



EUROPEAN PATENT APPLICATION

Application number: **92112112.5**

Int. Cl.⁵: **G10H 1/06, G10H 7/00**

Date of filing: **15.07.92**

Priority: **28.08.91 JP 216868/91**

Date of publication of application:
03.03.93 Bulletin 93/09

Designated Contracting States:
DE FR GB

Applicant: **CASIO COMPUTER CO., LTD.**
6-1, Nishi-Shinjuku 2-chome
Shinjuku-ku, Tokyo 163(JP)

Inventor: **Watanuki, Masatoshi, c/o Casio Computer Co., Ltd.**
Hamura R&D Center Pat Dept., 2-1 Sakae-cho 3-chome
Hamuramachi, Nishitamagun, Tokyo 190-11(JP)
Inventor: **Kawanishi, Toshiaki, c/o Casio Computer Co., Ltd.**
Hamura R&D Center Pat Dept., 2-1 Sakae-cho 3-chome
Hamuramachi, Nishitamagun, Tokyo 190-11(JP)

Inventor: **Hanzawa, Kohtaro, c/o Casio Computer Co., Ltd.**
Hamura R&D Center Pat Dept., 2-1 Sakae-cho 3-chome
Hamuramachi, Nishitamagun, Tokyo 190-11(JP)

Inventor: **Sasaki, Hiroyuki, c/o Casio Computer Co., Ltd.**
Hamura R&D Center Pat Dept., 2-1 Sakae-cho 3-chome
Hamuramachi, Nishitamagun, Tokyo 190-11(JP)

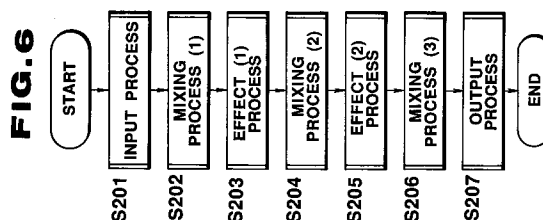
Inventor: **Yoshino, Jun, c/o Casio Computer Co., Ltd.**
Hamura R&D Center Pat Dept., 2-1 Sakae-cho 3-chome
Hamuramachi, Nishitamagun, Tokyo 190-11(JP)

Representative: **Patentanwälte Grünecker, Kinkeldey, Stockmair & Partner**
Maximilianstrasse 58
W-8000 München 22 (DE)

Effect adding apparatus.

When a plurality of effects are simultaneously applied to an input audio signal, a plurality of effect-algorithms each for applying solely an effect to the input signal are stored in a memory (2), and a plurality of combination-algorithms or combination-data which represent combinations of the effects and orders in which effects are applied to the input audio signal are stored in the memory. When one of the combination-algorithms or the combination-data is selected, effect-algorithms included in the selected combination-algorithm or combination-data are selectively read out from the memory. A program for applying effects in a predetermined combination and order is written by CPU (1) based on the read out effect-algorithms and combination-algorithm or combination-data. Receiving the program, DSP (4) successively applies effects to the input audio signal in accordance with the program. As a result, the

present effector needs less memory capacity than the conventional effector in which a number of programs consisting of combinations of effect-algorithms are previously stored.



The present invention relates to an effect adding apparatus which is capable of adding various sound effects to audio signals input from an electronic musical instrument and other audio equipment.

In an electronic musical instrument and audio equipment which generate and process sounds, it has been an important theme how to generate musical tones of rich tone color. An conventional electronic musical instrument and audio equipment generate musical tones which are richer in sound effects with the aid of effector devices which add sound effects such as chorus, delay, reverberation effects and the like to the generated musical tones.

In recent, a so called multi-effector which is capable of simultaneously adding a plurality of sound effects to musical tones has been proposed and put in use in place of an effector which adds a single sound effect to musical tones.

The multi-effector includes a plurality of effectors which add a single sound effect to musical tones. The multi-effector is composed of a series connection of effectors or a parallel connection of effectors, or is composed of a digital signal processor (DSP), to which a program involving algorithms for performing various effect processes is sent to obtain a plurality of sound effects.

Furthermore, when the multi-effector is used as effect adding device in an electronic musical instrument, method of playing the musical instrument is changed to alter an atmosphere of musical tones to be generated, but lately some trials have been made to realize the above by changing a way to apply the sound effects to musical tones or changing number of sound effects to be applied to the musical tones.

In the conventional effect adding device including a plurality of effectors, however, the physical connection of those effectors must be changed. But it will be easily understood that it is extremely hard to change the physical connection of the effectors while a performance of the musical instrument is being effected, and the effect adding device will not exhibit its features as the multi-effectors to a full extent. Usage of a switch may be also proposed for switching the connections of the effectors during the performance of the musical instrument, but a complicated circuitry arrangement will be invited for that purpose.

Meanwhile, when a user operates the multi-effector composed of DSP, he is simply required to change a program to be input thereto to obtain various sound effects, but sufficient number of algorithms must be prepared in advance for executing various effect processes combined in different ways, and a central processing unit (CPU) consequently needs a large capacity of memory for storing the algorithms to be transferred to the DSP.

The present invention has been made to overcome the above drawbacks, and has on object to provide an effect adding apparatus which is capable of changing a combination of various effects which are applied to input audio signal, requiring neither memory of a large capacity of memory nor change in circuitry connection.

According to one aspect of the invention, there is provided an effect adding apparatus which comprises:

effect-algorithm memory means for storing a plurality of effect-algorithms for applying sound effects to an input audio signal;

combination-algorithm memory means for storing a plurality of combination-algorithms for combining in different states the plurality of effect-algorithms stored in said effect-algorithm memory means;

algorithm combining means for reading out of the combination-algorithms from said combination-algorithm memory means, then selectively reading out effect-algorithms from said effect-algorithm memory means in accordance with the read out combination-algorithm, and combining the read out effect-algorithms in accordance with the read out combination-algorithm; and

effect adding means for applying relevant sound effects to the input audio signal based on the effect-algorithms combined by said algorithm combining means.

The effect adding apparatus with above structure allows change in combination of various sound effects to be applied to an input audio signal, without any restriction. There is no need in the effect adding apparatus to memorize as programs all of algorithms representative of all combinations of sound effects but only algorithms representative of respective sound effects and their combination are required to be memorized, resulting in a remarkable decrease in capacity of memory. Since the effect adding apparatus includes no connection of a plurality of hardware effectors, a user of the apparatus is not required to operate the apparatus to change a connection of the effectors while he is performing a musical instrument. The effect adding apparatus has no such a drawback that includes a complex wiring connection.

Another object of the present invention is to provide an effect adding apparatus which is capable of altering an order of application of various sound effects to an audio signal and a combination of these sound effects without requiring a large capacity of memory and without changing a circuitry connection.

According to still another aspect of the present invention, there is provided an effect adding apparatus which comprises:

effect-algorithm memory means for storing a

plurality of effect-algorithms for applying different sound effects to an input audio signal;

combination-data memory means for storing a plurality of combination-data which represent combinations in different states of a plurality of effect-algorithms stored in said effect-algorithm memory means;

algorithm combining means for reading out a combination-data from said combination-data memory means, then selectively reading out relevant effect-algorithms from said effect-algorithm memory means in accordance with the read out combination-data, and combining the read out effect-algorithms; and

effect adding means for applying relevant sound effects to the input audio signal in accordance with the effect-algorithms combined by said algorithm combining means.

With the effect adding apparatus of the above structure, an order of application of a plurality of sound effects to an input audio signal and a combination of these sound effects can be changed without any restriction. In the effect adding apparatus, since there is no need to memorize all algorithms representative of the combinations of the sound effects but algorithms representative of respective sound effects and combination data for indicating combinations of sound effects are memorized, a required capacity of memory can be decreased remarkably. Since the effect adding apparatus does not include a connection of a plurality of hardware effectors, the user of the apparatus is not required to operate to alter the connection of the effectors during his performance of an musical instrument. The effect adding apparatus is simple a wiring connection and allows the user to alter the order of application of sound effects to an input signal and the combination of the sound effects without any restriction.

Other objects and features of the present invention will be more fully understood by those with skill in the art from the following description of the preferred embodiments and the accompanying drawings.

Fig. 1 is a general circuit diagram of a first embodiment of an effect adding apparatus according to the present invention;

Fig. 2 is a circuit diagram of digital signal processor (DSP) in the first embodiment;

Fig. 3 is a flow chart of operation of a central processing unit (CPU) in the first embodiment;

Fig. 4 is a circuit diagram showing a hard circuit in the first embodiment for executing one form of multi-effect adding processes;

Fig. 5 is a circuit diagram showing hard circuit in the first embodiment for executing another multi-effect adding process;

Fig. 6 is a flow chart showing a general operation for executing a multi-effect adding process in the first embodiment;

Fig. 7 is a flow chart showing a detailed operation of an input process in the first embodiment;

Fig. 8 is a flow chart showing a detailed operation of a mixing process (1A) in the first embodiment;

Fig. 9 is a flow chart showing a detailed operation of a mixing process (2A) in the first embodiment;

Fig. 10 is a flow chart showing a detailed operation of a mixing process (3A) in the first embodiment;

Fig. 11 is a flow chart showing a detailed operation of a mixing process (1B) in the first embodiment;

Fig. 12 is a flow chart showing a detailed operation of a mixing process (2B) in the first embodiment;

Fig. 13 is a flow chart showing a detailed operation of a mixing process (3B) in the first embodiment;

Fig. 14 is a flow chart showing a detailed operation of an output process in the first embodiment;

Fig. 15 is a view showing data used in the first embodiment; and

Fig. 16 is a view showing coefficients used in the first embodiment.

Now, the first embodiment of the present invention will be described referring to the accompanying drawings.

Fig. 1 is a view showing a general circuit diagram of the first embodiment of an effect adding apparatus according to the present invention which is applied to an electronic musical instrument. In Fig. 1, a reference numeral 1 stands for a central processing unit (CPU), which is means for writing a program. CPU 1 uses RAM 3 as a work memory and controls a digital signal processor (DSP) 4 in accordance with a program stored in ROM 2. CPU 1 refers to a state of various switches provided in a switch section 5, and performs a control operation in accordance with changes in the state of the switches.

ROM 2 functions as an effect-algorithm memory means as well as a combination-algorithm memory means, and previously stores these algorithms. The effect-algorithm is an algorithm for adding or applying a predetermined sound effect such as an echo effect and reverberation effect to an input audio signal. The effect algorithm for applying a plurality of sound effects includes algorithms which correspond to various sound effects respectively.

The combination-algorithm is an algorithms for selecting and combining in different forms a plural-

ity of sound effects such as a chorus effect and the reverberation effect to be applied or added to an input signal.

The switches provided in the switch section 5 are operated to select one or a plurality of sound effects, and to select an appropriate form of combination of these sound effects. When one of the switches is operated, a control signal is sent to CPU 1 and CPU 1 allows a sound effect corresponding to the operated switch to be applied to an input audio signal.

When two switches are operated, a control signal is sent to CPU 1 and CPU 1 allows two sound effects corresponding to the operated switches respectively to be applied to an input audio signal with a desired order of their application. Further, the switches may be also operated such that one or two of sound effects are applied to a plurality of channels of audio signals.

CPU 1 judges a form of combination of a plurality of sound effects to be applied to an input audio signal, based on the control signal sent from various switches provided in the switch section 5, and CPU 1 reads out effect-algorithms and a combination-algorithm from ROM 2 in accordance with the read out the form of combination and writes a program based on these read out effect-algorithms and combination-algorithm. Then, the program is transferred to DSP 4.

DSP 4 executes a predetermined set of operation programs and applies a plurality of sound effects to a digital audio signal (such as a reproduced audio signal, hereafter, referred to as a musical-tone signal) to which an audio signal output from an electronic musical instrument (or from an audio reproducing apparatus) is converted by an analog/digital convertors (A/D convertors, ADC) 6, 7. DSP 4 has a function as an effect adding means, and performs an effect adding process in accordance with a program sent from CPU 1. The A/D convertor 6 converts a L-channel signal and R-channel signal of the musical-tone signal into a digital musical-tone signal and supplies it to an input terminal IN 1 of CPU 1 while A/D convertor 7 converts E-channel signal and T-channel signal into a digital musical-tone signal and input it to an input terminal IN 2.

The digital musical-tone signal which is applied with a plurality of sound effects is converted into analog musical-tone signals by digital/analog convertors (D/A convertors, DAC) 8, 9, and then the analog musical-tone signals are audibly output from speakers (not shown) through amplifiers (not shown).

The D/A convertor 8 converts the digital musical-tone signal, particularly an L-channel signal and R-channel signal, output from an output terminal OUT 1 of DSP 4 into analog musical-tone

signals while the D/A convertor 9 converts the digital musical-tone signal, particularly 1-channel signal and 2-channel signal, output from an output terminal OUT 1 of DSP 4 into analog musical-tone signals.

Fig. 2 is a view showing an internal structure of DSP 4.

In Fig. 2, a program memory 101 is a memory which stores predetermined micro-programs, and outputs a predetermined operation-program to a control circuit 102 in accordance with a program transferred from CPU 1 of Fig. 1. At this time, an address counter (not shown) is connected to the program memory. The program memory 101 successively supplies the control circuit 102 with contents of the program in accordance with address instructions sent from the address counter.

The control circuit 102 outputs various signals for controlling operations and data transfer between registers and memories as will be described later, and signals for open/close control of gates and latch circuits, and further outputs a counter value SC which is incremented every sampling time, performing predetermined signal processing operations.

A coefficient memory (P) 103 is a memory which stores various coefficients as will be described later with reference to Fig. 16. These coefficients are read out from RAM 3 of Fig. 1 and are stored in the coefficient memory (P) 103 under control of CPU 1.

A work memory (W) 104 is a memory for temporarily storing waveform signals and the like generated in DSP 4, as will be described later referring to Fig. 15.

An input register (PI1) 121 stores the digital musical-tone signal (L-channel signal and R-channel signal) input to DSP 4 from A/D convertor 6 of Fig. 1 through the input terminal IN1, and supplies the digital musical-tone signal to various sections via an internal bus 123.

In a similar manner, an input register (PI2) 122 stores the digital musical-tone signal (E-channel signal and T-channel signal) input to DSP 4 from A/D convertor 7 of Fig. 1 through the input terminal IN2, and supplies the digital musical-tone signal to various sections via an internal bus 123.

The output signals of the above coefficient memory (P) 103, the work memory (W) 104 and the output signals of the input registers (PI1) 121 and (PI2) 122 are input to gate terminals of gates 131 through 134 together with output signals from respective registers to be described later. The output signals of the gates 131 through 134 are input to registers (M0) 141, (M1) 142, (A0) 143 and (A1) 144.

The registers (M0) 141 and (M1) 142 store data under an operation which is to be supplied to a

multiplier 145 while the registers (A0) 143 and (A1) 144 stores data under an operation which is to be supplied to adder/subtractor 146.

The output signal of the register (M1) 142 and an output signal of a register (SR) 153 to be described later are input to the multiplier 145 through the gate 147 while the output signal of the register (A0) 143 and an output signal of a register (MR) 150 to be described later are input to the adder/subtractor 146 through the gate 148. Further, the output signal of the register (A1) 144 and an output signal of a register (AR) 151 to be described later are input to the adder/subtractor 146 through the gate 149.

The adder/subtractor 146 performs an addition and a subtraction and performs a process (a so-called through process) which allows data to pass through under an instruction from the control circuit 102.

A result of operation performed by the multiplier 145 is stored in the register (MR) 150 and the output of the register (MR) 150 are supplied to the gates 132 and 148. A result of operation performed by the adder/subtractor 146 is stored in the register (AR) 151, and the output of the register (AR) 151 are supplied to the gates 149 and to the register (SR) 153 through clipper circuit 152.

The clipper circuit 152 serves to prevent an over flow. The output of the register (SR) 153 is supplied to gate 147 and is transferred and stored as a result of an operation or a process performed on one tone in the work memory the internal bus 123.

When the results of the above operations are stored in the work memory (W) 104 and a series of processes are completed, data stored in the work memory (W) 104 are transferred to output registers (OR1) 154 and (OR2) 155, and further are transferred therefrom to D/A convertors 8 and 9 of Fig. 9.

The output register (OR1) 154 stores the L-channel signal and the R-channel signal, and outputs the same signals to D/A convertor 8 through the output terminals OUT 1 of DSP 4 while the output register (OR2) 155 stores the 1-channel signal and the 2-channel signal, and outputs the same signals to D/A convertor 9 through the output terminals OUT 2 of DSP 4.

Now, the principle of a multi-effect adding operation according to the present invention will be described with reference to the present embodiment.

Fig. 3 is a flow chart of a program routine of the multi-effect adding operation which is performed by CPU 1.

In Fig. 3, CPU 1 scans or refers to the switch section 5 at step S101, obtaining external data for judging whether or not a player (for example, a

player of an electronic musical instrument) has operated the switch section 5 to apply or add the multi-effects to an musical-tone signal. At step S102, CPU 1 judges if any change is occurs in the switch section 5, i.e., whether or not the player has operated the switch section 5 to apply or add the multi-effects to the signal. When it is judged that some changes have been found in the state of switch section 5, or when the result of the judgement at step S102 is YES, CPU 1 stores the state of the switch section 5 in its register (R) (not shown). On the contrary, when no change has been found in the state of the switch section 5, i.e., when the player has not operated the switch section 5, CPU 1 returns to step S101, waiting for a change.

CPU 1 judges at step S104 from the state of the switch section 5 stored in the register (R) whether or not the content of the register (R) is equivalent to a value A corresponding to an effect adding process (A). The effect adding process (A) is one example of combinations of a plurality of effect processes (1) and (2) (two effect processes in the present invention) which are to be applied in a different form to a plurality of channels of signals.

When the result of the judgement at step S104 is YES, CPU 1 judges at step S105 from the state of the switch section 5 that the effect adding process (A) is requested to be performed, transferring to DSP 4 programs for mixing processes (1A) to (3A) and effect processes (1) and (2), and also transferring contents of the coefficient memory (A) to DSP 4.

Then, DSP 4 executes a process for performing on a plurality of channels of signals one combination of effect adding process (A) among a plurality of combinations of effect adding processes.

When CPU 1 judges at step S104 from the state of the switch section 5 stored in the register (R) that the content of the register (R) is not equivalent to a value A corresponding to an effect adding process (A), i.e., when the result of the judgement at step S104 is NO, CPU 1 judges that an effect adding process other than the effect adding process (A) is requested to be performed, transferring at step S106 programs for the mixing processes (1B) to (3B) and effect processes (1) and (2) to DSP 4, and also transferring to DSP 4 contents of the coefficient memory (B) as coefficients which are necessary for performing an operation process.

Then, DSP 4 executes a process for performing one combination of effect adding process among a plurality of combinations of effect adding processes.

Algorithms (effect-algorithms) for performing the effect processes (1) and (2) have long descriptions and are previously stored in ROM 2 which

CPU 1 can control. The mixing processes (1A) to (3A) and the mixing processes (1B) to (3B) are algorithms (combination-algorithms) for determining how the algorithms of the effect processes (1) and (2) should be combined with respect to a plurality of channels of signals, and are stored in ROM 2 similarly to the effect-algorithms but have a relatively short description.

When a process of step S105 is performed, CPU 1 writes a program on the basis of the effect-algorithms for performing the effect processes (1) and (2), and combination-algorithms or mixing processes (1A) to (3A) for determining how the effect-algorithms should be combined with respect to a plurality of channels of signals, and then CPU 1 transfers the program to DSP 4.

As described above, the algorithms (effect-algorithms) for respective effect processes and algorithms (combination-algorithms) for combining in various forms these effect processes are previously stored in the present embodiment. When a selecting operation is externally executed to apply to a signal one of combinations of a plurality of effects, CPU 1 selects relevant algorithms for performing effect processes and also an algorithm for combining in various forms these algorithms in accordance with the selecting operation, and then CPU 1 writes one program using these algorithms and sends the program to DSP 4. DSP 4 perform a process for applying to the signal effects according to one of combinations of effects.

In the present embodiment, there is no need to previously store as programs in ROM 2 all the combinations of effects in different forms, to prepare required number of algorithms for changing the combination of effect processes and to change a program to be transferred to DSP 4, which increases a capacity of memory, but what are to be previously stored are effect-algorithms for performing effect processes and combination-algorithms for combining these effect-algorithms, which requires only a relatively small capacity of memory.

Further, the present embodiment is arranged such that, when a selecting manipulation is externally executed, an effect-algorithm for executing effect processes and a combination-algorithm for combining these effect-algorithms in different form are selected for prepare a program to be transferred to DSP 4.

In the present embodiment, therefore, there is no need to previously store as programs all the combinations of effects, so that a capacity of the memory can be remarkably reduced. In particular, since the effect-algorithms for performing effect processes have long descriptions, it is the main reason for the reduced capacity of memory that all these effect-algorithms are not necessary to be stored for respective combinations of effect-algo-

rithms.

Even if a plurality of effects are applied to an audio signal, the play is not required to perform troublesome operations during his performance for changing a wiring connection of a plurality of hardware effectors because the present embodiment has no plurality of hardware effectors connected to one another. The player is allowed to enjoy features of multi-effectors that apply multiple effects to sounds with a simple switching operation.

The present embodiment of the invention has no such a drawback that makes complex the wiring connection of the effectors because the player is not required during his performance to switch the wiring connection of the effectors with switching means.

As described above, the effect adding apparatus of the present invention is capable of changing a combination of sound effects to be applied to the input audio signal without need of a large capacity of memory and without switching the wiring connection of the effectors.

Figs. 4 and 5 are views schematically illustrating a hardware circuit performing one form of multi-effect adding processes in DSP 4. Contents of the coefficient memory (P) 103 and the work memory (WE) 104 of DSP 4 of Figs. 4 and 5 are shown in Figs. 15 and 16, respectively.

Fig. 4 is a view showing a first form of the effect adding process (A) for applying multi-effects to four input audio signals such as L-channel signal, E-channel signal, T-channel signal and R-channel signal.

In Fig. 4, the E-channel signal is subjected to a mixing (MIX) process (1A) 201, being separated into two systems of signals to be input to an effect (1) process 202. The effect (1) process 202 adds or applies a sound effect such as, for example, the reverberation effect to the input signals. The effect (1) process 202 applies the reverberation effect to the E-channel signals which are input thereto through the mixing process (1A) 201, and outputs the same to a mixing process (2A) 203.

The contents of the effect (1) process 202 for applying the reverberation effect is well known and therefore its detailed hardware circuit will be omitted.

In the mixing process (2A) 203, the two systems of E-channel signals with the reverberation effect applied are mixed with the T-channel signal at a predetermined rate, and then the two systems of E-channel signals are output to an effect (2) process 204. More specifically, one of two systems of E-channel signals with the reverberation effect applied is led to a multiplier 205, where the signal is multiplied by an effector (1) output multiplying coefficient P (EF1) shown in Fig. 16, and then is sent to an adder 206.

Meanwhile, the other one of two systems of E-channel signals with the reverberation effect applied is led to a multiplier 207, where the signal is multiplied by an effector (1) output multiplying coefficient P (EF1) shown in Fig. 16, and then is sent to an adder 208.

The T-channel signal is led to a multiplier 209, being multiplied by T-channel multiplying coefficient P (T), and then is transferred to the adders 206 and 208. One of two systems of E-channel signal which is multiplied by the effector (1) output multiplying coefficient P (EF1) in the multiplier 205, and the T-channel signal which is multiplied by the T-channel multiplying coefficient P (T) in the multiplier 209 are added in the adder 206 and the added signal is output to the effect (2) process 204.

The other one of two systems of E-channel signal with the reverberation effect applied, signal which is multiplied by the effector (1) output multiplying coefficient P (EF1) in the multiplier 207, and the T-channel signal which is multiplied by the T-channel multiplying coefficient P (T) in the multiplier 209 are added in the adder 206 and the added signal is output to the effect (2) process 204.

As described above, two systems of E-channel signals and the T-channel signal are mixed to each other at a predetermined rate.

The effect (2) process 204 is for applying a sound effect such as the chorus effect to input audio signals. In the effect (2) process 204, two systems of channel signals which have been processed in the mixing process (2A) 203 are applied with the chorus effect and then are transferred to a mixing process (3A) 201.

The contents of the effect (2) process 204 for applying the chorus effect are well known and therefore its detailed hardware circuit will be omitted.

In the mixing process (3A) 210, two systems of channel signals with the chorus effect applied are mixed with the L-channel signal and the R-channel signal at predetermined rates respectively, and are divided into four signals, i.e., 1-channel signal, L-channel signal, R-channel signal and 2-channel signal, which are output from DSP 4.

More specifically, one of two systems of signals output from the effect (2) process 204, signals which have been applied with the chorus effect, is led to a multiplier 211, where it is multiplied by an effector (2) output multiplying coefficient P (FL), and further is sent to an adder 214.

Meanwhile, the L-channel signal is led to a multiplier 215, being multiplied by an L-channel multiplying coefficient P (PL1), and then is sent to an adder 212. At the same time, the L-channel

signal is also led to a multiplier 216, being multiplied by an L-channel multiplying coefficient P (L1), and then is output as 1-channel signal from DSP 4.

In the adder 212, outputs of the multipliers 211 and 215 are added together. More specifically, one of two systems of output signals of the effect (2) process which has been applied with chorus effect and has been adjusted in the multiplier 211 to a level determined by the effector (2) output multiplying coefficient P (FL), and the L-channel signal which has been adjusted in the multiplier 215 to a level defined by the L-channel multiplying coefficient P (PL1) are added together to be output from DSP 4.

In a similar manner, the R-channel signal is led to a multiplier 217, being multiplied by an R-channel multiplying coefficient P (RP1), and then is sent to an adder 214. At the same time, the R-channel signal is also led to a multiplier 218, being multiplied by an R-channel multiplying coefficient P (R1), and then is output as 2-channel signal from DSP 4.

In the adder 214, outputs of the multipliers 213 and 217 are added together. More specifically, the other one of two systems of output signals of the effect (2) process which has been applied with chorus effect and has been adjusted in the multiplier 213 to a level determined by the effector (2) output multiplying coefficient P (FR), and the R-channel signal which has been adjusted in the multiplier 217 to a level that is defined by the R-channel multiplying coefficient P (RR1) are added together to be output from DSP 4.

During the above process, the E-channel signal is applied with the reverberation effect and then is mixed with the T-channel signal. The resultant signal is further applied with the chorus effect, then mixed with the L-channel signal and R-channel signal, and finally is divided into four systems of signals, i.e., L-channel signal, R-channel signal, 1-channel signal and 2-channel signal.

Fig. 5 is a view showing a second form of the effect adding process (B), which is different from the effect adding process (A), for applying multi-effects to four input audio signals such as L-channel signal, E-channel signal, T-channel signal and R-channel signal.

In Fig. 5, L-channel signal and R-channel signal are subjected to a mixing (MIX) process (1B) 301, being input to an effect (2) process 202 which is similar to that of Fig. 4. The effect (1) process 202 adds or applies the reverberation effect to the L-channel signal and the R-channel signal which have been transferred thereto through the mixing process (1B) 301, and outputs these signals to a mixing process (3B) 303.

Meanwhile, E-channel signal and T-channel signal are subjected to a mixing (MIX) process (2B)

304, being input to an effect (2) process 204 which is similar to that of Fig. 4. The effect (2) process 204 adds or applies the chorus effect to the E-channel signal and the T-channel signal which have been transferred thereto through the mixing process (2B) 304, and outputs these signals to a mixing process (3B) 303.

In the mixing process (3B) 303, the L-channel signal with the reverberation effect applied is mixed with the L-channel signal with no reverberation effect applied at a predetermined rate while the R-channel signal with the reverberation effect applied is mixed with the R-channel signal with no reverberation effect applied at a predetermined rate. Then, these two systems of signals are output from DSP 4 again as L-channel signal and R-channel signal respectively.

Meanwhile, the E-channel signal with the chorus effect applied is mixed with the original E-channel signal with no chorus effect applied at a predetermined rate while the T-channel signal with the chorus effect applied is mixed with the original T-channel signal with no chorus effect applied at a predetermined rate. Then, these two systems of signals are output from DSP 4 again as E-channel signal and T-channel signal respectively.

More specifically, the L-channel signal with the reverberation effect applied is led to a multiplier 305, where it is multiplied by an effector (1) output multiplying coefficient P (EL), and further is sent to an adder 306 while the R-channel signal with the reverberation effect applied is led to a multiplier 307, where it is multiplied by the effector (1) output multiplying coefficient P (ER), and then is sent to an adder 308.

Meanwhile, the L-channel signal is directly led to a multiplier 309, being multiplied by an L-channel multiplying coefficient P (LL2), and then is output to an adder 306. In the adder 306, the L-channel signal which has been applied with the reverberation effect and has been adjusted to a predetermined level that is defined by the multiplier 305 in accordance with the effector (1) output multiplying coefficient P (EL) is added to the L-channel signal which has been adjusted to a level that is defined by the multiplier 309 in accordance with the L-channel multiplying coefficient P (LL2), and then the resultant signal is output as L-channel signal from DSP 4 again.

The R-channel signal is directly led to a multiplier 310, being multiplied by an R-channel multiplying coefficient P (RR2), and then is output to an adder 308. In the adder 308, the R-channel signal which has been applied with the reverberation effect and has been adjusted to a predetermined level that is defined in the multiplier 307 by the effector (1) output multiplying coefficient P (ER) is added to the R-channel signal which has been

adjusted to a level that is defined in the multiplier 310 by the R-channel multiplying coefficient P (RR2), and then the resultant signal is output as R-channel signal from DSP 4 again.

Therefore, the L-channel signal and R-channel signal both with the reverberation effect applied are mixed with the original L-channel signal and R-channel signal both with no reverberation effect applied at a predetermined rate, respectively, and then are output from DSP 4 again as L-channel signal and R-channel signal.

In a similar manner, the E-channel signal with the chorus effect applied is led to a multiplier 311, where it is multiplied by an effector (2) output multiplying coefficient P (F1), and further is sent to an adder 312 while the T-channel signal with the chorus effect applied is led to a multiplier 313, where it is multiplied by the effector (2) output multiplying coefficient P (F2), and then is sent to an adder 314.

Meanwhile, the E-channel signal is directly led to a multiplier 315, being multiplied by an E-channel multiplying coefficient P (E1), and then is transferred to an adder 312. In the adder 312, the E-channel signal which has been applied with the reverberation effect and has been adjusted to a predetermined level that is defined by the multiplier 311 in accordance with the effector (2) output multiplying coefficient P (E1) is added to the E-channel signal which has been adjusted to a level that is defined by the multiplier 315 in accordance with the E-channel multiplying coefficient P (E1), and then the resultant signal is output as 1-channel signal from DSP 4 again.

The T-channel signal is directly led to a multiplier 316, being multiplied by an T-channel multiplying coefficient P (T2), and then is sent to an adder 314. In the adder 314, the T-channel signal which has been applied with the reverberation effect and has been adjusted to a predetermined level that is defined by the multiplier 313 in accordance with the effector (2) output multiplying coefficient P (F2) is added to the T-channel signal which has been adjusted to a level that is defined by the multiplier 316 by the T-channel multiplying coefficient P (T2), and then the resultant signal is output as 2-channel signal from DSP 4 again.

Therefore, the E-channel signal and T-channel signal both with the chorus effect applied are mixed with the original E-channel signal and T-channel signal both with no reverberation effect applied at a predetermined rate, respectively, and then are output from DSP 4 again as E-channel signal and T-channel signal, respectively.

During the above process, the L-channel signal and R-channel signal are applied with the reverberation effect, and then are mixed with the original L-channel signal and R-channel signal both with no

reverberation effect applied, respectively. The resultant signals are output as L-channel signal and R-channel signal, respectively. Meanwhile, the E-channel signal and T-channel signal are applied with the chorus effect, and then are mixed with the original E-channel signal and T-channel signal both with no chorus effect applied, respectively. The resultant signals are output as 1-channel signal and 2-channel signal, respectively. Finally, four channels of signals i.e., L-channel signal, R-channel signal, 1-channel signal and 2-channel signal are output from DSP 4.

Now, actual operation of DSP 4 structured as shown in Figs. 4 and 5 will be described in detail with reference to the operation flow charts of Figs. 6 to 14.

CPU 1 selects effect-algorithms for effect processes and combination-algorithm for combining the effect algorithms in the form as shown in Figs. 4 and 5, and writes a program, which is transferred to DSP 4. Then, DSP 4 stores the program transferred from CPU 1 in a program memory 101, and successively reads out the stored program as micro-programs to execute them as various processes.

In Figs. 15 and 16 are shown addresses in the coefficient memory (P) 103 where coefficients (constants) or variables are stored and addresses of the work memory 104 where data are temporarily stored. Names and contents of these coefficients, variables and data are also shown in Figs. 15 and 16.

Fig. 6 is a main flow chart of the multi-effects, the flow chart of which is illustrated in a flow for realizing various forms to explain the above effect-adding process (A) and effect adding process (B) to be executed.

In Fig. 6, an input process is executed at step S201, through which musical tone signals are input into DSP 4, that is, four musical tone signals are separately input to channels respectively.

At step S202, a mixing process (1) is executed. The mixing process (1) represents how these four channels of signals are mixed and applied with sound effects. The mixing process (1A) 201 or the mixing process (1B) 301 is executed in the mixing process (1).

At the following step S203, an effect process (1) is executed. In the effect process (1), the reverberation effect is applied to the signals based on how these four channels of signals are mixed.

At step S204, a mixing process (2) is executed. The mixing process (2) represents how these four channels of signals are mixed and applied with sound effects. The mixing process (2A) 203 or the mixing process (2B) 304 is executed in the mixing process (2).

At the following step S205, an effect process (2) is executed. In the effect process (2), the reverberation effect is applied to the signals based on how these four channels of signals are mixed.

5 A mixing process (3) is executed at step S206. More specifically, the mixing process (3A) 210 or the mixing process (3B) 303 is executed in the mixing process (3).

10 An output process is executed at step S207 to output from DSP 4 musical tone signals which have been subjected to the multi-effect process. Through the output process, four musical tone signals which have been subjected to the multi-effect process are separately output from DSP 4 to channels respectively.

15 Details of the above processes are shown in Fig. 7 through 14, and detailed contents of these processes will be described.

20 Fig. 7 is a view showing details of the input process (step S201).

25 At step S301 of Fig. 7, a musical tone signal retrieved in the input register (PI1) 121 is stored in the work memory(W) 104 as L-channel input data W (INL). At step S302, a musical tone signal retrieved in the input register (PI2) 122 is stored in the work memory(W) 104 as R-channel input data W (INR).

30 In a similar manner, at step S303, a musical tone signal retrieved in the input register (PI1) 121 is stored in the work memory(W) 104 as T-channel input data W (INT). At step S304, a musical tone signal retrieved in the input register (PI2) 122 is stored in the work memory(W) 104 as E-channel input data W (INE). In this manner, respective channels of input data are stored in the work memory (W) 104 at relevant addresses.

35 Figs. 8 to 10 are views showing details of the mixing processes (1A), (2A) and (3A). Fig. 8 is a view showing the mixing process (1A).

40 In Fig. 8, E-channel input data W (INW) is read out from the work memory (W) 104 and is stored in a register (A0) 143 at step S401. At step S402, the input data W (INE) stored in the register (A0) 143 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

45 At step S402, the input data W (INE) transferred to the register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152, and further the input data W (INE) stored in the register (A0) 143 is transferred to the register (AR) 151 again through the gate 148 and the adder/subtractor 146.

50 At the following step S403, the other input data W (INE) transferred to the register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152 while the former data SR (that is, the former input data W (INE)) previously stored in the register (SR) 153 is stored as effector (1) input

channel data W (EI1) in the work memory (W) 104 at the relevant address through an internal bus 123.

The latter input data W (INE) transferred to the register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152 at the same step S403. Then, at step S404, the latter data SR (that is, the latter input data W (INE)) previously stored in the register (SR) 153 is stored as effector (1) input channel data W (EI1) in the work memory (W) 104 at the relevant address through the internal bus 123.

As described above, a function similar to the mixing process (1A) 201 of Fig. 4 is performed.

Fig. 9 is a view showing the mixing process (2A) in detail.

In Fig. 9, the effector (1) output multiplying coefficient P (EF1) is read out from the coefficient memory (P) 201 and is stored in the register (M0) 141, and the effector (1) output channel data W (EO1) is read out from the work memory (W) 104 and is stored in the register (M1) 142 at step S501.

At step S502, the effector (1) output multiplying coefficient P (EF1) set in the register (M0) 141 is supplied to the multiplier 145, and the effector (1) output channel data W (EO1) set in the register (M1) 142 is supplied through the gate 147 to the multiplier 145. At the multiplier 145, the data W (EO1) is multiplied by the coefficient P (EF1), and the product is stored in the register (MR) 150. In this way, a process having a function equivalent to the function of the multiplier 205 of Fig. 4 will be realized through the above processes.

At the same step S502, the T-channel multiplying coefficient P (T) is read out from the coefficient memory (P) 103 and is stored in the register (M0) 141, and the T-channel input data W (INT) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At step S503, the product obtained by the multiplier 145, which has been stored in the register (MR) 150, is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146, and the T-channel multiplying coefficient P (T) set in the register (M0) 141 at step S502 is supplied to the multiplier 145, and further, the T-channel input data W (INT) set in the register (M1) 142 is supplied through the gate 147 to the multiplier 145. At the multiplier 145, the data W (INT) is multiplied by the coefficient P (T), and the product is stored in the register (MR) 150. Through the above processes, a process having a function equivalent to the function of the multiplier 209 of Fig. 4 is realized.

At step S504, the product obtained by the multiplier 145, which has been stored in the register (AR) 151, is transferred to one of the input terminals of the adder/subtractor 146 through the gate 149, and the result of the calculation at step

S503 (that is, the product of the T-channel multiplying coefficient P (T) and the T-channel input data W (INT)), which has been stored in the register (MR) 150, is supplied to the other one of the adder/subtractor 146 through the gate 148. Both data are added to each other at the adder/subtractor 146, and the sum of both data is stored in the register (AR) 151. Through these processes, a process having a function equivalent to the function of the adder 206 of Fig. 4 is realized.

At the same step S504, processes similar to a part of the process at step S503 are executed. That is, the T-channel multiplying coefficient P (T) set in the register (M0) 141 at step S502 is supplied to the multiplier 145, and the T-channel input data W (INT) set in the register (M1) 142 is supplied through the gate 147 to the multiplier 145. At the multiplier 145, the data W (INT) is multiplied by the coefficient P (T), and the product is stored in the register (MR) 150.

Further, at the same step S504, the effector (1) output multiplying coefficient P (EF1) is read out from the coefficient memory (P) 103 and is stored in the register (M0) 141, and the effector (1) output channel data W (EO2) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At step S505, the result of the calculation performed by the adder/subtractor 146, which has been stored in the register (AR) 151, is transferred to the register (SR) 152 through the clipper circuit 152, and the result of the calculation performed at step S504 (that is, the product of the T-channel multiplying coefficient P (T) and T-channel input data W (INT)), which has been stored in the register (MR) 150, is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

Further, the effector (1) output multiplying coefficient P (EF1), which has been set in the register (M0) 141 at step S504, is supplied to the multiplier 145, and the effector (1) output channel data W (EO2) set in the register (M1) 142 is supplied to the multiplier 145 through the gate 147. The product of both data obtained by the multiplier 145 is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 207 of Fig. 4 is realized.

At step S506, the data stored in the register (SR) 153 is stored as an effector (2) input channel data (1) W (EI1) via the internal bus 123 at a relevant address of the work memory (W) 104.

Further, at step S506, the product obtained at step S505 by the multiplier 145 (the product of the effector (1) output multiplying coefficient P (EF1) and the effector (1) output channel data W (EO2)), which has been stored in the register (MR) 150, is transferred through the gate 148 to one of the input

terminals of the adder/subtractor 146, and the product of the T-channel multiplying coefficient P (T) and the T-channel input data W (INT) obtained by the multiplier 145 is transferred through the gate 147 to the other input terminals of the adder/subtractor 146. The sum obtained by the adder/subtractor 146 is stored in the register (MR) 151. Through the above processes, a process having a function equivalent to the function of the adder 208 of Fig. 4 is realized.

At step S507, the result of the calculation performed by the adder/subtractor 146, which has been stored in the register (AR) 151, is stored in the register (SR) 153 through the clipper circuit 152, and at step S508, the data stored in the register (SR) 153 is stored as an effector (2) input channel data (2) W (FI2) via the internal bus 123 at a relevant address of the work memory (W) 104.

Through these processes, a process having a function equivalent to the function of the mixing process (2A) 203 of Fig. 4 is realized.

Fig. 10 is a view showing the mixing process (3A) in detail.

At step S601 of Fig. 10, the L-channel multiplying coefficient P (PL1) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141, while the L-channel input data W (INL) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At step S602, the L-channel multiplying coefficient P (PL1) set in the register (MO) 141 is supplied to the multiplier 145, and the L-channel input data W (INL) set in the register (M1) 142 is supplied through the gate 147 to the multiplier 145. At the multiplier 145, the data W (INL) is multiplied by the coefficient P (PL1), and the product is stored in the register (MR) 150. In this way, a process having a function equivalent to the function of the multiplier 216 of Fig. 4 will be realized through the above processes.

Further, at the same step S602, L-channel multiplying coefficient P (PLL1) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141.

At step S603, the product obtained by the multiplier 145 and stored in the register (MR) 150 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

At the same step S603, the L-channel multiplying coefficient P (PLL1) set in the register (MO) 141 is supplied to the multiplier 145, and the L-channel input data W (INL) set in the register (M1) 142 is supplied to the multiplier 145 through the gate 147. The multiplier 145 multiplies the data W (INL) by the coefficient P (PLL1) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 215 of Fig. 4 is

realized.

Further at the same step S603, the effector (2) output multiplying coefficient P (FL) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141, while an effector (1) output channel data (1) W (FO1) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S604, the product obtained by the multiplier 145, which has been stored in the register (AR) 151, is stored in the register (SR) 153 through the clipper circuit 152. Note that the data stored in the register (SR) 153 will be stored as 1-channel output data W (OT1) via the internal bus 123 at a relevant address of the work memory (W) 104 at the following step S605, and thereafter the data stored in the work memory (W) 104 will be output through an output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 216 of Fig. 4 is realized.

At the same step S604, the product (the product of the L-channel multiplying coefficient P (PLL1) and the L-channel input data W (INL)) obtained by the multiplier 145 and stored in register (MR) 150 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The effector (2) output multiplying coefficient P (FL) set in the register (MO) 141 is supplied to the multiplier 145, and the effector (1) output channel data (1) W (FO1) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data (1) W (FO1) by the coefficient P (FL) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 211 of Fig. 4 is realized.

Further at the same step S604, R-channel multiplying coefficient P (R1) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while R-channel input data W (INR) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S605, as described above, the data stored in the register (SR) 153 is stored as 1-channel output data W (OT1) via the internal bus 123 at a relevant address of the work memory (W) 104.

Then, the product obtained at step S605 by the multiplier 145 (the product of the effector (2) output multiplying coefficient P (FL) and the effector (1) output channel data (1) W (FO1)), which has been stored in the register (MR) 150, is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146, and the product of the L-channel multiplying coefficient P (PLL1) and the L-channel input data W (INL) obtained by the multiplier 145 is transferred through the gate 149 to the

other one of the input terminals of the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 212 of Fig. 4 is realized.

The R-channel multiplying coefficient P (R1) set in the register (MO) 141 is supplied to the multiplier 145, and the R-channel input data W (INR) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data (1) W (INR) by the coefficient P (R1) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 218 of Fig. 4 is realized.

Further at the same step S605, R-channel multiplying coefficient P (RR1) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141.

At step S606, the product obtained by the multiplier 145, which has been stored in the register (AR) 151, is stored in the register (SR) 153 through the clipper circuit 152. Note that the data stored in the register (SR) 153 will be stored as L-channel output data W (OTL) via the internal bus 123 at a relevant address of the work memory (W) 104 at the following step S607, and thereafter the data stored in the work memory (W) 104 will be output through the output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 212 of Fig. 4 is realized.

At the same step S606, the product (the product of the R-channel multiplying coefficient P (R1) and the R-channel input data W (INR)) obtained by the multiplier 145 and stored in register (MR) 150 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The R-channel multiplying coefficient P (RR1) set in the register (MO) 141 is supplied to the multiplier 145 while the R-channel input data W (INR) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data W (INR) by the coefficient P (RR1) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 217 of Fig. 4 is realized.

Further at the same step S606, the effector (2) output multiplying coefficient P (FR) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the effector (2) output channel data (2) W (FO2) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S607, as described above, the data stored in the register (SR) 153 is stored as L-channel output data W (OTL) via the internal bus 123 at a relevant address of the work memory (W) 104.

Then, the product obtained by the multiplier 145, which has been stored in the register (AR) 151, is stored in the register (SR) 153 through the clipper circuit 152. Note that the data stored in the register (SR) 153 will be stored as 2-channel output data W (OT2) via the internal bus 123 at a relevant address of the work memory (W) 104 at the following step S608, and thereafter the data stored in the work memory (W) 104 will be output through the output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 218 of Fig. 4 is realized.

At the same step S607, the product (the product of the R-channel multiplying coefficient P (RR1) and the R-channel input data W (INR)) obtained by the multiplier 145 and stored in register (MR) 150 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The effector (2) output channel multiplying coefficient P (FR) set in the register (MO) 141 is supplied to the multiplier 145 while the effector (2) output channel data W (FO2) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data W (FO2) by the coefficient P (FR) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 213 of Fig. 4 is realized.

At the following step S608, as described above, the data stored in the register (SR) 153 is stored as 2-channel output data W (OT2) via the internal bus 123 at a relevant address of the work memory (W) 104.

Then, the product obtained at step S607 by the multiplier 145 (the product of the effector (2) output multiplying coefficient P (FR) and the effector (2) output channel data (2) W (FO2)), which has been stored in the register (MR) 150, is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146, and the product of the R-channel multiplying coefficient P (RR1) and the R-channel input data W (INR) obtained by the multiplier 145 is transferred through the gate 149 to the other one of the input terminals of the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 214 of Fig. 4 is realized.

At the following step S609, the product obtained by the multiplier 145 and stored in the

register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152. Then, at step S610, data stored in the register (SR) 153 is transferred and stored as R-channel output data W (OTR) via the internal bus 123 at a relevant address of the work memory (W) 104.

Then, the data stored in the work memory (W) 104 will be output through the output process to be described later, whereby a process having a function equivalent to the function to externally supply the output of the multiplier 214 will be realized.

Through the above processes, a process having a function equivalent to that of the mixing process (3A) 210 of Fig. 4 is realized.

Figs. 11 to 13 are views showing mixing processes (1B), (2B) and (3B) in detail, respectively.

Fig. 11 is a view showing the mixing process (1B) in detail. At step S701 of Fig. 11, L-channel input data W (INL) is read out from the work memory (W) 104 and is stored in a register (A0) 143. At step S702, the input data W (INL) stored in the register (A0) 143 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

At the same step S702, the R-channel input data W (INR) is read out from the work memory (W) 104 and is stored in the register (AO) 143.

At step S703, L-channel input data W (INL) transferred to the register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152, and further R-channel input data W (INR) stored in the register (AO) 143 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

At the following step S704, the former data SR (L-channel input data W (INL)) transferred to the register (SR) 153 is stored as effector (1) input channel data (1) W (EI1) via the internal bus 123 at the relevant address of the work memory (W) 104.

In the similar manner, the latter R-channel input data W (INR) transferred to the register (AR) 151 at the previous step S703 is stored in the register (SR) 153 through the clipper circuit 152.

Then, at step S705, the latter data SR (that is, R-channel input data W (INR)) previously stored in the register (SR) 153 is stored as effector (1) input channel data W (EI2) via the internal bus 123 at the relevant address of the work memory (W) 104.

Through the described processes, a function equivalent to the mixing process (1B) 301 of Fig. 5 is realized.

Fig. 12 is a flow chart of the detailed mixing process (2B).

At step S801 of Fig. 12, E-channel input data W (INE) is read out from the work memory (W) 104 and is stored in a register (A0) 143. At step S802, the input data W (INE) stored in the register (A0) 143 is transferred to the register (AR) 151 through

the gate 148 and the adder/subtractor 146.

At the same step S802, the T-channel input data W (INT) is read out from the work memory (W) 104 and is stored in the register (AO) 143.

At step S803, E-channel input data W (INE) transferred to the register (AR) 151 is stored in the register (SR) 153 through the clipper circuit 152, and further T-channel input data W (INT) stored in the register (AO) 143 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

At the following step S804, the former data SR (E-channel input data W (INE)) transferred to the register (SR) 153 is stored as effector (2) input channel data (1) W (FI1) via the internal bus 123 at the relevant address of the work memory (W) 104.

In the similar manner, the latter T-channel input data W (INT) transferred to the register (AR) 151 at the previous step S803 is stored in the register (SR) 153 through the clipper circuit 152.

Then, at step S805, the latter data SR (that is, T-channel input data W (INT)) previously stored in the register (SR) 153 is stored as effector (2) input channel data (2) W (FI2) via the internal bus 123 at the relevant address of the work memory (W) 104.

Through the described processes, a function equivalent to the mixing process (2B) 304 of Fig. 5 is realized.

Fig. 13 is a flow chart of the detailed mixing process (3B).

At step S901 of Fig. 13, the L-channel multiplying coefficient P (LL2) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the L-channel input data W (INL) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S902, the L-channel multiplying coefficient P (LL2) set in the register (M0) 141 is supplied to the multiplier 145 while the L-channel input data W (INL) set in the register (M1) 142 is supplied through the gate 147 to the multiplier 145. At the multiplier 145, the data W (INL) is multiplied by the coefficient P (LL2), and the product is stored in the register (MR) 150. Through the process, a process having a function equivalent to the function of the multiplier 309 of Fig. 5 will be realized.

Further, at the same step S602, the effector (1) output multiplying coefficient P (EL) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the effector (1) output channel data (1) W (EO1) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At step S903, the product obtained by the multiplier 145 and stored in the register (MR) 150 is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

At the same step S903, the effector (1) output multiplying coefficient P (EL) set in the register (MO) 141 is supplied to the multiplier 145 while the effector (1) output channel data W (EO1) set in the register (M1) 142 is supplied to the multiplier 145 through the gate 147. The multiplier 145 multiplies the data W (EO1) by the coefficient P (EL) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 305 of Fig. 5 is realized.

Further at the same step S903, R-channel multiplying coefficient P (RR2) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while R-channel input data (1) W (INR) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S904, the data (the product of L-channel multiplying coefficient P (LL2) and L-channel input data W (INL)), which has been stored in the register (AR) 151, is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146 while the product obtained by the multiplier 145 (that is, the product of the effector (1) output multiplying coefficient P (EL) and the effector (1) output channel data (1) W (EO1)) is transferred through the gate 149 to the other one of the input terminals of the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 306 of Fig. 5 is realized.

At the same step S904, R-channel multiplying coefficient P (RR2) set in the register (MO) 141 is supplied to the multiplier 145 while R-channel input data W (INR) set in the register (M1) 142 is supplied to the multiplier 145 through the gate 147. The multiplier 145 multiplies the data W (INR) by the coefficient P (RR2) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 310 of Fig. 5 is realized.

Further, at the same step S904, the effector (1) output multiplying coefficient P (ER) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the effector (1) output channel data (2) W (EO2) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S905, the data (that is, the sum of the effector (1) output multiplying coefficient P (EL) and the effector (1) output channel data (1) W (EO1)), which has been stored in the register (AR) 151, is transferred and stored in the register (SR) 153 through the clipper circuit 152.

Note that the data stored in the register (SR) 153 will be stored as L-channel output data W (OTL) via the internal bus 123 at a relevant address of the work memory (W) 104 at the following step S906, and thereafter the data stored in the work memory (W) 104 will be output through the output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 306 of Fig. 5 is realized.

At the same step S905, the product obtained by the multiplier 145 (that is, the product of the L-channel multiplying coefficient P (PLL1) and the L-channel input data W (INL)), which has been stored in register (MR) 150, is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The effector (1) output multiplying coefficient P (ER) set in the register (MO) 141 is supplied to the multiplier 145 while the effector (1) output channel data (2) W (EO2) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data (2) W (EO2) by the coefficient P (ER) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 307 of Fig. 5 is realized.

Further at the same step S905, E-channel multiplying coefficient P (E1) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while E-channel input data W (INE) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S906, as described above, the data stored in the register (SR) 153 is read out and stored as 1-channel output data W (OTL) via the internal bus 123 at a relevant address of the work memory (W) 104.

Then, the data which has been obtained and stored in the register (MR) 150 at step S905 (that is, the product of the effector (1) output multiplying coefficient P (ER) and the effector (1) output channel data (2) W (EO2)) is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146 while the product which has been obtained by the multiplier 145 and transferred to the register (AR) 151 (that is, the product of R-channel multiplying coefficient P (RR2) and R-channel input data W (INR)) is transferred through the gate 149 to the other one of the input terminals of the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 308 of Fig. 5 is realized.

Further, at the same step S906, the effector (2) output multiplying coefficient P (F1) is read out

from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the effector (2) output channel data (1) W (FO1) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S907, the data which has been obtained by the multiplier 145 and stored in the register (AR) 151 is read out and stored in the register (SR) 153 through the clipper circuit 152. Note that the data stored in the register (SR) 153 will be stored as R-channel output data W (OTR) via the internal bus 123 at a relevant address of the work memory (W) 104 at the following step S908, and thereafter the data stored in the work memory (W) 104 will be output through the output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 308 of Fig. 5 is realized.

At the same step S907, the product which has been obtained by the multiplier 145 and stored in register (MR) 150 (that is, the product of the E-channel multiplying coefficient P (E1) and the E-channel input data W (INE)) is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The effector (2) output multiplying coefficient P (F1) set in the register (MO) 141 is supplied to the multiplier 145 while the effector (2) output channel data (1) W (FO1) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data (1) W (FO1) by the coefficient P (F1) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 311 of Fig. 5 is realized.

Further at the same step S907, T-channel multiplying coefficient P (T2) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while T-channel input data W (INT) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S908, as described above, the data stored in the register (SR) 153 is read out and stored as R-channel output data W (OTR) via the internal bus 123 in the work memory (W) 104 at a relevant address.

Then, the data which has been obtained and stored in the register (MR) 150 at step S907 (that is, the product of the effector (2) output multiplying coefficient P (F1) and the effector (2) output channel data (1) W (FO1)) is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146 while the product which has been obtained by the multiplier 145 and transferred to the register (AR) 151 (that is, the product of E-channel multiplying coefficient P (E1) and E-channel input data W (INE)) is transferred through the gate 149 to the other one of the input terminals of

the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 312 of Fig. 5 is realized.

The T-channel multiplying coefficient P (T2) set in the register (MO) 141 is read out and transferred to the multiplier 145 while T-channel input data W (INT) set to the register (M1) 142 is transferred through the gate 147 to the multiplier 145. The multiplier 145 multiplies both the transferred data, and the resultant product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to that of the multiplier 316 of Fig. 5 is realized.

Further, at the same step S908, the effector (2) output multiplying coefficient P (F2) is read out from the coefficient memory (P) 103 and is stored in the register (MO) 141 while the effector (2) output channel data (2) W (FO2) is read out from the work memory (W) 104 and is stored in the register (M1) 142.

At the following step S909, the data which has been obtained by the multiplier 145 and stored in the register (AR) 151 is read out and stored in the register (SR) 153 through the clipper circuit 152. Note that the data stored in the register (SR) 153 will be transferred to and stored as 1-channel output data W (OT1) via the internal bus 123 in the work memory (W) 104 at a relevant address at the following step S908, and thereafter the data stored in the work memory (W) 104 will be output through the output process to be described later. Through these processes, a process having a function equivalent to the function of the multiplier 312 of Fig. 5 is realized.

At the same step S909, the product which has been obtained by the multiplier 145 and stored in register (MR) 150 (that is, the product of the T-channel multiplying coefficient P (T2) and the T-channel input data W (INT)) is transferred to the register (AR) 151 through the gate 148 and the adder/subtractor 146.

The effector (2) output multiplying coefficient P (F2) set in the register (MO) 141 is supplied to the multiplier 145 while the effector (2) output channel data (2) W (FO2) is supplied through the gate 147 to the multiplier 145. The multiplier 145 multiplies the data (1) W (FO2) by the coefficient P (F2) and the product is stored in the register (MR) 150. Through these processes, a process having a function equivalent to the function of the multiplier 313 of Fig. 5 is realized.

At the following step S910, as described above, the data stored in the register (SR) 153 is read out and stored as 1-channel output data W (OT1) via the internal bus 123 in the work memory

(W) 104 at a relevant address.

Then, the data which has been obtained and stored in the register (MR) 150 at step S909 (that is, the product of the effector (2) output multiplying coefficient P (F2) and the effector (2) output channel data (2) W (FO2)) is transferred through the gate 148 to one of the input terminals of the adder/subtractor 146 while the product which has been obtained by the multiplier 145 and transferred to the register (AR) 151 (that is, the product of T-channel multiplying coefficient P (T2) and T-channel input data W (INT)) is transferred through the gate 149 to the other one of the input terminals of the adder/subtractor 146. The adder/subtractor 146 adds both data, and the sum obtained by the adder/subtractor 146 is stored in the register (AR) 151. Through the above processes, a process having a function equivalent to the function of the adder 314 of Fig. 5 is realized.

At the following step S911, the sum which has been obtained by the adder/subtractor 146 and stored in the register (AR) 151 is read out and stored in the register (SR) 153 through the clipper circuit 152. Data to be stored in the register (SR) 153 at step S912 will be transferred to and stored as 2-channel output data W (OT2) via the internal bus 123 in the work memory (W) 104 at a relevant address.

The data which is stored in the work memory (W) 104 will be output through an output process as will be described later. Through these processes, a process having a function equivalent to the function of the adder 314 of Fig. 5 is realized.

Through the above processes, a function equivalent to the function of the mixing process (3B) 303 is realized.

Fig. 14 is a flow chart of the output process executed at step S207.

In Fig. 14, at step S1001, the L-channel output data W (OTL) is read out from the work memory (W) 104 and is stored in the output register (OR1) 154. The data stored in the output register (OR1) 154 is further supplied to D/A convertor 8 of Fig. 1. Through these processes, a process having a function which outputs data from the multipliers 214 of Fig. 4 and 312 of Fig. 5 is realized.

At step S1002, the R-channel output data W (OTR) is read out from the work memory (W) 104 and is stored in the output register (OR2) 155. The data stored in the output register (OR2) 155 is further supplied to D/A convertor 9 of Fig. 1. Through these processes, a process having a function which outputs data from the adders 214 of Fig. 4 and 308 of Fig. 5 is realized.

At step S1003, the 1-channel output data W (OT1) is read out from the work memory (W) 104 and is stored in the output register (OR1) 154. The data stored in the output register (OR1) 154 is

further supplied to D/A convertor 8 of Fig. 1. Through these processes, a process having a function which outputs data from the adders 216 of Fig. 4 and 312 of Fig. 5 is realized.

At step S1004, the 2-channel output data W (OT2) is read out from the work memory (W) 104 and is stored in the output register (OR2) 155. The data stored in the output register (OR2) 155 is further supplied to D/A convertor 9 of Fig. 1. Through these processes, a process having a function which outputs data from the adders 210 of Fig. 4 and 314 of Fig. 5 is realized.

The present embodiment uses DSP (a digital signal processor) for the effect adding process in which an audio signal is processed in a digital fashion. With use of a combination of such DSP and other digital signal processing device, various musical tones other than the above musical tone signals, which are generated by an electronic musical instrument, may be applied with sound effects.

The embodiment has been described, in which the effect adding apparatus according to the present invention is applied to an electronic musical, but the embodiment is simply illustrative and not restrictive. The effect adding apparatus of the invention may be applied to other audio equipment other than an electronic musical instrument in a wide variety of fields.

The above described embodiments of the present invention are simply illustrative and not restrictive, and the present invention therefore may be modified in various manners. All the modification and applications of the present invention are within the scope and spirit of the invention, so that the scope of the invention should be determined only by what is recited in the appended claims and their equivalents.

Claims

1. An effect adding apparatus comprising:
 - effect-algorithm memory means (2) for storing a plurality of effect-algorithms for applying sound effects to an input audio signal;
 - combination-algorithm memory means (2) for storing a plurality of combination-algorithms for combining in different states the plurality of effect-algorithms stored in said effect-algorithm memory means;
 - algorithm combining means (1, 2, 3, 5) for reading out of the combination-algorithms from said combination-algorithm memory means (2), then selectively reading out effect-algorithms from said effect-algorithm memory means (2) in accordance with the read out combination-algorithm, and combining the read out effect-algorithms in accordance with the read out combination-algorithm; and

effect adding means (4) for applying relevant sound effects to the input audio signal based on the effect-algorithm combined by said algorithm combining means (1, 2, 3, 5).

2. An effect adding apparatus according to claim 1, wherein said effect-algorithms stored in said effect-algorithm memory means (2) and said combination-algorithm stored in said combination-algorithm memory means (2) are written in programs, and said algorithm combining means (1, 2, 3, 5) writes a combined program by means of combining the programs representative of the read out effect-algorithms in accordance with the program representative of the read out combination-algorithm. 5
3. An effect adding apparatus according to claim 1, wherein said effect-algorithm memory means (2) and said combination-algorithm memory means (2) are included in one and the same readable memory means (2). 10
4. An effect adding apparatus according to claim 1, wherein said effect adding means (4) comprises program memory means (101) for storing the effect algorithm combined by said algorithm combining means (1, 2, 3, 5) and signal processing means (145, 146) for performing an arithmetic process on the input audio signal in accordance with the program stored in said program memory means. 15
5. An effect adding apparatus according to claim 1, wherein said algorithm combining means (1, 2, 3, 5) comprises central processing means (1) for reading out a combination-algorithm from said combination-algorithm memory means (2) and effect-algorithms from said effect-algorithm memory means (2), and for combining the read out effect-algorithms in accordance with the read out combination-algorithm, and transferring the effect-algorithms to said effect adding means (4). 20
6. An effect adding apparatus according to claim 1, wherein said algorithm combining means (1, 2, 3, 5) includes externally operable means (5), wherein a combination-algorithm and effect-algorithms are selected from among those stored in said combination-algorithm memory means (2) and said effect-algorithm memory means (2) in accordance with operation of said externally operable means (5). 25
7. An effect adding apparatus according to claim 1, wherein said effect-algorithm memory means (2) stores a first effect-algorithm for 30

applying a first sound effect and a second effect-algorithm for applying a second sound effect, and said combination-algorithm memory means (2) stores a first combination-algorithm for combining in series said first effect-algorithm and said second effect-algorithm, and a second combination-algorithm for combining in parallel said first effect-algorithm and said second effect-algorithm. 35

8. An effect adding apparatus according to claim 7, wherein said algorithm combining means (1, 2, 3, 5) includes externally operable means (5), wherein the first combination-algorithm and a combination of the first effect-algorithm and the second effect-algorithm, or the second combination-algorithm and a combination of the first effect-algorithm and the second effect-algorithm are selected in accordance with operation of said externally operable means (5). 40
9. An effect adding apparatus for applying a plurality of sound effects to an input audio signal, comprising:
 - effect-algorithm memory means (2) for storing a plurality of effect-algorithms for applying different sound effects to the input audio signal; 45
 - combination-data memory means (2) for storing a plurality of combination-data which represent combinations in different states of a plurality of effect-algorithms stored in said effect-algorithm memory means (2);
 - algorithm combining means (1, 2, 3, 5) for reading out a combination-data from said combination-data memory means (2), then selectively reading out relevant effect-algorithms from said effect-algorithm memory means (2) in accordance with the read out combination-data, and combining the read out effect-algorithms; and
 - effect adding means (4) for applying relevant sound effects to the input audio signal in accordance with the effect-algorithm combined by said algorithm combining means (1, 2, 3, 5). 50
10. An effect adding apparatus according to claim 9, wherein said effect-algorithms stored in said effect-algorithm memory means (2) are written in programs, and said algorithm combining means (1, 2, 3, 5) writes a combined program by means of combining said programs representative of the read out effect-algorithms in accordance with said combination-data. 55
11. An effect adding apparatus according to claim 9, wherein said effect-algorithm memory 55

means (2) and said combination-data memory means (2) are included in one and the same readable memory means (2).

tion of said externally operable means (5).

12. An effect adding apparatus according to claim 9, wherein said effect adding means (4) comprises program memory means (101) for storing effect-algorithm combined by said algorithm combining means (1, 2, 3, 5) and signal processing means (145, 146) for performing an arithmetic process on the input audio signal in accordance with the program stored in said program memory means (101). 5
10
13. An effect adding apparatus according to claim 9, wherein said algorithm combining means (1, 2, 3, 5) comprises central processing means (1) for reading out a combination-data from said combination-data memory means (2) and effect-algorithms from said effect-algorithm memory means (2), and for combining the read out effect-algorithms in accordance with the read out combination-data, and transferring the effect-algorithms to said effect adding means (4). 15
20
25
14. An effect adding apparatus according to claim 9, wherein said algorithm combining means (1, 2, 3, 5) includes externally operable means (5), wherein a combination-data and effect-algorithms are selected from among those stored in said combination-data memory means (2) and said effect-algorithm memory means (2) in accordance with operation of said externally operable means (5). 30
35
15. An effect adding apparatus according to claim 9, wherein said effect-algorithm memory means (2) stores a first effect-algorithm for applying a first sound effect and a second effect-algorithm for applying a second sound effect, and said combination-data memory means (2) stores a first combination-data for combining in series said first effect-algorithm and said second effect-algorithm, and a second combination-data for combining in parallel said first effect-algorithm and said second effect-algorithm. 40
45
16. An effect adding apparatus according to claim 15, wherein said algorithm combining means (1, 2, 3, 5) includes externally operable means (5), wherein the first combination-data and a combination of the first effect-algorithm and the second effect-algorithm, or the second combination-data and a combination of the first effect-algorithm and the second effect-algorithm are selected in accordance with opera-

FIG.1

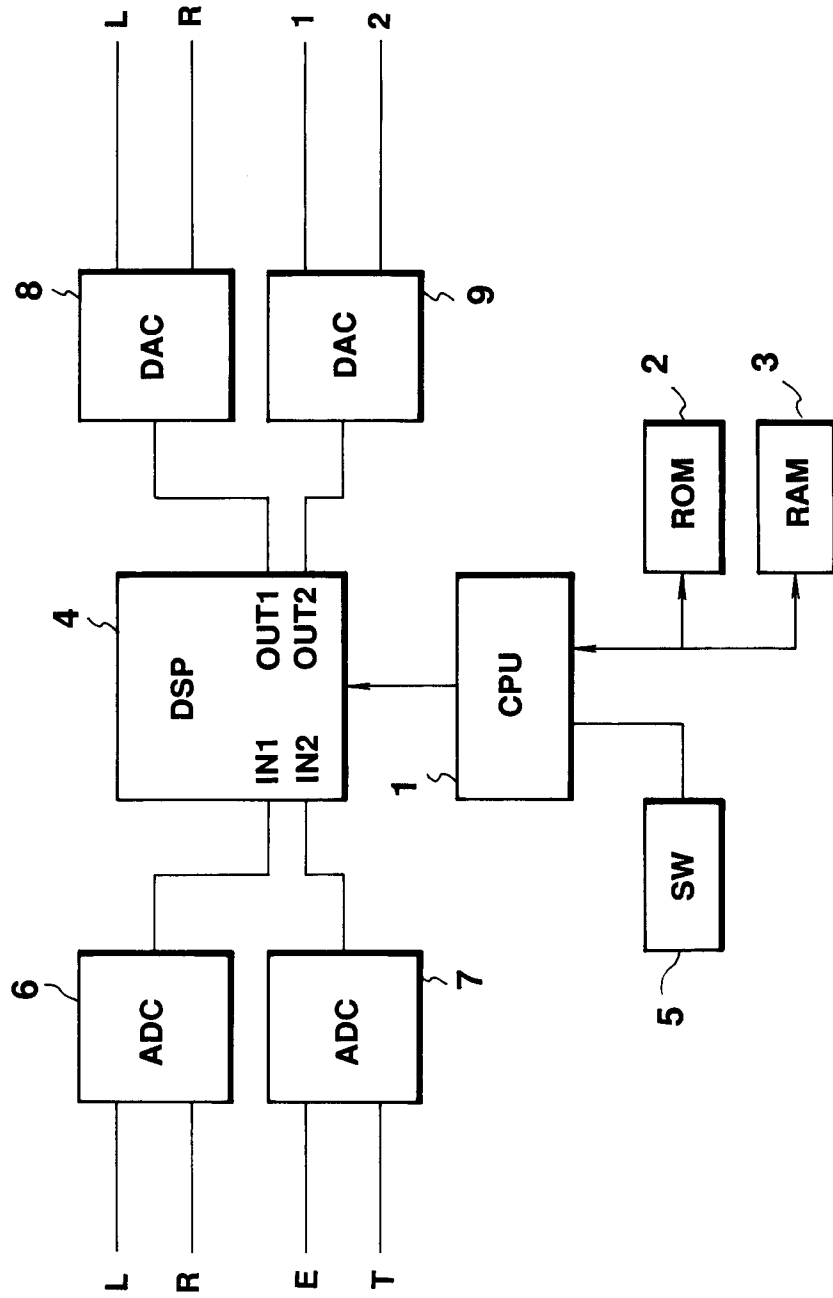


FIG. 2

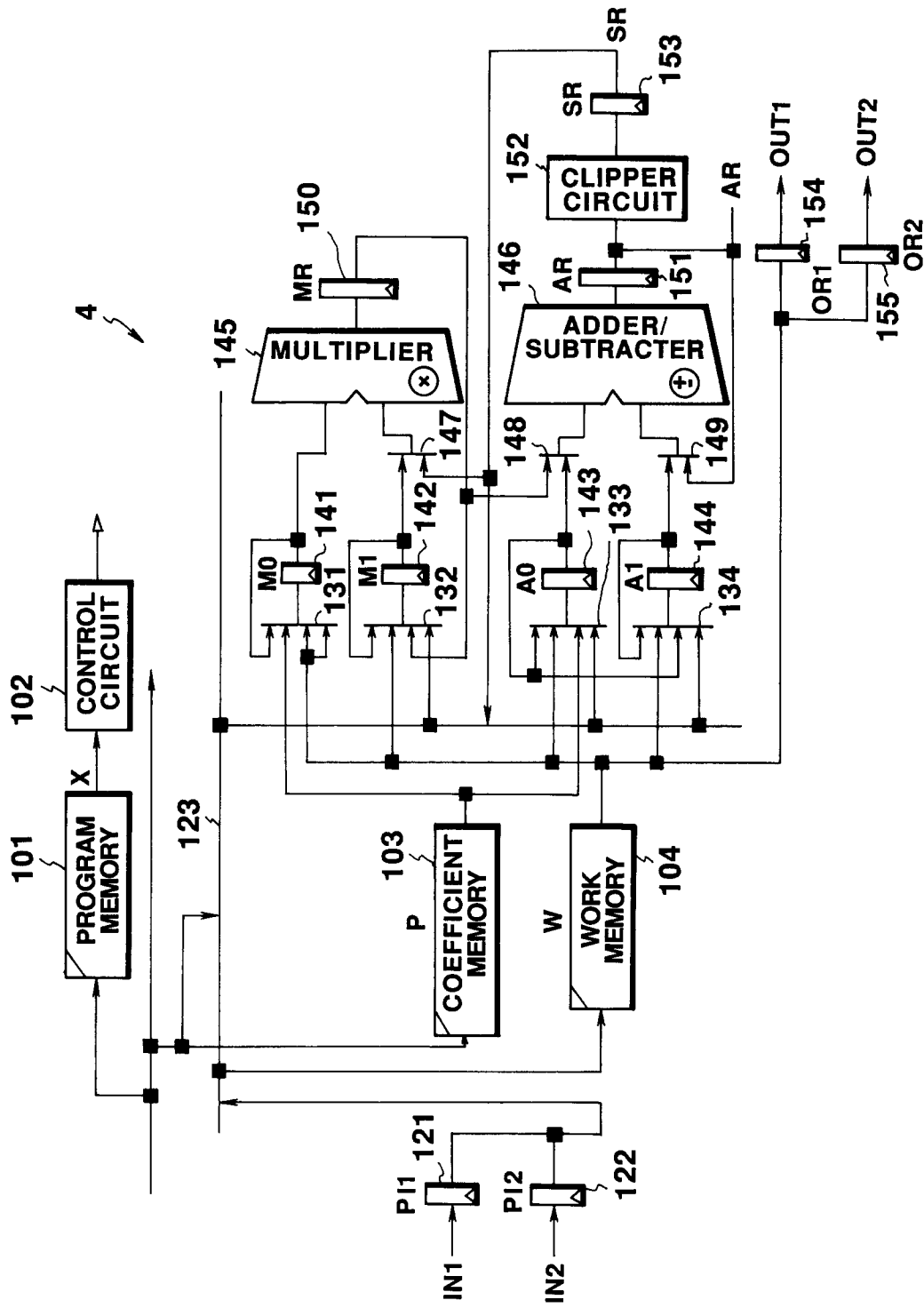


FIG. 3

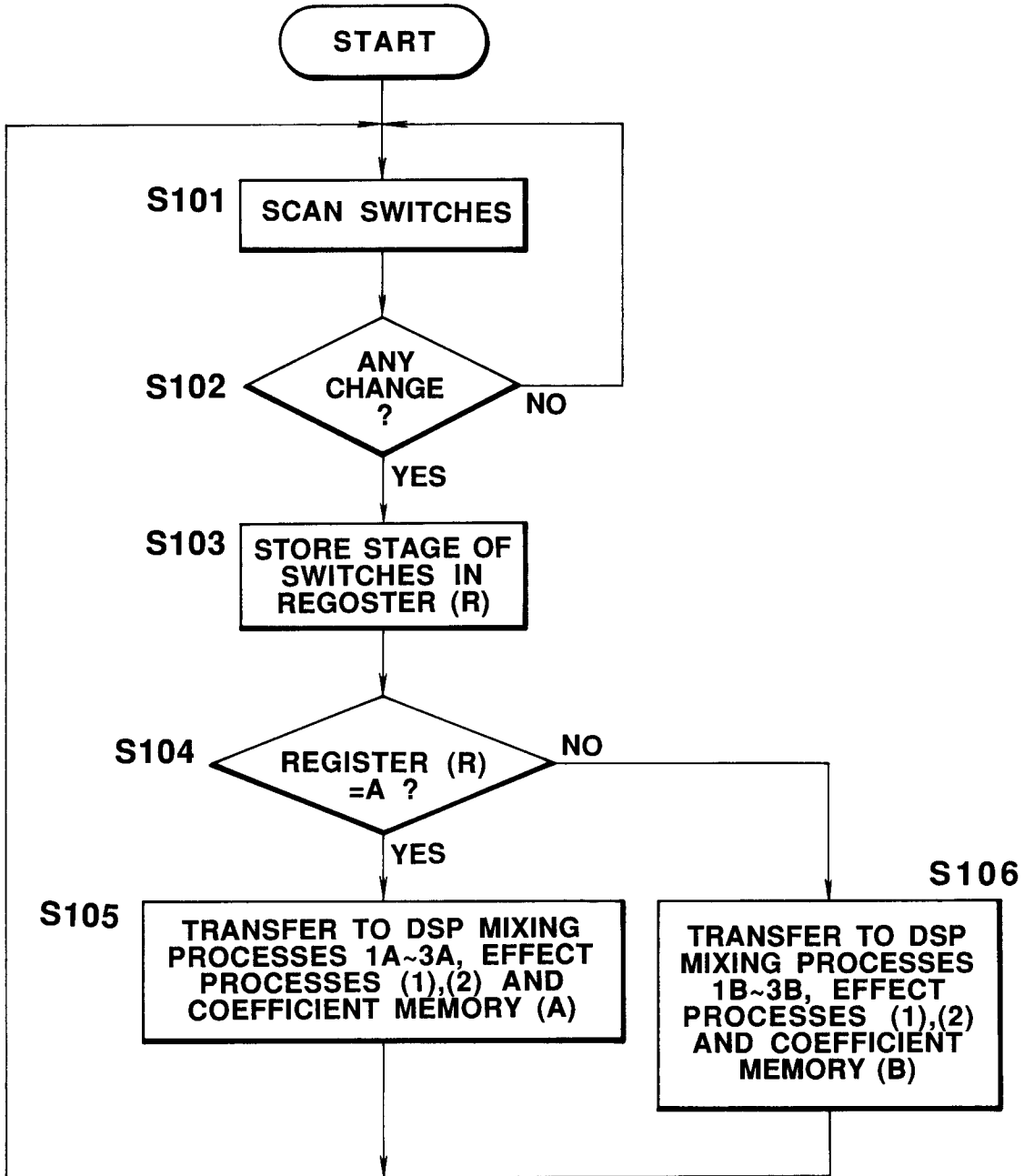


FIG. 4

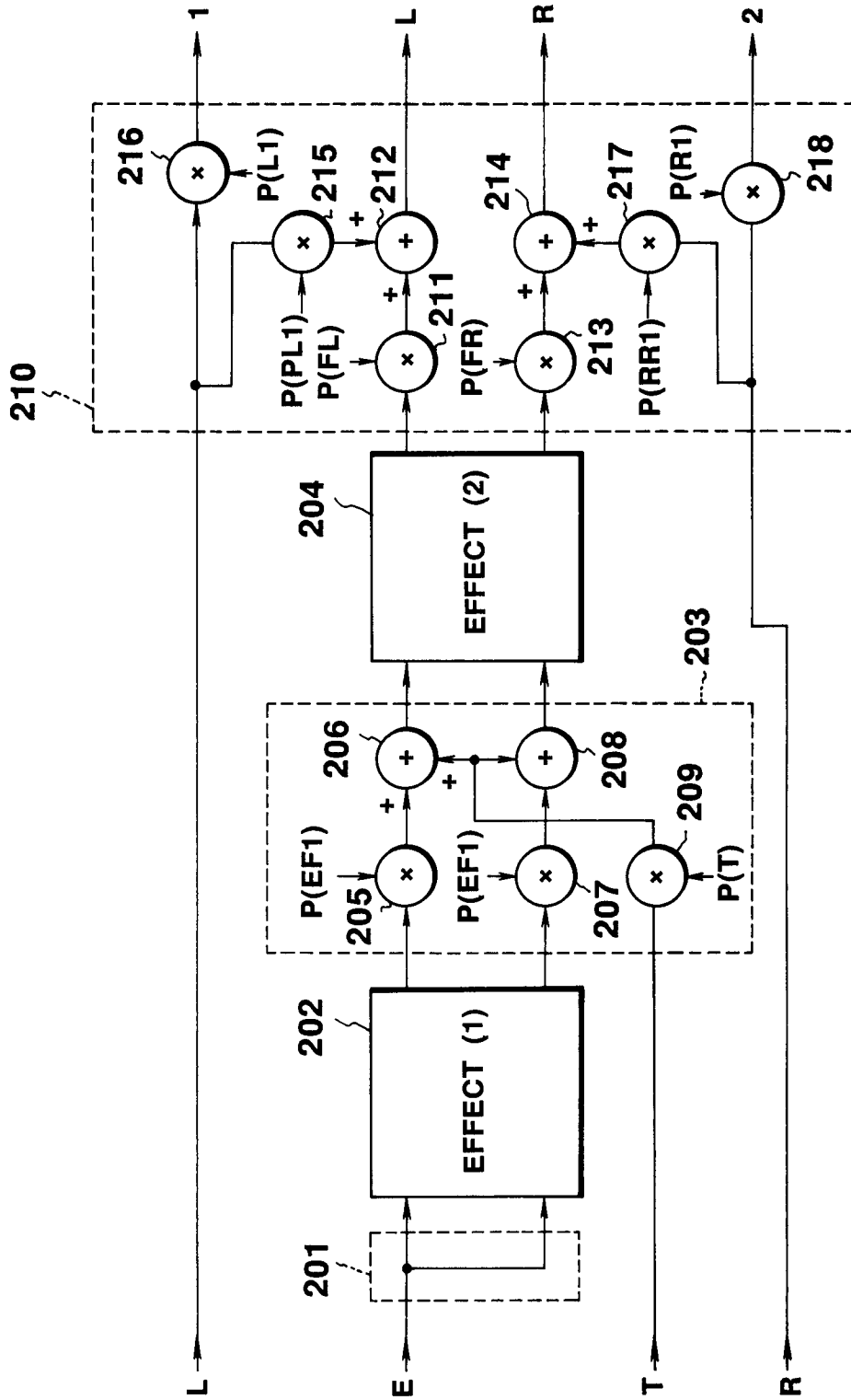


FIG. 5

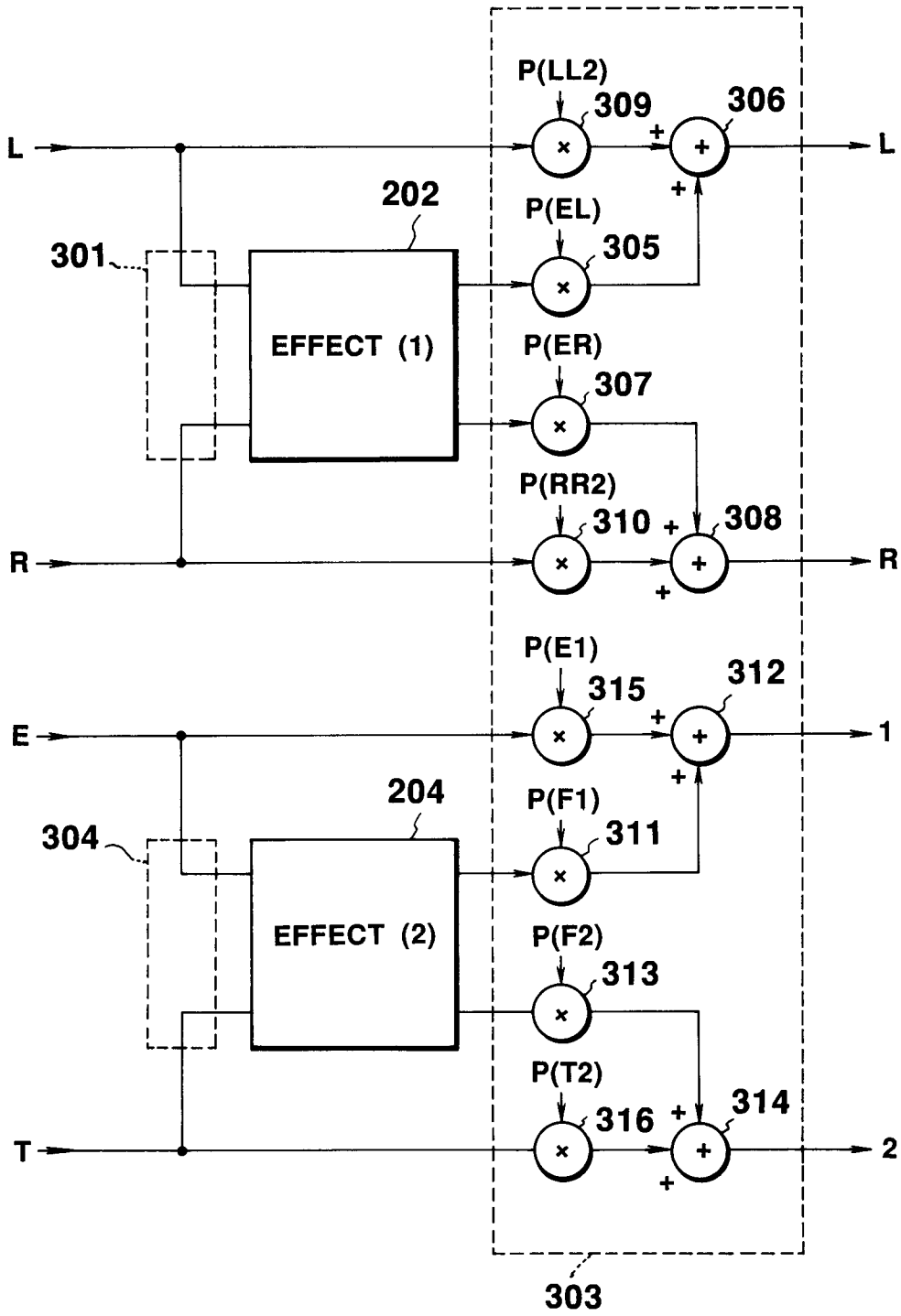


FIG. 6

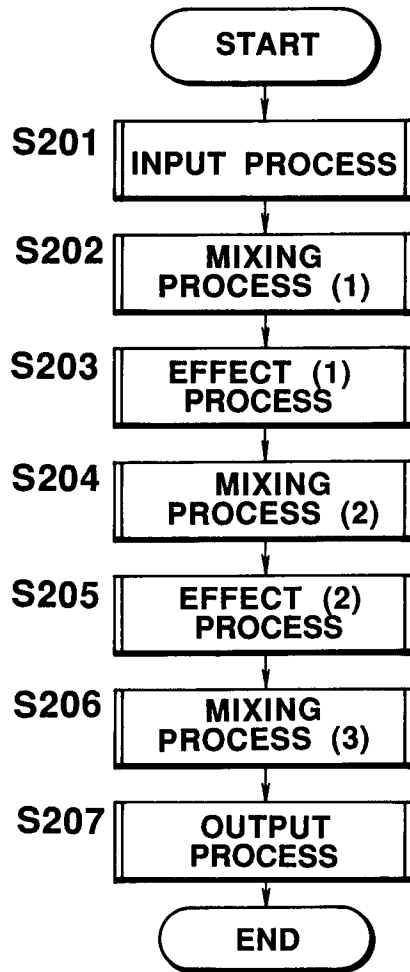


FIG. 7

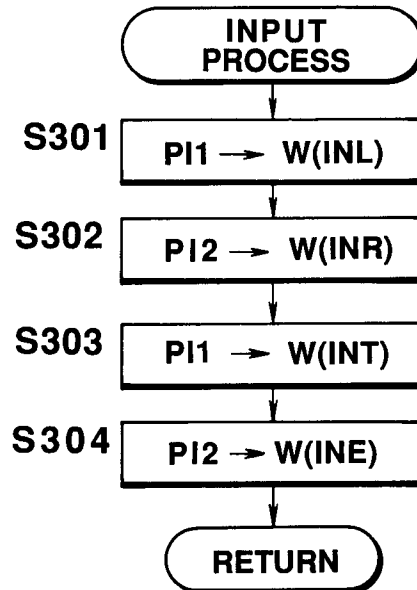


FIG. 8

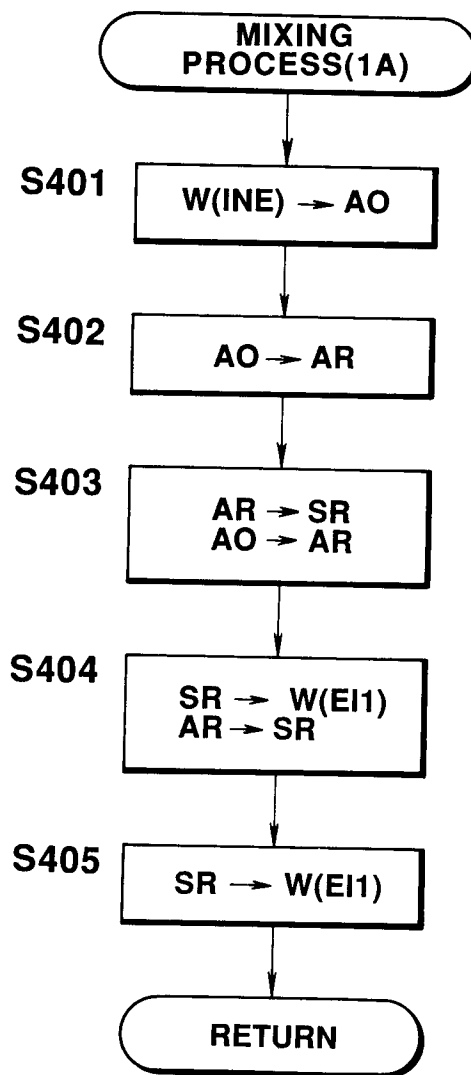


FIG. 9

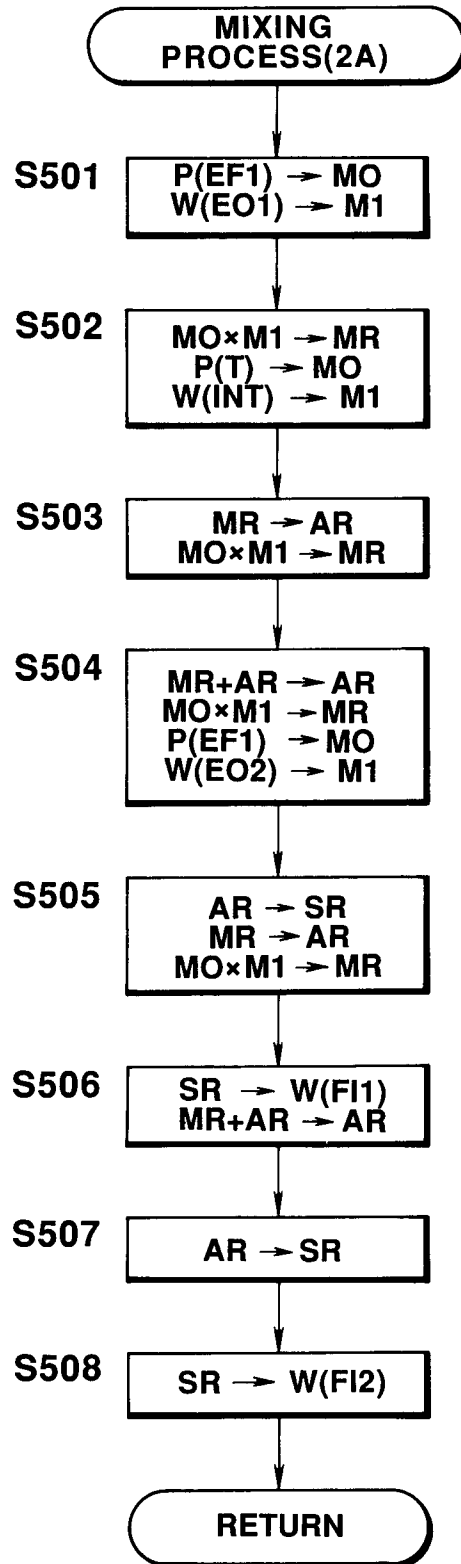


FIG.10

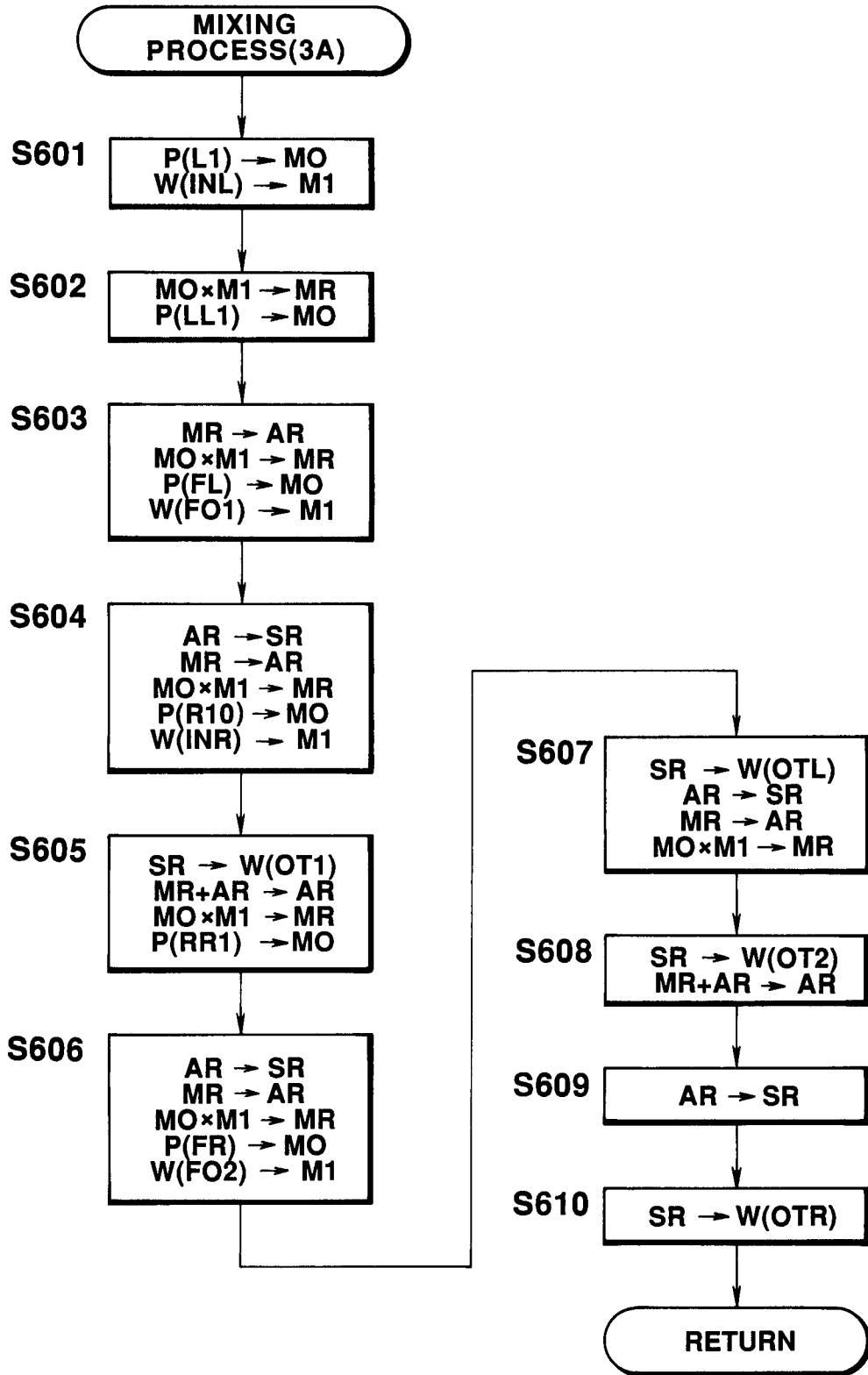


FIG.11

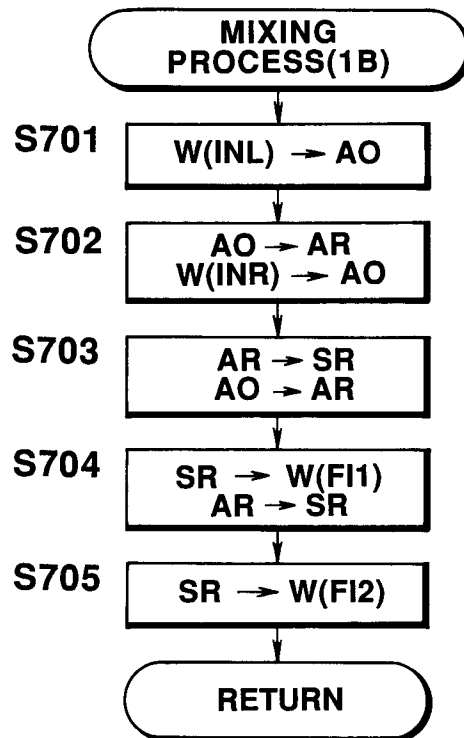


FIG.12

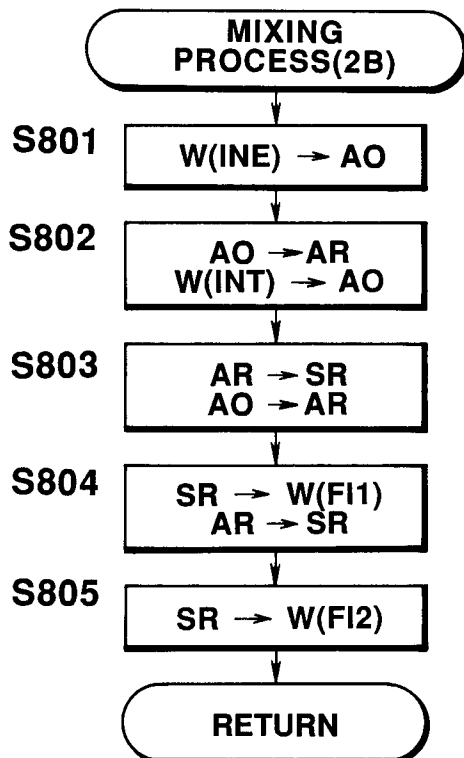


FIG.13

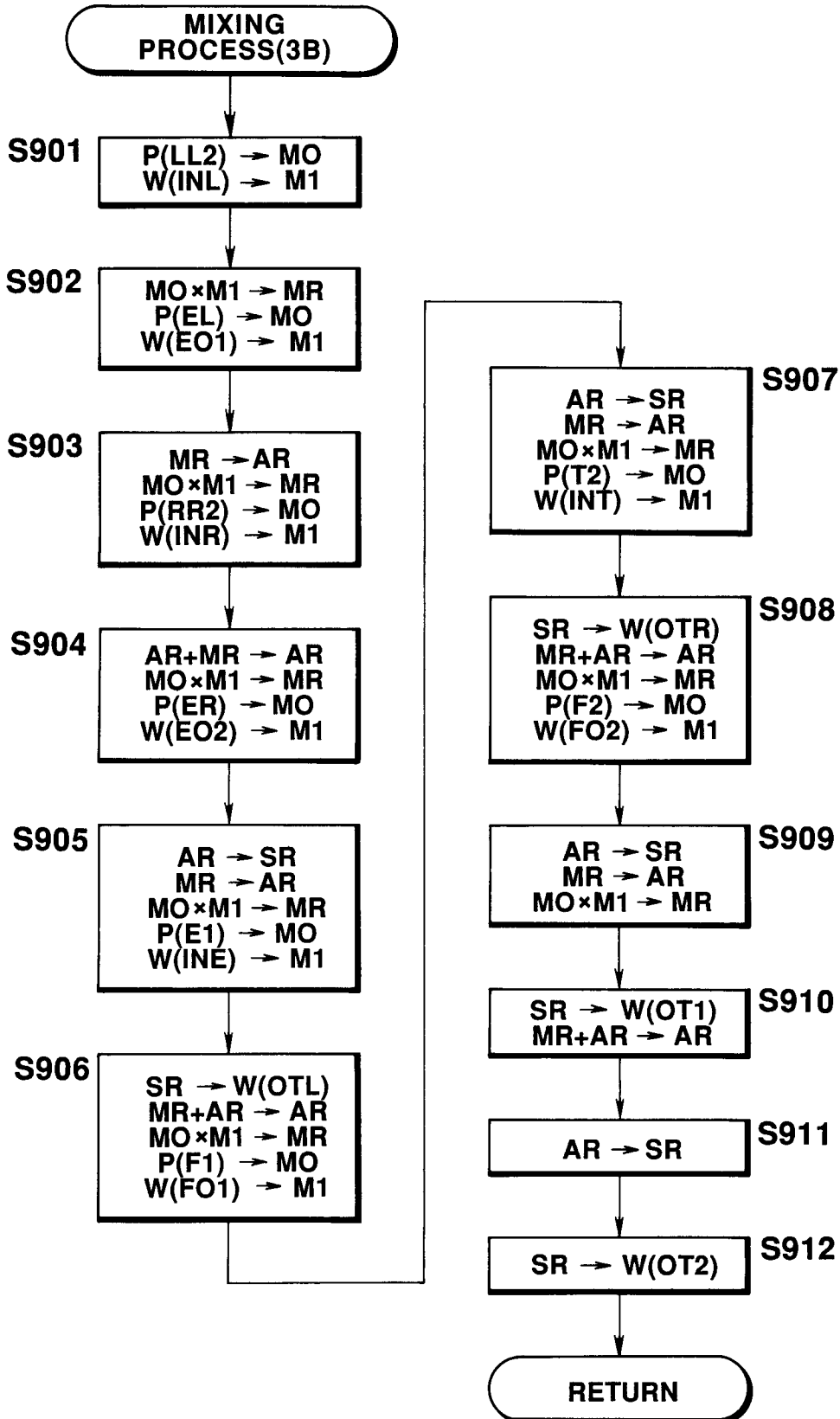


FIG. 14

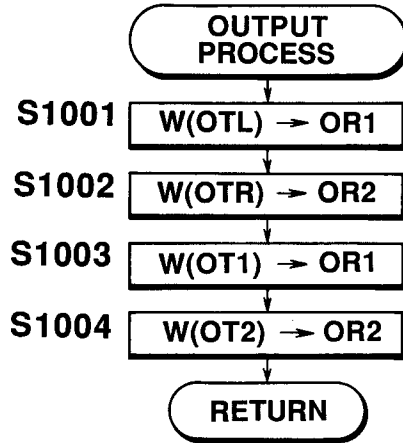


FIG. 15

ADDRESSES	COEFFICIENTS	CONTENTS
0	W(INL)	L CHANNEL INPUT DATA
1	W(INR)	R CHANNEL INPUT DATA
2	W(INT)	T CHANNEL INPUT DATA
3	W(INE)	E CHANNEL INPUT DATA
4	W(EI1)	EFFECTOR1 INPUT CHANNEL DATA1
5	W(EI2)	EFFECTOR1 INPUT CHANNEL DATA2
6	W(EO1)	EFFECTOR1 OUTPUT CHANNEL DATA1
7	W(EO2)	EFFECTOR1 OUTPUT CHANNEL DATA2
8	W(FI1)	EFFECTOR2 INPUT CHANNEL DATA1
9	W(FI2)	EFFECTOR2 INPUT CHANNEL DATA2
10	W(FO1)	EFFECTOR2 OUTPUT CHANNEL DATA1
11	W(FO2)	EFFECTOR2 OUTPUT CHANNEL DATA2
12	W(OTL)	L CHANNEL OUTPUT DATA
13	W(OTR)	R CHANNEL OUTPUT DATA
14	W(OTT)	T CHANNEL OUTPUT DATA
15	W(OTE)	E CHANNEL OUTPUT DATA
16	W(OT1)	1 CHANNEL OUTPUT DATA
17	W(OT2)	2 CHANNEL OUTPUT DATA

FIG.16 (a)

ADDRESSES	COEFFICIENTS	CONTENTS
0	P(EF1)	EFFECTOR1 OUTPUT MULTIPLYING COEFFICIENT
1	P(T)	T CHANNEL MULTIPLYING COEFFICIENT
2	P(PL1)	L CHANNEL MULTIPLYING COEFFICIENT
3	P(PLL1)	L CHANNEL MULTIPLYING COEFFICIENT
4	P(FL)	EFFECTOR2 OUTPUT MULTIPLYING COEFFICIENT
5	P(FR)	EFFECTOR2 OUTPUT MULTIPLYING COEFFICIENT
6	P(RR1)	R CHANNEL MULTIPLYING COEFFICIENT
7	P(RR)	R CHANNEL MULTIPLYING COEFFICIENT

FIG.16 (b)

ADDRESSES	COEFFICIENTS	CONTENTS
0	P(LL2)	L CHANNEL MULTIPLYING COEFFICIENT
1	P(EL)	EFFECTOR1 OUTPUT MULTIPLYING COEFFICIENT
2	P(ER)	EFFECTOR1 OUTPUT MULTIPLYING COEFFICIENT
3	P(RR2)	R CHANNEL MULTIPLYING COEFFICIENT
4	P(E1)	E CHANNEL MULTIPLYING COEFFICIENT
5	P(F1)	EFFECTOR2 OUTPUT MULTIPLYING COEFFICIENT
6	P(F2)	EFFECTOR2 OUTPUT MULTIPLYING COEFFICIENT
7	P(T2)	T CHANNEL MULTIPLYING COEFFICIENT