(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 264 955 B1**

(12) EUROPEAN PATENT SPECIFICATION

(45) Date of publication of patent specification: **17.03.93**

(51) Int. Cl.^{5}: **G10H 3/00**, G10H 1/00

(21) Application number: **87115594.1**

(22) Date of filing: **23.10.87**

(54) Apparatus for determining the pitch of a substantially periodic input signal.

(30) Priority: **24.10.86 JP 253487/86**
28.11.86 JP 282142/86
28.11.86 JP 283292/86
02.12.86 JP 285985/86
03.12.86 JP 286745/86
12.01.87 JP 4714/87
06.03.87 JP 50381/87

(43) Date of publication of application:
27.04.88 Bulletin 88/17

(45) Publication of the grant of the patent:
17.03.93 Bulletin 93/11

(84) Designated Contracting States:
DE FR GB IT

(56) References cited:
GB-A- 2 138 988
US-A- 4 627 323

(73) Proprietor: **CASIO COMPUTER COMPANY LIMITED**
6-1, 2-chome, Nishi-Shinjuku
Shinjuku-ku Tokyo 160(JP)

(72) Inventor: **Uchiyama, Shigeru Pat. Dept. Dev. Hamura**
R&D Center CASIO COMP. CO. LTD. 3-2-1, Sakae-cho
Hamura-machi Nishitama-gun Tokyo 190-11(JP)
Inventor: **Obata, Katsuhiko Pat. Dept. Dev. Hamura**
R&D Center CASIO COMP. CO. LTD. 3-2-1, Sakae-cho
Hamura-machi Nishitama-gun Tokyo 190-11(JP)

(74) Representative: **KUHNEN, WACKER & PARTNER**
Alois-Steinecker-Strasse 22 Postfach 1553
W-8050 Freising (DE)

EP 0 264 955 B1

Description

The present invention relates to an apparatus for determining the pitch of a substantially periodic input signal, and more particularly to an electronic string instrument such as an electronic guitar or a guitar synthesizer incorporating such an apparatus.

In recent days, there have been developed musical instruments of the type in which a pitch (frequency) is extracted from a waveform signal generated by a natural or conventional musical instrument and, under control of the extracted pitch, a sound source of an electronic circuitry is driven to artificially generate a sound such as a musical tone.

In this type of musical instruments, the pitch extraction still involves some problems to urgently be solved. Typical pitch extraction systems are a zero-cross point detection system and a peak detection system.

The zero-cross point detection system detects the time intervals between zero-cross points in the input waveform, and uses them as periods of the artificial sound. The input waveform frequently contains harmonics which should have been removed by filters such as low-pass filters. The detection system does not operate well for such input waveforms. If it is applied, the detected pitches contain many errors. In order to prevent these errors, complicated software processings, for example, to check the duty of the input waveform, are required. It is technically difficult to realize this.

The peak-detection system detects maximum and minimum peak points of the input waveform signal, and sets and resets a flip-flop at the peak points, to generate a period signal, e.g., a rectangular wave signal. Peak detection systems are disclosed in KOKOKU Nos. 57-37074 (corresponding to US-A-4,117,757) and 57-58672, KOKAI Nos. 55-55398, 55-152597 (Utility Model), and 61-26090, and in US-A-4 627 323.

The peak detection system determines the period of the artificial sound waveform by the time interval between the adjacent maximum peaks, for example. This feature causes reduction of the reliability of the detected pitches, and a slow response to frequency change of the input natural sound waveform. Use of the R-S flip-flop makes the instrument circuitry inflexible in generic use and makes it difficult to construct so-called intelligent musical instruments having data processing functions.

Other proposals of a pitch detection are disclosed in KOKAI Nos. 55-87196 and 55-159495, and KOKOKU No. 61-51793.

In KOKAI 55-87196 it is suggested to measure the period of the input natural sound waveform, and then converts the measured value into a frequency number, which in turn is sent to the sound source. Any novel technical proposal for the period measurement is not disclosed in this specification.

KOKAI 55-159495 and KOKOKU 61-51793 disclose a frequency stabilizing technique in which, when the adjacent extracted periods are substantially the same, the sounding of the musical instrument starts. A sounding command is not sent to the sound source until at least two periods elapse. In this respect, the systems of these documents involve a response performance problem. To obtain a quick response, the sounding should start as soon as possible.

Utility Model KOKOKU No. 62-20871 (corresponding to US-A-4,606,255) discloses another frequency stabilizing technique. In a string musical instrument, a vibration of one string affects the vibration of another string. Extremely, the latter vibrates resonating at the vibration frequency of the former. The system proposed therein yields high costs because mechanical parts are used and, moreover, can imperfectly remove the resonance.

It is therefore the object of the present invention to improve a pitch determining apparatus according to the preamble of claim 1 in such a way that the pitch of a periodic input signal can be determined in a short time and with more precision.

According to the present invention this object is solved by the advantageous measures indicated in new claim 1.

Advantageous developments of the invention are subject-matter of the subclaims.

The invention will now be described in more detail with reference to the accompanying drawings; in which:

Fig. 1 illustrates in block form, an overall arrangement of a pitch determining apparatus;

Fig. 2 shows a circuit diagram of a maximum or positive peak detector used in the Fig. 1 circuit;

Fig. 3 shows a circuit diagram of a minimum or negative peak detector used in the Fig. 1 circuit;

Fig. 4 shows a timing chart useful in explaining the operation of the Figs. 2 and 3 circuits;

Fig. 5 shows a graphical representation of a cut-off frequency of a low-pass filter used in the Fig. 1 circuit;

Fig. 6 shows a flowchart of a main routine used by a CPU in the Fig. 1 circuit;

Fig. 7 shows a flowchart of an interrupt routine executed by the CPU when the maximum or positive peak point MAX is detected;

Fig. 8 shows a flowchart of an interrupt routine executed by the CPU when the minimum or negative peak point MIN is detected;

Fig. 9 shows waveforms of an input signal to the Fig. 1 circuit, and of flags;

Figs. 10 and 11 illustrate in block form an overall arrangement of another pitch determining apparatus;

Fig. 12 shows a circuit diagram of a zero-cross point detector used in the Fig. 10 circuit;

Fig. 13 shows a timing chart useful in explaining the operation of the Fig. 10 circuit when it receives an waveform of the input signal;

Fig. 14 shows a flowchart of an interrupt routine executed by the CPU of the Figs. 10 and 11 circuit when the zero-cross point is detected immediately after a maximum peak point is detected;

Fig. 15 shows a flowchart of an interrupt routine executed by the CPU of the Figs. 10 and 11 circuit immediately when the zero-cross point is detected after a minimum peak point is detected;

Fig. 16 shows a timing chart useful in explaining the operation of the Figs. 10 and 11 circuit when it receives another waveform of the input signal;

Fig. 17 illustrates in block form an overall arrangement of a first embodiment of the present invention;

Fig. 18 shows a flowchart of a main routine of a CPU in the Fig. 17 circuit;

Fig. 19 shows a timing chart useful in explaining the operation of the Fig. 17 circuit when it receives an waveform of the input signal;

Fig. 20 shows a flowchart of an interrupt routine executed by the CPU of the Fig. 17 circuit when the zero-cross point is detected immediately after a maximum or positive peak point is detected;

Fig. 21 shows a flowchart of an interrupt routine executed by the CPU of the Fig. 17 circuit immediately when the zero-cross point is detected after a minimum or negative peak point is detected;

Fig. 22 shows a timing chart useful in explaining the operation of the Fig. 17 circuit when it receives another waveform of the input signal;

Fig. 24 shows a block diagram of a modification of the Fig. 17 circuit;

Fig. 23 shows a timing chart for explaining the operation of the Fig. 24 circuit;

Fig. 25 shows a waveform possibly causing an erroneous operation in extracting the pitch of an input signal waveform, this erroneous operation problem being solved by a second embodiment of the present invention;

Figs. 26A and 26B show a main routine executed by a CPU in the second embodiment;

Fig. 27 shows a timing chart for explaining the operation of the second embodiment; and

Fig. 28 a circuit diagramm of the second embodiment.

At first, with reference to Figures 1 through 16, there will be described a pitch determining apparatus which does not contain the core features of the present invention. The construction and operation of that pitch determining apparatus is however believed to be useful for the complete understanding of the present invention.

An overall circuit arrangement of a musical instrument which is embodied in an electronic guitar is illustrated in Fig. 1. Vibrations of six strings of the electronic guitar are picked up by pickups (not shown). The pickups convert the mechanical vibrations of the strings into electrical signals. These signals as musical tone signals are input to six input terminals 1. These input signals are amplified by amplifiers 2, and applied to low-pass filters 3. These filters respectively remove the high frequency components of these signals in order to extract the fundamental waveform of these input signals. Each of the input signal of the fundamental waveform is applied to a maximum peak detector (MAX) 4 and a minimum peak detector (MIN) 5. Each low-pass filter 3 is designed so that, as shown in Fig. 5, the cut-off frequency is set at 4f, four times the frequency f of the vibration of the open string of each string, when each string is picked without the fret operation. Such selection of the cutoff frequency is due to the fact that the guitar has generally frets of about two octaves. Each of the maximum peak detectors 4 detects the maximum peak points of the input signal. The pulse signals representing the peak detections are used as interrupt command signals INTa1 to INTa6, and transferred to a CPU 6. Minimum peak detectors 5 detect each the minimum peak point of the input signal. The pulse signals representative of peak detections are transferred to CPU 6 in the form of interrupt command signals INTb1 to INTb6.

When CPU 6 receives the interrupt command signals INTa1 to INTa6 representing the maximum peak points detections appearing immediately after the interrupt command signals INTb1 to INTb6 representing the minimum peak points detections, CPU 6 calculates a count difference between the present maximum peak points and those previously obtained in a similar way. When CPU 6 receives the interrupt command signals INTb1 to INTb6 appearing immediately after the interrupt command signals INTa1 to INTa6, CPU 6 similarly obtains a count difference between the minimum peak points and those previously obtained in a similar way. Every time both the command signals INT are applied to counter 7, the counts of counter 7 are stored into a maximum memory 14 and a minimum memory 15, respectively.

The time count data representative of the difference of the counts as counted by counter 7 is transferred by CPU 6, directly or after it is converted into a pitch code, to a frequency ROM 8. The frequency data representing a frequency whose period is defined by this count data, is read out, and transferred to a sound source circuit 9. Sound source circuit 9 forms a musical tone signal based on the frequency data. Receiving this signal, sound system 10 sounds the musical tone. The sound source circuit 9 may be set inside or outside the guitar main frame.

Six musical tone signals from low-pass filters 3 are transferred through transfer gates T to an A/D converter 11. The A/D converter 11 converts these signals into digital data based on the waveform levels. The digital data is applied to CPU 6. When the absolute values of the waveform level data exceed a predetermined level, CPU 6 sends the data as obtained from the count of counter 7, to frequency ROM 8, to start the sounding. When the absolute values are below the predetermined value, CPU 6 terminates the sending of the data to frequency ROM 8, to stop the sounding. The waveform level data is also applied to a sound source circuit 9 to control the musical tone sounding level. This indicates that the volume is controllable according to the touch force to the string.

Staggered channel timing signals t1 to t6 are applied to the transfer gates T, so that the sounding levels of the six musical tone signals for the six strings of the guitar are performed in a time-sharing manner. Concurrently, six channel musical tone generating systems are formed in sound source circuit 9, in a time-sharing manner.

A circuit configuration of the maximum peak detector 4 is illustrated in Fig. 2. As shown, the musical tone signal is derived from low-pass filter 3, and input to the + terminal of an operational amplifier 12. The output terminal of operational amplifier 12 is coupled with the anode of a diode D1. The cathode of diode D1 is grounded through parallel arranged capacitor C and resistor R1. The cathode of diode D1 is also coupled with the - terminal of operational amplifier 12. The output signal of operational amplifier 12 is output through a resistor R2 and an operational amplifier 13, as interrupt command signals INTa1 to INTa6, which are representative of maximum peak detections.

When a signal with a waveform as shown in Fig. 4a is applied to the + terminal of operational amplifier 12, the capacitor C is charged when the signal waveform rises, and discharged when it falls. In turn, the signal with a waveform as shown in Fig. 4b is applied to the - terminal of operational amplifier 12. Only when the waveform rises, the operational amplifier 12 produces a difference between the signal levels at the + and - terminals. This signal is output as interrupt command signal INTa shown in Fig. 4c. The interrupt processing starts at the trailing edge of the pulsative signal of Fig. 4c.

A specific configuration of minimum peak detector 5 is shown in Fig. 3. The circuit configurations thereof is substantially the same as that of the maximum peak detector 4 as mentioned above, except for the connection of a diode D2 being opposite. Accordingly, capacitor C repeats the charge/discharge in the opposite directions as shown in Fig. 4d. Finally, the interrupt command signal INTb as shown in Fig. 4e is obtained.

The operation of the above-mentioned apparatus will now be described.

1) In Waveform Rising:

Upon power on, CPU 6 starts a main routine as illustrated in Fig. 6. In step A1, CPU 6 initializes the related circuits in the electronic musical instrument of the embodiment. In step A2, it fetches the value of A/D converter 11. CPU 6 continues the note off processing unless the signal from the converter 11 exceeds a predetermined level. See step A3 and A4. If a player picks the guitar string, a musical tone signal at a level larger than the predetermined level as shown in Fig. 9 is input to A/D converter 11 (step A3), CPU 6 advances to step A5 where it executes the frequency control processing. The frequency control processing is a musical tone sounding processing in which the data from counter 7 is directly applied to frequency ROM 8 or the pitch code as indicated by the fret number extracted on the basis of that data is applied to frequency ROM 8. So long as the musical tone level is above the predetermined level, CPU 6 continues this sounding processing (steps A2, A3 and A5). The count data of counter 7 is set through the interrupt processing as given later.

2) At maximum peak point MAX1:

Let us consider now that the waveform of the musical tone signal rises and reaches the first maximum peak point denoted as MAX1 in Fig. 9. At the maximum peak point, maximum peak detector 4 produces interrupt command signal INTa representing the maximum peak point detection, and applies it to CPU 6. In response to this signal, CPU 6 starts the interrupt processing shown in Fig. 7. CPU 6 fetches the count of counter 7 in step B1. In step B2, the CPU checks whether or not the waveform having the detected maximum peak point is the first wave. Since it is now immediately after the tone waveform rises, that waveform is the first wave. The CPU advances to step B5 to set flag "1", and further to set in the maximum memory 14 the count of counter 7 as fetched in step B1. "1" of flag indicates that the maximum peak point has been detected. The cleared flag "0" indicates that the minimum peak point has been detected.

2) At maximum peak point MAX2;

It is assumed that the waveform level reaches the second maximum peak point denoted as MAX2 in Fig. 9. At the maximum peak point, maximum peak point detector 4 applies interrupt command signal INTa indicating the maximum peak detection to CPU 6. Then, CPU 6 starts the interrupt processing shown in Fig. 7, again. CPU 6 fetches the count of counter 7, and checks if the present waveform is the first wave (step B2). Since the present wave is not the first one, the CPU advances to step B3, and checks if the flag is "0". Since the flag is "1" at the first maximum peak point MAX1, the CPU terminates the present processing, without executing the period computing processing for obtaining the frequency data.

In this way, the frequency data is correctly obtained, whose one period is the difference between the count data at the maximum peak points MAX1 and MAX2, and the time count at the maximum peak point MAX1 is left stored in the maximum memory 14.

3) At minimum peak point MIN:

When the minimum peak point MIN is reached (Fig. 9), minimum peak detector 5 produces interrupt command signal INTb representing the minimum peak point detection, and applies it to CPU 6. Upon receipt of this, CPU 6 starts the interrupt processing shown in Fig. 8. In step C1, CPU 6 fetches the count of counter 7, and check if the waveform at the minimum peak point is the first wave (step C2). In this instance, the waveform is the first waveform, and then CPU 6 proceeds to step C5. In this step, the flag is cleared to be "0", and sets in the minimum memory 15 the count of counter 7, which has been fetched in step C1.

The judging processes in step B2 is performed in the following way. When the waveform level data derived from A/D converter 11 exceeds a predetermined level, the first wave flag "1" is set. When interrupt command signal INTb for minimum peak detection is then applied to the CPU, the first wave flag is cleared to "0". In steps B2 and C2, the CPU 6 checks whether or not the wave flag is set.

4) Period Calculation at the Maximum peak Point MAX1:

When the input signal waveform increases to reach the maximum peak point MAX1, maximum peak detector 4 applies interrupt signal INTa representing the maximum peak point detection to CPU 6. CPU 6 fetches the count of counter 7 in step B1, and confirms in step B2 that the waveform at the maximum peak point MAX1 is not the first wave, and checks if the flag is "0" in step B3. Since the flag is "0" at the minimum peak point MIN immediately before the maximum point MAX1, CPU 6 advances step B4, and in this step reads out the time count data from the maximum memory 14, and subtracts it from the this-time count data as has been read out in step B1, and finally obtains the data of the subtraction result. As a result, in step A5, CPU 6 applies the subtraction result data to frequency ROM 8, and controls the system of the musical instrument so that it produces a musical tone with the frequency whose period ranges from the maximum peak point MAX1 to the next maximum peak point MAX1. Following the above processing, CPU 6 sets the flag to "1", and sets the this-time time count data into the maximum memory 14 (step B5).

In steps C5 and B3, CPU 6 checks the maximum peak point following the minimum peak point, and measures the time interval between the maximum peak points. In step B4, the period calculation is progressively executed.

Therefore, if the input signal waveform contains two maximum peak points MAX1 and MAX2 successively, only the time interval between two positive peak points MAX1 and MAX1 is used to determine the sounding frequency. The time interval between the successive maximum peak points MAX1 and MAX2 is never used for the same purpose.

5) Period Calculation at Minimum Peak point MIN :

When the input signal waveform reaches to the minimum peak point MIN, minimum peak detector 5 applies interrupt signal INTb representing the minimum peak point detection to CPU 6. CPU 6 fetches the count of counter 7 in step C1, and confirms in step C2 that the waveform at the minimum peak point MIN is not the first wave, and checks if the flag is "1" in step C3. Since the flag is "1" at the maximum peak point MAX1, CPU 6 advances step C4, and in this step reads out the time count data from the minimum memory 15, and subtracts it from the this-time count data as has been read out in step C1, and finally obtains the data of the subtraction result. According to the result data, CPU 6 reads out from ROM 8 the frequency data to give a musical tone at the frequency whose period ranges between the previous and current minimum peak points MIN and MIN. Following step C4, CPU 6 clears the flag to "0", and sets this-time count data into the minimum memory 15 (step C5).

In this way, the time count of one period of the minimum peak points MIN is obtained and the frequency data is output. When a plurality of minimum peak points continuously appear without crossing the zero level, as in the previous case of the maximum peak points MAX1 and MAX2, CPU 6 picks up the minimum peak point following the maximum peak point in steps B5 and C3, measures the time interval between such minimum peak points. In this way, an exact period computation is performed in step C4.

As described above, on the basis of the time count data computed in steps B4 and C4, the corresponding frequency data is output from frequency ROM 8 in step A5. As a result, the frequency is changed two times (MAX1-MAX1 and MIN-MIN) during one period. This implies that the system of the musical instrument can quickly respond to the frequency variation of the input signal.

To stabilize the frequency, the previous time count data is stored into the memory in step A5, and the this-time time count data and the previous one are averaged. If the previous time data is greater than the this-time data by, for example, 20% or more, the previous data is output.

Now, still another pitch extraction system will be described. In this pitch extraction system, the maximum and/or minimum peaks of the waveform of a musical tone, i.e., an input signal, are first detected. Then, zero-cross points immediately after the maximum or minimum peaks are detected. A time interval between the detected zero-cross points is obtained. Finally, the pitch of the input signal waveform is obtained using the time interval.

More specifically, the time interval (t1) between the zero-cross points first appearing respectively after the maximum peaks are detected, or the time interval (t2) between the zero-cross points first appearing after the the minimum peaks are detected, is detected. The detected time interval or intervals are used for the pitch extraction. The pitch extraction is exact with a simple construction.

An arrangement of this pitch determining apparatus will now be described referring to Figs. 10 to 15. In those figures, like symbols are used for designating like or equivalent portions of Figs. 1 to 9.

Reference is made to Figs. 10 and 12 illustrating an overall circuit arrangement of that apparatus. Each one of a plurality of maximum peak detectors 4 detects the maximum peak point of an input signal waveform. At the trailing edge of the detected pulse signal, a flip-flop (FF) 214 located at the post stage of each detector 4 outputs the Q output at high level. The output signal of FF 214 and the inverted output of an inverter 230 coupled with a zero-cross point detector 206 are applied to an AND gate 224. The output signal from AND gate 224 is applied as interrupt signal INTan (where n is any of 1 to 6 figures) to a CPU 200.

Similarly, each one of a plurality of minimum peak detectors 5 detects the minimum peak point of an input signal waveform. At the leading edge of the detected pulse signal, a flip-flop (FF) 215 located at the post stage of each detector 5 outputs the Q output at high level. The output signal of FF 215 and the inverted output of the inverter 230 coupled with zero-cross point detector 206 are applied to an AND gate 225. The output signal from AND gate 225 is applied as interrupt signal INTbn (where n is any of 1 to 6 figures) to CPU 200.

Thus, when the maximum peak point is detected and FF 214 is logical high at the Q output, if the waveform crosses the zero level from the positive domain to the negative domain, interrupt signal INTan is applied to CPU 200. When the minimum peak point is detected and FF 215 is logical high at the Q output, if the waveform crosses the zero level from the negative domain to the positive domain, interrupt signal INTbn is applied to CPU 200.

Immediately after receiving the interrupt signal INTan or INTbn, CPU 200 applies clear signal CLan (where n is any of 1 to 6) or CLbn to FF214 or 215. In turn, the corresponding FF is reset. Accordingly, until the next maximum or minimum peak point is detected, even if the waveform crosses the zero level any number of times, the corresponding FF remains reset, and therefore CPU 200 is never interrupted.

When CPU 200 receives interrupt signal INTan for the zero-cross point encountered immediately after the maximum peak point is detected, CPU 200 computes a difference between the present count of counter 7 and that of counter 7 as the waveform previously crosses the zero level immediately after the maximum peak point. Similarly, when interrupt signal INTbn is generated, CPU 200 computes a difference between the present count of counter 7 and that of counter 7 as the waveform previously crosses the zero level immediately after the minimum peak point. Every time interrupt signals INTan and INTbn are generated, CPU 200 stores the counts of counter 7 into a maximum memory 201 and a minimum memory 202, respectively. The time count data as the count difference is transferred from CPU 200 to frequency ROM 8, directly or after it is converted into a key code. The frequency data with one period of the counter data is read out, and sent to sound source circuit 9, where a musical tone signal is generated. Sound system 10 receives this and starts the sounding of the musical tone.

The arrangement of zero-cross point detector 206 will be described. In Fig. 12 illustrating a specific circuit arrangement of zero-cross point detector 206, the musical tone signal derived from low-pass filter 3 is applied to the + terminal of an operational amplifier 206-1. The - terminal of the operational amplifier is grounded. The output signal of operational amplifier 206-1 is output by way of a resistor R5 and another operational amplifier 206-2. For a positive input signal, operational amplifier 206-2 outputs a logical high signal. For a negative input signal, amplifier 206-2 outputs a logical low signal. This indicates that every time the waveform of the signal signal, i.e., musical tone, crosses the zero level, the logical level at the output of the operational amplifier 206-2 is inverted.

The operation of the aforementioned apparatus will be described.

The main routine used in this apparatus is the same as that of Fig. 6, and hence it is not described here. The description of operation will be given in connection with the interrupt routines.

A player picks one of strings of the guitar, and a musical tone with a waveform as shown in Fig. 13(a) is generated. The musical tone waveform varies and ascends to reach a maximum peak point MAX1. At this time, maximum peak detector 4 generates a signal as shown in Fig. 13(b). This signals renders the output of FF 214 high in logical level (Fig. 13(e)). At the zero-cross point Zero 1 (Fig. 13(c)), the zero-cross point detection output of zero-cross point detector 206 is inverted (Fig. 13-(c)). At this time, AND gate 224 applies interrupt signal INTa to CPU 200. Upon receipt of this signal, CPU 200 starts the interrupt processing.

The interrupt routine is the same as that of Fig. 7, except for step B10. Hence, the interrupt routine will be described in brief.

CPU 200 resets FF 214 in step B10, and reads in the count of counter 7, and checks if the present wave of the input signal waveform is the first wave in step B2. Since the musical tone waveform has been just risen and therefore the present waveform is the first wave, CPU 200 advances to step B5 and sets flag to "1", and sets the count of counter 7 as read out in step B1 into the maximum memory 201. When the input signal with the waveform as shown in Fig. 13(a) is input to the musical instrument, zero-cross point detector 6 produces the inverted output signal as shown in Fig. 13(c) every time the zero-cross points Zero 2 and Zero 3 are successively detected.

At this time, however, FF 214 has been reset in step B10, and no interrupt signal INTb is produced.

When the minimum peak point MIN1 shown in Fig. 13(a) is reached, minimum peak detector 5 produces a peak detection signal, and sets FF 215 by the signal. At the next zero-cross point (Zero 4), the output signal of zero-cross point detector 206 is inverted, and AND gate 225 applies interrupt signal INTb to CPU 200. CPU 200 starts the interrupt processing shown in Fig. 15.

This interrupt routine of Fig. 15 is also substantially the same as that of Fig. 8, except step C10. This routine will be described in brief. CPU 200 resets FF 215 in step C10, fetches the count of counter 7, and checks if the present wave of the waveform is the first wave in step C2. The wave containing the zero-cross point succeeding to the minimum peak point is the first wave. Then, CPU 200 advances to step C5 and clears the flag to "0", and stores the count of counter 7 as read out in step C10 into the minimum memory 202.

When the zero-cross point (Zero 5) succeeding to the maximum peak point MAX2 shown in Fig. 13(a) is reached, interrupt signal INTa for the detection of the zero-cross point following the maximum peak point is generated, and CPU 200 fetches the count of counter 7 in step B10. In step B2, the CPU 200 confirms whether the present wave is not the first wave. Then, in step B3, the CPU checks if the flag is "0". Since the flag has been cleared to "0" at the zero-cross point (Zero 4) immediately after the previous minimum peak point MIN1, CPU 200 advances to step B4, and reads out the time count data, which has been saved into the maximum memory 201 at the zero-cross point (Zero 1) immediately after the maximum peak point MAX1 one period before, and subtracts the time count data read out this time from that read out in step B1, to have the subtraction result data. As a result, in step A5 of the main routine, the resultant data is applied to frequency ROM 8, and the musical instrument is controlled so as to sound the musical tone at the frequency whose period ranges from the zero-cross point (Zero 1) to the zero-cross point (Zero 5). Succeeding to this processing, CPU 200 sets the flag "1", and sets the this-time time count data into the maximum memory 201 (step B5).

In this way, the zero-cross point immediately after the maximum peak point is checked in steps C5 and B3 and the time interval (t1) only between these zero-cross points is measured, and the period calculation is performed in step B4.

Similarly, the zero-cross points (Zero 6 and Zero 7) are ignored. In response to the interrupt signal INTb, which is generated upon detection of the zero-cross point (Zero 8) immediately after the minimum peak point detection, the CPU 200 performs the routine of Fig. 15. The time interval (t2) from the previous zero-cross point (Zero 4) to the present zero-cross point (Zero 8) is used as the pitch data.

Thus, the time interval (t1), i.e., Zero 1 → Zero 5, between the zero-cross points immediately after the maximum detection points, and the time interval (t2), i.e., Zero 4 → Zero 8, between the zero-cross points immediate after the minimum point detections, are obtained. In other words, the frequency change is performed two times during one period in the same manner as in the first embodiment. Therefore, the musical instrument can quickly respond to the frequency change of the input signal.

In the afore described musical instrument, if the waveform as shown in Fig. 16(a) is input, the zero-cross points Zero 12 and Zero 14 are ignored with the function of the flag in the Fig. 15 flow.

More specifically, if the signals for zero-cross points Zero 12 and Zero 14 come in as the interrupt signal INTa, the flag has been set to "1" when these signals arrive (step B5), and therefore, the answer to step B3 is NO, and hence no period calculation is performed. In this way, the zero-cross points are successively detected after the maximum peaks are detected, these are ignored by the flag, removing the effects by the harmonics.

To stabilize the frequency, in step A5, the previous time count data is stored into the memory, and the this-time time count data and the previous one are averaged. If the previous time data is greater than that of this-time by, for example, 20% or more, the previous data is output. The period calculations based on the zero-cross point detection immediately after the maximum and minimum peak points detections, may be selectively performed in a manner that the period calculation based on the zero-cross point immediately after the maximum peak point detection is executed for the waveform which rises at the start point and that based on the zero-cross point immediately after the minimum peak point detection is executed for the waveform which falls at the start point. Such arrangement provides a quick response at the start of the sounding.

As described above, CPU 6 can execute an appropriate processing as the step A5 processing shown in Fig. 6. The selection of the appropriate processing can be made by modifying the program for CPU 6, and not requiring any modification of the hardware, i.e., the external circuit of CPU 6. Therefore, the musical instrument is excellent in flexibility of use, and can easily be designed so as to have so-called intelligent functions.

The time intervals (t1) and (t2) respectively between the zero-cross points immediately after the maximum and minimum peak points are both obtained, but the musical instrument is operable with satisfactory performances, if one of them is used. If so done, the combination of maximum peak detector 4, FF 214, AND gate 224 and inverter 230 or the combination of maximum peak detector 5, FF 215, and AND gate 225 may be omitted, simplifying the circuit arrangement of the musical instrument.

First Embodiment

1. General

A first embodiment of a pitch determining apparatus according to the core features of the invention will now be described referring to Figs. 17 to 23.

The apparatus of this first embodiment is advantageous in that the time from the inputting of an input signal waveform till the generation of an actual acoustic sound is shortened to have a good response for eliminating unnatural feelings of sounded musical tones.

The first embodiment is based on the fact that three items of distinctive period information exist between two periods of the input signal waveform. Within the time interval of less than two periods of the input signal waveform, two periods of the waveform are detected. If the detected two periods are almost equal to each other, a controller issues a command to start the sounding.

More specifically, the two periods are detected by detecting a time interval t1 between the zero-cross points each first appearing after the maximum peak of the input signal waveform is detected, and by detecting a time interval t2 between the zero-cross points first appearing after the minimum peak point of the input signal waveform is detected.

Alternatively, the two periods are detected by detecting a time interval T1 between the maximum peak points of the input signal waveform and by detecting a time interval T2 between the minimum peak points of the same waveform.

The above technical features reduces the time from the arrival of the input signal to the generation of artificial acoustic wave, thereby providing an attractive performance by the guitar.

2. Description of Arrangement and Operation

1) Arrangement

The first embodiment is configured like the afore described apparatus shown in, Figs. 10 and 11. Thus like or equivalent portions are designated by like reference numerals in Figs. 10 and 11, for simplicity.

Six pitch extraction circuits P1 to P6 are used. An A/D converter 411 is contained in each pitch extraction circuit P1 to P6. Output signals of FFs 214 and 215 are applied as read-in signals to a latch 412, via an OR gate 413. In response to the signal from OR gate 413, latch 412 fetches the digital output of A/D converter 411.

With such an arrangement, each latch 412 latches the maximum or minimum peak point of the input signal waveform, so that a CPU 400 can readily fetch these peak values.

The output signal of each OR gate 413 is applied as a respective signal L1 to L6 to CPU 400.

2) Operation

Fig. 17 shows a main routine executed by CPU 400. This routine is similar to those of the early mentioned apparatus. The main routine illustrated is for processing the musical tone generated by one string of the guitar. This processing is correspondingly applied for other five strings. Actually, CPU 400 executes these routines in a time divisional manner. CPU 400 first executes step A401 to initialize the system of the electronic guitar. Following this, CPU 400 reads the content of A/D converter 411 in step A402. The CPU continues the tone-off processing of musical tone until the output level of A/D converter 41 reaches a predetermined level (steps A403, 404, and 405). If the guitar string is picked, and the musical tone signal above the predetermined level as shown in Fig. 19 is input to A/D converter 411 (step A403), the CPU advances to step A405. In this step, the CPU checks whether or not the already extracted time intervals t1 and t2 are equal. Since both the intervals are not yet detected, however, the answer in step A406 is NO, and the CPU returns to the original flow. As described later, when starting the sounding, the CPU advances to step A407 and starts the sounding of the musical tone at the pitch as determined by the time interval t1 (nearly equals t2). Subsequently, the CPU goes through the route of A403 → A404 → A408. In the step A408, the CPU executes the frequency control processing, i.e., the sounding processing to apply the data for designating the musical tone to frequency ROM 8. This sounding processing is continued so long as the musical tone signal level is above the predetermined level (steps A402 → A403 → A404 → A408). When the output level A/D converter 411 is below the predetermined level, step A405 is executed and the sounding starts.

The operation of the electronic guitar, when a string of it is picked, will now be described. A string is picked and the waveform of the musical tone as generated rises, and its level reaches the first maximum peak point MAX1 shown in Fig. 19-(a). Maximum peak detector 4 produces a signal as shown in Fig. 19(b). This signal sets the output of FF 214 high (Fig. 19(d)). The zero-cross point detection output signal from zero-cross point detector 6 is inverted at the zero-cross point Zero 1 (see Figs. 19(a) and 19(c)). At this time interrupt signal INTan is transferred to CPU 400, from AND gate 224. Upon receipt of this, the CPU starts the interrupt processing of Fig. 20. This processing is substantially the same as the interrupt processing of Fig. 14. The period obtained in the step B4 in the Fig. 20 routine is used as the interval t1.

When at the zero-cross point Zero 4 in Fig. 19-(a), interrupt signal INTbn is applied to CPU 400, the CPU executes the Fig. 21 processing. The Fig. 21 interrupt processing is substantially the same as that of the Fig. 15 processing.

In Fig. 19(a), CPU 400 first obtains the period data t1 through the interrupt processing at the zero-cross point Zero 5, and obtains the period data t2 by the interrupt processing at the zero-cross point Zero 8.

Following the detection of this zero-cross point Zero 8, the CPU returns to the main routine. In this routine, the CPU executes steps A403 and A406 in Fig. 18, successively. If the time data t1 and t2 are approximately equal, and the difference between them is within a tolerable range, the answer in step A406 is YES, and in step A407 the CPU causes the sound source circuit to start the sounding of musical tone at the frequency as defined by the time data t1 or t2 (see Fig. 19 (f)).

If the answer to the step A406 is NO, that is, when interval t1 between zero-cross points Zero 1 and Zero 5 is not equal to the interval t2 between zero-cross points Zero 4 and Zero 8, no sounding starts and the CPU waits for the next interrupt processing. Then, the CPU checks if the time interval t2 between the points Zero 4 and Zero 8 is equal to the next time interval t1 ranging from zero-cross points Zero 5 to Zero 9 (Fig. 19(a)). This is made in step A406. If the check result is YES, the CPU advances to step A407 where it directs the start of sounding.

Subsequently, the CPU measures the time interval t1 between the zero-cross points immediately after the maximum peak point detection, and the time interval t2 immediately after the minimum peak point detection. The CPU executes the frequency change processings two times for one period, if necessary. Therefore, the guitar can respond to the frequency change of the input signal quickly.

In this first embodiment, with the flag function in the flows of Figs. 20 and 21, if the waveform as shown in Fig. 15 is input, the zero-cross points 12 and 14 are ignored.

In the first embodiment, the waveform input is that of the type in which the waveform first rises, that is the amplitude of the waveform increases from zero level to positive. The same operation can be obtained when the waveform first falling, or going from zero level to negative is input.

Such an example is shown in Fig. 22. When the waveform as shown in Fig. 22(a) comes in, zero-cross point detector 206 outputs a signal with a waveform as shown in Fig. 22(b). As a result, interrupt signal INTb is applied to CPU 400. At this time point, the Fig. 21 routine starts. Count 7 starts the counting of the time interval t2 (see zero-cross point Zero 21 in Fig. 22(a)), then interrupt signal INTa is generated at the zero-cross point Zero 22 in the same figure. The CPU starts the processing of the Fig. 20 flow chart. At the next zero-cross point Zero 23, the first period calculation is completed to obtain interval t2. At the next zero-cross point Zero 24, time t1 is obtained (see Fig. 22(c)).

The CPU executes the Fig. 20 processing as the interrupt at zero-cross point Zero 24. After completion of this processing, the CPU returns to the main flow of Fig. 18, and executes steps A403 and A406 in a successive manner. Step A406 is used for checking whether the intervals t2 and t1 fall within a time length within a given tolerance (Fig. 22(c)). If the answer to this step is YES, the CPU advances to step A407 and gives the command to generate a musical tone with the period of that time length to frequency ROM 8 and sound source circuit 9 (Fig. 22(d)).

If the answer to step A406 is NO the CPU checks if the previous time interval t1 (between Zero 22 and Zero 24) is equal to the next interval t2 (ranging from zero-cross points Zero 22 to Zero 24), after about half the period. If these intervals are approximately equal, the CPU executes the sounding start processing.

Thus, even when the waveform falls, the sounding can be started during the period of time less than two periods (for a sine wave, it is 1.5 period).

To stabilize the frequency, in step A408 in Fig. 18, after sounding is started, the time count data previously stored and the this-time time count data are averaged and the averaged data is output. Alternatively, when the difference between the previous data and the this time data is great, for example, 20% or more, the previous data is output. The period calculations based on the zero-cross point detection immediately after the maximum and minimum peak points detections, may be selectively performed in a manner that the period calculation based on the zero-cross points immediately after the maximum peak points detection is executed for the waveform which rises at the start point, and that based on the zero-cross points immediately after the maximum peak points detection is executed for the waveform which falls at the start point.

As described above, the CPU may select an appropriate processing in the step A405 in Fig. 18. It is noted that this selection can be made by some modification of the software, and not modifying the external circuit of the CPU. Therefore, the musical instrument of this invention can be flexibly used.

In the above-mentioned first embodiment, the period of the waveform is calculated in step B4 in Fig. 20 and in step C4 in Fig. 21. The sounding control based on the calculated period and the musical tone frequency are performed in the step A407 or A408 in the main flow of Fig. 18. These processings may be performed in the interrupt processing (Figs. 20 and 21), if necessary. If so done, the response to the input signal is still more quick.

3. Modification of the first Embodiment

A modification of the first embodiment will now be described, an overall circuit configuration of which is shown in Fig. 24. In the modification, time intervals T1 and T2 respectively between the maximum and minimum peak points are detected. The sounding control is made on the basis of the thus detected time intervals.

The output signals from a microphone and pick-ups are applied to input terminals 1, and then to pitch extraction circuits P11 to P16. The specific configuration of each pitch extraction circuit P11 to P16 is as shown in Fig. 24. As shown, each one of plural amplifiers 2 receives the musical tone signal from the corresponding input terminal 1. Each one of plural low-pass filters 3 cuts off the high frequency components of the output signal from the amplifier. The output signal of the filter is applied to a maximum peak detector 4, a minimum peak detector 5, and an A/D converter 411.

These detectors 4 and 5 may be the same as those of the first embodiment.

The signals of these detectors 4 and 5 and of A/D converter 411 are applied to CPU 400. On the basis of these signals, the CPU appropriately processes the sounding start and frequency change.

Fig. 23 shows a timing chart of related signals when the input signal waveform rises. When the waveform as shown in Fig. 23(a) comes in, maximum peak detector 4 has the output waveform as shown in Fig. 23(b). The waveform of the output signal of minimum peak detector 5 is as shown in Fig 23(c).

CPU 400 determines the time intervals T1 and T2 on the basis of the output signals from those detectors, as shown in Fig. 23(d). If the time intervals T1 and T2 each representing one period of the waveform are substantially equal to each other and the difference between them is within a tolerable range, a sounding start command as shown in Fig. 23(e) is applied to the sound source circuit.

If these time intervals T1 and T2 are not equal, the time interval T1 is measured about half period later, and it is compared with the old time interval T1. If these are substantially equal, the sounding start processing is executed.

It should be understood that any suitable measure may be employed for the pitch extraction. In the abovementioned first embodiment, either of the two time intervals t1 and t2 is used for the pitch extraction. As recalled, the time interval t1 is between the zero-cross point first appearing after the maximum peak values of the input waveform. The time interval t2 is between the zero-cross point first appearing after the minimum peak values of the input waveform. Another possible pitch extraction is, as described in the modification, to compare the time interval T1 between two adjacent maximum peak values of the input waveform with the time interval T2 between two adjacent minimum peak points. A further possible pitch extraction is to use the time interval between two adjacent zero-cross points. An additional pitch extraction is to use an auto-correlation function or other suitable function of the period calculation.

The first embodiment and its modification use three types of period data t1, t2, t3 or t2, t1, t2 and T1, T2, T1 or T1, T2, T1. During the time interval of less than two periods of the input waveform (if it is a perfect sine wave, the interval is 1.5 period), two periods of the waveform are detected. If these values of periods are coincident with each other, the CPU directs the sound source circuit to start the sounding.

While in the above-mentioned first embodiment, the present invention is applied for an electronic guitar, it can be applied for any other systems of the type in which pitches are extracted from a sound signal or an electrical vibration as input from a microphone, for example, and an acoustic signal, which is different from the original signal, is generated at the pitches or note frequencies corresponding to those of the original signal. Specific examples of such are electronic pianos with key boards, electronic wind instruments, electronic string instruments such as electronic violins and koto (Japanese string instrument).

As described above, during the time interval of less than two periods of the input signal waveform, two periods of the input signal waveform are detected. If these are substantially equal, the CPU directs the sounding of a musical tone at the frequency as defined by the period. Therefore the time required from the inputting of the waveform to the actual sounding start is reduced. The result is an improvement of the response performance and the elimination of unnatural feeling of sounded musical tones.

Second Embodiment

1. General

A second embodiment of an electronic guitar according to this invention will be described referring to Figs. 25 through 27. This embodiment is able to extract the pitch of an input signal waveform even then when it contains much harmonic components as shown in Fig. 25.

As shown in Fig. 25, the waveform contains much intensive harmonics and has many peak points and cross points. If the electronic guitar receives such a waveform, it mistakenly measure one wave length time interval P as intervals PS and PR, which are each different from the true time interval P. The guitar sounds the musical tone at incorrect pitches. This is problematic in the technique disclosed in KOKOKU No. 61-51793. In this technique, when the adjacent waveform periods are substantially equal, the frequency control is executed on the basis of the time length of the period compared.

This embodiment is directed to solve such a problem, and provides an apparatus which can exactly extract the pitch of the much harmonic contained waveform and sounds the musical tone at the correct pitch.

To achieve this object, measurement is made of the time length of a first wave of an input waveform which partially overlaps one wave of the same waveform, and the time length of a second wave also overlapping the first wave. If the measured time lengths are substantially equal, a control directs the sounding of a musical tone at the frequency as defined by the time length.

2. Description of Arrangement and Operation

An arrangement of the second embodiment is shown in Fig. 28 (Fig. 33).

The operation of the second embodiment will be given with emphasis on the main routine.

Before proceeding with the detailed description of the CPU 600 operation, main registers in a work memory 601 will first be described.

PS, PR and P registers store respectively the measuring time of one period PS, the measuring time of one period PR partially overlapping the period PS, and the measuring time of one period P partially overlapping the period PR. When these three pitch data PS, PR and P are substantially coincident, CPU 600 executes the frequency control for a frequency ROM 8 and a sound source circuit 9 on the basis of the last pitch data P. This will be described in detail later. At other time than the sounding start time, every one period, the pitch data of the P register is transferred intactly to the PS register. In the subsequent description, PS, PR and P will be referred to the first, second and third periods, respectively.

The TS register stores the count of a counter 7 at the start of the period P. The TR register stores the count of counter 7 at the start of the PR period. These registers are used to obtain the respective pitch data. The counter 7 free runs to count at a given clock signal.

A TRIGGER register is set to "1" during one period just before the sounding starts, as shown in Fig. 26A. In other periods than this, this register has "0" and stores the data indicating "preparatory processing for sounding start being progressed".

A SIGN register is used to indicate that the zero-cross point for period measurement is located after the maximum peak point or the minimum peak point. The "1" of the register indicates that the zero-cross point after the maximum peak point, and the "2" of the register indicates the zero-cross point after the minimum peak point.

An AMP (i) register stores the maximum or minimum peak value, more exactly, its absolute value, which is applied from an A/D converter 411 and latched in a latch 412. An AMP (1) register stores the maximum peak value, and an AMP (2) register for minimum peak value storage.

Additionally, CPU 600 further contains three threshold levels ONLEVI, ONLEVII and ONLEV.

In a main routine (Figs. 26A and 26B), CPU 600 clears each register of TRIGGER, AMP(1), and AMP(2) in steps S31 to S33 first. In step S1, the CPU executes the interrupt processing as mentioned above, and judges if the contents of a', b', and t' (corresponding to to a, b, and t' and indicating the previous recording) have been saved into a work memory 101. If no interrupt processing has been executed, the answer of NO is given, and this step S1 is executed repeatedly.

If step S1 gives YES, the next step S2 is executed to read out the contents a', b' and t'. In the next step S3, CPU 600 reads out the value of peak point of the same type (maximum or minimum from AMP(a') register, and loads it into the s' register in the CPU. Further, the CPU sets the peak value b' now extracted in the AMP(a') register.

In step S4, the CPU checks whether the contents of TRIGGER register is "1" or not. If the system of the electronic guitar is in the initial condition, "1" cannot be set in the register, and the CPU goes to step S5 where it checks if the note-on condition has been set up or not. In this case, however, the note-on condition has not been set up, and the CPU returns to step S1. If the input signal whose level is larger than ONLEVI comes in, step S6 gives the answer of YES and then step S7 will be executed.

In step S7, the a' value is input to the SIGN register. The value of a' is 1 at the zero-cross point immediately after the maximum peak point, and is 2 at the zero-cross point immediately after the minimum peak point.

In steps S8 and S9, the CPU stores the value of t' as the count value at the start of the first period PS, into the TS register, and loads "1" into the TRIGGER register. As the result, the value of a' is loaded into the SIGN register (in the case of Fig. 41(a), this register contains "1"). The value of b' is set in the AMP register, and the value of t' to the TS register. Then the CPU returns to step S1.

The processing of main routine immediately after the zero-cross point Zero 1 (Fig. 27(a)) is completed at this point.

The processing in the main routine immediately after the zero-cross point Zero 2 will now be discribed. In steps S1 to S3 the data set processing is executed and in step S4 the answer is YES and step S10 is then executed.

When the waveform goes positiv at the time of waveform inputting, as shown in Fig. 27(a), the SIGN register contains "1". The present waveform having the negative peak in the past and therefore the a'-register contains "2". At this condition, NO is given and step S11 is executed. In this step, the value of t' as the count value at the start of the second period PR is set in the TR register. Then, the CPU returns to step S1.

After the next zero-cross point Zero 3, if the main routine is executed again, the answer of YES is given in step S4 and then the CPU jumps to step S10. The present value of a' is "1" and SIGN is "1". Therefore step S10 gives the answer YES and step S12 is executed to clear the TRIGGER register. In the succeeding step S13, the value of the TS register, i.e., the time at zero-cross point Zero 1 is subtracted from the count of counter 7 which is accepted in the present interruption and is contained in the t'-register. The result of subtraction is loaded into the PS register.

In this way the pitch data of the first period PS whose length is PS shown in Fig. 27(c) is obtained.

In step S14, as shown in Fig. 27(d), CPU 600 gives a note-on command to the frequency ROM 8 and sound source circuit 9, according to the contents of the PS register. And the sounding starts at this time point. Alternatively, the musical tone as given by the open string in place of the contents of the PS register may be generated.

In the main routine processing immediately after the next zero-cross point Zero 4, the note-on mode is set up. The CPU proceeds from step S5 to step S15 and checks whether the value of b' as the peak value fetched just now is above the OFFLEV level. If it is above the OFFLEV, the CPU checks if the relative-on processing is executed. More specifically, the CPU checks if the this-time peak value b' is by ONLEVII larger than the previous peak value "c", that is to say, checks if the extracted peak value abruptly increases during the sounding.

When the string is picked, the string gradually damps toward zero. The answer of this step S16 is NO. In case where before the vibration of a string is damped to zero, the string is picked again, by the tremolo playing, the answer of step S16 may be YES, although this is a rare case. In this case step S16 gives YES and flow jumps to step S17 and steps S8 and S9 are executed consecutively. As a result, "1" is set in the TRIGGER register and exactly the same operation as that at the sounding start will be executed. In other words, steps S12 to S14 are successively executed to effect the relative-on processing.

In a normal condition, step S17 following step S16 is executed to compare the value of a' and the value of the SIGN register. If these are not coincident as in the case of zero-cross points Zero 4, Zero 6, ..., steps S18 and S19 are executed. In these steps the content of the TR register is subtracted from the t' register to obtain the second period PR. The PR is set in the PR register, and the value of t' is set in the TS register in order to obtain the pitch data of the next period PR.

At the zero-cross point Zero 5, Zero 7, ..., whose characteristics are reverse to those of zero-cross points Zero 4 and Zero 6, sequences of steps S1 to S5 and S15 to S17 are executed. In this instance, the content of the a'-register coincides with the content of the TS register. Therefore, step S20 is executed. In this step the content difference between the t'-register and TS register is obtained to have the pitch data of the next third period. The difference is set in the P register. The value t' is set in the TS register in order to obtain the pitch data of the next third period P, in step S21.

Subsequently, in step S22, CPU 600 checks if the data difference between the first and third periods PS and P is within a predetermined error tolerance which is smaller than the third period P ($0<\alpha<0.4$). Further, in step S23 the CPU checks if the data difference between the second and third periods PR and P falls within a predetermined error tolerance which is smaller than the third period P ($0<\alpha<0.4$). If these three items of pitch data PS, PR and P are coincident within the tolerance, in step S24 the CPU executes the frequency control (pitch change) for frequency ROM 8 and sound source circuit 9 as shown in Fig. 5(d). If these are not coincident with each other, the CPU does not execute the pitch change processing in step S24, and maintains the pitch thus far used.

As described above, by comparing three items of pitch data, exact pitch extraction is performed, with sounding of the musical tone at the correct pitch.

The electronic guitar discretely picks up the frequency variation in the string vibration time to time, and executes the frequency control on the basis of the picked up data in a real time manner.

The CPU goes from step S24 to step S25, and transfers the pitch data of the third period in the P register into the PS register. That pitch data is used as the pitch data of the next new first period.

As described above, when the string vibration gradually damps and its level is below the OFFLEV level, the CPU goes from step S15 to step S26, and executes the note-off processing (sounding-off processing), as shown in the right portion of Fig. 27. The CPU gives a note-off command to sound source circuit 9 to stop the sounding.

In this way, the sounding preparation and sounding start processings in the steps S7 to S9, step S11, and steps 12 to S14 are performed at the respective zero-cross points of the first period. At the zero-cross points immediately after the minimum peak points of the respective periods, that means at the even-numbered zero-cross points of the input waveform of Fig. 27, the pitch change processing is executed in steps S18 and S19. At the zero-cross points immediately after the maximum peak points, i.e., at the odd-numbered zero-cross points of the input waveform of Fig. 27, the same processing is executed in steps S20 to S25. Finally, the note-off processing is performed in step S26.

As seen from the description relating to Fig. 25, even if the input waveform contains much harmonics the frequency change processing is not performed unless the periods PS, PR and P are equal to one another. The result reduces the possibility of the erroneous pitch extraction.

In the pitch comparing processing in steps S22 and S23, the comparison of PS and PR is allowed in addition to the comparison of PS and P, PR and P. The essential is to check if these three factors are equal or not. The pitch data used in the pitch changing processing in step S24 following the detection of three factors equality may be the arithmetic mean or the geometrical mean of PS, PR and P, in addition to the P. Additionally, two or three of these factors are appropriately weighted, and then averaged for that data.

In the above-mentioned embodiment, the sounding of the musical tone is started according to the content of the PS register, improving the response of the musical instrument system. Alternatively, after detection of the coincidence of these three factors PS, PR and P, the musical tone at the extracted frequency may be sounded.

While in the above-mentioned embodiment, the pitch between the zero-cross points is measured, it is allowed to measure the period between any other suitable points, such as the maximum peak point and zero-cross point, the minimum peak point and zero-cross point, and the maximum and minimum peak points. In such a case, the signal output from maximum or minimum peak detector 4 or 5 is used as the interrupt signal INT, and the same effects can be obtained. Additionally, a similar processing may be executed upon detection of the zero-cross point immediately before the peak point. Any other measure may be used to set up the reference point for pitch extraction.

In the above-mentioned embodiment, the respective processings are executed in the main flow, but these may also be executed in the interrupt processings.

While in the above-mentioned embodiment, the present invention is applied for an electronic guitar, it can also be applied for any other systems of the type in which pitches are extracted from a sound signal or an electrical vibration as input from a microphone, for example, and an acoustic signal, which is different from the original signal, is generated at the pitches or note frequencies corresponding to those of the original signal. Specific examples of such are electronic pianos with key boards, electronic wind instrument electronic string instruments such as electronic violins and koto (Japanese string instrument).

As seen from the foregoing description, in the second embodiment, measurement is made of the time length of a first wave of an input waveform which partially overlaps one wave of the same waveform, and the time length of a second wave also overlapping the first wave. If the measured time lengths are substantially equal, the CPU controls the sounding of a musical tone at the frequency as defined by the time length. Therefore, the system of the electronic guitar, which is best on the three periods comparison, can exactly extract the pitch of the waveform even if it contains much harmonics.

Claims

1. Apparatus for determining the pitch of a substantially periodic input signal,

characterized by:

detection means (P1-P6) for performing a plurality of detections of the period of said input signal, the samples of said input signal used for said detection overlapping in time; and

determining means (CPU) determining the pitch of said input signal if at least two of the detected periods are approximately equal, and determining the pitch of said input signal to be the reciprocal of one of the approximately equal detected periods.

2. The apparatus according to Claim 1,

characterized in that

said detection means (P1-P6) execute two detections of the period of said input signal within a time shorter than two periods of said input signal; and

said determining means (CPU) determine the pitch of said input signal if the two detected periods within the two-period timespan are approximately equal, and determine the input-signal pitch as the reciprocal of one of the two, approximately equal, detected periods.

3. The apparatus according to Claim 1 or 2, ***characterized by*** commanding means (CPU) for giving a command to generate a musical tone at a frequency whose period corresponds to said determined period.

4. The apparatus according to Claim 2 or 3, ***characterized in*** that said detecting means (P1-P6) detect said two periods by detecting a time duration (t1) between two zero-cross points following two successive positive peak points (MAXi) of said input signal, and by detecting a time duration (t2) between two zero-cross points following two successive negative peak points (MINi) of said input signal.

5. The apparatus according to Claim 2 or 3, ***characterized in*** that said detecting means (P1-P6) detect said two periods by detecting a time duration (T1) between two successive positive peak points (MAXi) and by detecting a time duration (T2) between two successive negative peak points (MINi).

6. The apparatus according to Claim 1, ***characterized in*** that said determining means (CPU) include first measuring means (7) for measuring a given time period (PS) of one period of said input signal, second measuring means (7) for measuring a given time period (PR) of one period of said input signal partially overlapping said one period (PS) measured by said first measuring means, third measuring means (7) for measuring a given time period (P) of one period of said input signal partially overlapping said one period (PR) measured by said second measuring means (7), and judging means for judging the period of said input signal when the given time periods (PS, PR, P) measured by said first to third measuring means (7) are coincident with one another.

7. The apparatus according to Claim 6, ***characterized by*** commanding means (CPU) for giving a command to generate a musical tone at a frequency based on said given time periods as measured by said first to third measuring means (7), when said given time periods are substantially coincident.

8. The apparatus according to one of Claims 1 to 7, ***characterized in*** that said input signal is produced by the vibration of a string.

9. The apparatus according to one of Claims 1 to 7, ***characterized in*** that said input signal is an audio input signal.

Patentansprüche

1. Vorrichtung zur Bestimmung der Tonhöhe eines im wesentlichen periodischen Eingangssignals,
gekennzeichnet durch:
ein Erfassungsmittel (P1-P6) zum Durchführen einer Mehrzahl von Erfassungen der Periode des Eingangssignals, wobei die Abtastwerte des Eingangssignals für die Erfassung verwendet werden, welche zeitlich überlappt; und
ein Bestimmungsmittel (CPU), welches die Tonhöhe des Eingangssignals bestimmt, wenn wenigstens zwei der erfaßten Perioden nahezu gleich sind, und welches die Tonhöhe des Eingangssignals bestimmt, um reziprok einer der nahezu gleichen erfaßten Perioden zu sein.

2. Die Vorrichtung nach Anspruch 1,
dadurch gekennzeichnet, daß
das Erfassungsmittel (P1-P6) zwei Erfassungen der Periode des Eingangssignals innerhalb einer Zeit, welche kürzer ist als die zwei Perioden des Eingangssignals, ausführt; und
das Bestimmungsmittel (CPU) die Tonhöhe des Eingangssignals bestimmt, wenn die zwei erfaßten Perioden innerhalb der Zwei-Perioden-Zeitspanne ungefähr gleich sind, und die Eingangssignaltonhöhe als das Reziproke eines der zwei ungefähr gleichen erfaßten Perioden bestimmt.

3. Die Vorrichtung nach Anspruch 1 oder 2, gekennzeichnet durch ein Befehlsmittel (CPU) zum Geben eines Befehls, um einen Musikton bei einer Frequenz zu erzeugen, dessen Periode der bestimmten Periode entspricht.

4. Die Vorrichtung nach Anspruch 2 oder 3, dadurch gekennzeichnet, daß das Erfassungsmittel (P1-P6) die zwei Perioden erfaßt durch Erfassen einer Zeitdauer (t1) zwischen zwei Nulldurchgangspunkten, welche zwei aufeinanderfolgenden Spitzenpunkten (MAXi) des Eingangssignals folgen, und durch Erfassen einer Zeitdauer (t2) zwischen zwei Nulldurchgangspunkten, welche zwei aufeinanderfolgenden negativen Spitzenpunkten (MINi) des Eingangssignals folgen.

5. Die Vorrichtung nach Anspruch 2 oder 3, dadurch gekennzeichnet, daß das Erfassungsmittel (P1-P6) die zwei Perioden durch Erfassen einer Zeitdauer (P1) zwischen den zwei aufeinanderfolgenden positiven Spitzenpunkten (MAXi) und durch Erfassen einer Zeitdauer (T2) zwischen den zwei aufeinanderfolgenden negativen Spitzenpunkten (MINi) erfaßt.

6. Die Vorrichtung nach Anspruch 1, dadurch gekennzeichnet, daß das Bestimmungsmittel (CPU) ein erstes Meßmittel (7) zum Messen einer gegebenen Zeitperiode (PS) einer Periode des Eingangssignals, ein zweites Meßmittel (7) zum Messen einer gegebenen Zeitperiode (PR) einer Periode des Eingangssignals, welches teilweise die eine Periode (PS) überlappt, welche durch das erste Meßmittel gemessen worden ist, ein drittes Meßmittel (7) zum Messen einer gegebenen Zeitperiode (P) einer Periode des Eingangssignals, welches teilweise die eine Periode (PR) überlappt, welche durch das zweite Meßmittel (7) gemessen worden ist, und ein Beurteilungsmittel enthält zum Beurteilen der Periode des Eingangssignals, wenn die gegebenen Zeitperioden (PS, PR, P), welche durch das erste bis dritte Meßmittel (7) gemessen worden ist, miteinander übereinstimmen.

7. Die Vorrichtung nach Anspruch 6, gekennzeichnet durch Befehlsmittel (CPU) zum Geben eines Befehls, um einen Musikton bei einer Frequenz zu erzeugen, welche auf den gegebenen Zeitperioden wie durch das erste bis dritte Meßmittel (7) gemessen basieren, wenn die gegebenen Zeitperioden im wesentlichen übereinstimmen.

8. Die Vorrichtung nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, daß das Eingangssignal durch die Vibration einer Saite erzeugt wird.

9. Die Vorrichtung nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, daß das Eingangssignal ein Toneingangssignal ist.

Revendications

1. Appareil destiné à déterminer le fondamental d'un signal d'entrée pratiquement périodique,
caractérisé par :
des moyens de détection (P1-P6) destinés à effectuer un ensemble de détections de la période du signal d'entrée, les échantillons du

signal d'entrée qui sont utilisés pour cette détection étant en chevauchement dans le temps; et

des moyens de détermination (CPU) qui déterminent le fondamental du signal d'entrée si deux au moins des périodes détectées sont approximativement égales, et qui déterminent pour le fondamental du signal d'entrée une valeur égale à l'inverse de l'une des périodes approximativement égales qui sont détectées.

2. L'appareil selon la revendication 1,

caractérisé en ce que

les moyens de détection (P1-P6) accomplissent deux détections de la période du signal d'entrée en une durée plus courte que deux périodes du signal d'entrée; et

les moyens de détermination (CPU) déterminent le fondamental du signal d'entrée si les deux périodes détectées pendant l'intervalle de temps de deux périodes sont approximativement égales, et ils déterminent pour le fondamental du signal d'entrée une valeur égale à l'inverse de l'une des deux périodes, approximativement égales, qui sont détectées.

3. L'appareil selon la revendication 1 ou 2, caractérisé par des moyens de génération d'ordre (CPU) qui sont destinés à donner un ordre de génération d'un son musical à une fréquence dont la période correspond à la période déterminée.

4. L'appareil selon la revendication 2 ou 3, caractérisé en ce que les moyens de détection (P1-P6) détectent les deux périodes en détectant une durée (t1) entre deux points de passage par zéro qui suivent deux crêtes positives successives (MAXi) du signal d'entrée, et en détectant une durée (t2) entre deux points de passage par zéro qui suivent deux crêtes négatives successives (MINi) du signal d'entrée.

5. L'appareil selon la revendication 2 ou 3, caractérisé en ce que les moyens de détection (P1-P6) détectent les deux périodes en détectant une durée (T1) entre deux crêtes positives successives (MAXi) et en détectant une durée (T2) entre deux crêtes négatives successives (MINi).

6. L'appareil selon la revendication 1, caractérisé en ce que les moyens de détermination (CPU) comprennent des premiers moyens de mesure (7) qui sont destinés à mesurer une durée donnée (PS) d'une période du signal d'entrée, des seconds moyens de mesure (7) qui sont destinés à mesurer une durée donnée (PR) d'une période du signal d'entrée chevauchant partiellement la période (PS) qui est mesurée par les premiers moyens de mesure, des troisièmes moyens de mesure (7) qui sont destinés à mesurer une durée donnée (P) d'une période du signal d'entrée qui chevauche partiellement la période (PR) qui est mesurée par les seconds moyens de mesure (7), et des moyens d'évaluation qui sont destinés à évaluer la période du signal d'entrée lorsque les durées données (PS, PR, P) qui sont mesurées par les premiers à troisièmes moyens de mesure (7) coïncident mutuellement.

7. L'appareil selon la revendication 6, caractérisé par des moyens de génération d'ordre (CPU) qui sont destinés à donner un ordre de génération d'un son musical à une fréquence qui est basée sur les durées données qui sont mesurées par les premiers à troisièmes moyens de mesure (7), lorsque les durées données coïncident pratiquement.

8. L'appareil selon l'une quelconque des revendications 1 à 7, caractérisé en ce que le signal d'entrée est produit par la vibration d'une corde.

9. L'appareil selon l'une quelconque des revendications 1 à 7, caractérisé en ce que le signal d'entrée est un signal d'entrée audio.

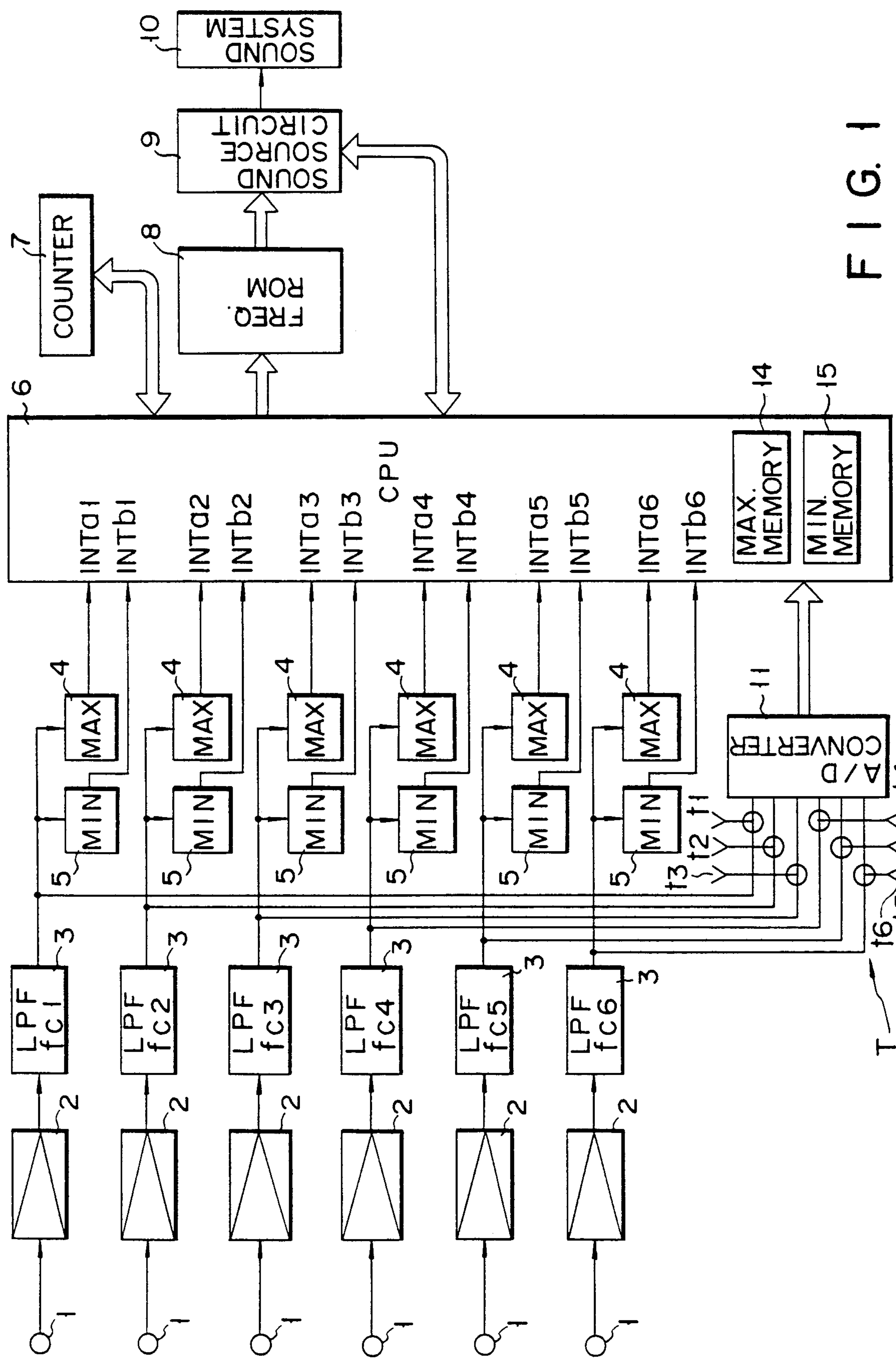
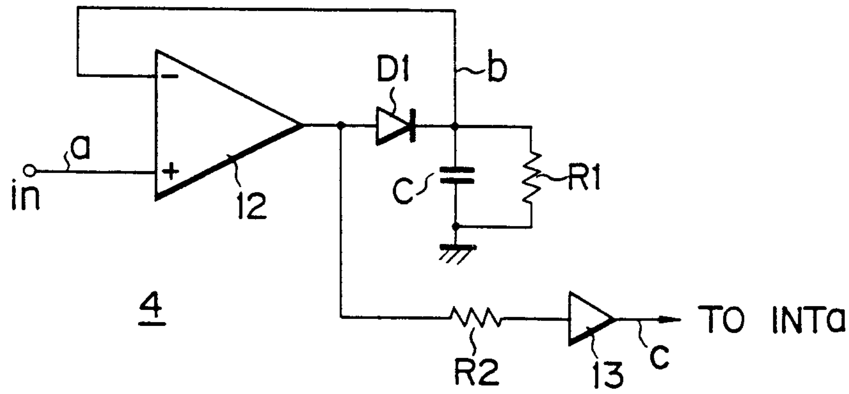
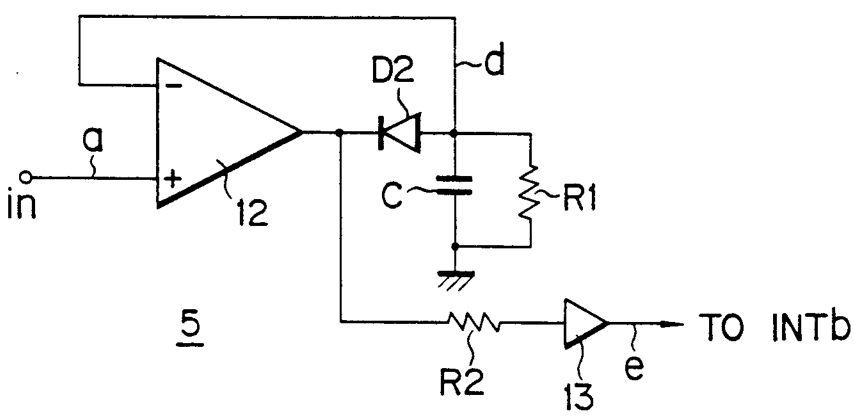

LPF fc1
LPF fc2
LPF fc3
LPF fc4
LPF fc5
LPF fc6
MIN
MAX
INTa1
INTb1
INTa2
INTb2
INTa3
INTb3
INTa4
INTb4
INTa5
INTb5
INTa6
INTb6
CPU
MAX. MEMORY
MIN. MEMORY
COUNTER
FREQ. ROM
SOUND SOURCE CIRCUIT
SOUND SYSTEM
A/D CONVERTER
t1
t2
t3
t4
t5
t6
T

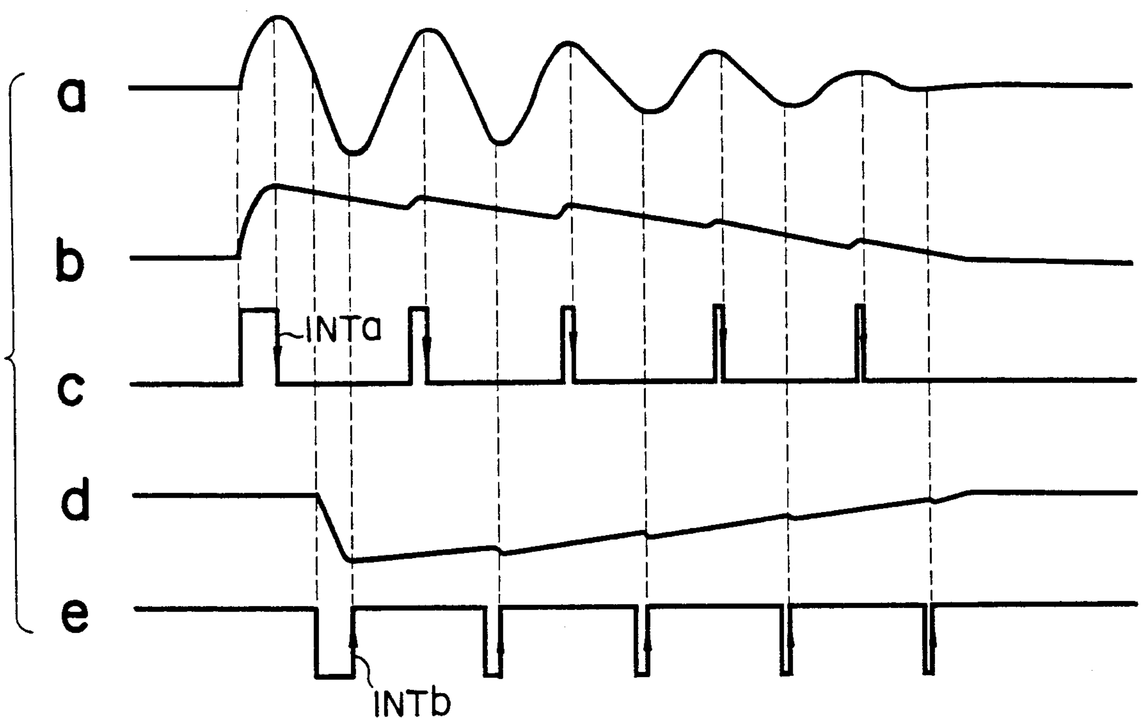
FIG. 1



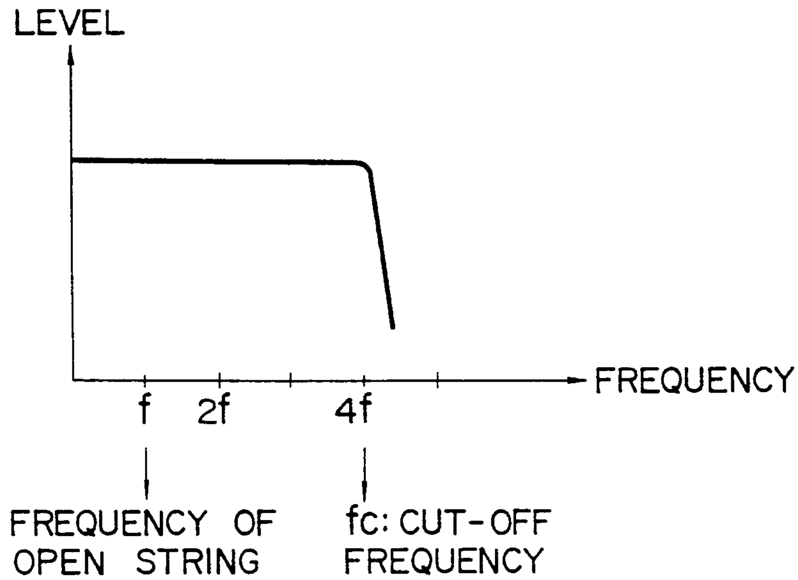
F I G. 2



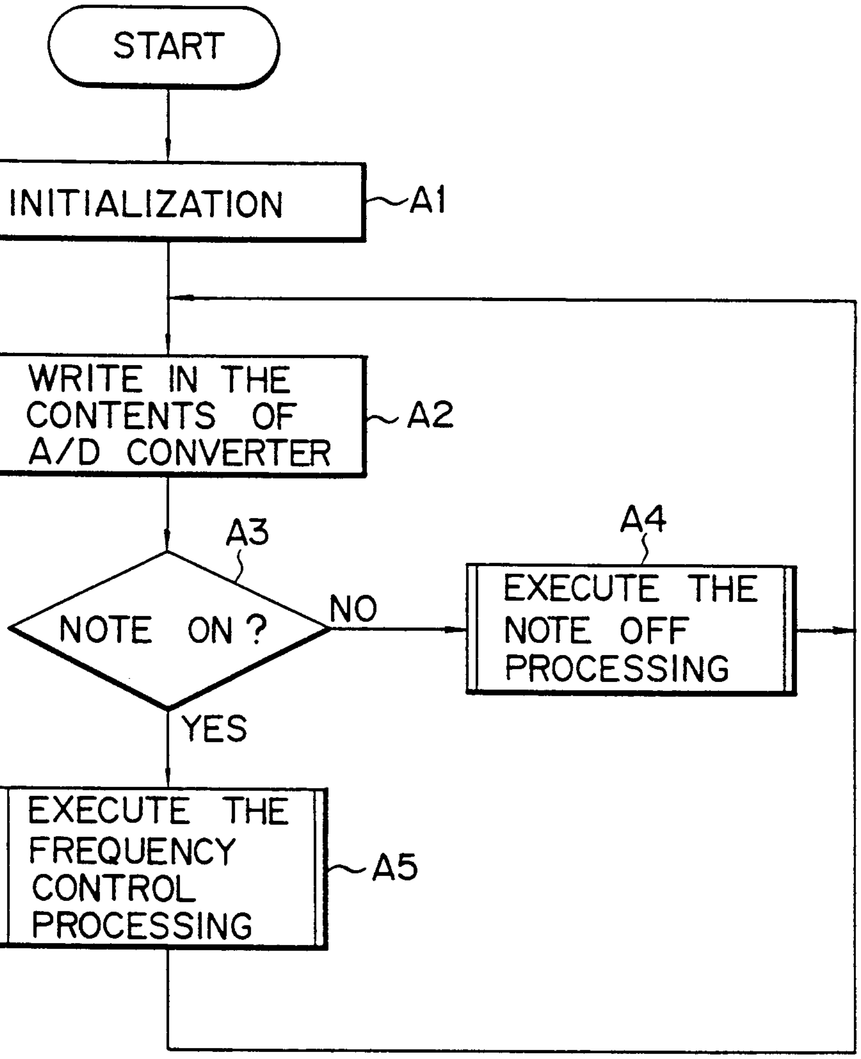
F I G. 3


a
b
c
d
e
INTa
INTb

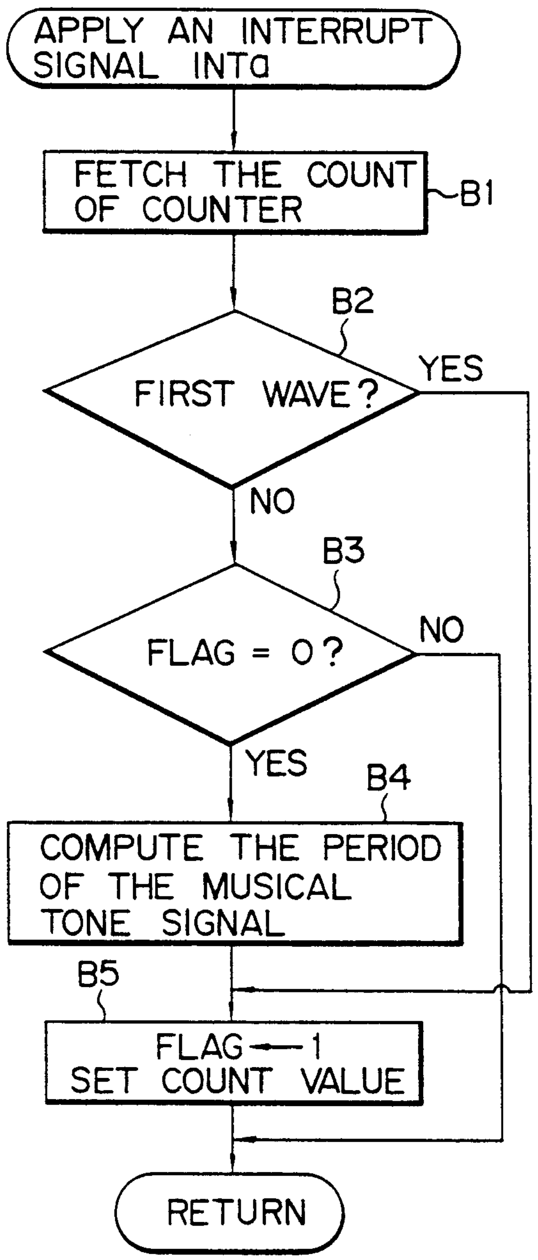
F I G. 4


LEVEL
FREQUENCY
f
2f
4f
FREQUENCY OF OPEN STRING
fc: CUT-OFF FREQUENCY

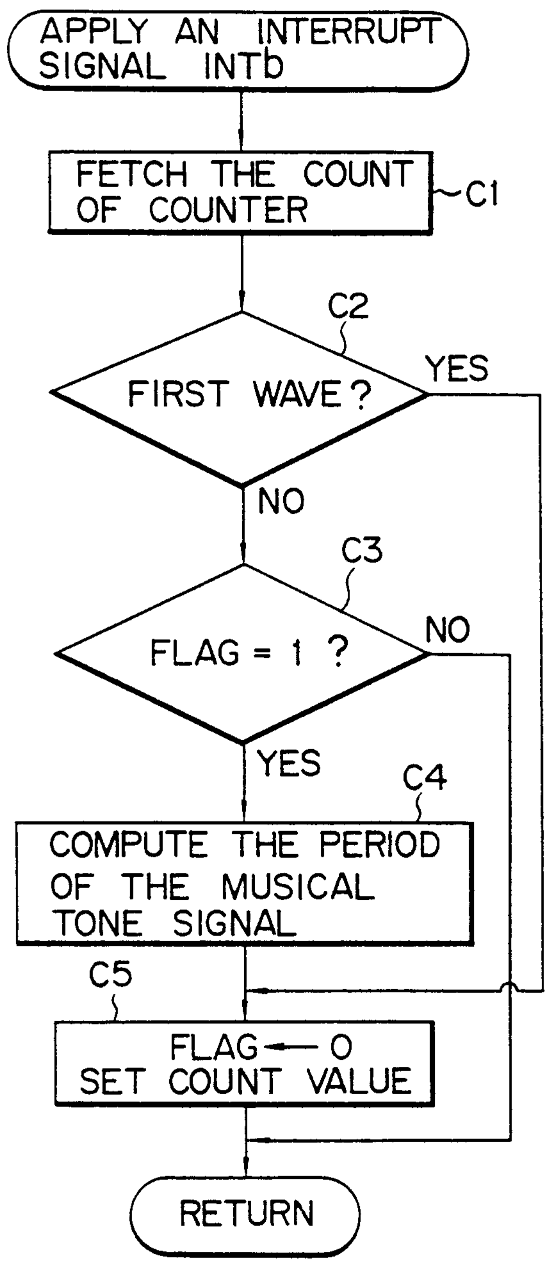
F I G. 5


START
INITIALIZATION
A1
WRITE IN THE CONTENTS OF A/D CONVERTER
A2
A3
NOTE ON ?
NO
YES
A4
EXECUTE THE NOTE OFF PROCESSING
EXECUTE THE FREQUENCY CONTROL PROCESSING
A5

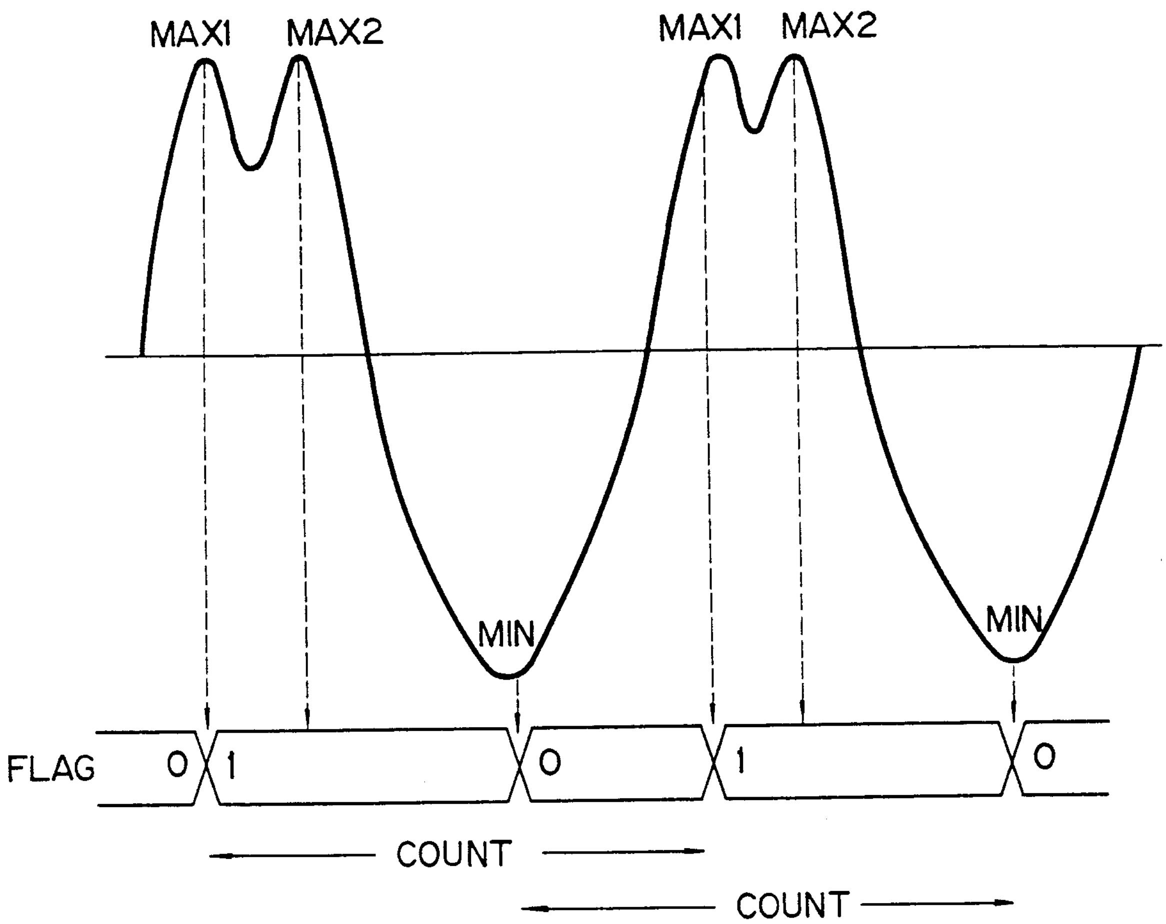
F I G. 6


APPLY AN INTERRUPT SIGNAL INTa
FETCH THE COUNT OF COUNTER
B1
B2
FIRST WAVE?
YES
NO
B3
FLAG = 0?
NO
YES
B4
COMPUTE THE PERIOD OF THE MUSICAL TONE SIGNAL
B5
FLAG ← 1
SET COUNT VALUE
RETURN

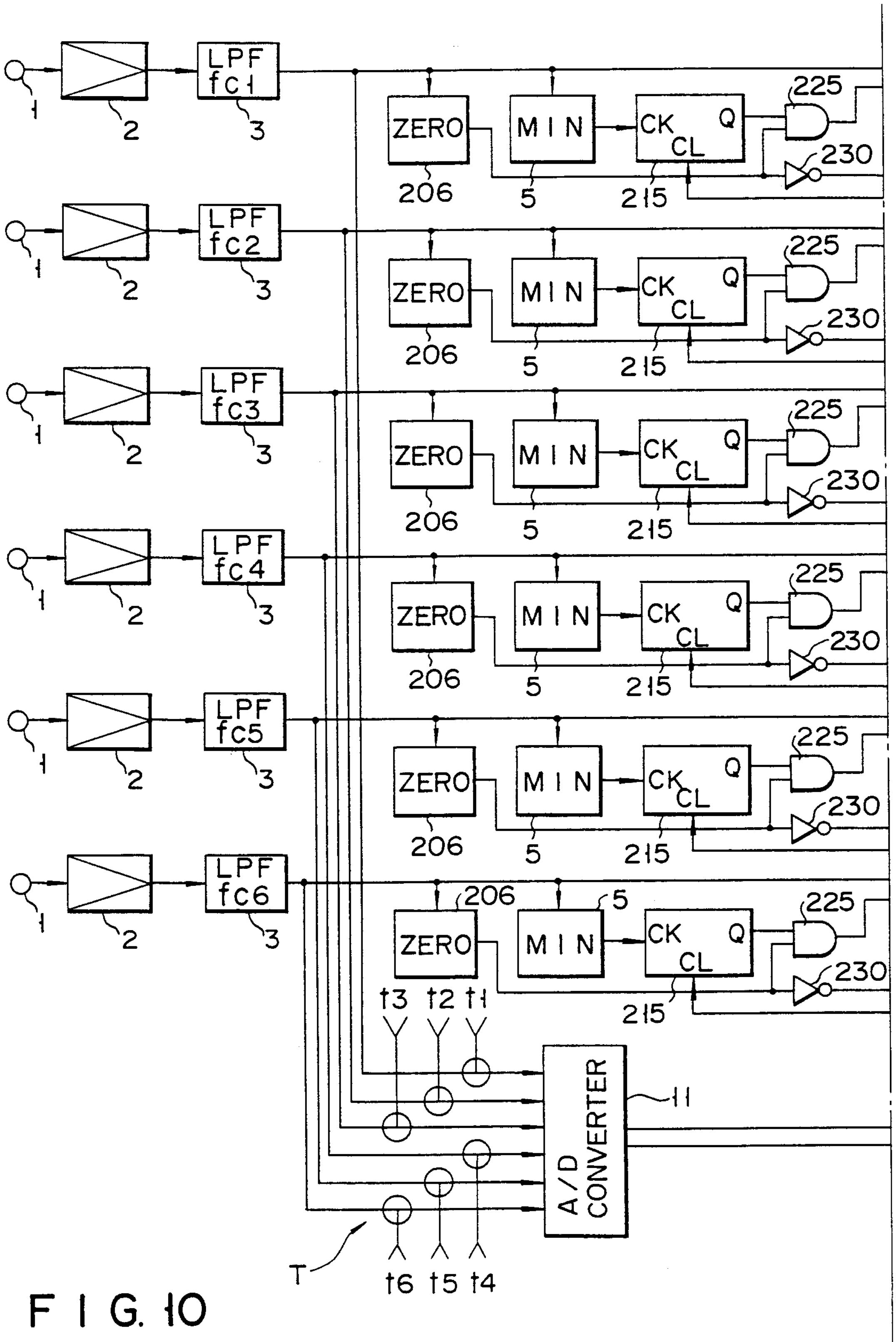
F I G. 7


APPLY AN INTERRUPT SIGNAL INTb
FETCH THE COUNT OF COUNTER
C1
C2
FIRST WAVE?
YES
NO
C3
FLAG = 1 ?
NO
YES
C4
COMPUTE THE PERIOD OF THE MUSICAL TONE SIGNAL
C5
FLAG ← 0
SET COUNT VALUE
RETURN

F I G. 8



F I G. 9


LPF fc1
LPF fc2
LPF fc3
LPF fc4
LPF fc5
LPF fc6
ZERO
MIN
CK
CL
Q
1
2
3
5
206
215
225
230
t1 t2 t3
t4 t5 t6
A/D CONVERTER
11
T

F I G. 10

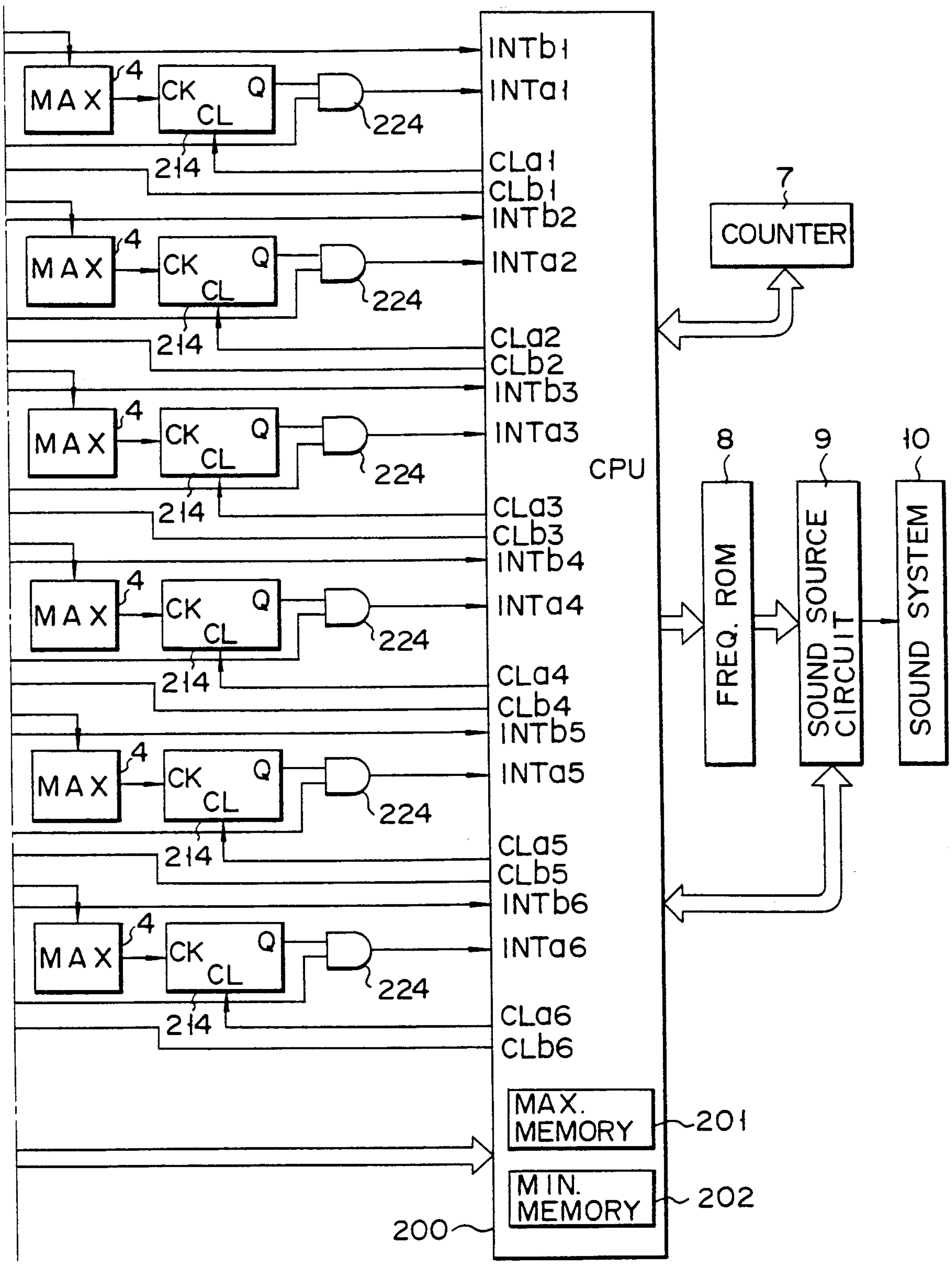
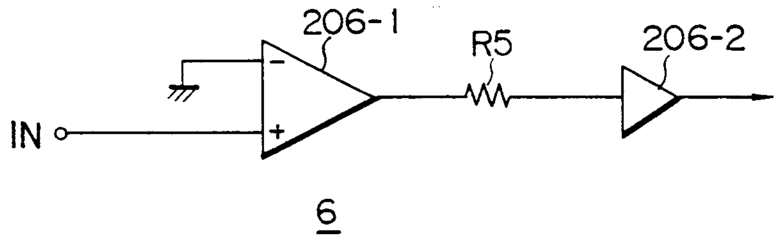
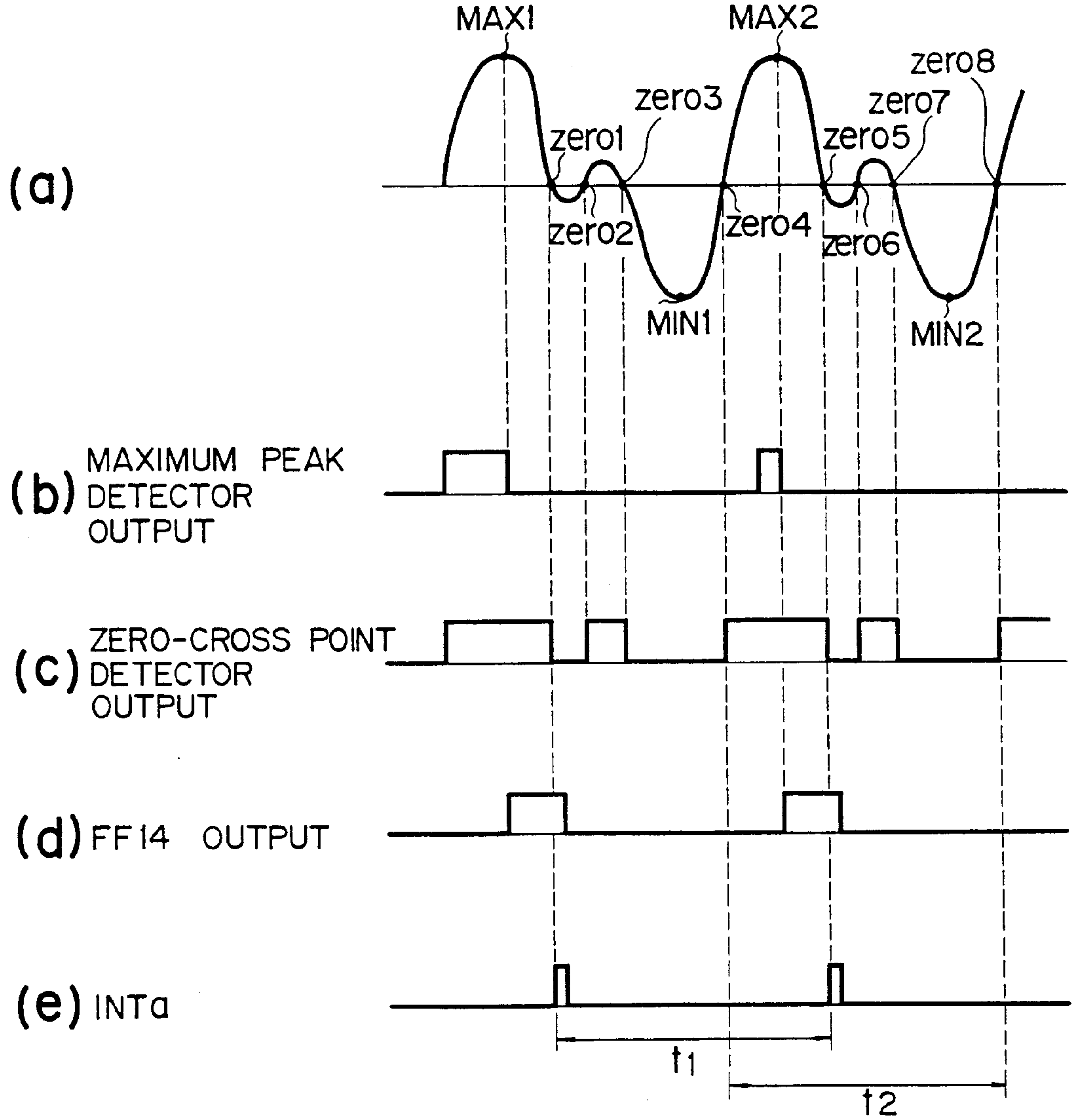

INTb1
MAX
4
CK
Q
CL
INTa1
224
214
CLa1
CLb1
INTb2
MAX
4
CK
Q
CL
INTa2
224
214
CLa2
CLb2
INTb3
MAX
4
CK
Q
CL
INTa3
224
CPU
214
CLa3
CLb3
INTb4
MAX
4
CK
Q
CL
INTa4
224
214
CLa4
CLb4
INTb5
MAX
4
CK
Q
CL
INTa5
224
214
CLa5
CLb5
INTb6
MAX
4
CK
Q
CL
INTa6
224
214
CLa6
CLb6
MAX. MEMORY
201
MIN. MEMORY
202
200
7
COUNTER
8
FREQ. ROM
9
SOUND SOURCE CIRCUIT
10
SOUND SYSTEM

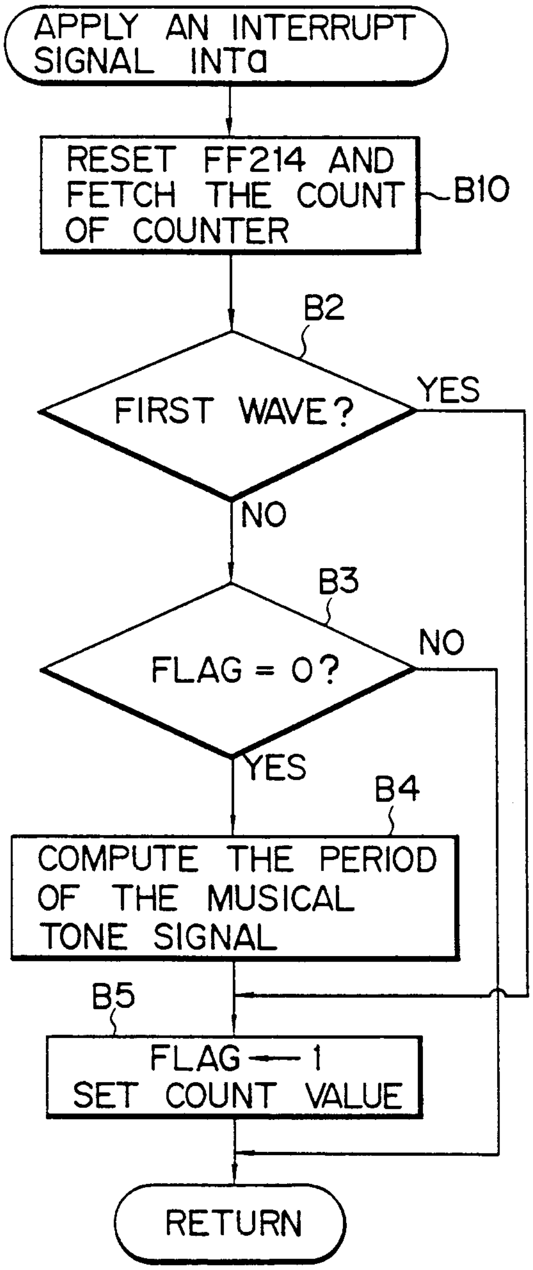
FIG. 11


206-1
R5
206-2
IN
6

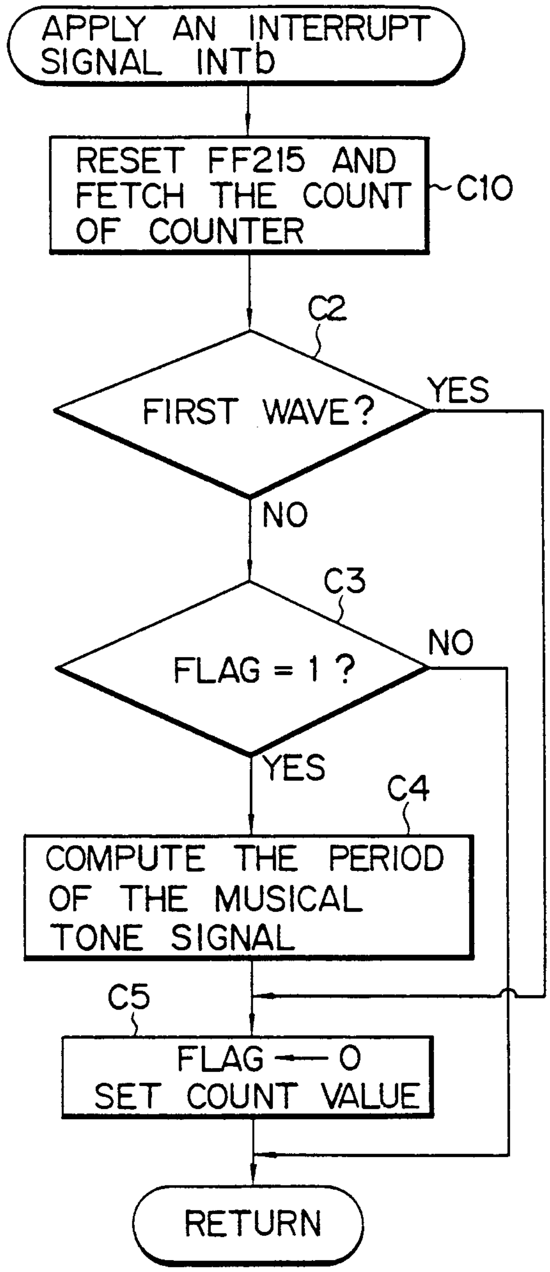
F I G. 12


MAX1
MAX2
zero8
zero3
zero7
zero1
zero5
(a)
zero2
zero4
zero6
MIN1
MIN2
(b) MAXIMUM PEAK DETECTOR OUTPUT
(c) ZERO-CROSS POINT DETECTOR OUTPUT
(d) FF14 OUTPUT
(e) INTa
t1
t2

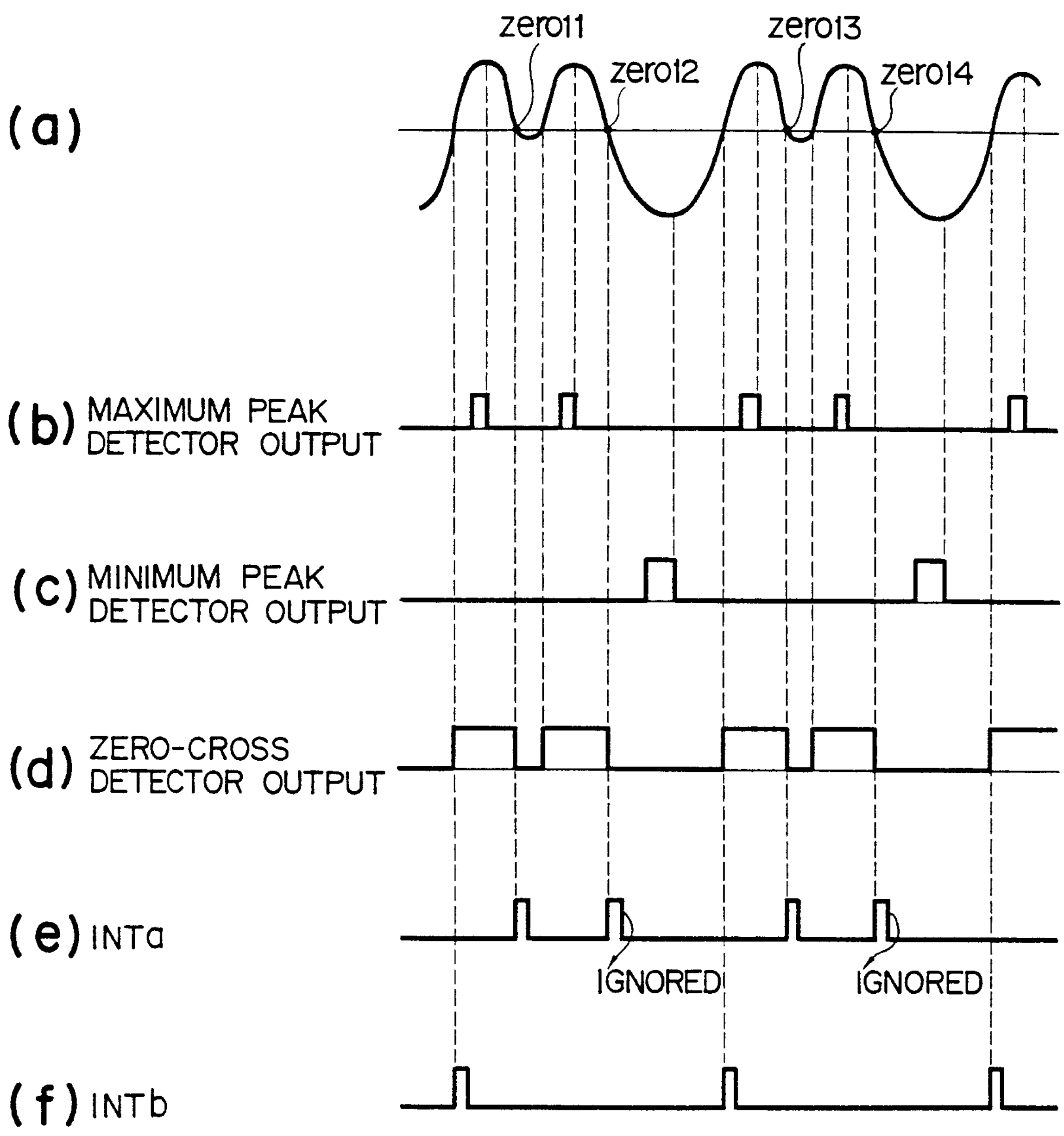
F I G. 13


APPLY AN INTERRUPT SIGNAL INTa
RESET FF214 AND FETCH THE COUNT OF COUNTER
B10
B2
FIRST WAVE?
YES
NO
B3
FLAG = 0?
NO
YES
B4
COMPUTE THE PERIOD OF THE MUSICAL TONE SIGNAL
B5
FLAG ← 1 SET COUNT VALUE
RETURN

F I G. 14


APPLY AN INTERRUPT SIGNAL INTb
RESET FF215 AND FETCH THE COUNT OF COUNTER
C10
C2
FIRST WAVE?
YES
NO
C3
FLAG = 1 ?
NO
YES
C4
COMPUTE THE PERIOD OF THE MUSICAL TONE SIGNAL
C5
FLAG ← 0 SET COUNT VALUE
RETURN

F I G. 15


zero11
zero12
zero13
zero14
(a)
(b) MAXIMUM PEAK DETECTOR OUTPUT
(c) MINIMUM PEAK DETECTOR OUTPUT
(d) ZERO-CROSS DETECTOR OUTPUT
(e) INTa
IGNORED
IGNORED
(f) INTb

F I G. 16

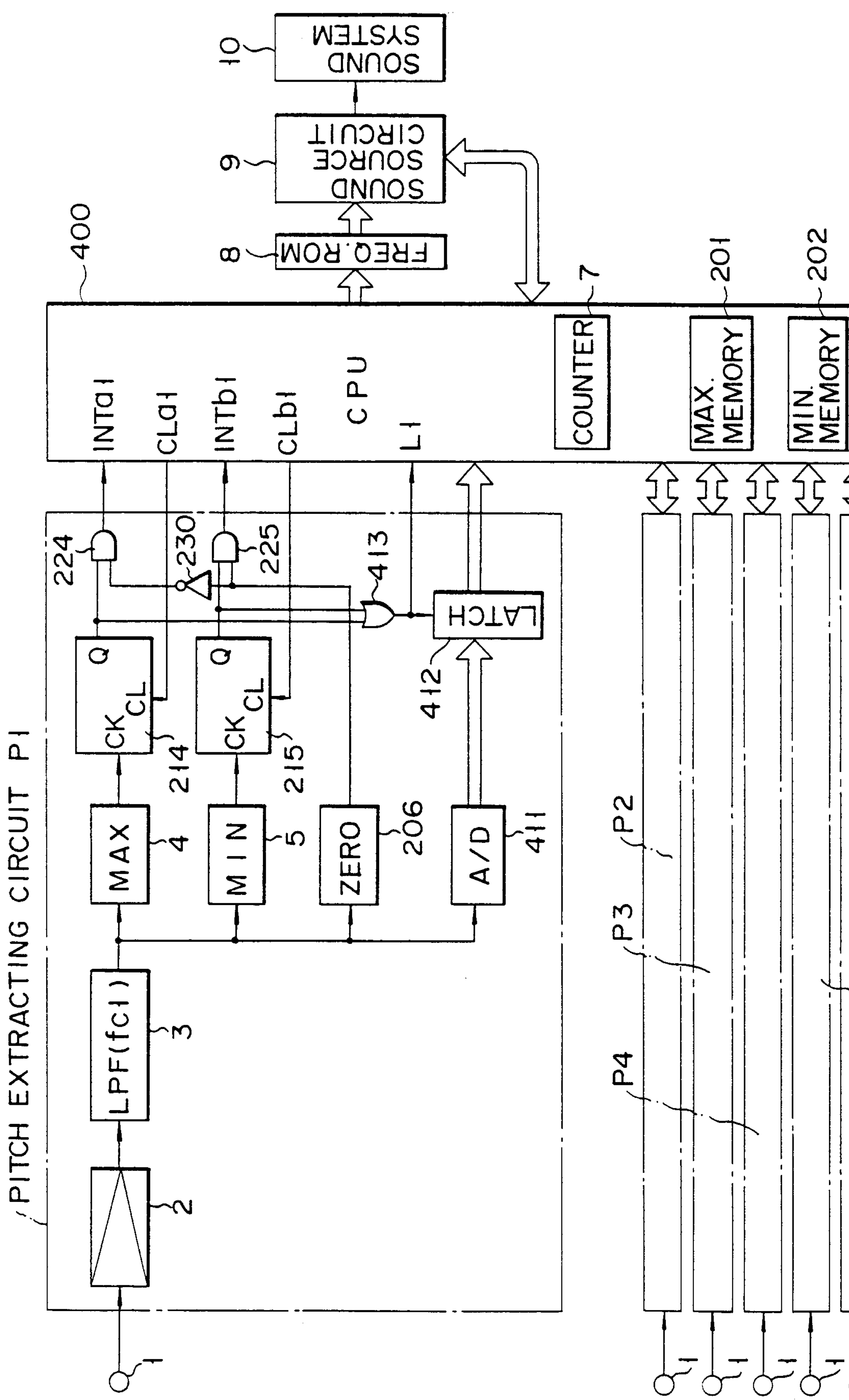
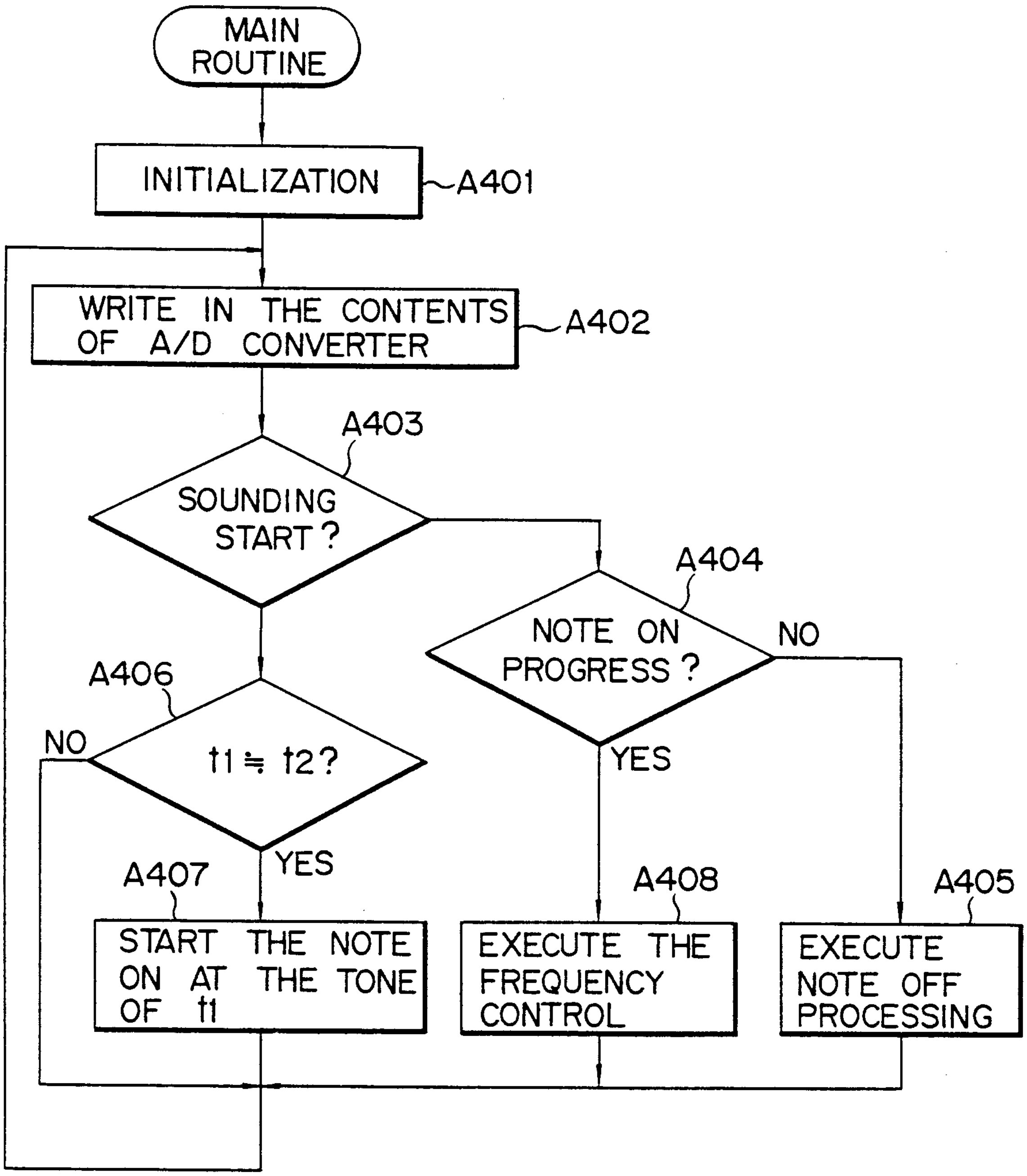
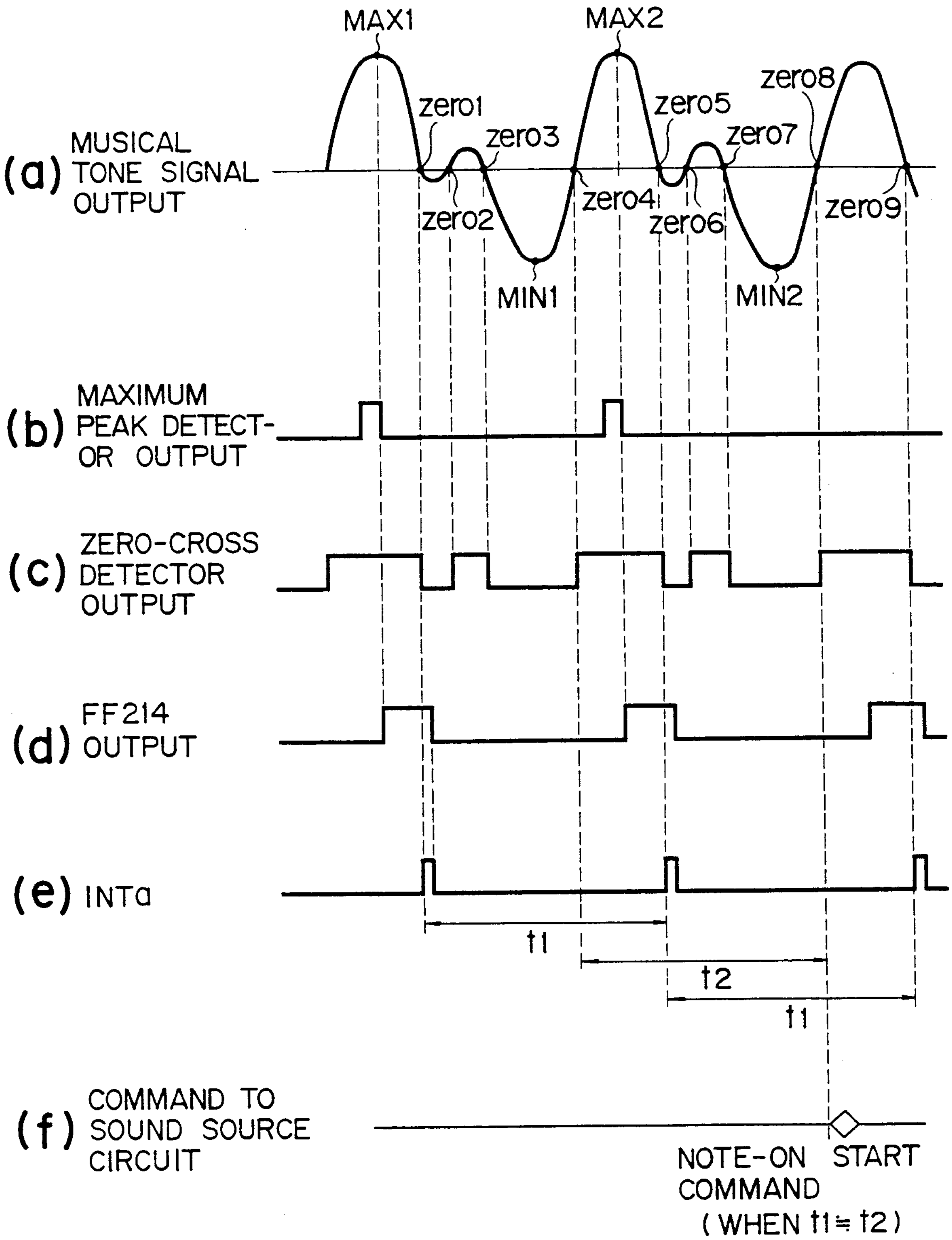

PITCH EXTRACTING CIRCUIT P1
LPF(fc1)
MAX
MIN
ZERO
A/D
CK
CL
Q
LATCH
CPU
INTa1
CLa1
INTb1
CLb1
L1
COUNTER
MAX. MEMORY
MIN. MEMORY
FREQ. ROM
SOUND SOURCE CIRCUIT
SOUND SYSTEM
1
2
3
4
5
7
8
9
10
201
202
206
214
215
224
225
230
400
411
412
413
P2
P3
P4
P5
P6

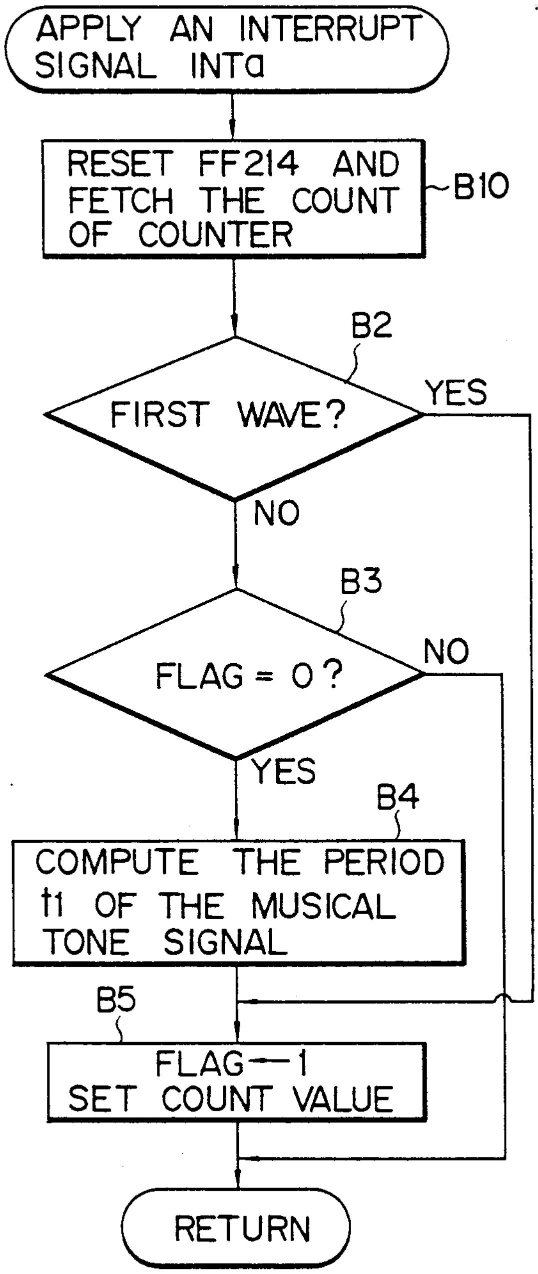
FIG. 17


MAIN ROUTINE
INITIALIZATION
A401
WRITE IN THE CONTENTS OF A/D CONVERTER
A402
A403
SOUNDING START ?
A404
NOTE ON PROGRESS ?
NO
YES
A406
NO
t1 ≒ t2?
YES
A407
START THE NOTE ON AT THE TONE OF t1
A408
EXECUTE THE FREQUENCY CONTROL
A405
EXECUTE NOTE OFF PROCESSING

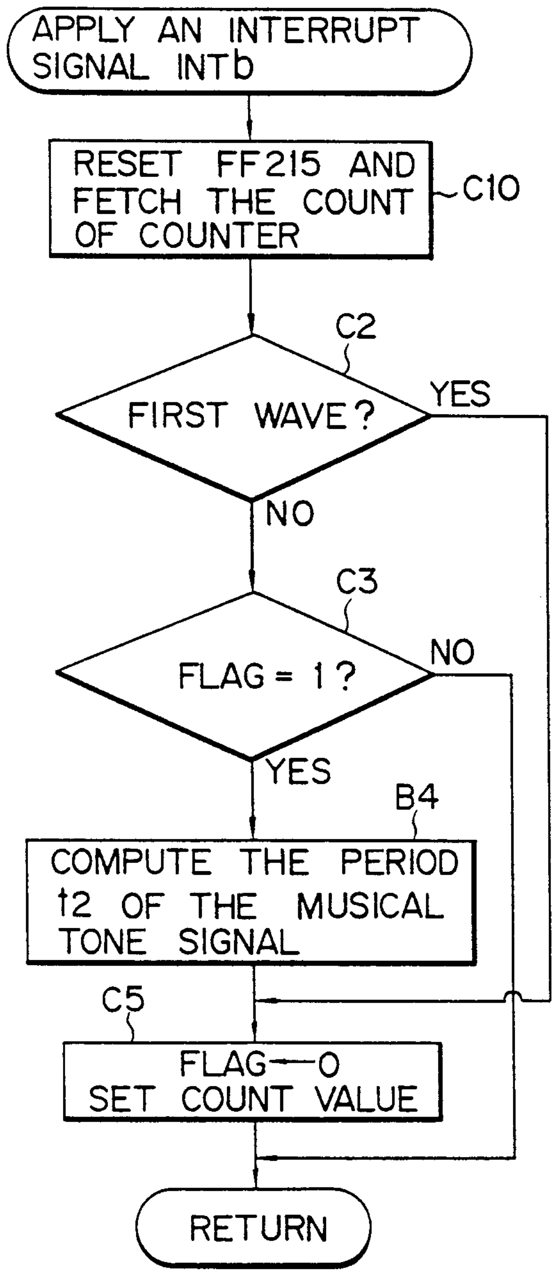
F I G. 18


MAX1
MAX2
zero1
zero2
zero3
zero4
zero5
zero6
zero7
zero8
zero9
MIN1
MIN2
(a) MUSICAL TONE SIGNAL OUTPUT
(b) MAXIMUM PEAK DETECT-OR OUTPUT
(c) ZERO-CROSS DETECTOR OUTPUT
(d) FF214 OUTPUT
(e) INTa
t1
t2
t1
(f) COMMAND TO SOUND SOURCE CIRCUIT
NOTE-ON COMMAND (WHEN t1≒t2)
START

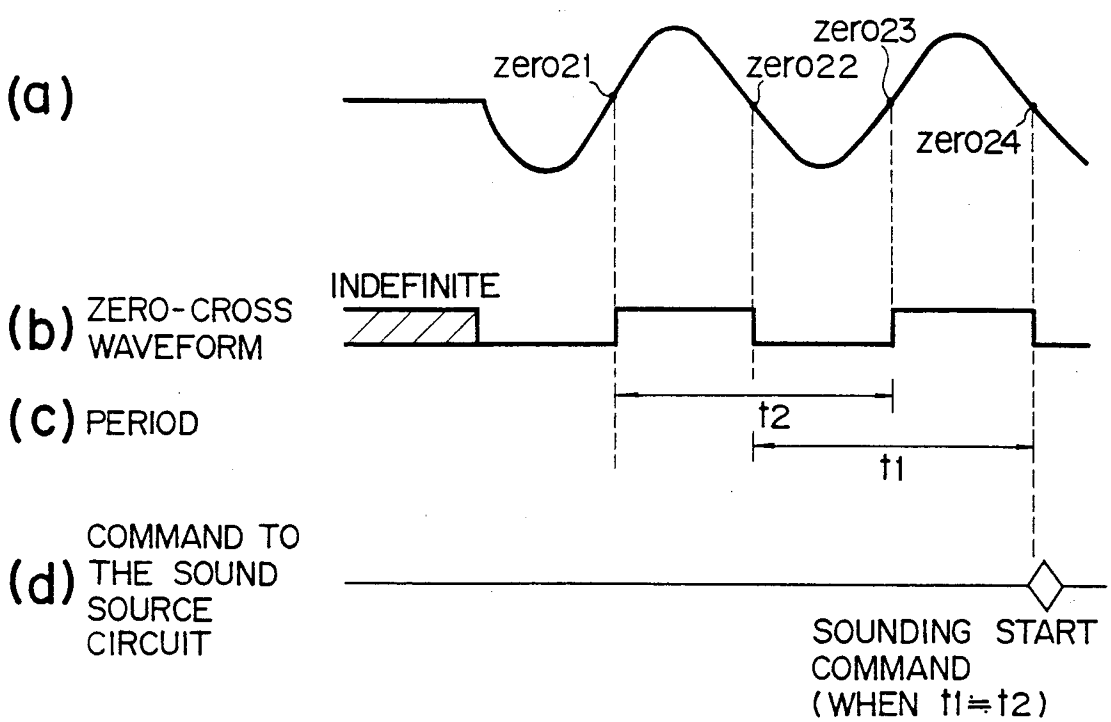
F I G. 19


APPLY AN INTERRUPT SIGNAL INTa
RESET FF214 AND FETCH THE COUNT OF COUNTER
B10
B2
FIRST WAVE?
YES
NO
B3
FLAG = 0?
NO
YES
B4
COMPUTE THE PERIOD t1 OF THE MUSICAL TONE SIGNAL
B5
FLAG←1 SET COUNT VALUE
RETURN

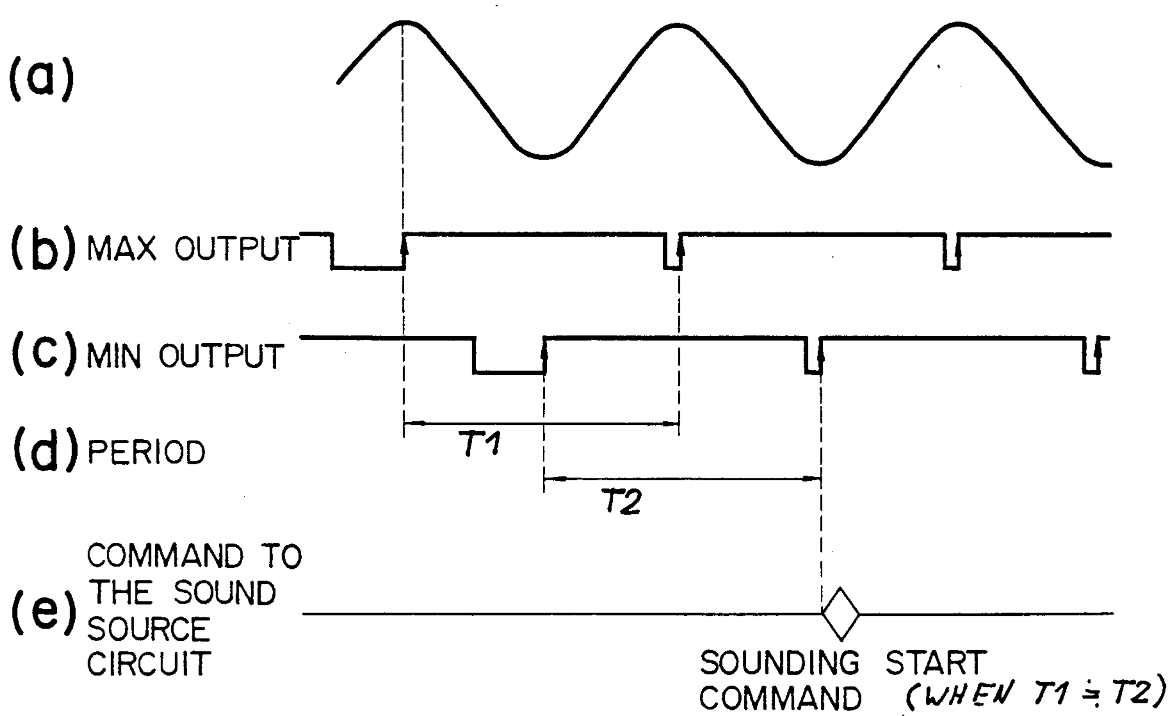
F I G. 20


APPLY AN INTERRUPT SIGNAL INTb
RESET FF215 AND FETCH THE COUNT OF COUNTER
C10
C2
FIRST WAVE?
YES
NO
C3
FLAG = 1?
NO
YES
B4
COMPUTE THE PERIOD t2 OF THE MUSICAL TONE SIGNAL
C5
FLAG←0 SET COUNT VALUE
RETURN

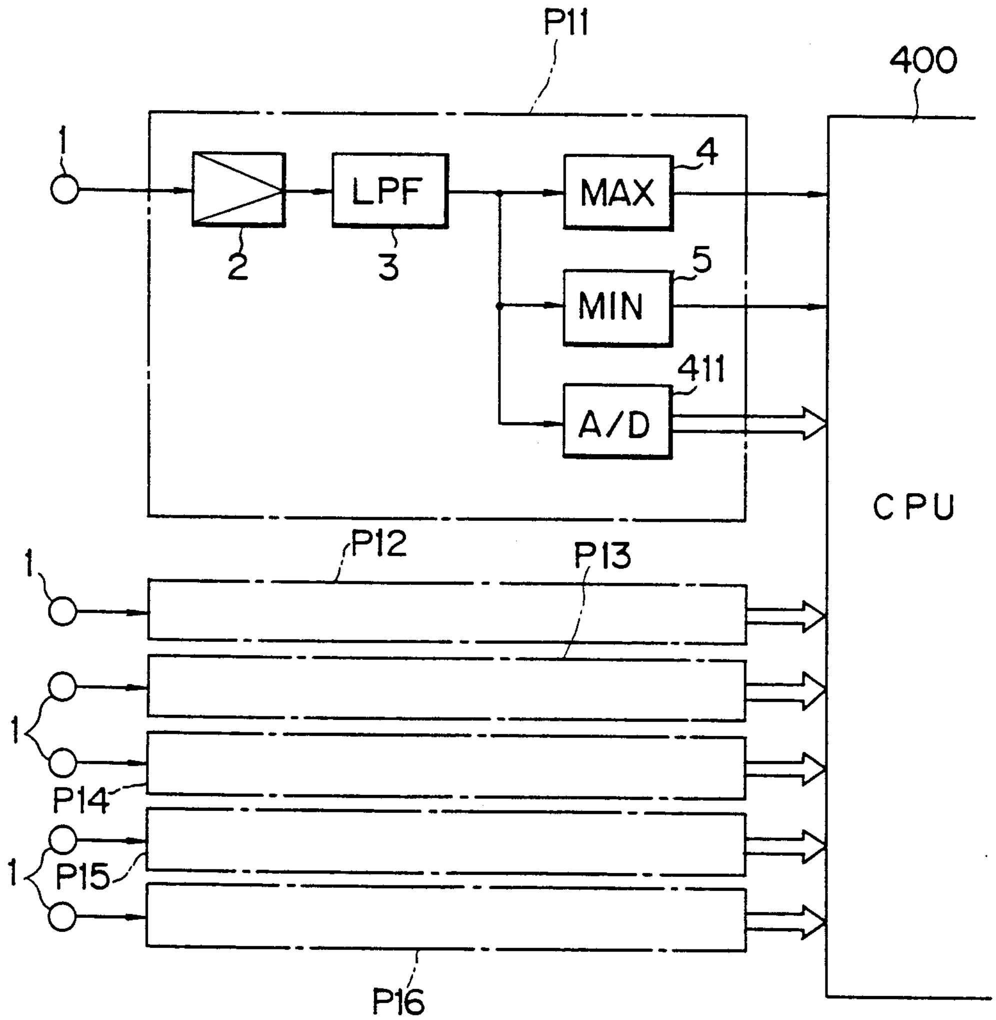
F I G. 21


(a)
zero21
zero22
zero23
zero24
(b) ZERO-CROSS WAVEFORM
INDEFINITE
(c) PERIOD
t2
t1
(d) COMMAND TO THE SOUND SOURCE CIRCUIT
SOUNDING START COMMAND (WHEN t1≒t2)

F I G. 22


(a)
(b) MAX OUTPUT
(c) MIN OUTPUT
(d) PERIOD
T1
T2
(e) COMMAND TO THE SOUND SOURCE CIRCUIT
SOUNDING START COMMAND (WHEN T1≒T2)

F I G. 23



F I G. 24

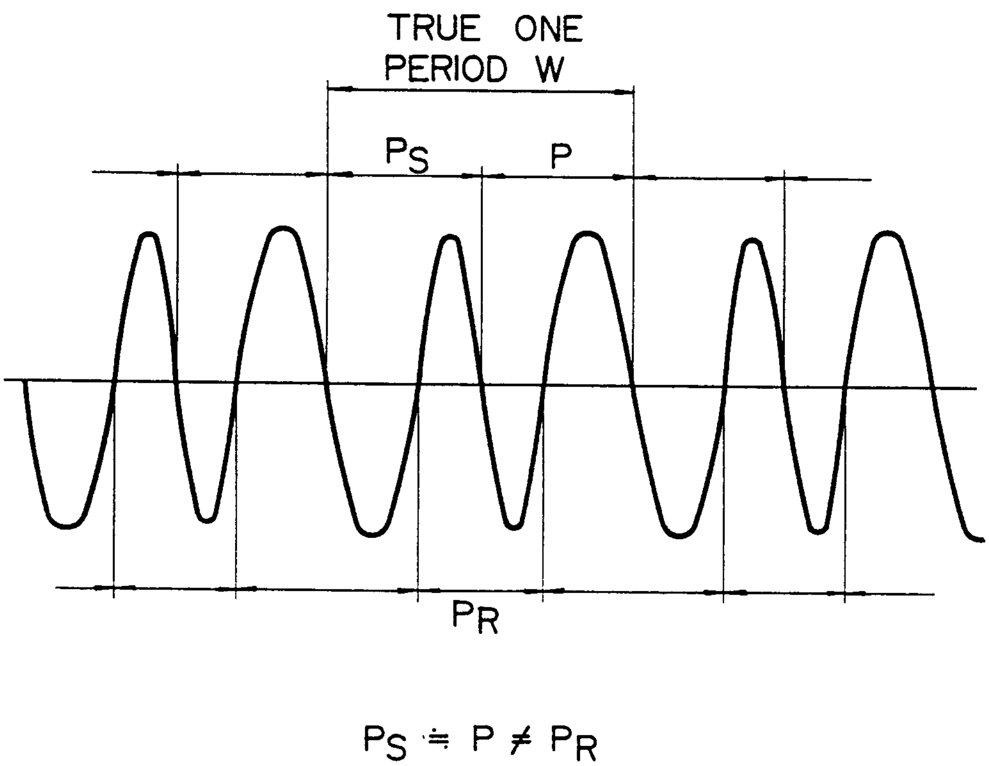


FIG. 25

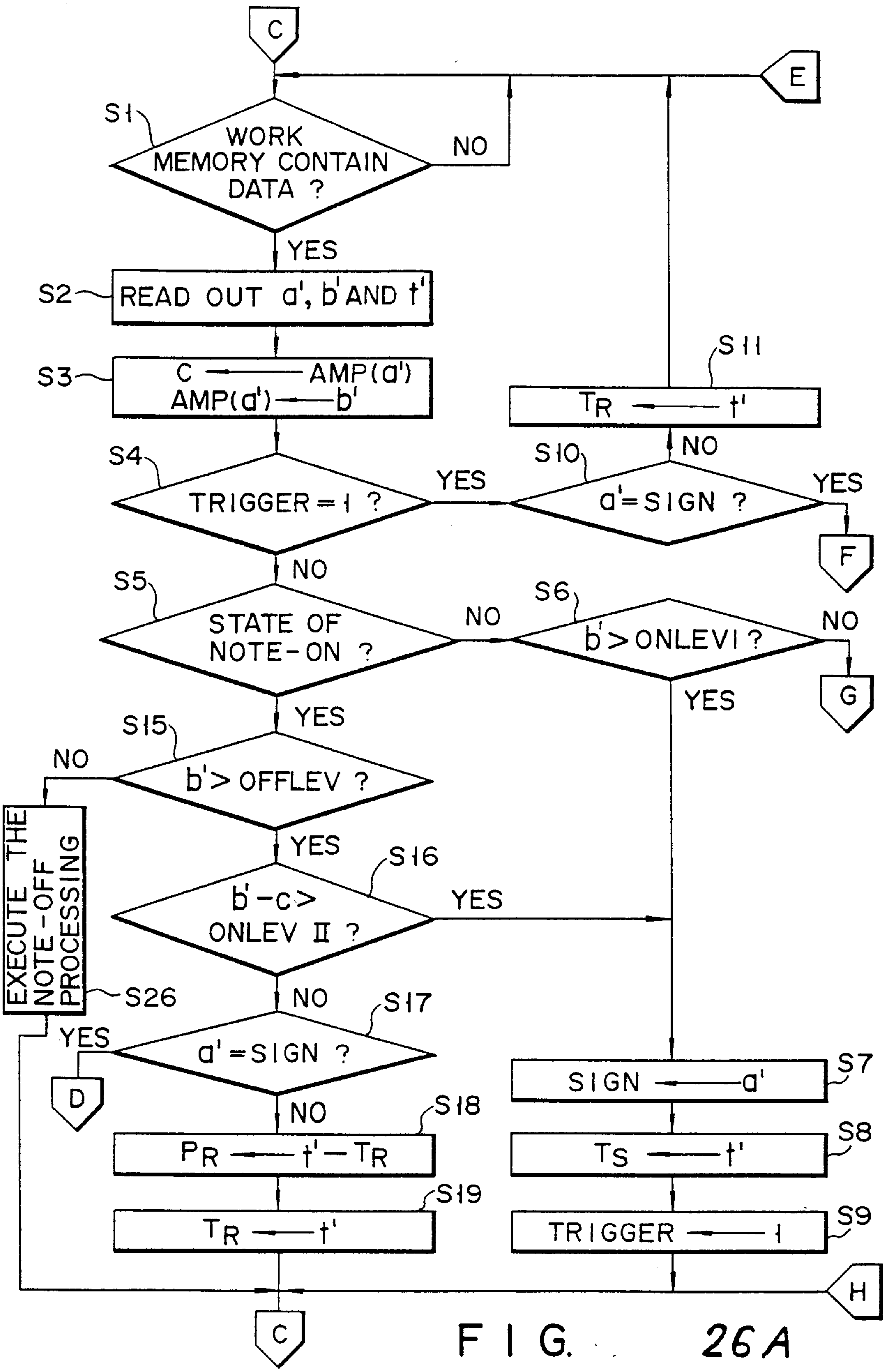

C
E
S1
WORK MEMORY CONTAIN DATA ?
NO
YES
S2
READ OUT a', b' AND t'
S3
C ← AMP(a')
AMP(a') ← b'
S11
TR ← t'
S4
TRIGGER = 1 ?
YES
S10
a' = SIGN ?
NO
YES
F
NO
S5
STATE OF NOTE-ON ?
NO
S6
b' > ONLEVI ?
NO
G
YES
YES
S15
b' > OFFLEV ?
NO
EXECUTE THE NOTE-OFF PROCESSING
S26
YES
S16
b' − c > ONLEV II ?
YES
NO
S17
a' = SIGN ?
YES
D
NO
S18
PR ← t' − TR
S19
TR ← t'
S7
SIGN ← a'
S8
TS ← t'
S9
TRIGGER ← 1
H
C

FIG. 26A

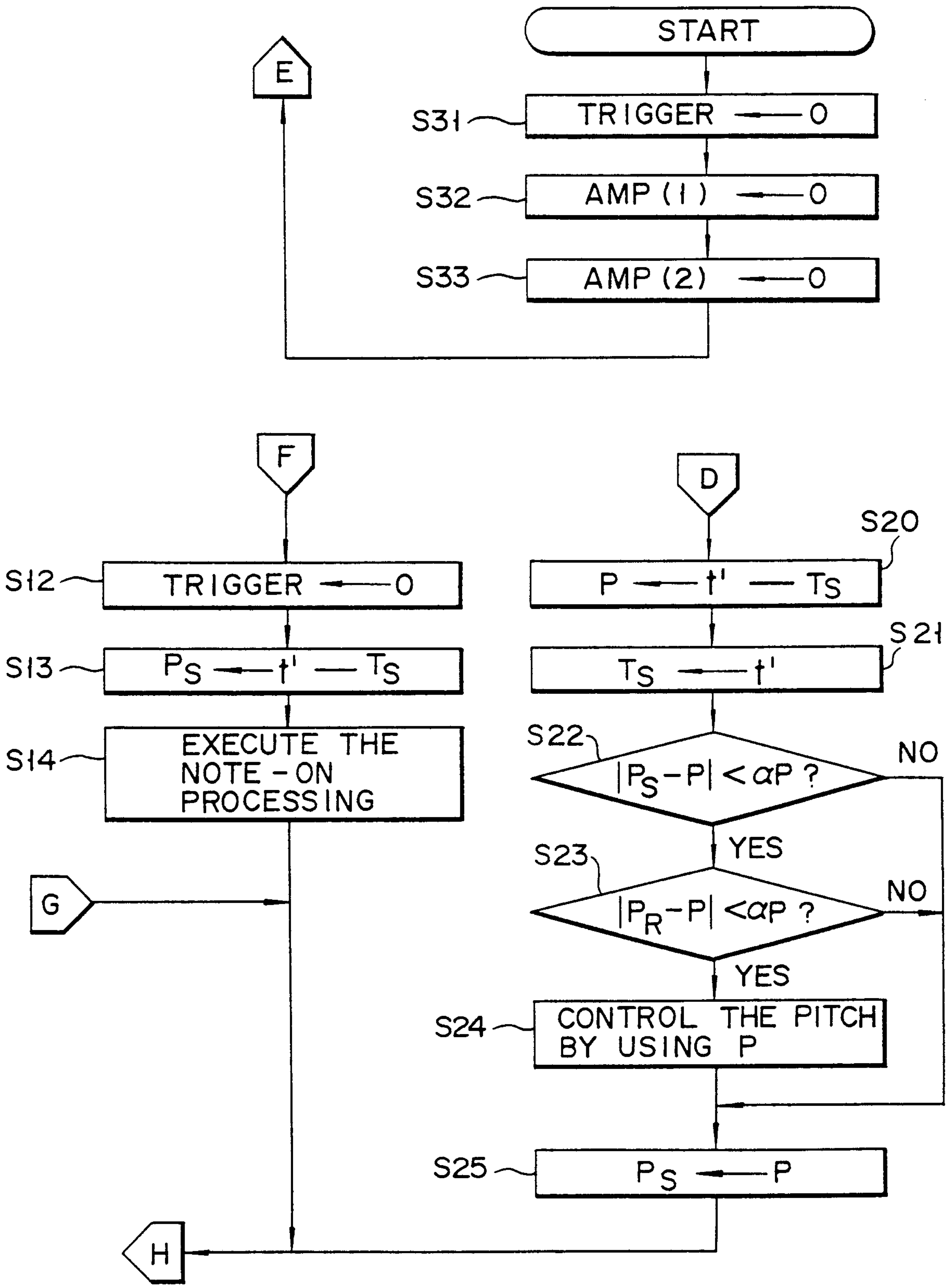

START
S31
TRIGGER ← 0
S32
AMP (1) ← 0
S33
AMP (2) ← 0
E
F
S12
TRIGGER ← 0
S13
PS ← t' − TS
S14
EXECUTE THE NOTE - ON PROCESSING
G
H
D
S20
P ← t' − TS
S21
TS ← t'
S22
|PS−P| < αP ?
NO
YES
S23
|PR−P| < αP ?
NO
YES
S24
CONTROL THE PITCH BY USING P
S25
PS ← P

FIG. 26B

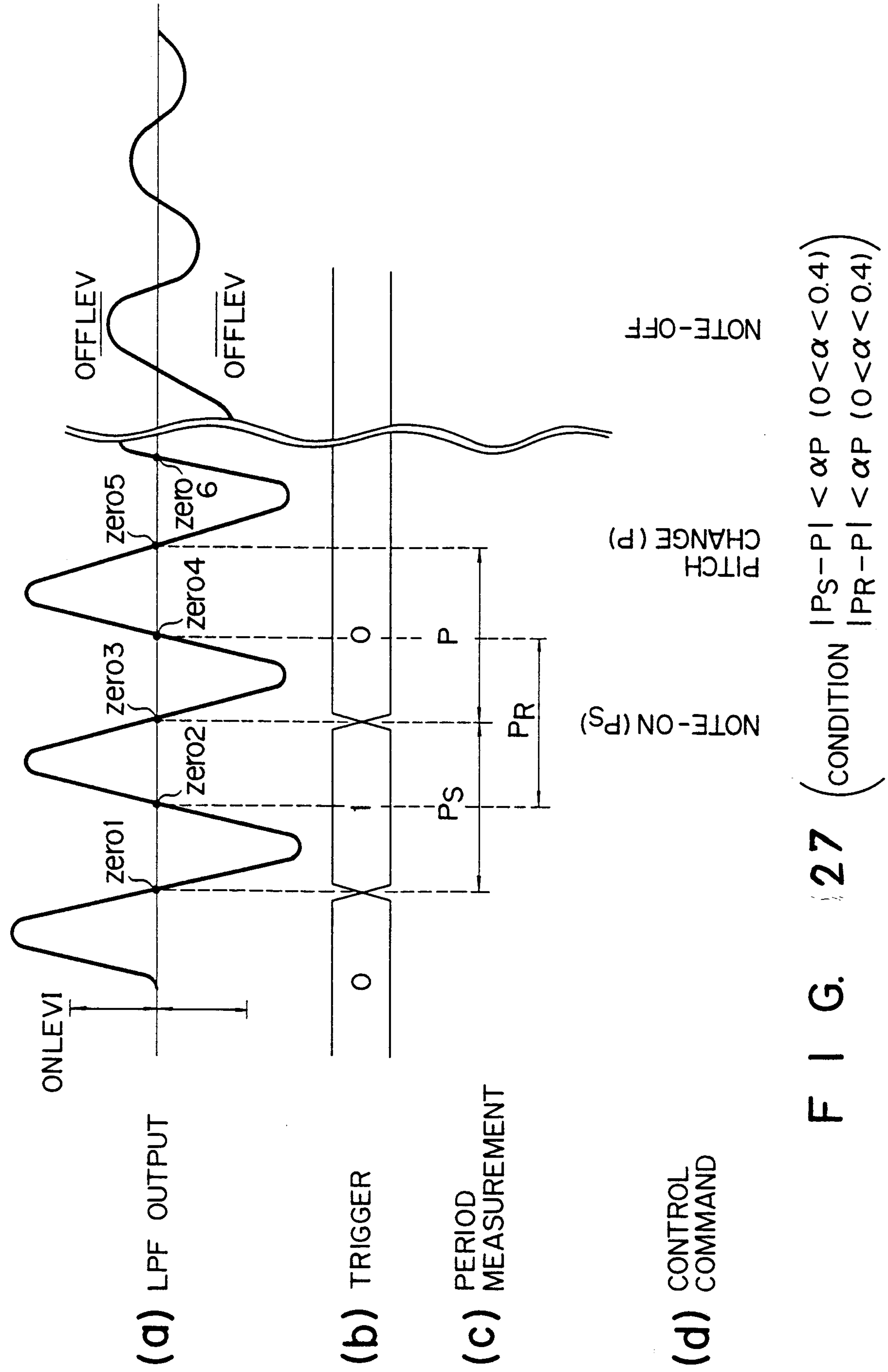

(a) LPF OUTPUT
(b) TRIGGER
(c) PERIOD MEASUREMENT
(d) CONTROL COMMAND
ONLEVI
zero1
zero2
zero3
zero4
zero5
zero 6
OFFLEV
OFFLEV
0
1
0
PS
P
PR
NOTE-ON (PS)
PITCH CHANGE (P)
NOTE-OFF
(CONDITION |PS-P| < αP (0<α<0.4)
|PR-P| < αP (0<α<0.4))

FIG. 27

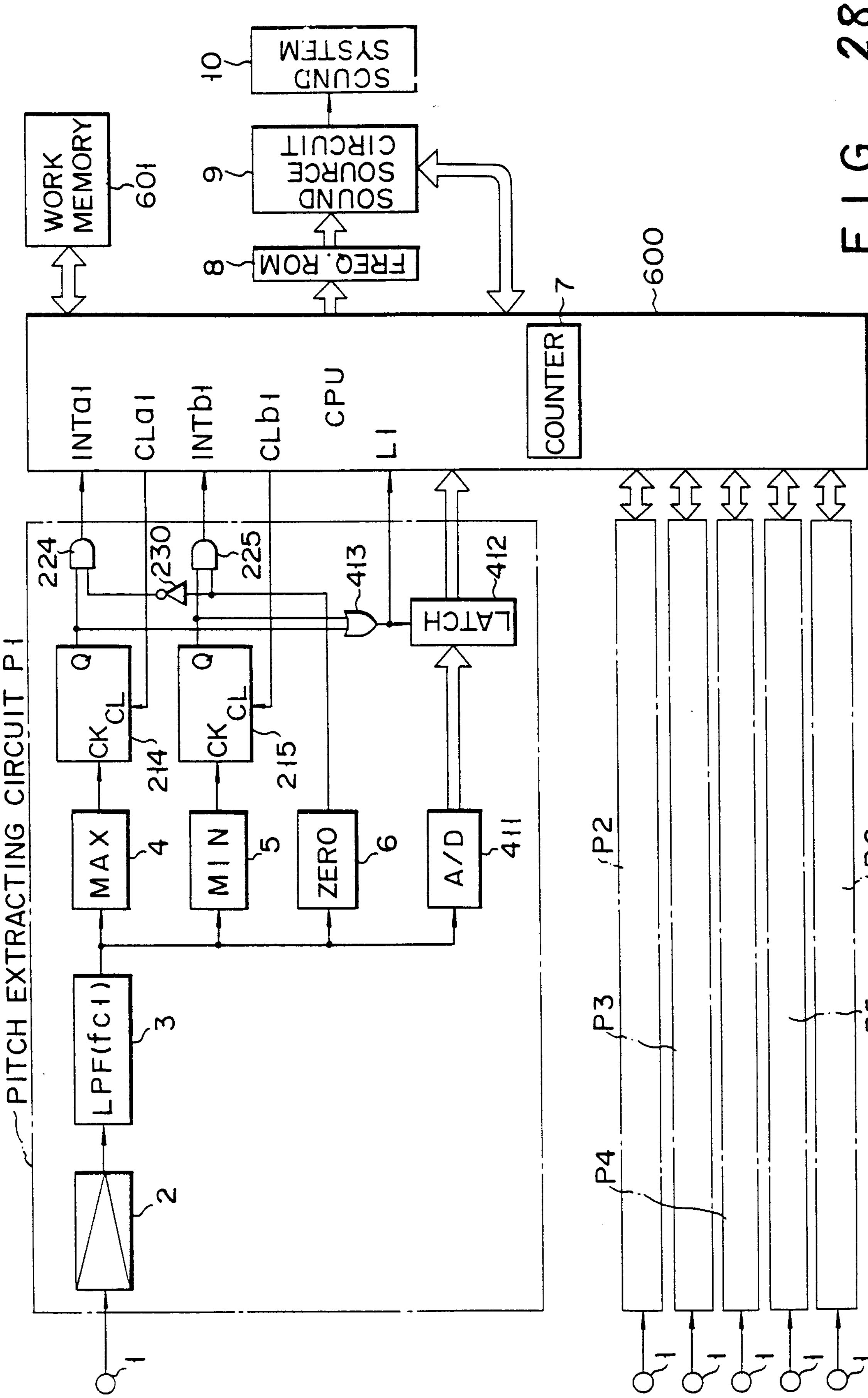

PITCH EXTRACTING CIRCUIT P1
1
2
LPF(fc1)
3
MAX
4
MIN
5
ZERO
6
A/D
411
CK
CL
Q
214
215
224
225
230
413
LATCH
412
INTa1
CLa1
INTb1
CLb1
CPU
L1
COUNTER
7
600
WORK MEMORY
601
FREQ. ROM
8
SOUND SOURCE CIRCUIT
9
SOUND SYSTEM
10
P2
P3
P4
P5
P6

FIG. 28