), a tone color number as well as performance information such as tone pitch and touch are supplied to the tone generating device (1) every key-on. Accordingly, if a performer designates tone color at each sound productions, each of the sound production channels of the tone generating device (1) directly access the corresponding area of the waveform memory and read waveform data from it. Thus, as stated above, it is an easy matter for one sound production channel to produce sound with the tone color of a piano at one timing and to produce sound with the tone color of a violin at the next timing by means of timesharing.

In contrast, in the above conventional electronic musical





instrument comprising the above conventional tone generating device (2), in the case of changing tone color at each key-on, there is a necessity either to supply a microprogram to the sound production channel at each key-on or to previously store a plurality of microprograms in each sound channel. Since the microprogram as shown in Fig. 10 is the microprogram corresponding to very a fundamental circuit construction as shown in Fig. 9, it does not take long to supply this microprogram to the sound production channel at each key-on. However, since the microprogram which accurately simulates the sound production algorithm in the target non-electronic musical instruments consists of a large number of data, when it is supplied to the sound production channel at each key-on, there is a drawback in that the key-on response is reduced due to the limitation on the data transmitting rate. In the case where previously storing a plurality of microprogram in each sound channel, there is a drawback in that the use efficiency of memory becomes lower and the system become expensive because great deal of memory is necessary.

Summary of the invention

In consideration of the above, it is an object of the present invention to provide an electronic musical instrument which is capable of efficiently using memory, and which is capable of producing the sounds of a plurality of tone colors every key-on without reducing key-on response.

To satisfy this object, the present invention provides an electronic musical instrument comprising tone generating means

comprising an excitation means for generating an excitation signal and sound producing means having input means for producing a musical tone signal in response to the excitation signal, delaying the musical tone signal and feeding the musical tone signal back to the input means;

memory means for storing a plurality of sound production algorithms;

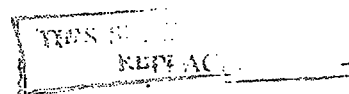
assignment designating means for designating one of the plurality of sound production algorithms and assigning the designated sound production algorithm to the musical tone generating means, the tone generating means further comprising operation means for performing the assigned sound production algorithm on the musical tone signal; and

extracting means for extracting the musical tone signal.

According to such a structure, when a performer designates one of the plurality of sound production algorithms and assigns the designated sound production algorithm to the musical tone generating means using the assignment designating means, the operation means performs the assigned sound production algorithm on the musical tone signal.

Accordingly, the excitation means generates the excitation signal and the input means produces the musical tone signal in response to the excitation signal and the sound producing means delays the musical tone signal and feeds the musical tone signal back to the input means. Thus, the extracting means extracts the musical tone signal.

According to the present invention, there is the positive effect that the volume of the tone color buffer memory of each of the sound production channel can be minimized. Furthermore,



there is the positive effect that a system having an efficient use of memory can be constructed. Moreover, there is the positive effect that the response to key-on is not less than in the conventional art. In addition, there is the positive effect that the generation of the musical forced tone that is caused by the limited number of sound production channel can be prevented because the order of priority of each of the tone colors in each of the sound production channels is prescribed.

Brief Explanation of the Drawings

Fig. 1 shows a block diagram of the electrical structure of an electronic musical instrument based on the preferred embodiment of the present invention.

Fig. 2 shows an example of the external structure of the panel 21 of Fig. 1.

Fig. 3 is a flow chart showing the main procedure routine of the CPU 18 based on the preferred embodiment of the present invention.

Fig. 4 is a flow chart showing the note on procedure routine of the CPU 18 based on the preferred embodiment of the present invention.

Fig. 5 is a flow chart showing the note off procedure routine of the CPU 18 based on the preferred embodiment of the present invention.

Fig. 6 is a flow chart showing the procedure routine in



connection with the note color of the CPU 18 based on the preferred embodiment of the present invention.

Fig. 7 shows a display example of the display 22 of Fig. 2.

Fig. 8 shows another display example of the display 22 of Fig. 2.

Fig. 9 shows a block diagram of a structural example of the linear portion of the physical model tone generating device of the prior art.

Fig. 10 shows an example of the microprogram of the physical model tone generating device of Fig. 9.

Detailed Description of the Preferred Embodiments

Hereinafter, an explanation of the preferred embodiment of the present invention is given by referring to the figures. Fig. 1 shows a block diagram of the structure of an electronic musical instrument in accordance with the preferred embodiment of the present invention. In this figure, a central processing unit (CPU) 18, which controls all apparatuses, a ROM 19, which has stored various control programs used in the CPU 18 and various microprograms loaded in the hereinafter described tone generating circuit 29 consisting of DSP and a RAM, and a RAM 20 are provided. Additionally, in the RAM 20, within which all types of registers, flags, and working buffers are maintained for use when the CPU 18 carries out any type of procedure, and MIDI data buffers are maintained for storing the MIDI data.

Furthermore, in Fig. 1, a panel 21 is provided, which consists of a display 22 such as a liquid crystal display, ten



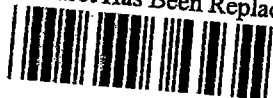
keys 23, an enter key 24 for designating, for example, the change of display or the like, cursor keys 25 for designating the movement of cursors on the display 22 and the like as shown in Fig. 2. The panel 21 supplies information in response to the operation of keys via a panel interface 26 and a system bus 27 to the CPU 18.

Moreover, in Fig. 1, a MIDI interface 28 is provided. The CPU 18 exchanges data such as MIDI data via the MIDI interface 28 and the system bus 27 with another electronic musical instrument or the like. A tone generating circuit 29 is provided, which synthesizes tones which effectively simulate the sound of wind instruments such as the clarinet, rubbed stringed instruments such as the violin, plucked stringed instruments such as the guitar, and beat stringed instruments such as the piano by simulating the sound production algorithm in these. The tone generating circuit 29 consists of a plurality of DSPs and a plurality of RAMs in which temporarily store the various computing data of the plurality of DSPs, respectively. The set of DSP and RAM correspond to the hereafter described sound production channels. A sound system 30 is provided comprising amplifiers, etc., which amplify a plurality of musical tone signals supplied from the tone generating circuit 29. A speaker 31 is provided which transduces a plurality of the musical tone signals to the musical tone and delivers.

Next the flow of operation of the CPU 18 in the electronic musical instrument of the present invention will be described with reference to the flow charts of Figs. 3 through 6.

When power is supplied to the device shown in Fig. 1, the CPU 18 begins to execute the main procedure routine shown in Fig.

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED

3 starting with step SA1. In step SA1, the initialization of all apparatuses is carried out. This initialization consists of the setting of the initial tone color in the tone generating circuit 29, and the clearing of the registers of RAM 20. Next, MIDI interface 28 is scanned and the input state of MIDI data is detected in step SA2.

Next, in step SA3, judgment is made as to whether or not a MIDI event based on the input state of MIDI data detected in the MIDI scanning procedure of step SA2 exists. When the result of the judgment in SA3 is [YES], the routine proceeds to step SA4. In contrast, when the result of the judgment in step SA3 is [NO], that is, when the MIDI event is not detected, the routine proceeds to step SA8 described below.

In step SA4, the values corresponding to their respective detected states are stored in register EV, which temporarily stores a note on event NON or a note off event NOFF, register NC, which temporarily stores note code NC, and in register NV, which temporarily stores velocity.

Next, in step SA5, judgment is made as to whether or not the stored data in the register EV corresponds to a note on event NON. When the result of the judgment in SA5 is [YES], the routine proceeds to step SA6 and note on procedure (sound production procedure) is carried out in step SA6. In contrast, when the result of the judgment in step SA5 is [NO], that is, when the stored data in the register EV corresponds to the note off event NOFF, the routine proceeds to step SA7 and note off procedure (sound silencing procedure) is carried out in step SA7. The sound production procedure and the sound silencing procedure will be described below in detail. Next, when the sound



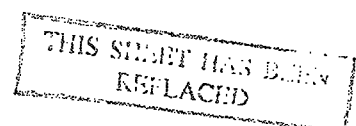
production procedure or the sound silencing procedure have been carried out, the routine proceeds to step SA8.

In step SA8, the panel 21 is scanned to detect the operation state of the panel 21. Next, in step SA9, judgment is made as to whether or not there exists a panel event based on the state of panel 21 detected in the panel scanning procedure of step SA8. When the result of the judgment in SA9 is [YES], the routine proceeds to step SA10. In contrast, when the result of the judgment in step SA9 is [NO], in other words, when the panel event is not detected, the routine returns to step SA2.

In step SA10, judgment is made as to whether or not the panel event detected in step SA8 is in connection with tone color. When the result of the judgment in SA10 is [YES], the routine proceeds to step SA11 and the procedure in connection with the tone color is carried out in step SA11. In contrast, when the result of the judgment in SA10 is [NO], namely, when the panel event detected in step SA8 is not in connection with tone color, the routine proceeds to step SA12 and the procedures in step SA12 is carried out. The procedures relating to the tone color will be described below in detail. Next, when the procedures relating to the tone color and other procedures have been carried out, the routine returns to step SA2 and steps SA2 through SA12 are repeatedly carried out until the power is turned off.

Next, the note on procedure of CPU 18 will be described with reference to the flow chart in Fig. 4. When the routine proceeds to step SA6 shown in Fig. 3, the CPU 18 begins to execute the note on procedure routine shown in Fig. 4 starting with step SB1. In step SB1, the number of the MIDI channel for which an event

This Sheet Has Been Replaced



was detected is stored in the register MCH. Next, in step SB2, "0" is stored in a register CH, storing the number of the sound production channel so as to search the state of all of the sound production channels.

Next, in step SB3, "7FFF" (maximum value in hexadecimal) is stored in the register MIN so as to truncate the sound production channel having the envelope value minimum when an open sound production channel does not exist.

In step SB4, a judgment is made as to whether or not the value in the register AMC[CH], in which what number MIDI channel has been assigned for the sound production channels set in register CH is stored, is identical to the value set in register MCH. When the result of the judgment in SB4 is [YES], the routine proceeds to step SB5. In contrast, when the result of the judgment in SB4 is [NO], namely, when the value stored in the register AMC[CH] is not equal to the value stored in the register MCH, the routine proceeds to step SB10 described below because the sound production channel corresponding to the value stored in the register AMC[CH] can not be assigned.

Next, in step SB5, judgment is made as to whether or not the value stored in the register ST[CH] (ST is a state signal), storing the state of the sound production channel corresponding to the number stored in the register CH, equals "0", namely, whether or not this sound production channel is in a channel standby state. When the result of the judgment in SB5 is [NO], the routine proceeds to step SB6. In contrast, when the result of the judgment in SB5 is [YES], in other words, when the value stored in the register ST[CH] equals "0", the routine proceeds to step SB14 described below because the open sound production



channel corresponding to the value stored in the register ST[CH] exists.

In step SB6, the envelope value of the sound production channel in the tone generating circuit 29 corresponding to the number stored in the register CH is stored in the register ENV. Next, in step SB7, judgment is made as to whether or not the value stored in the register ENV is smaller than the value stored in the register MIN. When the result of the judgment in SB7 is [YES], the routine proceeds to step SB8. In contrast, when the result of the judgment in SB7 is [NO], that is, when the value stored in the register ENV is equal to or larger than the value stored in the register MIN, the routine proceeds to step SB10 described below.

In step SB8, the value stored in the register ENV is stored in the register MIN. Next, in step SB9, the value stored in the register CH is stored in the register TCH. In step SB10, "1" is added to the value stored in the register CH in order to search the next sound production channel. Next, in step SB11, judgment is made as to whether or not the new value stored in the register CH is equal to the total number of sound production channels CHMAX (for example, 32). When the result of the judgment in SB11 is [NO], the routine returns to step SB4 and the above-mentioned procedure is repeatedly carried out until the value stored in the register CH is equal to the total number of sound production channels. In contrast, when the result of the judgment in SB11 is [YES], that is, when the value stored in the register CH is equal to the total number sound production channels, the routine proceeds to step SB12.

In step SB12, the sound silencing procedure is carried out



for silencing the sound of the sound production channel in the tone generating circuit 29 corresponding to the number stored in the register TCH. Next, in step SB13, the value stored in the register TCH is stored in the register CH. Next, in step SB14, "1" indicating the continuation state of sound producing based on note on, is stored in the register ST[CH].

In step SB15, the key code KC corresponding to the tone pitch to be produced is stored in the register AKC[CH], storing a key code KC in response to the sound production channel. Next, in step SB16, the note code NC, the velocity NV and the note on NON are supplied to the open sound production channel in the tone generating circuit 29 corresponding to the number stored in the register CH, and the routine returns to step SA8 of the main procedure routine shown in Fig. 4.

Next, the note off procedure of the CPU 18 will be described with reference to the flow chart of Fig. 5. When the routine proceeds to step SA7 shown in Fig. 3, the CPU 18 begins to execute the note off procedure routine shown in Fig. 5 starting with step SC1. In step SC1, the number of MIDI channels for which a MIDI event was detected is stored in the register MCH. Next, in step SC2, "0" is stored in register CH, storing the number of sound production channels in order to search the state of all of the sound production channel.

Next, in step SC3, judgment is made as to whether or not the value stored in the register AMC[CH] is equal to the value stored in the register MCH. When the result of the judgment in SC3 is [YES], the routine proceeds to step SC4. In contrast, when the result of the judgment in SC3 is [NO], namely, when the value stored in the register AMC[CH] does not equal the value stored in



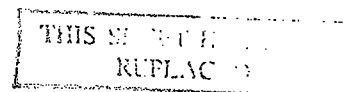
the register MCH, the routine proceeds to step SC5 described below.

Next, in step SC4, judgment is made as to whether or not the value stored in the register AKC[CH] is equal to the key code KC. When the result of the judgment in SC4 is [NO], the routine proceeds to step SC5. In contrast, when the result of the judgment in SC4 is [YES], in other words, when the value stored in the register AKC[CH] is equal to the key code KC, the routine proceeds to step SC7 described below.

In step SC5, "1" is added to the value stored in the register CH in order to search the next sound production channel. Next, in step SC6, judgment is made as to whether or not the new value stored in the register CH is equal to the total number sound production channels CHMAX (for example, 32). When the result of the judgment in SC6 is [NO], the routine returns to step SC3 and the above-mentioned procedure is repeatedly carried out until the value stored in the register CH is equal to the number of all of sound production channel. In contrast, when the result of the judgment in SC6 is [YES], that is, when the value stored in the register CH is equal to the total number of sound production channels, the routine returns to step SA8 of the main procedure routine shown in Fig. 4.

In step SC7, "0" indicating the channel standby state, is stored in the register ST[CH]. In step SC8, "0" is stored in the register AKC[CH]. Next, in step SC9, the note off NOFF is supplied to the sound production channel in the tone generating circuit 29 corresponding to the number stored in the register CH, and the routine returns to step SA8 of the main procedure routine shown in Fig. 4.

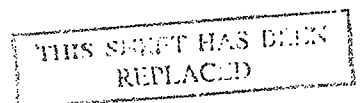
This Sheet Has Been Replaced



Next, the procedure in connection with the tone color of the CPU 18 will be described with reference to the flow chart in Fig. 6. When the routine proceeds to step SA11 shown in Fig. 3, the CPU 18 begins to execute the procedure in connection with the tone color routine shown in Fig. 6 starting with step SD1. In step SD1, the number of sound production channel and the tone color number for each MIDI channel is stored in the registers based on the operation of the panel 21 by operator. Namely, when the operator selects the number of the sound production channel and the tone color number for each MIDI channel using the ten keys 23, the enter key 24 and the cursor key 25 of the panel 21 shown in Fig. 2, the CPU 18 stores the number of sound production channel and the tone color number for each MIDI channel in the corresponding registers of the RAM 20. The CPU18 displays the number of sound production channel and the tone color number selected for each MIDI channel on the display 22, as shown, for example, in Figs. 7 and 8. In the example shown in Fig. 7, 4 sound production channels are assigned to MIDI channel 0, 2 sound production channels are assigned to MIDI channel 1, ..., and 4 sound production channels are assigned to MIDI channel 7. In the example shown in Fig. 8, tone color corresponding to the tone color number 02, that is, the tone color of a grand piano is assigned to MIDI channel 3.

In step SD2, "0" is stored in the register MCH in order to decide the state of the sound production channel of each MIDI channel for which the number of sound production channels and the tone color number are selected by the operator. Next, in step SD3, "0" is stored in register CH in order to decide the state of all of the sound production channels selected for the MIDI.

This Sheet Has Been Replaced



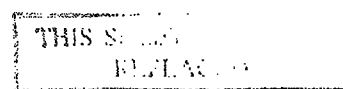
channels.

Next, in step SD4, the number of sound production channels assigned to the MIDI channel corresponding to the number stored in the register MCH, for example, 4 in case of MIDI channel 0, is stored in the register N. In step SD5, the tone color number of the sound production channel assigned to the MIDI channel corresponding to the number stored in the register MCH, for example, 02 in case of MIDI channel 3, is stored in the register TC.

In step SD6, the microprogram corresponding to the tone color number stored in the register TC, for example, the microprogram of a violin, is supplied to the sound production channel in the tone generating circuit 29 corresponding to the number stored in the register CH. Next, in step SD7, the value stored in the register MCH is stored in the register AMC[CH], in which is recorded what MIDI channel number is assigned for the sound production channel stored in the register CH.

In step SD8, "1" is added to the value stored in the register CH in order to decide the state of the next sound production channel. Next, in step SD9, "1" is subtracted from the value stored in the register N so as to decide the state of the next sound production channel assigned to the same MIDI channel. In step SD10, judgment is made as to whether or not the new value stored in the register N is equal to "0". When the result of the judgment in SD10 is [NO], the routine returns to step SD6 and the above-mentioned procedure is repeatedly carried out for all of the sound production channels assigned to the same MIDI channel. In contrast, when the result of the judgment in SD10 is [YES], that is, when the value stored in the register N

This Sheet Has Been Replaced



is equal to "0", the routine proceeds to step SD11.

In step SD11, "1" is added to the value stored in the register MCH in order to decide the state of the next MIDI channel. Next, in step SD12, judgment is made as to whether or not the new value stored in the register MCH is equal to "8". When the result of the judgment in SD12 is [NO], the routine returns to step SD4 and the above-mentioned procedure is repeatedly carried out for all MIDI channel. In contrast, when the result of the judgment in SD12 is [YES], that is, when the value stored in the register MCH is equal to "8", the routine returns to step SA2 of the main procedure routine shown in Fig. 4.

With the electronic musical instrument of the embodiment of the present invention as thus described above, a plurality of tone colors are preassigned to each of the sound production channels limited in number, a plurality of microprogram corresponding to a plurality of tone colors are presupplied to each of the sound production channels assigned and sounds are produced in the assigned sound production channel in response to the MIDI data. Accordingly, it is possible to minimize the volume of memory and construct a system having an efficient utilization of memory. Furthermore, response to a key-on can be carried out more quickly than in the conventional art. Moreover, generation of a forced musical tone which is caused by the limited number of the sound production channels can be prevented because the order of priority of each of the tone colors in each of the sound production channels is prescribed.

This Sheet Has Been Replaced



THIS SHEET HAS BEEN REPLACED



What is claimed is:

1. An electronic musical instrument comprising:

tone generating means comprising an excitation means for generating an excitation signal and sound producing means having input means for producing a musical tone signal in response to said excitation signal, delaying said musical tone signal and feeding said musical tone signal back to said input means;

memory means for storing a plurality of sound production algorithms;

assignment designating means for designating one of said plurality of sound production algorithms and assigning said designated sound production algorithm to said musical tone generating means, said tone generating means further comprising operation means for performing said assigned sound production algorithm on said musical tone signal; and

extracting means for extracting said musical tone signal.

2. An electronic musical instrument according to claim 1 wherein each of said plurality of sound production algorithms, imparts a characteristic of a musical sound to said musical tone signal.

3. An electronic musical instrument according to claim 1 wherein said musical tone generating means consists of a physical model tone generating device.

4. An electronic musical instrument according to claim 1 wherein said musical tone generating means consists of a digital signal processor.

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED