

Thus, in the operating mode setup in step Sa1, the operating mode is set up and recognized before executing the waveform synthesizing program.

[0026] In step Sa2, waveform loading process is executed. In the waveform loading process, typical or basic waveforms are loaded into the area WAVE allocated in the RAM 13 or 20 in case that the basic waveform is used in the waveform memory readout mode. In step Sa3, it is tested whether a flag SETFLG is "1", or not. The flag SETFLG is initially set to "0", but may be turned to "1", if the sampling frequency f_s is set up in step Sa21, or if the waveform memory readout mode is designated in a backup waveform computation mode. The procedure advances forward to step Sa4 if the flag SETFLG is "1". Otherwise, if the flag SETFLG is "0", the procedure jumps to step Sa5. In case that the waveform synthesizing program is executed in the loop shown in Figure 7 at the first time, the flag SETFLG is "0", and the procedure unconditionally advances forward to step Sa5. However, the process of step Sa4 is also described here, for convenience of description. In step Sa4, it is tested whether the sound synthesizing should be carried out entirely by the CPU synthesizing mode or not. It is possible to use the result of the hardware check in step Sa1 to detect whether any external sound source device is recognized or not. If there is possibility of using any sound source other than CPU for the waveform synthesizing, step Sa4 branches to "No" direction. In step Sa5, a flag ENBLFLG is tested if it is "1" or not, in order to detect a nonoperable state. The nonoperable state means that the sampling frequency f_s is neither set up in the waveform generation procedure, nor the backup waveform computation mode is enabled. Therefore, the CPU synthesizing mode is not yet ready in the nonoperable state.

[0027] When the waveform synthesizing program is executed in the main loop shown in Figure 7 at the first time, the flag ENBLFLG is "0" so that the procedure advances forward to step Sa11. However, the process under an operable state in case that the flag ENBLFLG is "1" is described here just for convenience of description. The operable state means that all the preparation for the CPU synthesizing is already completed. In this state, the procedure advances forward to step Sa6, where processing of performance information is done. In step Sa7, existence of a CPU waveform generation command is checked. The CPU waveform generation command is generated in response to a key-on event contained in the keyboard information KBD, MIDI information or performance information fed from the I/F under the CPU synthesizing mode. If the CPU waveform generation command is detected in step Sa7, the procedure responsive to the command is executed in step Sa8. In the procedure of step Sa8, the wave data is generated by a specific synthesizing mode selected out of the available CPU synthesizing modes. The wave data is then distributed to the D/A converter 23 via the bus 12. Thus, the sound generation is accomplished

according to the synthesized wave data or waveform.

[0028] In broader definition, the CPU waveform generation command may include a key-off event commanding note-off, though explanation for the procedures relevant to the key-off event is omitted here. These note-off procedures are very simple process such as release of the waveform generation and termination of the waveform generation. If no CPU waveform generation command is detected in step Sa7, the waveform synthesizing computation in step Sa8 is skipped, since there is no job to be executed. In step Sa9, it is tested whether terminating operation of the waveform synthesizing program is conducted by the user or not. If the termination of the program is not indicated, the procedure immediately returns to prepare for a next CPU waveform generation command. On the other hand, in case that the termination command is inputted, the waveform synthesizing program is terminated by setting the flag SETFLG to "0" in step Sa10.

[0029] If the waveform synthesizing program is executed at the first time in the main loop of Figure 7, the flag ENBLFLG is "0". Then, in step Sa11, it is tested whether the allocation mode is either of the CPU select mode and the sound source select mode. If the allocation mode is the manual mode or compulsory mode, the detection result is "No" so that the procedure advances forward to step Sa12, where the flags ENBLFLG, DACENBL, and SETFLG are all set to "1". Then, the procedure returns. The flag DACENBL is set to "1" to enable the D/A converter 23. Thus, if the waveform synthesizing program is executed next time and the external sound source is used, the check of step Sa3 results in "Yes" and the check of step Sa4 results in "No" so that the procedure shown in Figure 11 is executed. Otherwise steps Sa6 to Sa10 are executed if the waveform synthesizing is effected by only the CPU.

[0030] On the other hand, when it is detected that the allocation mode is either of the CPU or sound source select mode in step Sa11, the procedure advances forward to step Sa13. The processing in steps Sa13 to Sa27 shown in Figures 9 and 10 may be done if the waveform generation program is invoked at the first time. In this process, the sampling frequency f_s is determined for synthesis only by the CPU. In step Sa13, the flag ENBLFLG is switched to "0", a flag BUSY is set to "1", and the flag DACENBL is set to "0" explicitly, the flag BUSY is set to "1" to enable counting up in timer process described later. Setting of the flag DACENBL to "0" is done in order to disable the output operation of the D/A converter 23 to prevent sound generation during the trial waveform computation described later. After step Sa13, registers SCOUNT and TCOUNT are reset to "0". The content of the register SCOUNT indicates loop cycles of the following trial waveform computation. The register TCOUNT incrementally counts up while the flag BUSY is switched to "1". Thus, this register indicates a lapse time required to generate one waveform by m cycles of the synthesis computation. In step Sa15,