EXHIBIT 2

EXHIBIT 2

US005488196A

United States Patent [19]

Zimmerman et al.

[11] **Patent Number: 5,488,196**

[45] **Date of Patent: Jan. 30, 1996**

[54] **ELECTRONIC MUSICAL RE-PERFORMANCE AND EDITING SYSTEM**

[76] Inventors: **Thomas G. Zimmerman**, 7611 Hollanderry Pl., Cupertino, Calif. 95014; **Samuel P. Wantman**, 1124 Page St. Apt. #3, San Francisco, Calif. 94117

[21] Appl. No.: **183,489**

[22] Filed: **Jan. 19, 1994**

[51] **Int. Cl.**^{6} **G10H 1/02**; G10H 1/42; G10H 1/46

[52] **U.S. Cl.** **84/612**; 84/622; 84/629; 84/633; 84/DIG. 12

[58] **Field of Search** .... 84/600, 609–614, 84/622–633, 615, DIG. 12

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,336,734	6/1982	Polson	84/1.01
4,704,682	11/1987	Clynes	84/622
4,771,671	9/1988	Hoff	84/1.01
4,953,439	9/1990	Newell	84/655
4,969,384	11/1990	Kawasaki	84/612
4,974,486	12/1990	Wallace	84/609
4,980,519	12/1990	Mathews	178/19
4,981,457	1/1991	Iimura	446/408
5,125,313	6/1992	Hiyoshi	84/600
5,140,887	8/1992	Chapman	84/646
5,288,938	2/1994	Wheaton	84/600
5,355,762	10/1994	Tabata	84/609

OTHER PUBLICATIONS

Virtual Guitar User's Manual, 1994, pp. 1–25.

Mathews & Pierce, editors "Current Directions in Computer Music Research", 1989, MIT Press pp. 118, 119, 224, 225, 240, 241, 244, 245 ibid pp. 254–289.

Chabot, "Gesture Interfaces and Software Toolkit for Performance with Electronics", Summer 1990, Computer Music Journal pp. 15–27.

Boulange, "Conducting the Midi Orchestra", Summer 1990, Computer Music Journal pp. 34–46.

Rothstein & Metlay, "Products of Interest", Summer 1990, Computer Music Journal pp. 73–83, 88, 91.

J. Pressing, "Cybernetic Issues in Interactive Performance Systems" Computer Music Journal Spring 1990 pp. 12–25.

Robine & McAvinney "Programmable Finger Tracking Instrument Controller" Computer Music Journal Spring 1990 pp. 26–41.

Knapp & Lusted "A Bielectric Controller for Computer Music Application" Computer Music Journal Spring 1990 pp. 42–47.

Rosenboom "The Performing Brain" Computer Music Journal Spring 1990 pp. 48–50.

Rothstein & Metlay "Products of Interest" pp. 69–72, 79, 82, 83.

Freff "Midi Wind Controller" Keyboard Magazine Mar. 1989 pp. 114–115.

Krefeld "The Hand in the Web" Computer Music Journal Summer 1990 pp. 28–33.

Vercoe & Puckette "Synthetic Rehearsal" ICMC 1985 Proceedings pp. 275–278.

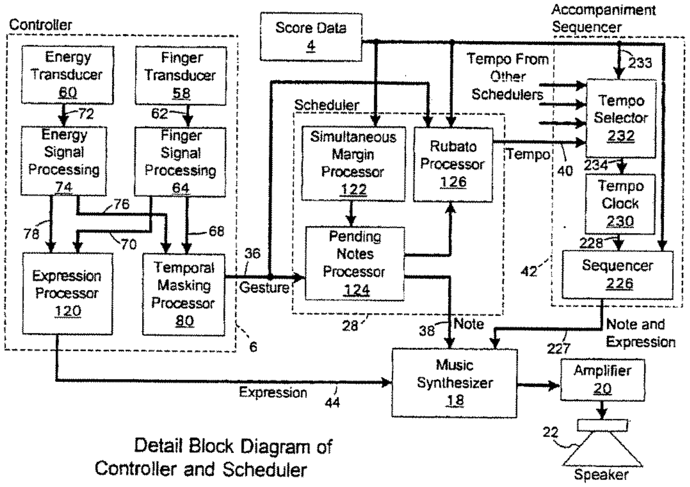
Machover Hyperinstruments: A Progress Report 1987–1991 MIT Media Laboratory.

Primary Examiner—Stanley J. Witkowski

[57] **ABSTRACT**

A music re-performance system allows a plurality of untrained instrumentalist to play pre-stored music using traditional playing techniques along with an automatic accompaniment at a tempo controlled by a selected-instrumentalist. Instrumentalist's gestures start and stop pre-stored score notes and temporal restrictions limit gestural timing errors. Expression parameters, including volume, timbre, and vibrato, are selectively updated, allowing editing of music sound files. A finger manipulation and energy driver controller model, including transducers and signal processing, accommodates wind and string instruments. Temporal masking prevents substantially concurrent finger and energy gestures, intended as simultaneous, from producing multiple false gestures.

34 Claims, 21 Drawing Sheets



Detail Block Diagram of Controller and Scheduler

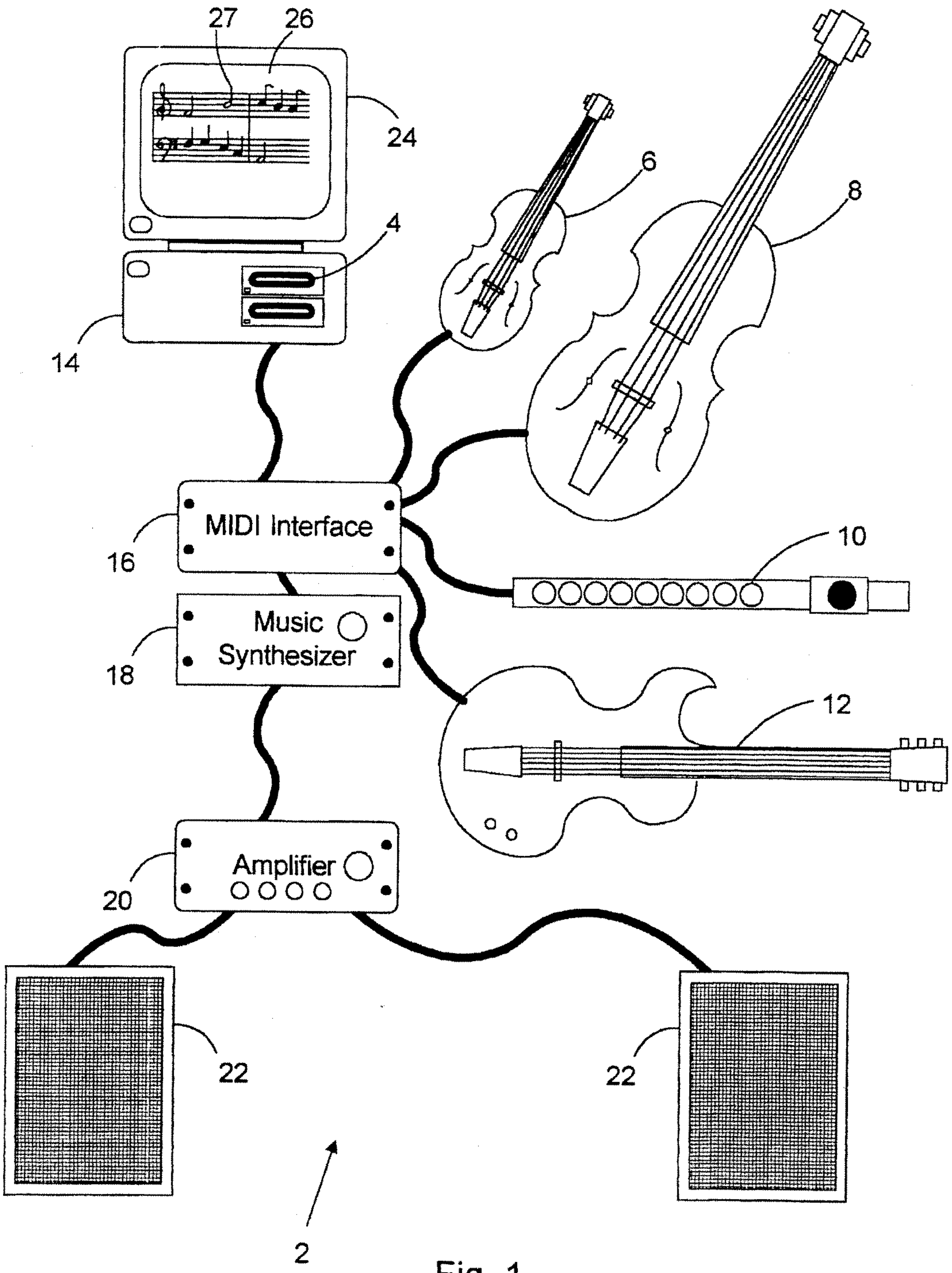

27
26
24
6
8
4
14
MIDI Interface
16
10
Music
Synthesizer
18
12
Amplifier
20
22
22
2

Fig. 1

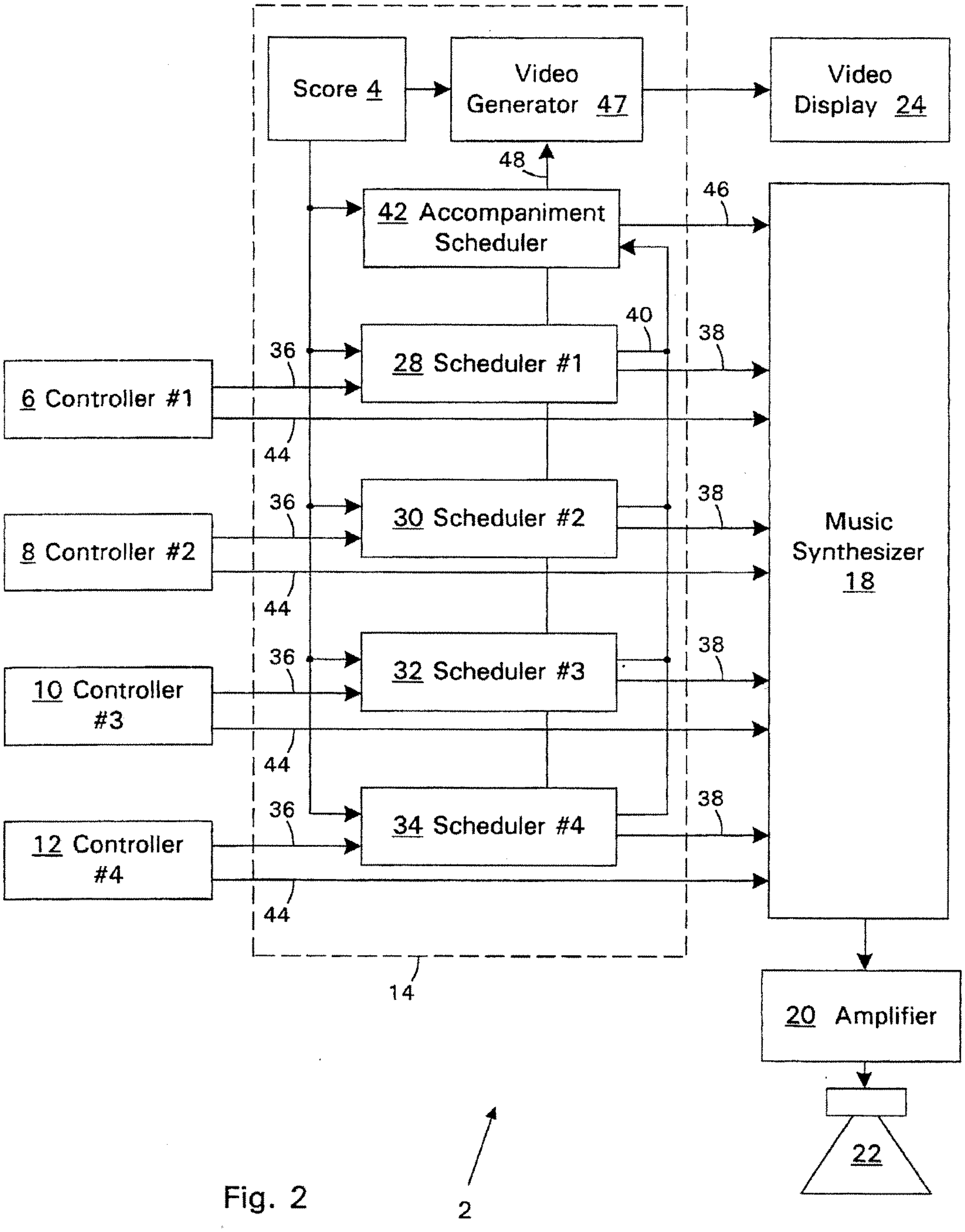

Score 4
Video Generator 47
Video Display 24
48
46
42 Accompaniment Scheduler
40
38
36
28 Scheduler #1
6 Controller #1
44
30 Scheduler #2
8 Controller #2
32 Scheduler #3
10 Controller #3
34 Scheduler #4
12 Controller #4
Music Synthesizer 18
14
20 Amplifier
22
2

Fig. 2

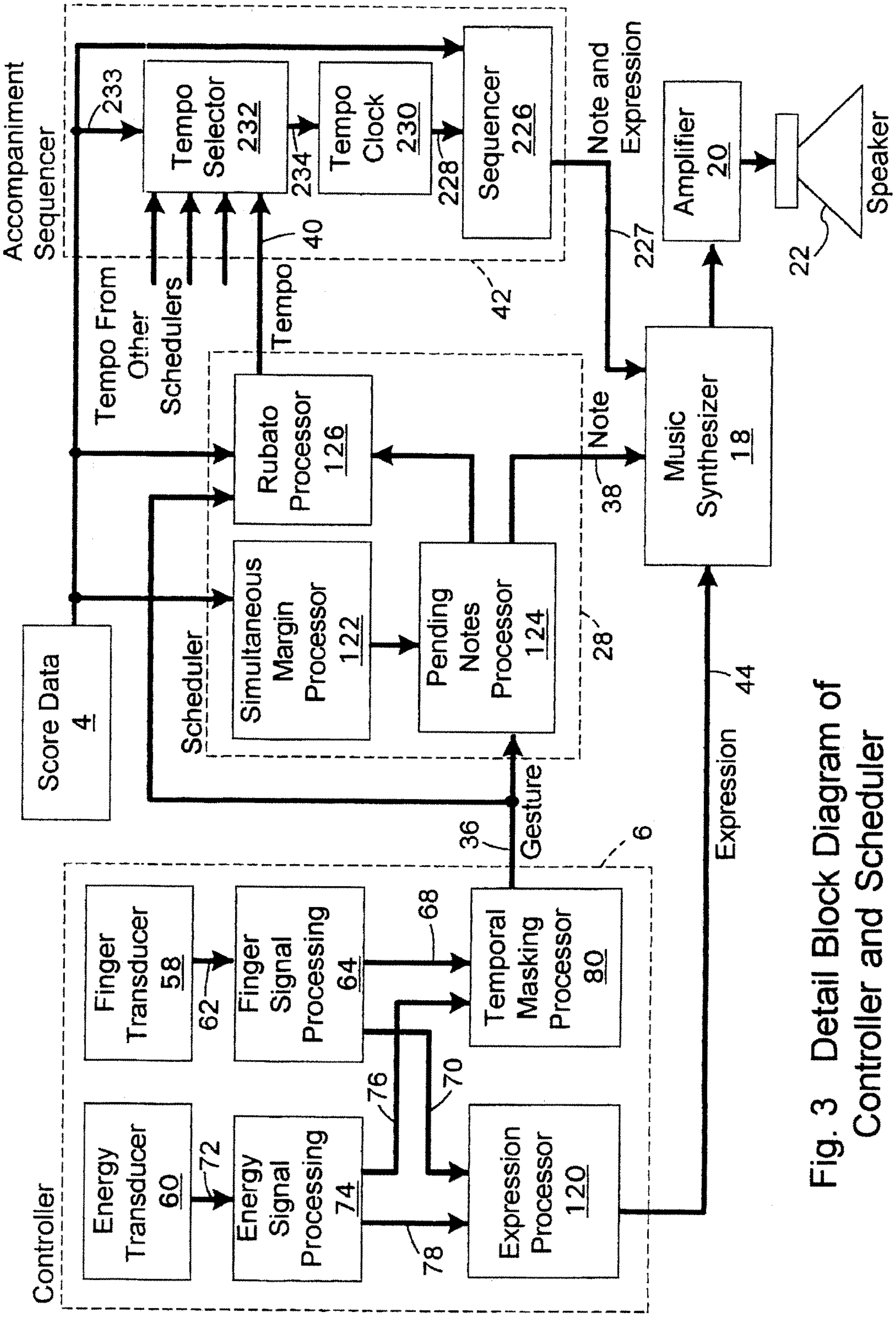
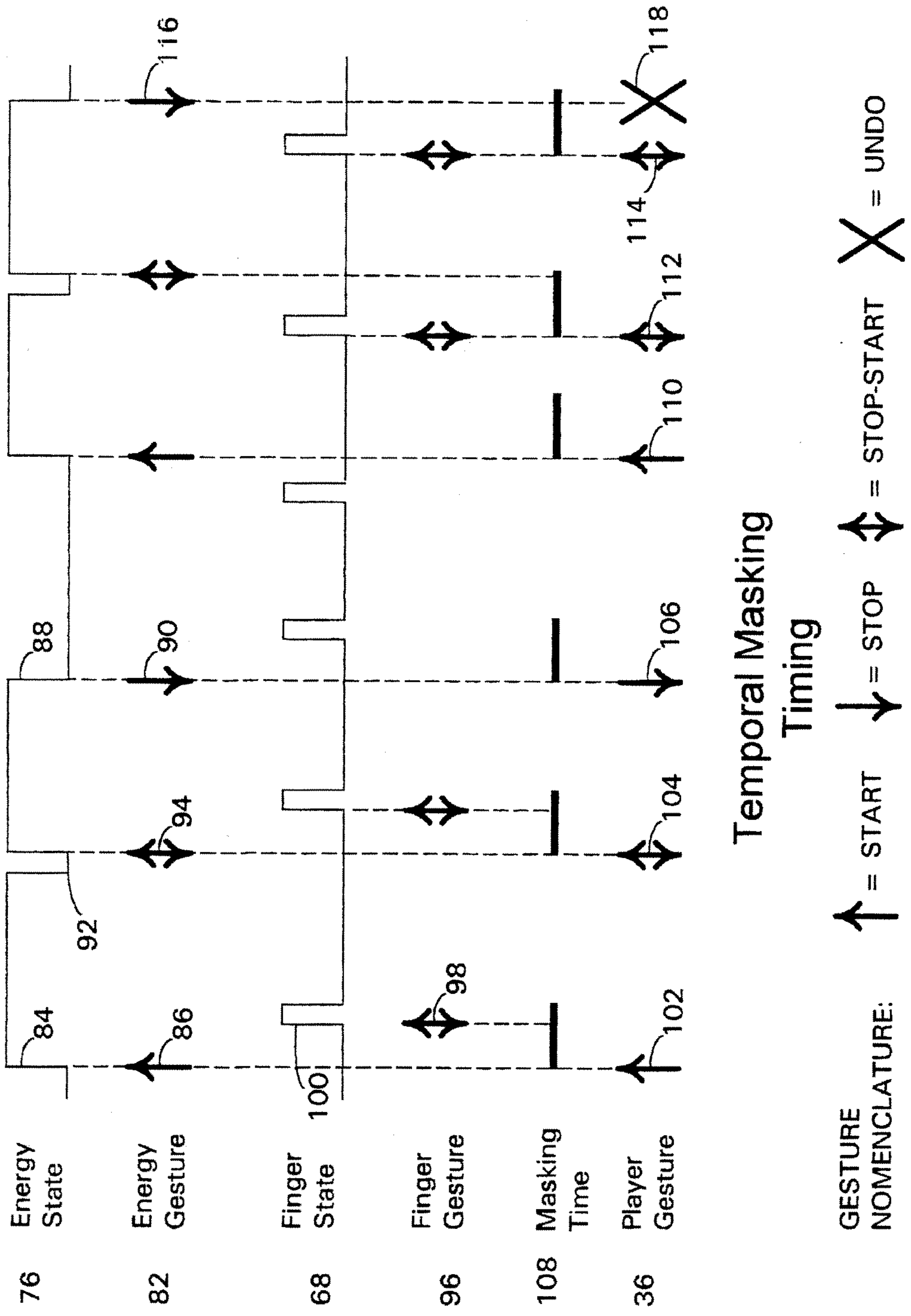


Fig. 3 Detail Block Diagram of Controller and Scheduler



Temporal Masking Timing

Fig. 4

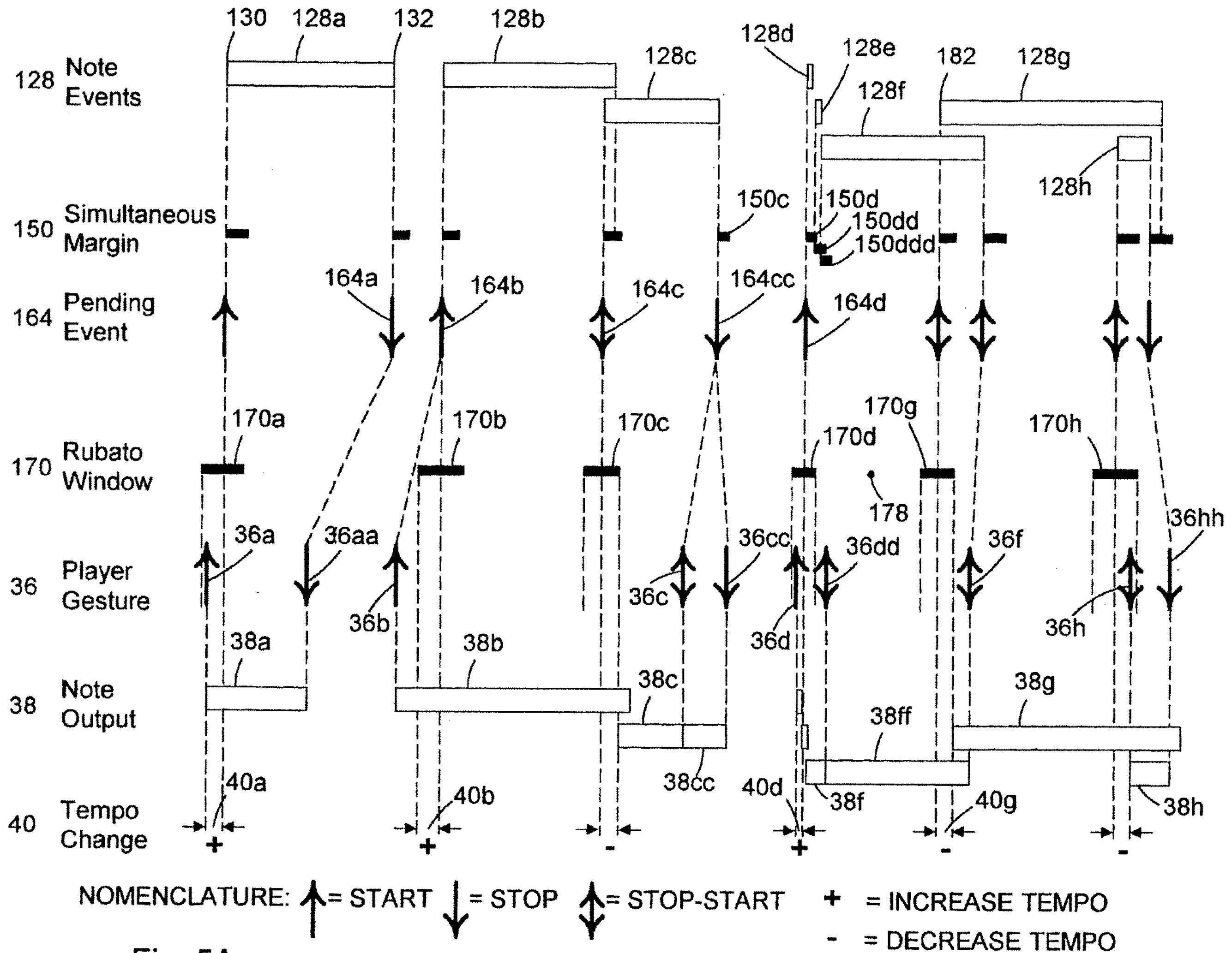

128 Note Events
150 Simultaneous Margin
164 Pending Event
170 Rubato Window
36 Player Gesture
38 Note Output
40 Tempo Change
130
128a
132
128b
128c
128d
128e
128f
182
128g
128h
150c
150d
150dd
150ddd
164a
164b
164c
164cc
164d
170a
170b
170c
170d
170g
170h
178
36a
36aa
36b
36c
36cc
36d
36dd
36f
36h
36hh
38a
38b
38c
38cc
38f
38ff
38g
38h
40a
40b
40d
40g
+
+
-
+
-
-
NOMENCLATURE:
= START
= STOP
= STOP-START
+ = INCREASE TEMPO
- = DECREASE TEMPO

Fig. 5A

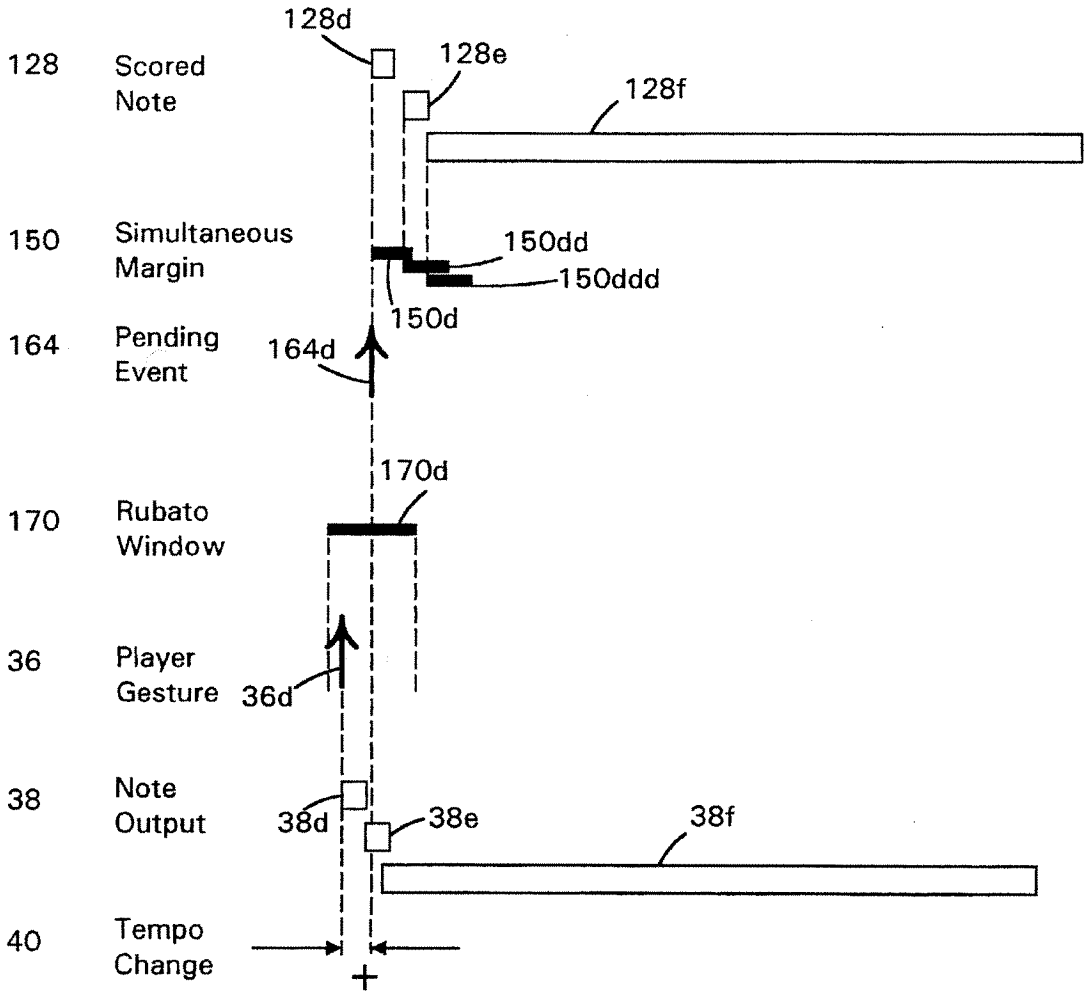

128d
128 Scored Note
128e
128f
150 Simultaneous Margin
150dd
150ddd
150d
164 Pending Event
164d
170d
170 Rubato Window
36 Player Gesture
36d
38 Note Output
38d
38e
38f
40 Tempo Change
+

Fig. 5B

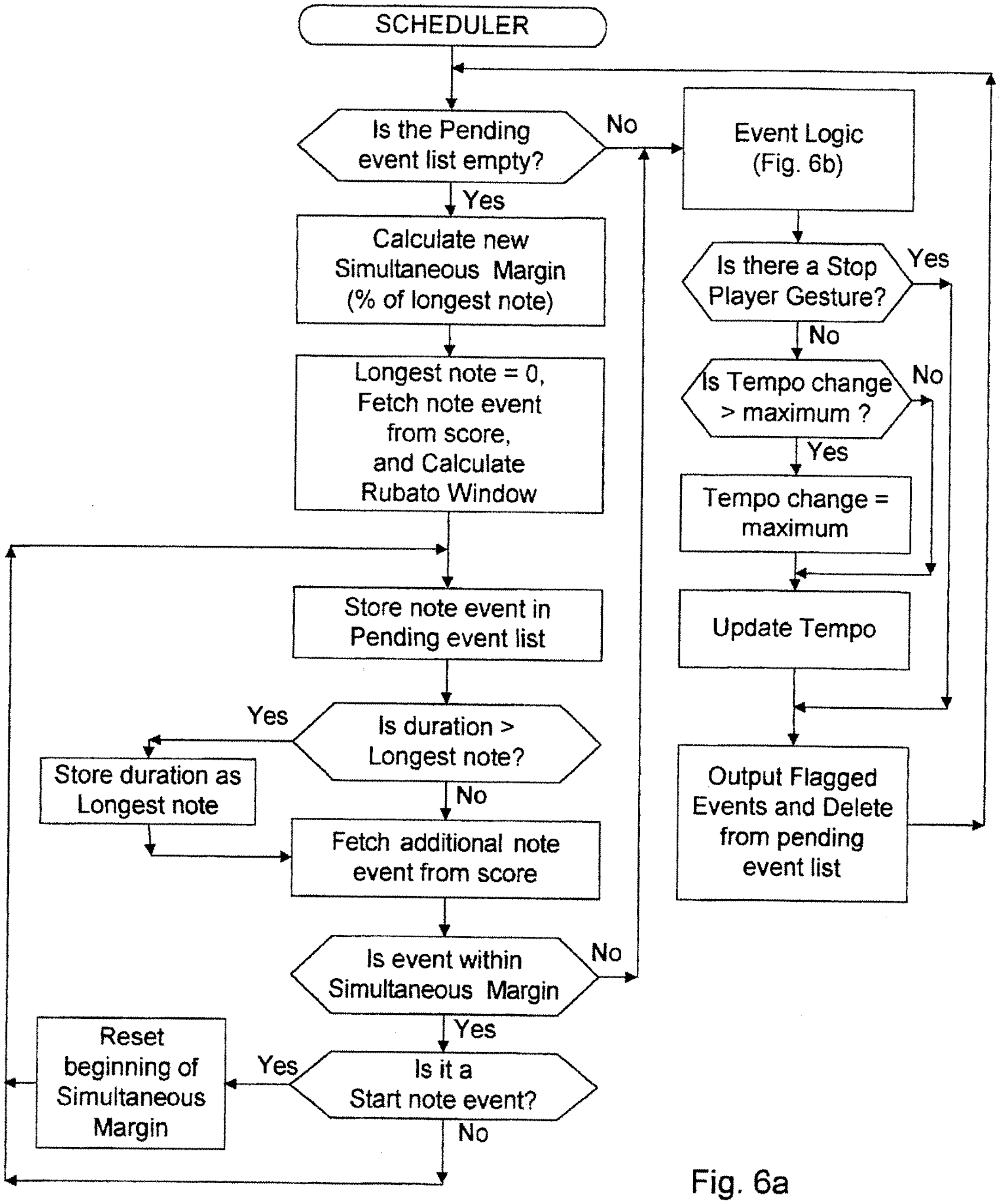

SCHEDULER
Is the Pending event list empty?
No
Yes
Event Logic (Fig. 6b)
Calculate new Simultaneous Margin (% of longest note)
Is there a Stop Player Gesture?
Yes
No
Longest note = 0, Fetch note event from score, and Calculate Rubato Window
Is Tempo change > maximum ?
No
Yes
Tempo change = maximum
Update Tempo
Store note event in Pending event list
Yes
Is duration > Longest note?
Store duration as Longest note
No
Output Flagged Events and Delete from pending event list
Fetch additional note event from score
Is event within Simultaneous Margin
No
Yes
Reset beginning of Simultaneous Margin
Yes
Is it a Start note event?
No

Fig. 6a

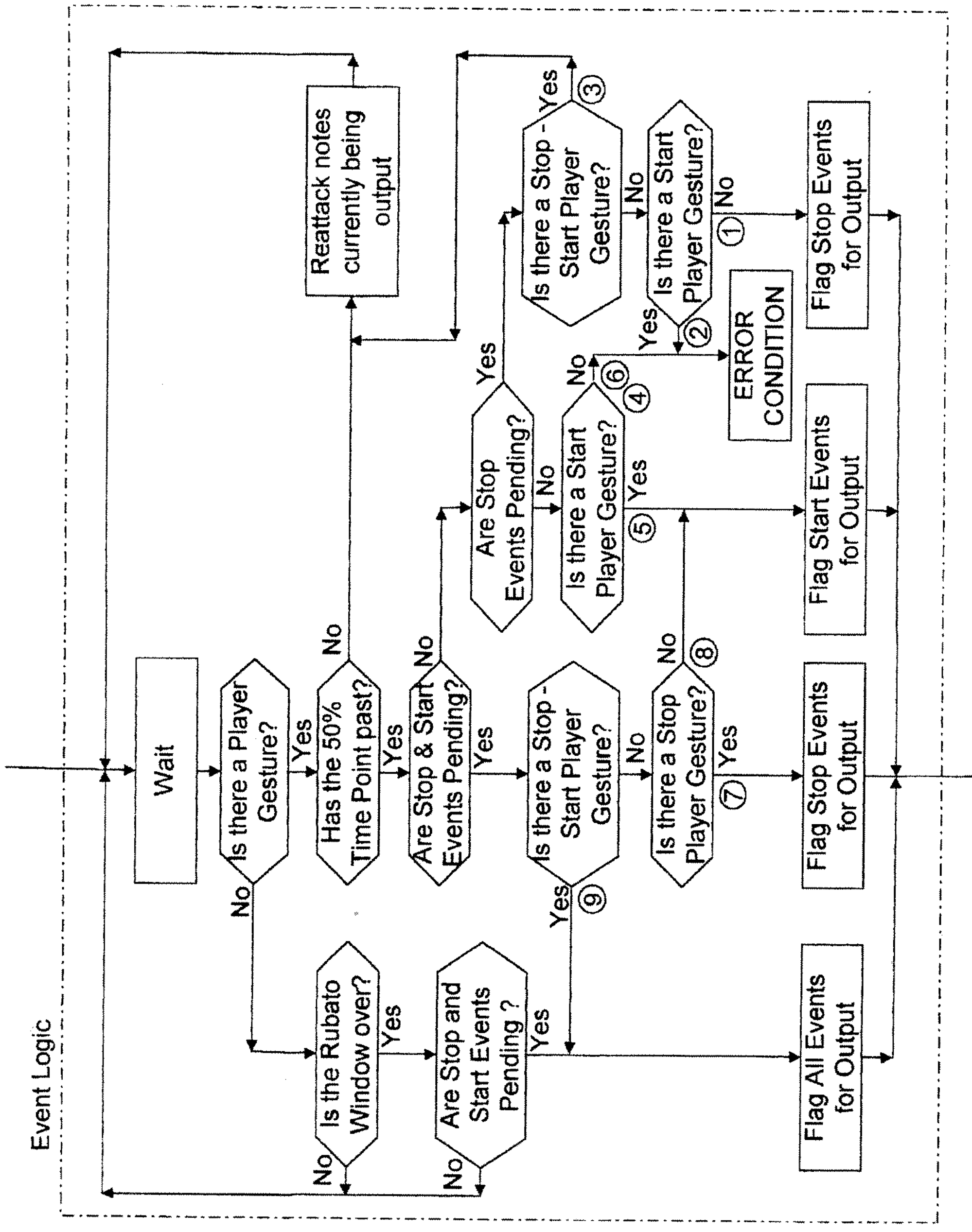

Event Logic
Wait
Is there a Player Gesture?
Yes
No
Has the 50% Time Point past?
Yes
No
Reattack notes currently being output
Are Stop & Start Events Pending?
Yes
No
Are Stop Events Pending?
Yes
No
Is there a Stop - Start Player Gesture?
Yes
3
No
Is there a Start Player Gesture?
No
1
Yes
2
ERROR CONDITION
Flag Stop Events for Output
Is there a Start Player Gesture?
No
6
4
Yes
5
Flag Start Events for Output
Is there a Stop - Start Player Gesture?
No
Yes
9
Is there a Stop Player Gesture?
No
8
Yes
7
Flag Stop Events for Output
Is the Rubato Window over?
Yes
No
Are Stop and Start Events Pending ?
Yes
No
Flag All Events for Output

Fig. 6b

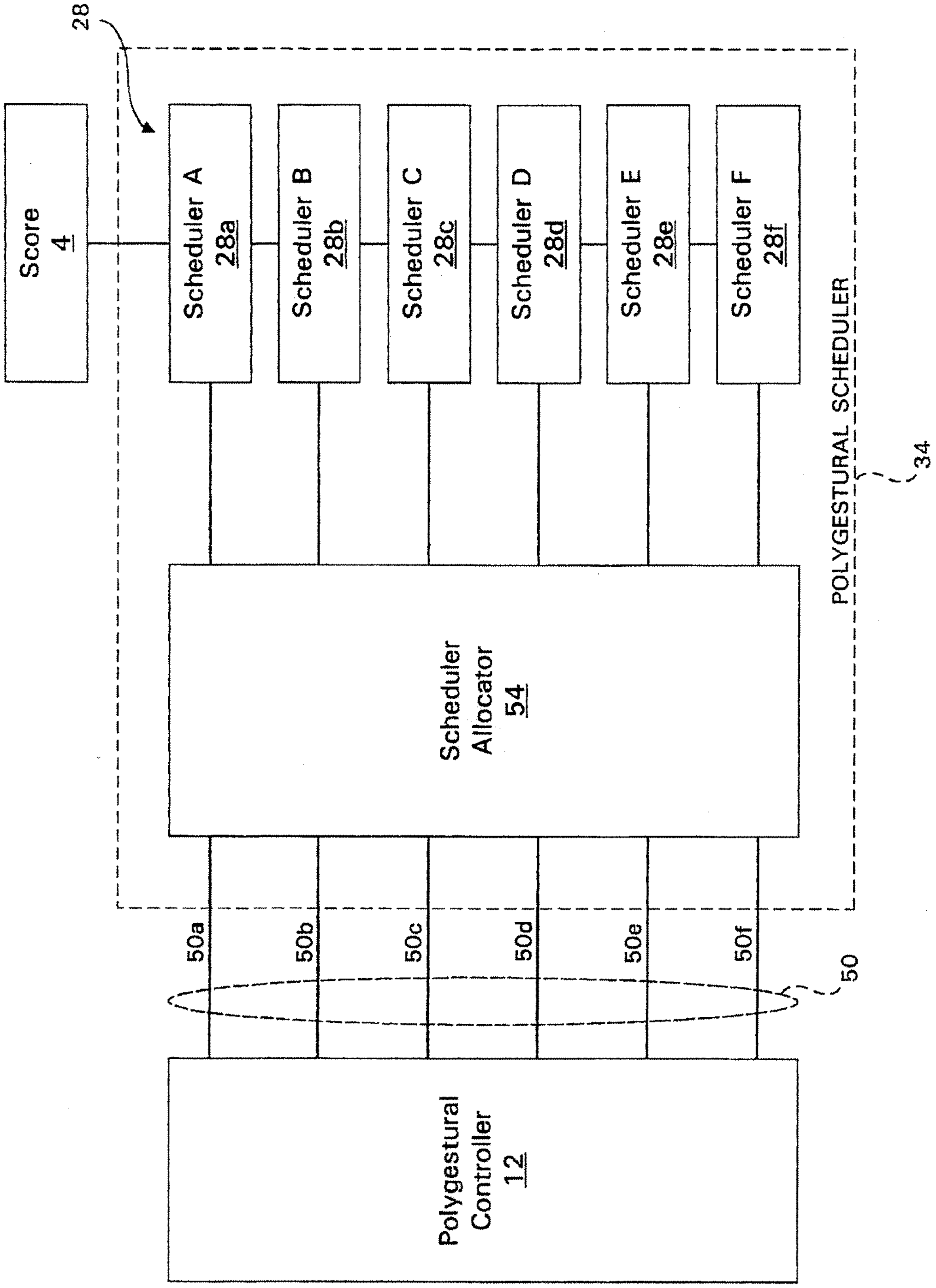

28
Score
4
Scheduler A
28a
Scheduler B
28b
Scheduler C
28c
Scheduler D
28d
Scheduler E
28e
Scheduler F
28f
POLYGESTURAL SCHEDULER
34
Scheduler
Allocator
54
50a
50b
50c
50d
50e
50f
50
Polygestural
Controller
12

Fig. 7

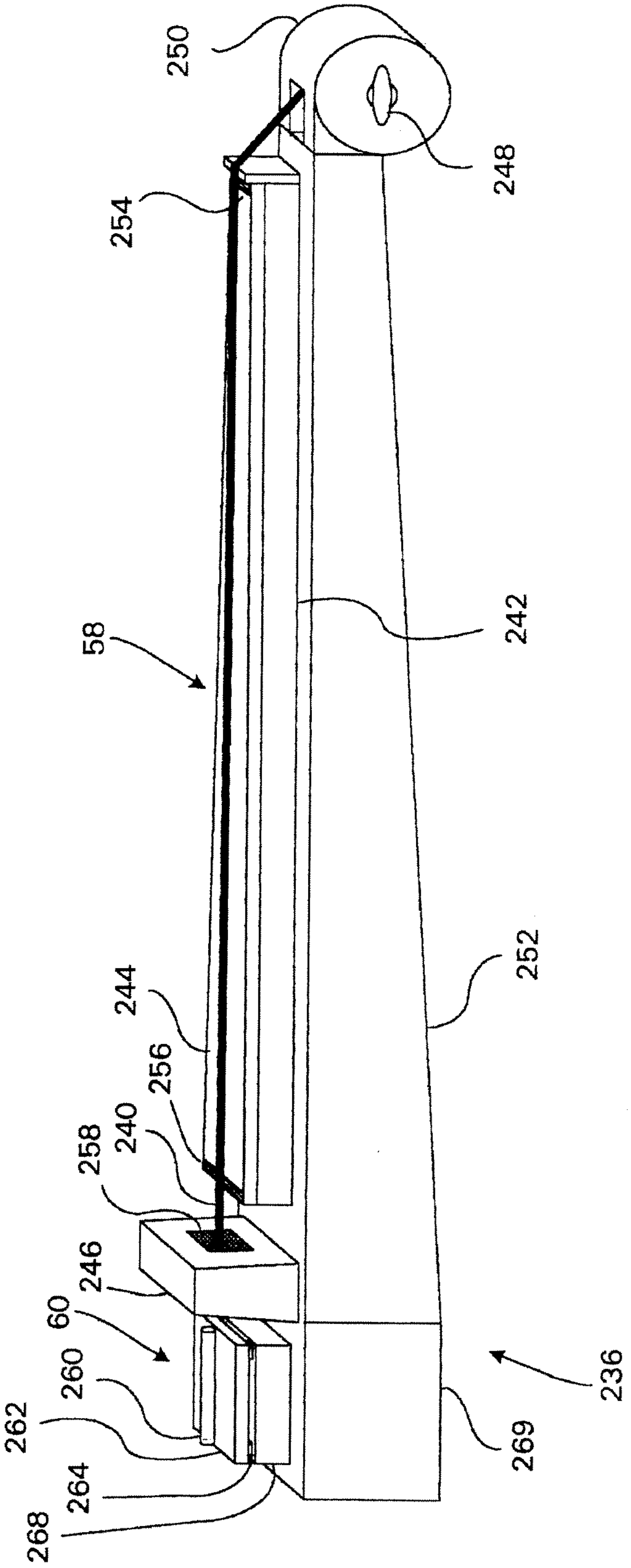

250
248
254
58
242
252
244
256
240
258
246
60
260
262
264
268
236
269

Fig. 8

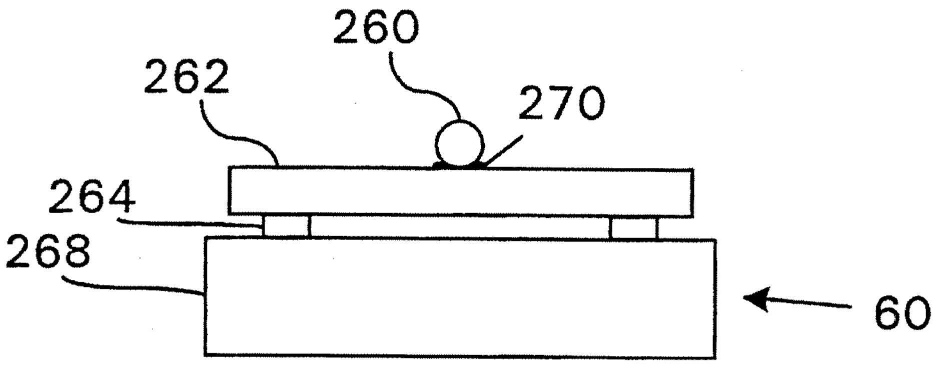

260
262
270
264
268
60

Fig. 9b

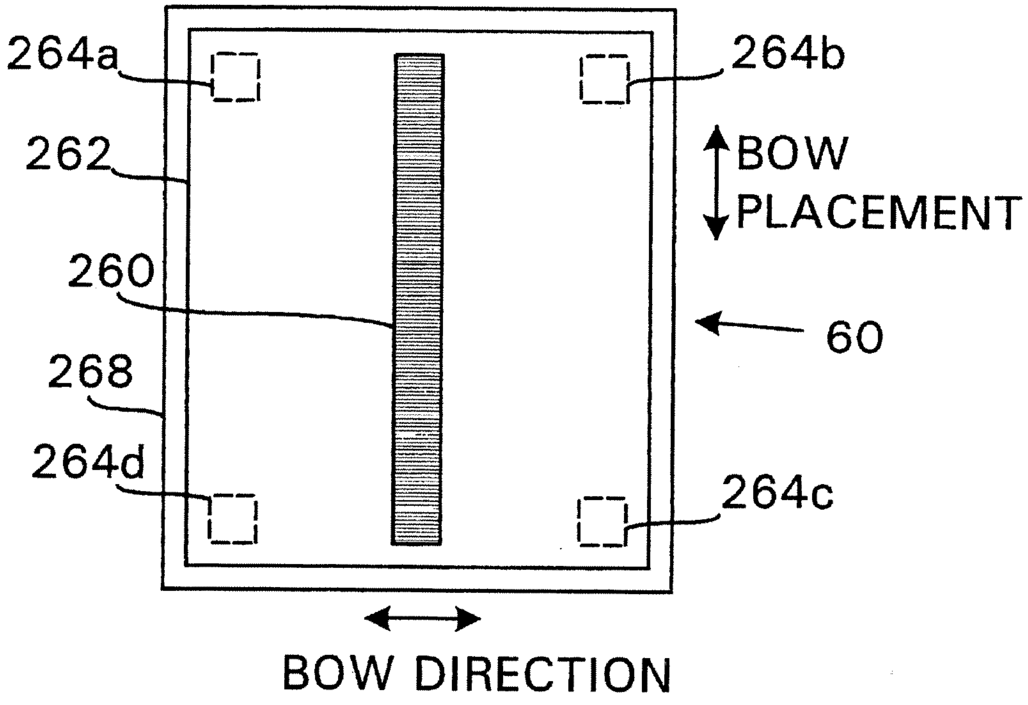

264a
264b
262
BOW PLACEMENT
260
60
268
264d
264c
BOW DIRECTION

Fig. 9a

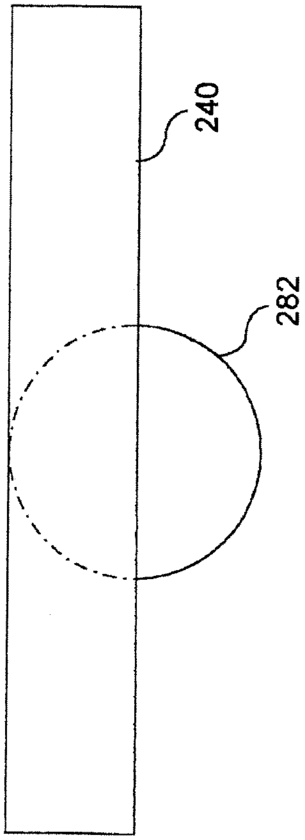


Fig. 10B

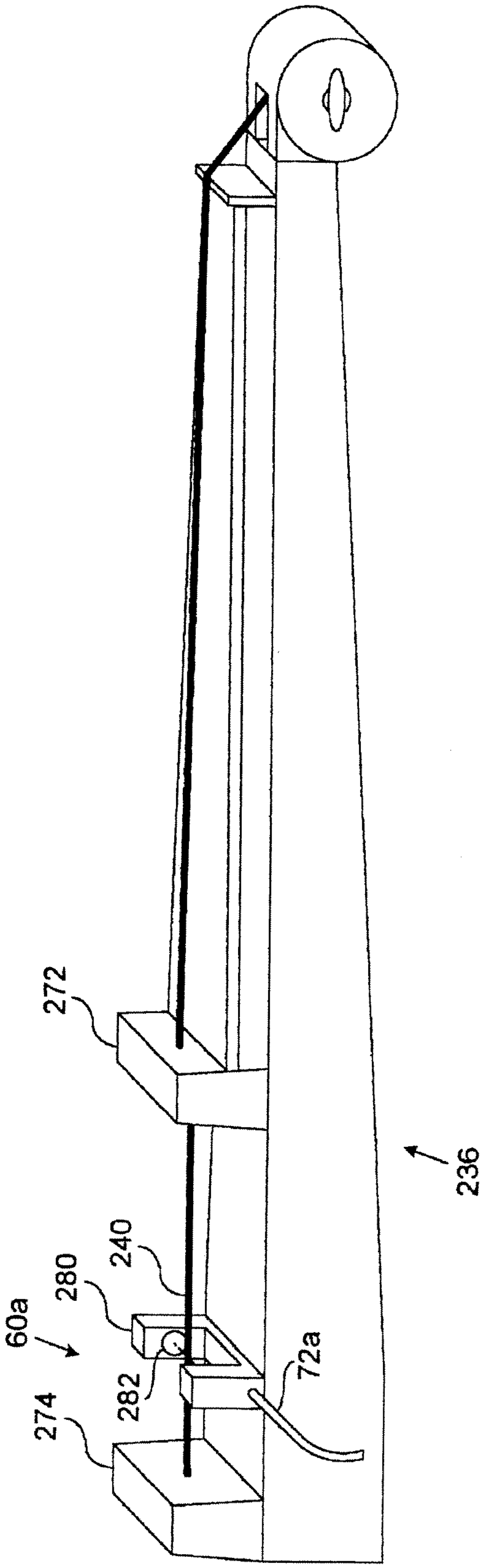


Fig. 10A

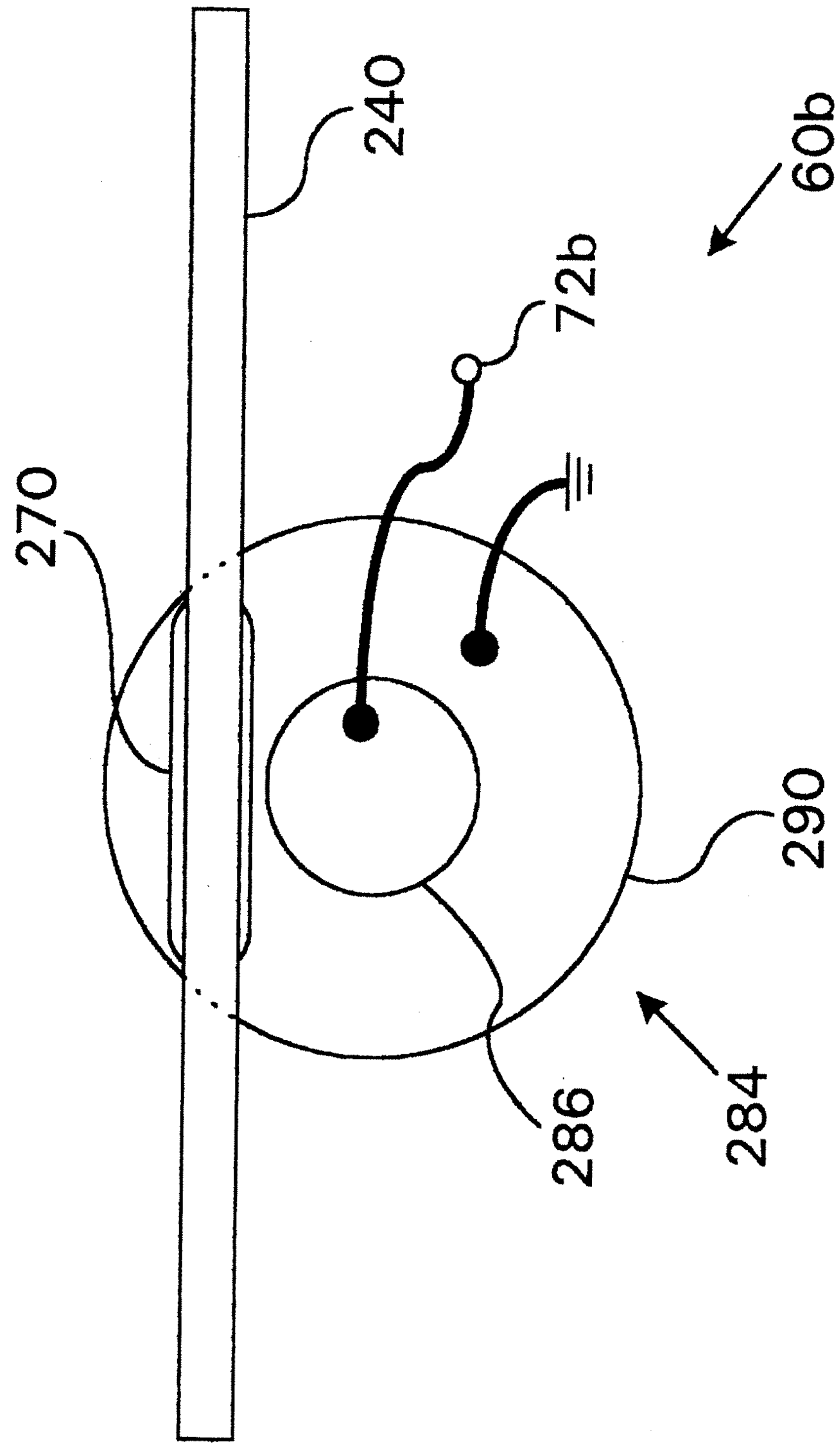


Fig. 11

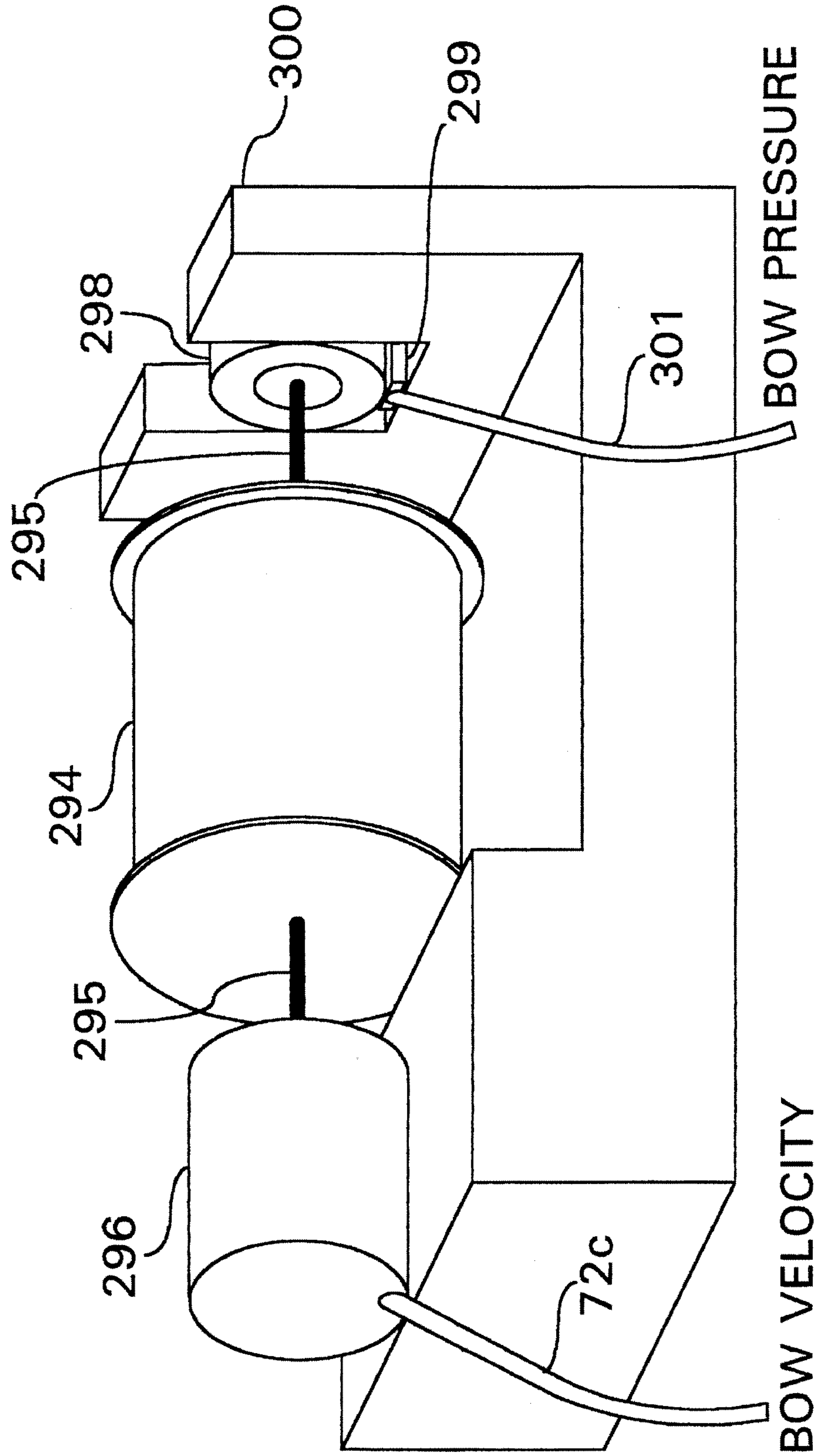


Fig. 12

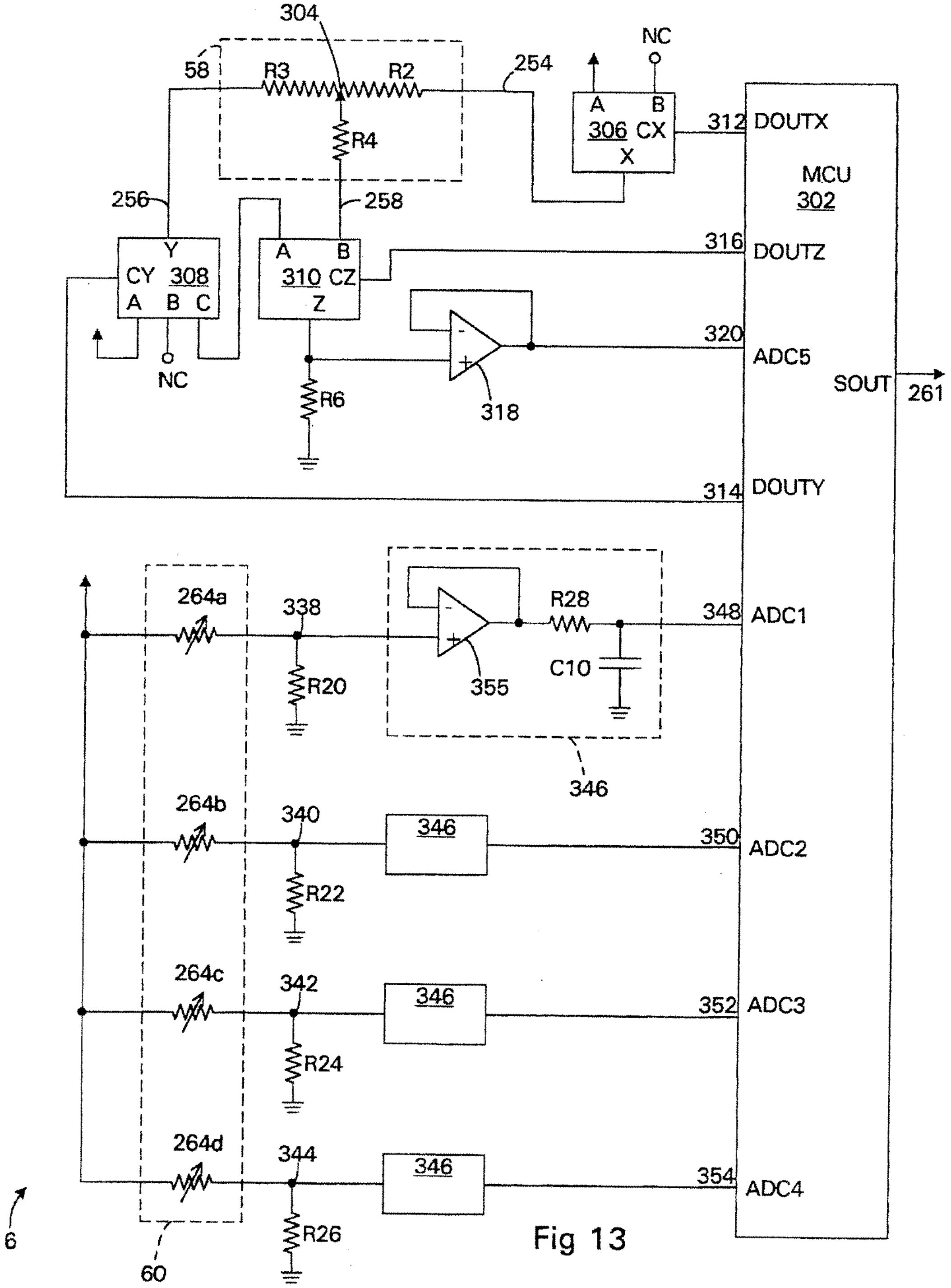

304
58
R3
R2
254
NC
A
B
306
CX
X
312
DOUTX
MCU
302
R4
256
258
Y
CY
308
A
B
C
A
B
310
CZ
Z
316
DOUTZ
320
ADC5
NC
R6
318
SOUT
261
314
DOUTY
264a
338
R28
348
ADC1
R20
355
C10
346
264b
340
346
350
ADC2
R22
264c
342
346
352
ADC3
R24
264d
344
346
354
ADC4
R26
6
60

Fig 13

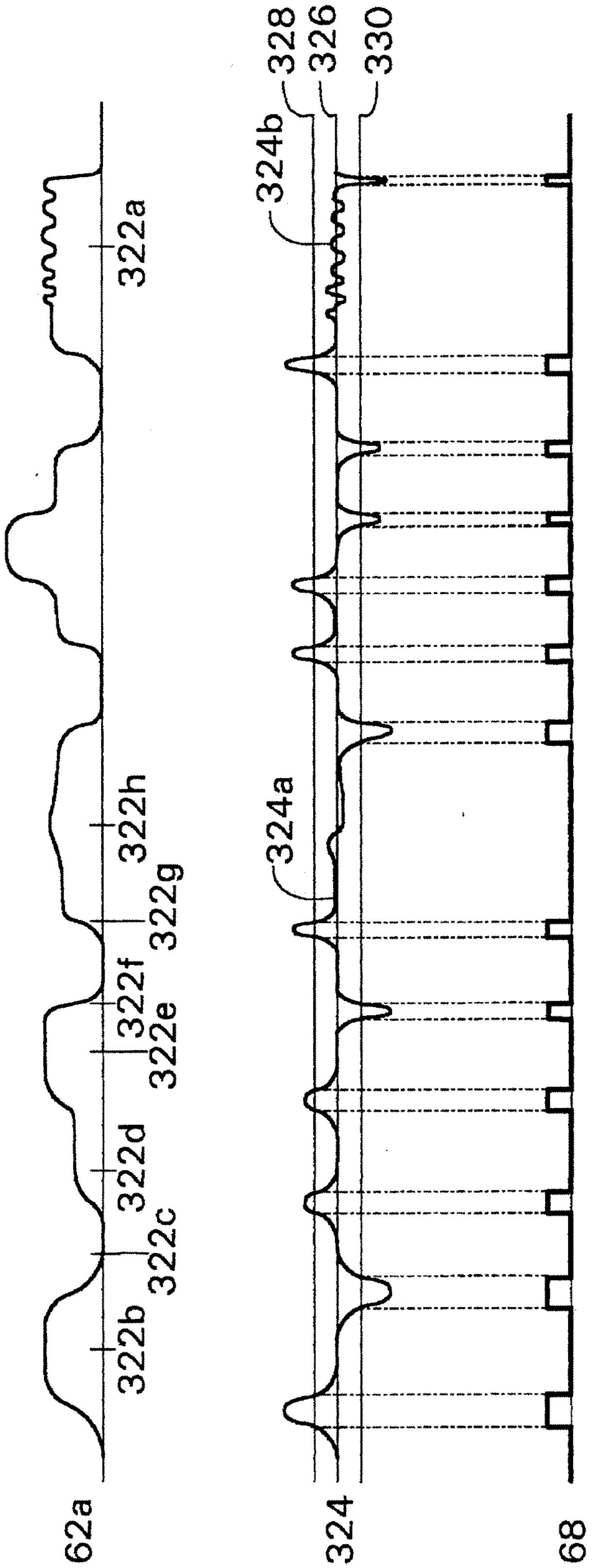

62a
322a
322b
322c
322d
322e
322f
322g
322h
324
324a
324b
326
328
330
68

Fig. 14

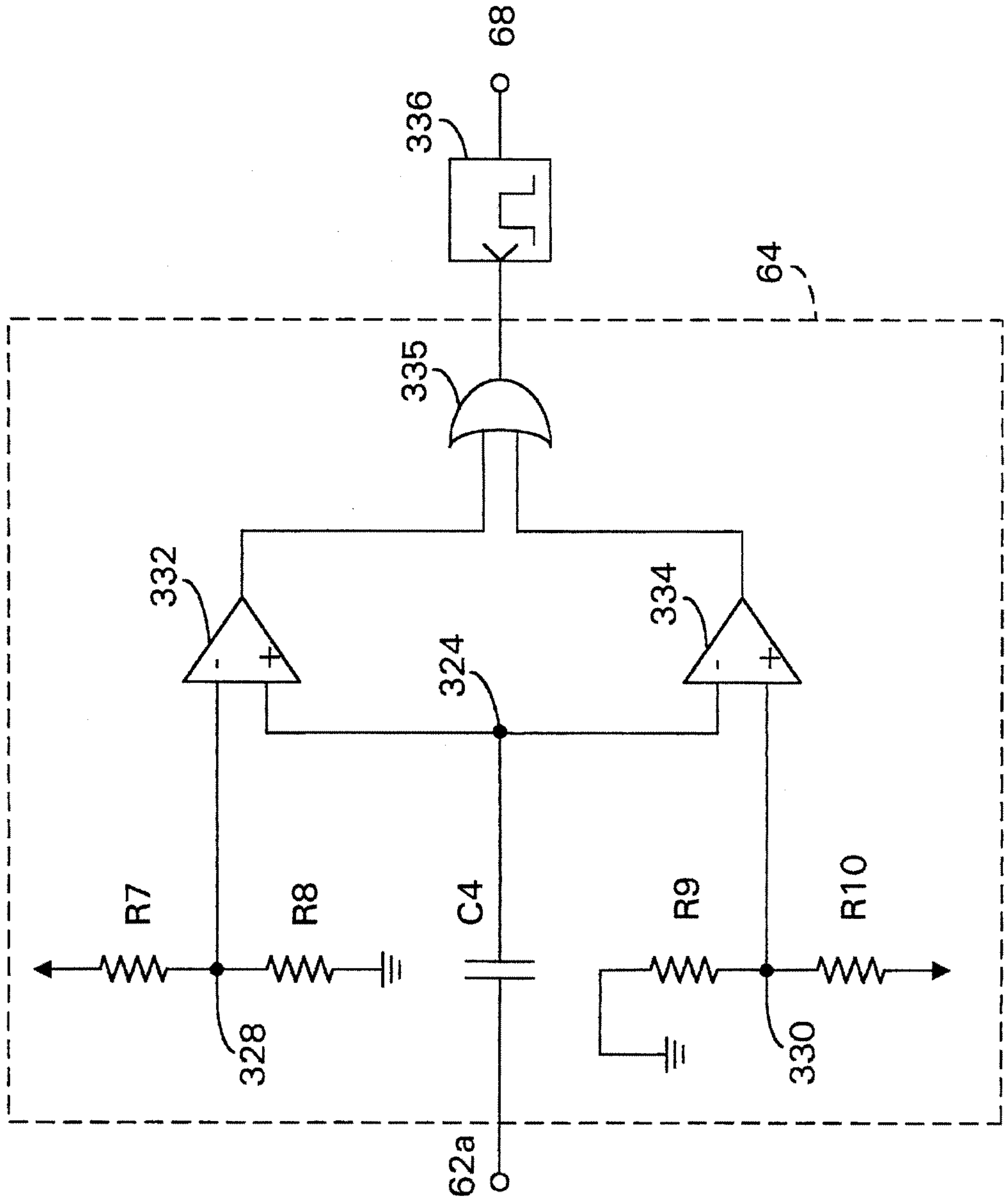


Fig. 15

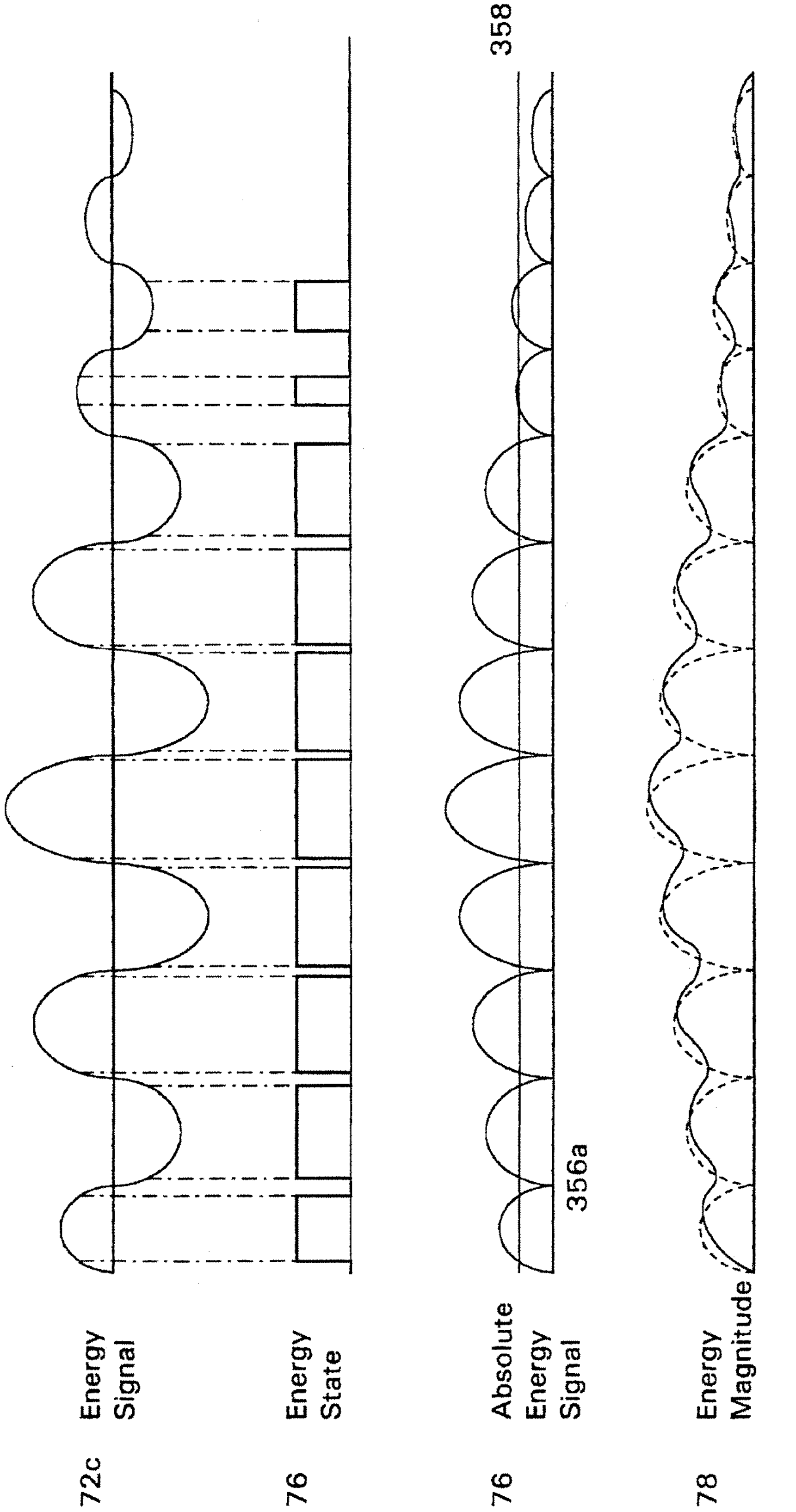

72c Energy Signal
76 Energy State
76 Absolute Energy Signal
78 Energy Magnitude
358
356a

Fig. 16

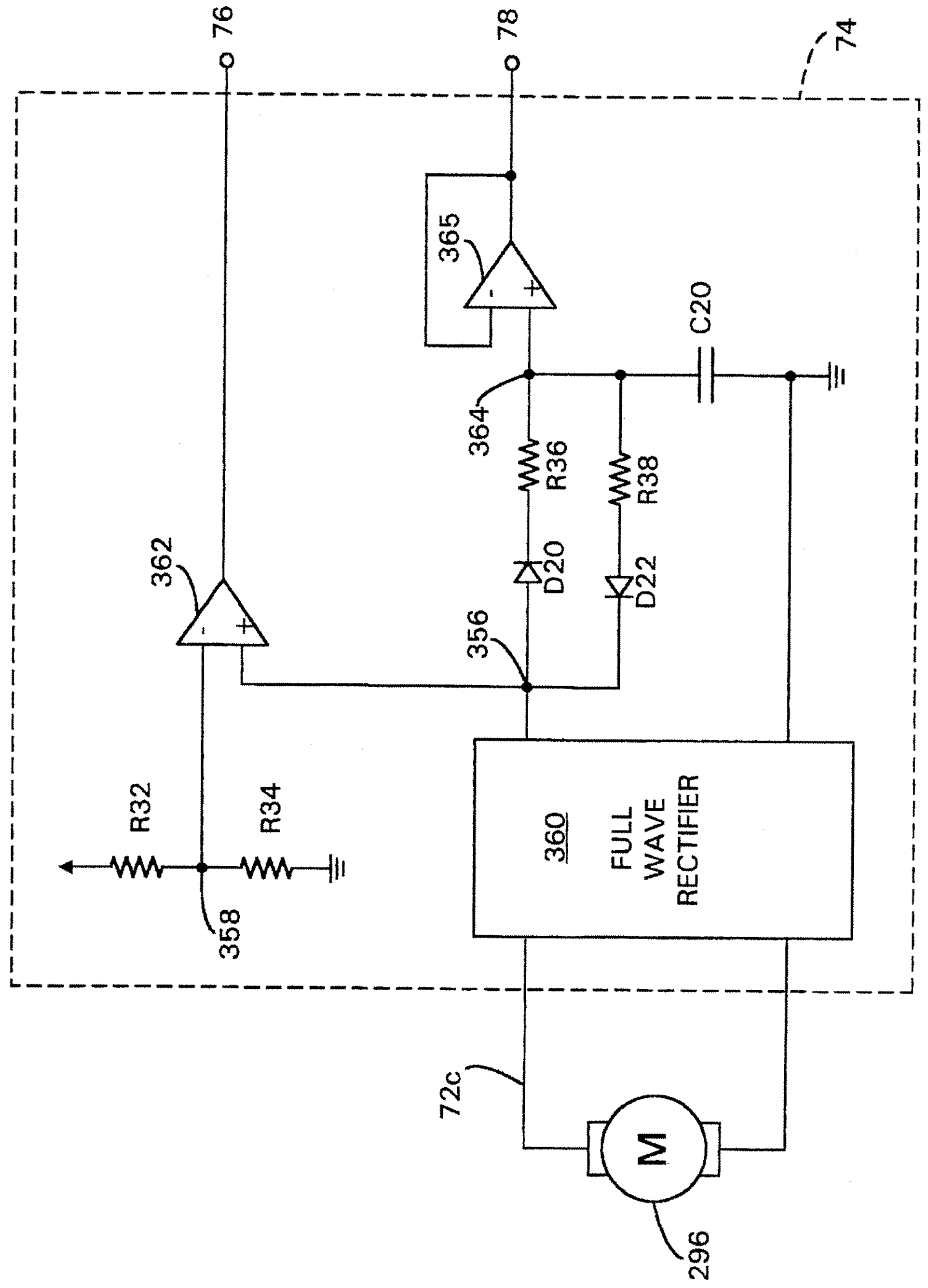

76
78
74
365
C20
364
R36
R38
362
D20
D22
356
R32
R34
360
FULL
WAVE
RECTIFIER
358
72c
M
296

Fig. 17

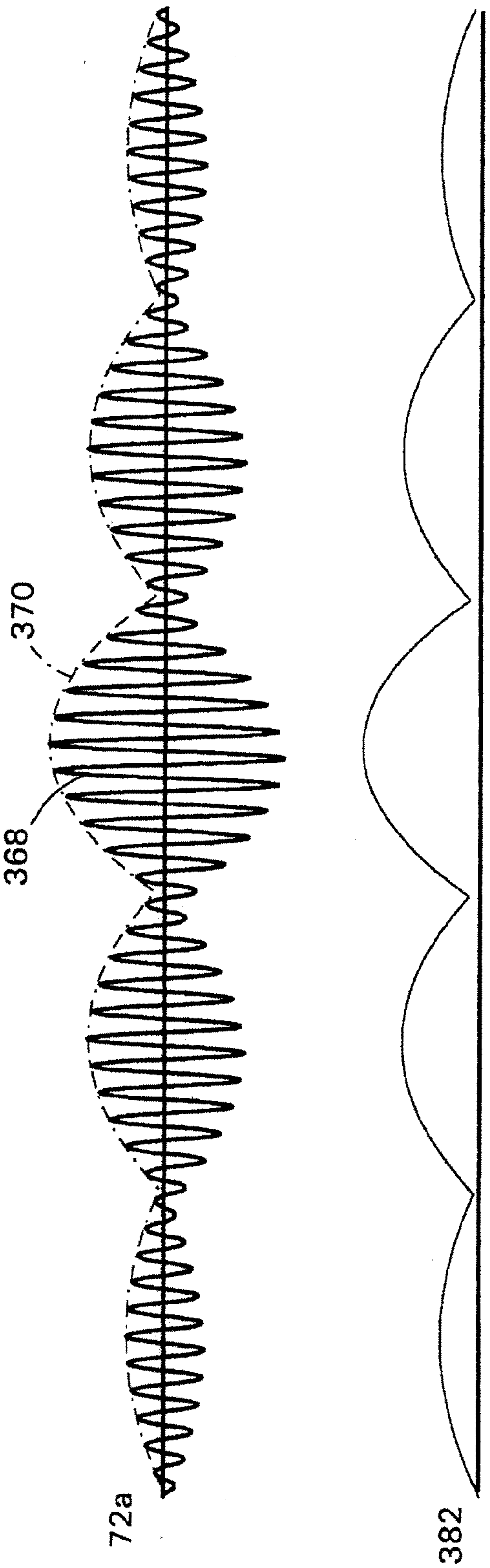


Fig. 18

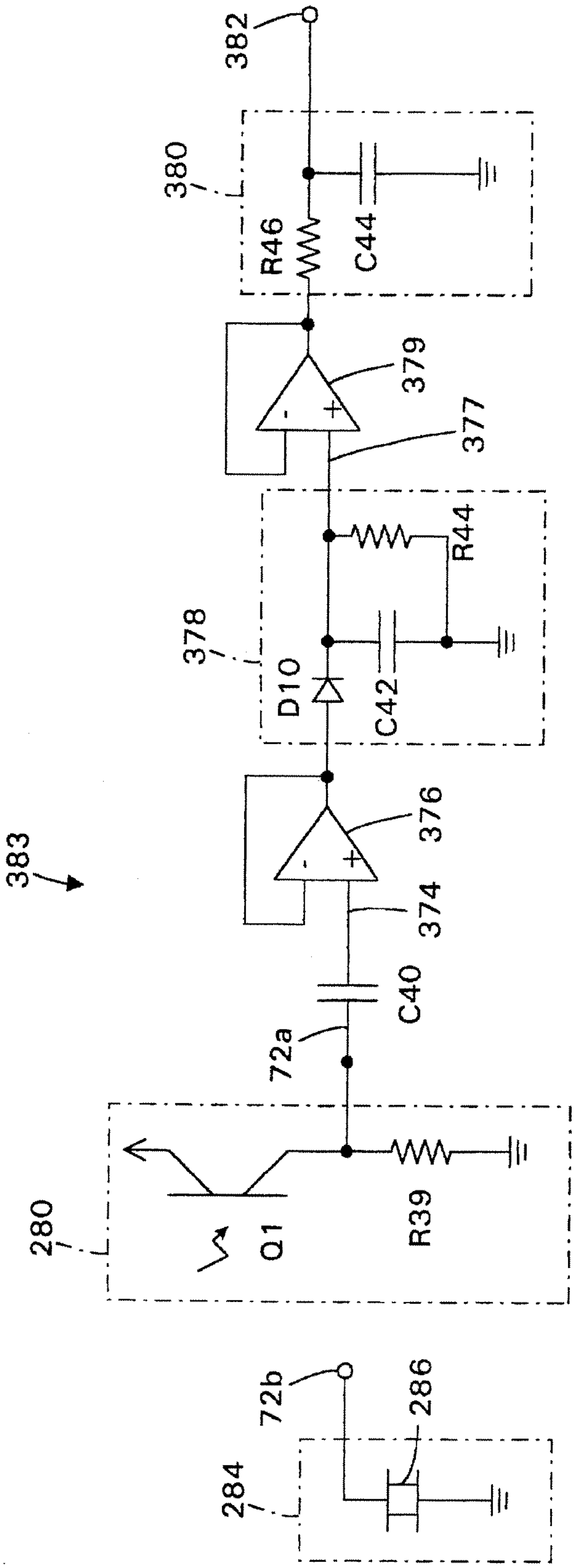

382
380
R46
C44
379
377
R44
378
D10
C42
376
383
374
C40
72a
280
Q1
R39
72b
286
284

Fig. 19

ELECTRONIC MUSICAL RE-PERFORMANCE AND EDITING SYSTEM

BACKGROUND

1. Field of the Invention

The present invention relates generally to an electronic musical performance system that simplifies the playing of music, and more particularly, to methods and systems for using traditional music gestures to control the playing of music.

2. Description of the Prior Art

TRADITIONAL MUSICAL INSTRUMENTS

Musical instruments have traditionally been difficult to play. To play an instrument a student must simultaneously control pitch, timbre (sound quality), and rhythm. To play in an ensemble, the student must also keep in time with the other musicians. Some instruments, such as the violin, require a considerable investment of time to develop enough mechanical skill and technique to produce a single note of acceptable timbre. Typically a music student will start with simple, often uninspiring, music.

Once a musician becomes proficient at playing a sequence of notes in proper pitch, timbre, and rhythm, the musician can start to develop the skills of expression. Slight variations in the timing of notes, called rubato, and the large scale speeding and slowing called tempo are both temporal methods of bringing life to a musical score. Variations of volume and timbre also contribute to the expression of a musical piece. Musical expression distinguishes a technically accurate, yet dry, rendition of a piece of music from an exciting and moving interpretation. In both instances the correct sequence of notes as specified in a musical score are played, but in the latter, the musician, through manipulation of timing and timbre, has brought out the expressive meaning of the piece which is not fully defined in the score.

For those people who want to experience the pleasures of playing a musical instrument but do not have the necessary training, technique, and skills, they must postpone their enjoyment and endure arduous practice and music lessons. The same applies for those who want to play with others but are not proficient enough to play the correct note at the correct volume, time, and timbre, fast enough to keep up with the others. Many beginning music students abandon their study of music along the way when faced with the frustration and demands of learning to play a musical instrument.

ELECTRONIC MUSIC CONTROLLERS

The introduction of electronic music technology, however, has made a significant impact on students participation in music. A music synthesizer, such as the Proteus from E-mu Systems of Santa Cruz, Calif., allows a novice keyboard player to control a variety of instrument sounds, including flute, trumpet, violin, and saxophone. With the standardization of an electrical interface protocol, Musical Instrument Digital Interface (MIDI), it is now possible to connect a variety of controllers to a synthesizer.

A controller is a device that sends commands to a music synthesizer, instructing the synthesizer to generate sounds. A wide variety of commercially available controllers exist and can be categorized as traditional and alternative. Traditional controllers are typically musical instruments that have been instrumented to convert the pitch of the instrument into MIDI commands. Examples of traditional controllers include the violin, cello, and guitar controllers by Zeta Systems (Oakland, Calif.); Softwind's Synthaphone saxophone controller; the stringless fingerboard synthesizer controller, U.S. Pat. No. 5,140,887, dated Aug. 25, 1992, issued to Emmett Chapman; the digital high speed guitar synthesizer, U.S. Pat. No. 4,336,734, dated Jun. 29, 1982, issued to Robert D. Polson; and the electronic musical instrument with quantized resistance strings, U.S. Pat. No. 4,953,439, dated Sep. 4, 1990, issued to Harold R. Newell.

A technology which is an integral part of many traditional controllers is a pitch tracker, a device which extracts the fundamental pitch of a sound. IVL Technologies of Victoria, Canada manufactures a variety of pitch-to-MIDI interfaces, including The Pitchrider 4000 for wind and brass instruments; Pitchrider 7000 for guitars; and Steelrider, for steel guitars.

Some traditional controllers are fully electronic, do not produce any natural acoustic sound, and must be played with a music synthesizer. They typically are a collection of sensors in an assembly designed to look and play like the instrument they model. Commercial examples of the non-acoustic traditional controllers which emulate wind instruments include Casio's DH-100 Digital Saxophone controller, Yamaha's WXll and Windjamm'r wind instrument controller, and Akai's WE1000 wind controller. These controllers sense the closing of switches to determine the pitch intended by the player.

Alternative controllers are sensors in a system that typically control music in an unconventional way. One of the earliest, pre-MIDI, examples is the Theremin controller where a person controlled the pitch and amplitude of a tone by the proximity of their hands to two antenna. Some examples of alternative controllers include Thunder (trademark), a series of pressure pads controlled by touch, and Lightening (trademark), a system in which you wiggle an infrared light in front of sensors, both developed and Sold by Don Buchla and Associates (Berkeley, Calif.); Videoharp, a controller that optically tracks fingertips, by Dean Rubine and Paul McAvinney of Carnegie-Mellon University; Biomuse, a controller that senses and processes brain waves and muscle activity (electromyogram), by R. Benjamin Knapp of San Jose State University and Hugh S. Lusted of Stanford University; Radio Drum, a three dimensional baton and gesture sensor, U.S. Pat. No. 4,980,519, dated Dec. 25, 1990, issued to Max V. Mathews; and a music tone control apparatus which measures finger bending, U.S. Pat. No. 5,125,313, dated Jun. 30, 1992, issued to Teruo Hiyoshi, et al.

The traditional controllers enable a musician skilled on one instrument to play another. For example, a saxophonist using Softwind's Synthaphone saxophone controller can control a synthesizer set to play the timbre of a flute. Cross-playing becomes difficult when the playing technique of the controller does not convert well to the timbre to be played. For example a saxophonist trying to control a piano timbre will have difficulty playing a chord since a saxophone is inherently monophonic. A more subtle difference is a saxophonist trying to control a violin. How does the saxophonist convey different bowing techniques such as reversal of bow direction (detache and legato), the application of significant bow pressure before bow movement (martele, marcato, and staccato), and dropped, lifted or ricocheted strokes of the bow (pique, spiccato, jete and flying staccato). Conventional violin controllers do not make sufficient measurements of bow contact, pressure, and velocity to respond to these bowing techniques. To do so would encumber the playablity of the instrument or affect its ability to produce a good quality acoustic signal. However, these bow gestures have an important effect on the timbre of sound and are used to convey expression to music.

Tod Machover and his students at M.I.T. have been extending the playing technique of traditional musical instruments by applying sensors to acoustic instruments and connecting them to computers (Machover, T., "Hyperinstrument, A Progress Report 1997–1991", MIT Media Laboratory, January 1992). These extended instruments, called hyperinstruments, allow a trained musicians to experiment with new ways of manipulating synthesized sound. Once such instrument, the Hyperlead Guitars, the timbre of a sequence of notes played by a synthesizer is controlled by the position of the guitarist's hand on the fret board. In another implementation, the notes of guitar chords automatically selected from a score stored inside a computer, are assigned to the strings of a guitar. Picking a string triggers the note assigned to the string, with a timbre determined by fret position. Neither of these implementations allows traditional guitar playing technique where notes are triggered by either hand.

EASY-TO-PLAY MUSICAL ACOUSTIC INSTRUMENTS

Musical instruments have been developed that simplify the production of sound by limiting the pitches that can be produced. The autoharp is a harp with racks of dampers that selectively mute strings of un-desired pitch, typically those not belonging to a particular chord. A harmonica is a series of vibrating reeds of selected pitches. Toy xylophones and piano exists that only have the pitches of a major scale.

VOICE CONTROLLED SYNTHESIZER

Marcian Hoff in U.S. Pat. NO. 4,771,671, dated Sep. 20, 1988, discloses an electronic music instrument that controls the pitch of a music synthesizer with the pitch of a human voice, later manufactured as the Vocalizer by Breakaway Systems (San Mateo, Calif.). The Vocalizer limits pitches to selected ones, similar to an autoharp. The Vocalizer includes a musical accompaniment which dynamically determines which pitches are allowed. If the singer produces a pitch that is not allowed, the device selects and plays the closest allowable pitch.

The difficulty in adopting Hoff's method to play a musical melody is that a vocalized pitch must be produced for each note played. Fast passages of music would require considerable skill of the singer to produce distinct and recognizable pitches. Such passages would also make great demands of the system to distinguish the beginning and ending of note utterances. The system has the same control problems as a saxophone controller mentioned above: singing technique does not convert well to controlling other instruments. For example, how does one strum a guitar or distinguish between bowing and plucking a violin with a voice controller?

ACCOMPANIMENT SYSTEMS

Accompaniment systems exist that allow a musician to sing or play along with a pre-recorded accompaniment. For the vocalist, karaoke is the use of a predefined, usually prerecorded, musical background to supply contextual music around which a person sings a lead part. Karaoke provides an enjoyable way to learn singing technique and is a form of entertainment. For the instrumentalist, a similar concept of "music-minus-one" exists, where, typically, the lead part of a musical orchestration is absent. Hundreds of classical and popular music titles exist for both karaoke and music-minus-one. Both concepts require the user to produce the correct sequence of notes, with either their voice or their instrument, to play the melody.

Musical accompaniment also exists on electronic keyboards and organs, from manufacturers such as Wurlitzer, Baldwin, Casio, and Yamaha, which allow a beginner to play a simple melody with an automatic accompaniment, complete with bass, drums, and chord changes.

A more sophisticated accompaniment method has been designed independently by Barry Vercoe (Vercoe, B., Puckette, M., "Synthetic Rehearsal: Training the Synthetic Performer", ICMC 1985 Proceedings, pages 275–278; Boulanger, R., "Conducting the MIDI Orchestra, Part 1", Computer Music Journal, Vol. 14, No. 2, Summer 1990, pages 39–42) and Roger Dannenberg (ibid., pages 42–46). Unlike previous accompaniment schemes where the musician follows the tempo of the accompaniment, they use the computer accompaniment to follow the tempo of the live musician by monitoring the notes played by the musician and comparing it to a score stored in memory. In Vercoe's system a flute and a violin were used as the melody instruments. In Dannenberg's system a trumpet was used.

In all of the cases of accompaniment mentioned, the person who plays the melody must still be a musician, having enough skill and technique to produce the proper sequence of pitches at the correct times and, where the instrument allows, with acceptable timbre, volume, and other expressive qualities.

SYSTEMS WITH STORED MELODY

In order to reduce the simultaneous tasks a person playing music must perform, a music re-performance system can store a sequence of pitches, and through the action of the player, output these pitches. A toy musical instrument is described in U.S. Pat. No. 4,981,457, by Taichi Iimura et al, where the closing of a switch by a moveable part of the toy musical instrument is used to play the next note of a song stored in memory. Shaped like a violin or a slide trombone, the musical toy is an attempt to give the feeling of playing the instrument the toy imitates. The switch is closed by moving a bow across the bridge, for the violin, or sliding a slide tube, for the trombone. The length of each note is determined by the length of time the switch is closed, and the interval between notes is determined by the interval between switch closing. No other information is communicated from the controller to the music synthesizer.

The toy's limited controller sensor, a single switch makes playing fast notes difficult, limiting expression to note timing, and does not accommodate any violin playing technique that depends on bow placement, pressure, or velocity, and finger placement and pressure. Similarly the toy does not accommodate any trombone playing techniques that depends on slide placement, lip tension, or air pressure. The limited capability of the toy presents a fixed level of complexity to the player which, once surpassed, renders the toy boring.

The melody for a song stored in the toy's memory has no timing information, making it impossible for the toy to play the song itself, to provide guidance for the student, and does not contain any means to provide any synchronized accompaniment. The toy plays monophonic music while a violin, having four strings, polyphonic. The toy has no way to deal with a melody that starts a note before finishing the last, or ornamentations a player might add to a re-performance, such as playing a single long note as a series of shorter notes.

Another system that simplifies the tasks of the person playing music is presented by Max Mathews in his Conductor Program (Mathews, M. and Pierce, J., editors, "The Conductor Program and Mechanical Baton", Current Directions in Computer Music Research, The MIT Press, 1989, Chapter 19; Boulanger, R., "Conducting the MIDI Orchestra, Part 1", Computer Music Journal, Vol. 14, No. 2, Summer 1990, page 34–39). In Mathews' system a person conducts a score, which is stored in computer memory, using special batons, referred to earlier as the alternative controller Radio Drum.

Mathews' system is basically a musical sequencer with synchronization markers distributed through the score. The sequencer plays the notes of the score at the times specified, while monitoring the actions of the batons. If the sequencer reaches a synchronization marker before a baton gesture, the sequencer stops the music and waits for a gesture. If the baton gesture comes in advance of the marker, the sequencer jumps ahead to the next synchronization marker, dropping the notes in between. The system does not tolerate any lapses of attention by the performer. An extra beat can eliminate a multitude of notes. A missed beat will stop the re-performance.

Expressive controls of timbre, volume, pitch bend are controlled by a combination of spatial positions of the batons, joystick and knobs. Designed primarily as a control device for the tempo and synchronization of an accompaniment score, there are no provisions for controlling the relative timing of musical voices in the score. The controller is a cross between a conductor's baton and a drum mallet and does not use the gestures and playing techniques of the instruments being played. There is no way for several people to take part in the re-performance of music. Mathews' conductor system is a solo effort with no means to include any other players.

None of the systems and techniques presented that are accessible to non-musicians provides an adequate visceral and expressive playing experience of the instrument sounds they control. The natural gestural language people learn and expect from watching instruments being played are not sufficiently utilized, accommodated, or exploited in any of these system.

MIDI SEQUENCERS

With the advent of standardization of the electronic music interface, MIDI, many software application programs called sequencers became available to record, store, manipulate, and playback music. Commercial examples include Cakewalk by Twelve Tone Systems and Vision by Opcode Systems. One manipulation technique common to most sequencers is the ability to change the time and duration of notes. One such method is described in U.S. Pat. No. 4,969,384, by Shingo Kawasaki, et al., where the duration of individual sections of music can be shortened or lengthened.

Music can be input into sequencers by typing in notes and durations, drawing them in using a mouse pointing device, or more commonly, using the sequencer as a tape recorder and "playing live". For those not proficient at playing keyboard it is often difficult to play the correct sequence of notes at the correct time, with the correct volume. It is possible to "play in" the correct notes without regard for time and edit the time information later. This can be quite tedious as note timing is edited "off line", that is non-real time, yet music is only perceived while it is being played. Typically this involves repeatedly playing and editing the music in small sections, making adjustments to the location and duration of notes. Usually the end result is stilted for it is difficult to "edit-in" the feel of a piece of music.

It is therefore desirable to have a music editing system where selected music parameters (e.g. volume, note timing, timbre) can be altered by a musician re-playing the piece. Such a system, called a music re-performance system, would allow a musician to focus on the selected parameters being edited.

SUMMARY DESCRIPTION OF THE INVENTION

An object of the invention is to provide a musical re-performance system to allow a person with a minimum level

of skill to have a first-hand experience of playing a musical instrument using familiar playing techniques. The music re-performance system is easy enough to operate that a beginner with little musical skill can play a wide variety of musical material, with recognizable and good sounding results. The system can tolerate errors and attention lapses by the player. As the student gains more experience, the system can be adjusted to give the student greater control over note timing and expression.

To accomplish these goals the music re-performance system provides an instrument controller that is played using traditional playing techniques (gestures), a scheduler that plays preprogrammed notes in response to gestures from the controller, and an accompaniment sequencer that synchronizes to the tempo of the player. The scheduler maintains a tolerance for gesture timing error to handle missed and extra gestures. Expressive parameters including volume, timbre, and vibrato and can be selectively controlled by the score, the player's gestures, or a combination of the two. The system takes care of the note pitch sequence and sound generation, allowing the player to concentrate on the expressive aspects of music.

The similarity between the playing technique of the controller and the traditional instrument allows experiences learned on one to carry over to the other, providing a fun entry into music playing and study. A beginner can select a familiar piece of music and receive the instant gratification of playing and hearing good sounding music. As the player gains skill, more difficult music can be chosen and greater control can be commanded by the player, allows the system to track the development of the player. Music instruction, guidance and feedback are given visually, acoustically, and kinesthetically, providing a rich learning environment.

Another object of the invention is to allow a plurality of people with a minimum level of skill to have the first-hand experience of playing in an ensemble, from a string quartet to a rock-and-roll band. The music re-performance system can take over any of the players parts to assist with difficult passages or fill in for an absent musician. A video terminal displays multi-part scores, showing the current location of each player in the score. The system can accommodate any number of instrument controller, monophonic or polyphonic, conventional MIDI controllers or custom, and accept scores in standard MIDI file format.

To accomplish these goals a scheduler is assigned to each controller. If a controller is polyphonic, like a guitar, a scheduler containing multiple scheduler, one for each voice (e.g. six for a guitar) is assigned. To play a part automatically, the scheduler for that part is set with zero tolerance for gesture error. The scheduler can automatically analyze a score and determine when a sequence of notes should be played with one gesture, making fast passages easier to play. The system can accommodate accompaniment that is live, recorded audio, or stored in memory.

Another object of the invention is to provide controllers that play like traditional instruments, provide greater control and are less expensive to manufacture then MIDI controllers, and are interchangeable in the system. To accomplish these goals traditional instruments are modeled as having two components; an energy source that drives the sound and finger manipulation that changes the pitch of the instrument. Transducers appropriate to each instrument are used to convert these components into electric signals which are processed into standardized gesture outputs. The common model and standardized gestures allow the system to accommodate a variety of instruments. Wind controllers have been developed, particularly the Casio DH-100 Digital Saxophone, that can easily be adapted to the music re-performance system.

Commercially available string controllers, including guitars and violin, suffer from one or more of the following problems:

They are to difficult for non-musicians to play.

They do not allow enough expressive control of the music.

They hinder the development of skill and technique.

They do not use traditional playing techniques.

They are expensive.

Another object of the invention is to address these problems by making expressive, responsive, and inexpensive string controllers that use traditional playing techniques, with a music performance system that is easy to use and can be adjusted to match the skill level of the player.

Another object of the invention is to be able to edit selected parameters of a score (e.g. timing, volume, brightness) by playing those parameters live, without having to worry about the accuracy of the unselected parameters. Such editing can give life and human feel to a musical piece that was, for example, transcribed from a written score. To accomplish this only the parameters selected to be edited (e.g. note volume) are updated when playing the controller, leaving all other parameters unchanged.

These and other advantages and features of the invention will become readily apparent to those :skilled in the art after reading the following detailed description of the invention and studying the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a schematic block diagram of an embodiment of the music re-performance system for four instruments with accompaniment;

FIG. **2** is a block diagram of the system component of the embodiment of FIG. **1**;

FIG. **3** is a detail block diagram of a portion of the embodiment of FIG. **1**, showing the components of the controller, scheduler, and accompaniment sequencer;

FIG. **4** illustrates by means of icons and timing information the operation of the temporal masking processor shown in the controller of FIG. **3**;

FIG. **5**A pictorially illustrates the operation of the scheduler shown in FIG. **3**;

FIG. **5**B shows a detail of FIG. **5**A to illustrate the operation of the simultaneous margin processor;

FIG. **6**A and **6**B illustrates by means of a flow chart the operation of the scheduler;

FIG. **7** is a schematic block diagram of an embodiment of a polygestural scheduler capable of processing a plurality of simultaneous input gestures;

FIG. **8** is a perspective view of an embodiment of a string controller preferred for bowing;

FIG. **9**A is a perspective view of an energy transducer preferred for bowing used in the string controller shown in FIG. **8**;

FIG. **9**B is a side view of the energy transducer of FIG. **9**A;

FIG. **10**A is a perspective view of an alternate embodiment of a string controller using an optical interrupter to measure string vibrations;

FIG. **10**B is a side view of a detail of FIG. **10**A, showing the optical aperture of the optointerrupter partially eclipsed by a string;

FIG. **11** is a perspective view of an alternate embodiment of a string energy transducer using a piezo-ceramic element to measure string vibrations;

FIG. **12** is a perspective view of an alternate embodiment of a string energy transducer using a tachometer to measure bow velocity;

FIG. **13** is a schematic of an embodiment of controller electronics using the preferred energy and finger transducers illustrated in FIG. **8**.

FIG. **14** illustrates with wave forms and timing diagrams the signal processing for the preferred finger transducer of FIG. **8**;

FIG. **15** is a schematic of an embodiment of an electronic circuit to perform signal processing for the preferred finger transducer of FIG. **8**;

FIG. **16** illustrates by means of wave forms and timing diagrams signal processing for the tachometer to convert bow velocity to bow gestures and bow energy;

FIG. **17** is a schematic of an embodiment of an electronic circuit to perform signal processing for the tachometer to convert bow velocity to bow gestures and bow energy;

FIG. **18** illustrates by means of wave forms and timing diagrams signal processing for the optical interrupter and piezo-ceramic element, to convert string vibrations into energy magnitude and energy gestures;

FIG. **19** is a schematic of an embodiment of an electronic circuit perform signal processing of the optical interrupter and piezo-ceramic element, to convert string vibrations into an energy envelope;

DESCRIPTION OF THE INVENTION AND PREFERRED EMBODIMENT

OVERVIEW

FIG. **1** Shows an embodiment of the music re-performance system **2** to allow four people to play a musical piece stored in a score memory **4**. Each person plays a musical controller **6**,**8**,**10**,**12** which is shaped like a traditional musical instrument. The quartet of musical controllers **6**,**8**,**10**,**12** assembled in FIG. **1** are a violin controller **6**, cello controller **8**, flute controller **10**, and guitar controller **12**. These controllers can be conventional MIDI instrument controllers, which are available for most traditional instruments, or ones embodied in the invention that will be discussed later.

In order to describe the operation of the music re-performance system **2**, the concept of gestures and expression commands are introduced. When a person plays a musical instrument their actions (e.g. strumming, bowing, and fretting a string) are converted by the instrument into; acoustic sound. Some actions start, stop, and change the pitch of the sound (e.g. fretting and picking strings), others change the loudness and timbre of the sound (e.g. changing bowing pressure). For the purposes of the music re-performance system **2**, the former actions are called player gestures, the latter actions are called expression commands.

There are three types of player gestures: START, STOP, and STOP-START. The player gestures describe the action they produce. A START starts one or more notes, a STOP stops all the notes that are on (i.e. sounding), and a STOP-START stops one or more notes and starts one or more notes. From silence (all notes off), the only possible player gesture is START. When at least one note is on, a STOP-START or

STOP player gesture is possible. After a STOP player gesture, only a START is possible.

When conventional MIDI controllers are used in the music reperformance system **2**, a START corresponds to the MIDI commands NOTE ON, a STOP corresponds to NOTE OFF, and a STOP-START corresponds to a NOTE OFF immediately followed by a NOTE ON. Expression commands include the MIDI commands PROGRAM CHANGE, PITCH BEND, and CONTROLLER COMMANDS.

Each controller, **6**,**8**,**10**,**12**; transmits gesture and expression commands of a player (not shown) to the computer **14** though a MIDI interface unit **16**. The computer **14** receives the gesture and expression commands, fetches the appropriate notes from the musical score **4**, and sends the notes with the expression commands to a musical synthesizer **18**, whose audio output is amplified **20** and played out loudspeakers **22**.

The MIDI interface unit **16** provides a means for the computer **14** to communicate with MIDI devices. MIDI is preferred as a communication protocol since it is the most common musical interface standard. Other communication methods include the RS-232 serial protocol, by wire, fiber, or phone lines (using a modem), SCSI, IEEE-488, and Centronics parallel interface.

The music score **4** contains note events which specify the pitch and timing information for every note of the entire piece of music, for each player, and may also include an accompaniment. In a preferred embodiment, score data **4** is stored in the Standard MIDI File Format 1 as described in the MIDI File Specification 1.0. In addition to pitch and timing information, the score may include expressive information such as loudness, brightness, vibrato, and system commands and parameters that will be described later. In a preferred embodiment system commands are stored in the MIDI file format as CONTROLLER COMMANDS.

Examples of the computer **14** in FIG. **1** include any personal computer, for example an IBM compatible personal computer, or an Apple Macintosh computer.

The media used store the musical score data **4** can be read-only-memory ROM circuits, or related circuits, such as EPROMs, EEROMs, and PROMs; optical storage media, such as videodisks, compact discs CD ROM's, CD-I discs, or film; bar-code on paper or other hard media; or magnetic media such as floppy disks of any size, hard disks, magnetic tape; audio tape cassette or otherwise; or any other media which can store score data or music, or any combination of media above. The medium or media can be local, for example resident in the embodiment of the music reperformance system **2**, or remote, for example separately housed from the embodiment of the music re-performance system **2**.

A video display **24** connected to the computer **14** displays a preferred visual representation **26** of the score in traditional music notation. As each player gestures a note **27**, the gestured note **27** changes color, indicating the location of the player in the score. An alternative representation of the score is a horizontal piano scroll (not shown) where the vertical position of line represent pitch, and the length of the lines represents sustain time.

Many music synthesizers **18** exist that would be suitable including the PROTEUS from E-UM Systems, Sound Canvas from Roland, and TX-81Z from Yamaha.

The media which is used to store the accompaniment include any of the score storage media discusses above or can be live or prerecorded audio, on optical storage media such as videodisks, compact discs CD ROM's, CD-I discs, or film; magnetic media such as floppy disks of any size, hard disks, magnetic tape; audio tape cassette or otherwise; phonograph records; or any other media which can store digital or analog audio, or any combination of media above. The medium or media can be local or remote.

FIG. **2** shows a block diagram of the music re-performance system **2**. The following discussion of the operation of the controller **6** and scheduler **28** applies to controllers **8**,**10**,**12** and schedulers **30**,**32**,**34** as well. The scheduler **28** collects note events from the score **4** that occur close together in time, groups them as pending events, and determines what type of player gesture is required by the group. For example, the first NOTE ON event of a piece is a pending event requiring a START player gesture, two events that happen close together that stop a note and start another form a pending events group requiring a STOP-START player gesture, and an event that stops all the notes currently on requires a STOP player gesture.

The controller **6** sends player gestures **36** to the scheduler **28**. The scheduler **28** matches player gestures **36** to the gestures required by the pending events, and sends the matched events as note output commands **38** to the music synthesizer **18**. When all the pending events are successively matched up and sent out, the scheduler **28** selects the next collection of pending events. The scheduler **28** calculates tempo changes by comparing the time of the player gestures **36** with the times of the note events as specified in the score **4**. These tempo change calculations are sent as tempo change commands **40** to the accompaniment sequencer **42**.

The controller **6** also sends expression commands **44** directly to the music synthesizer **18**. These expression commands **44** include volume, brightness, and vibrato commands which change corresponding parameters of the synthesizer **18**. For example if the controller **6** is a violin, bowing harder or faster might send a volume expression command **44** telling the music synthesizer **18** play the notes louder.

The accompaniment sequencer **42** is based on a sequencer, a common software program, which reads the score **4** and sends note and expression commands **46** the music synthesizer **18**, at the times specified by the score **4**, and modified to work at a tempo specified by one of the schedulers **28**, **30**, **32**, **34**.

Examples where the accompaniment sequencer **42** may not be required include an ensemble where all the parts are played by controllers **6**, when the accompaniment is provided by an audio source, or when the accompaniment is live musicians. In one embodiment of the music re-performance system **2**, a solo player using one controller **6** plays the lead part of a piece of music, accompanied by a "music-minus-one" audio recording.

The video generator **47** displays the current page of the music score **4** on the video display **24**, and indicates the location of the accompaniment sequencer **42** and all the controllers **6**, **8**, **10**, **12** in the musical score **4**, by monitoring the note output commands **38** of the controllers **6**, **8**, **10**, **12** and accompaniment sequencer **42**, sent on the note data bus **48**. Methods to display the score **4** and update the locations of the controllers **6**, **8**, **10**, **12** and accompaniment sequencer **42** in the score **4**, are well known to designers of commercial sequencer programs like Cakewalk from Twelve Tone Systems and will not be reviewed here.

FIG. **3** shows a detailed block diagram of the three main components of the music re-performance system: the controller **6**, scheduler **28**, and accompaniment sequencer **42**. Each of these components will be examined. If the controller **6** is a conventional MIDI instrument controller, the functional blocks inside the controller **6** are performed by the MIDI controller. A MIDI instrument controller serving as the controller **6** will be considered first.

CONTROLLER **6**

The MIDI output from the controller **6** is separated into two streams; player gestures **36** and expression commands **44**. The expression commands **44** are passed from the controller **6** directly the music synthesizer **18** and control the expression (e.g. volume, brightness, vibrato) of the instrument sound assigned to the controller **6**.

An alternative to using a MIDI controller is provided by the invention. Since the pitch is determined by the score 4 and not the controller **6**, the invention offered the opportunity to design controllers that are less expensive and easier play than conventional MIDI controllers. One skilled in the art of instrument design and instrumentation need only construct a controller **6** that provides player gestures **36** and expression commands **44** to the invention to play music. The blocks inside the controller **6** illustrate a preferred means of designing a controller **6** for the invention.

The controller **6** for any music instrument is modeled as a finger transducer **58** and an energy transducer **60**. Table 1 classifies common musical instruments into four categories. Table 2 lists the measurable phenomena for the energy transducer **60** of each instrument class. Table 3 lists the measurable phenomena for the finger transducers **58** of each instrument class.

TABLE 1

INSTRUMENT CLASSIFICATION

Class	Examples
bowed strings	violin, viola, cello, bass
picked strings	guitar, bass, banjo, ukulele
blown	recorder, clarinet, oboe, flute, piccolo, trumpet, French horn, tuba
blown slide valve	trombone

TABLE 2

ENERGY MEASUREMENT PARAMETERS

Class	Phenomena
bowed string	bow position, velocity, pressure
picked string	string vibration amplitude
blown	air pressure, velocity
blown slide valve	air pressure, velocity

TABLE 3

FINGER MEASUREMENT PARAMETERS

Class	Phenomena
bowed string	string contact position and pressure
picked string	string contact position and pressure
blown	switch closure and pressure
blown slide valve	valve position

The music instrument model is general enough to include all the instruments listed in Table 1. Many sensors exist to measure the phenomena listed in Table 2 and Table 3. To design a controller **6** for a particular instrument sensors are selected to measure the energy and finger phenomena particular to the instrument, preferably utilizing traditional playing techniques. Signal processing is chosen to generate gestures and expression from these phenomena. Gestures are :intentional actions done by the player on their instrument to start and end notes. Expression are intentional actions done by the player on their instrument to change the volume and timbre of the sound they are controlling.

FINGER TRANSDUCER **58**

Referring to FIG. **3**, the finger transducer **58** senses finger manipulation of the controller **6** and produces a finger manipulation signal **62** responsive to finger manipulation. The finger signal processing **64** converts the finger manipulation signal **62** into a binary finger state **68**, indicating the application and removal of a finger (or sliding of a valve for a trombone) and a continuous finger pressure **70**, indicating the pressure of one or more fingers on the finger transducer **58**.

ENERGY TRANSDUCER **60**

The energy transducer **60** senses the application of energy to the controller **6** and converts the applied energy to an energy signal **72**. The energy signal processing **74** converts the energy signal **72** into a binary energy state **76**, indicating energy is being applied to the controller **6**, and into a continuous energy magnitude **78**, indicating the amount of energy applied to the controller **6**.

TEMPORAL MASKING PROCESSOR **80**

In a musical instrument, notes can be started, stopped, and changed by the energy source (e.g. bowing a string or blowing a flute), and changed by finger manipulation (e.g. fretting a string or pushing or releasing a valve on a flute). In the controller model **6**, these actions correspond to energy gestures **82** (not shown) and finger gestures **96** (not shown), respectively. In a traditional instrument when these gestures are done close together in time (substantially simultaneous), the acoustic and mechanical properties of the instrument produces a graceful result. In an electronic system capable of high speed responses, an energy gesture **82** and finger gesture **96** intended by the player to be simultaneous, will more likely be interpreted as two distinct gestures, producing unexpected results. The temporal masking processor **80** is designed to combine the two gestures into the single response expected by the player.

In the embodiment of the music re-performance system **2** shown in FIG. **1**, the implementation of the scheduler **28**, accompaniment sequencer **42**, and the task of separating the player gestures **36** from the expression commands **44** from the MIDI controller **6**, is performed in software in the computer **14**. The MIDI interface unit **16** is not shown explicitly in FIG. **3** but provides for the communication of player gestures **36**, expression commands **44**, and note output commands **38** to the computer **14** and music synthesizer **18**.

FIG. **4** shows a pictorial timing diagram of gestures applied to and output from the temporal masking processor **80**. The energy state **76** is a binary level applied to the temporal masking processor **80** that is high only when energy is being applied to the controller **6** (e.g. blowing or bowing). The temporal masking processor **80** internally generates an energy gesture **82** in response to changes in the energy state **76**. A rising edge **84** of the energy state **76** produces a START energy gesture **86** (represented by an arrow pointing up), a falling edge **88** produces a STOP energy gesture **90** (arrow pointing down), and a falling edge followed by a rising edge. **92**, within a margin of time, produces a STOP-START energy gesture **94** (two headed arrow). The margin of time can be fixed, variable, a fraction of a note duration, or based on the tempo of the song. In a preferred embodiment, the margin of time is fixed (e.g. 50 milliseconds).

The finger state **68** is a binary level applied to the temporal masking processor **80** that is pulsed high when a finger is lifted or applied, or in the case of a trombone, the slide valve is moved in or out an appreciable amount. The temporal masking processor **80** internally generates a finger gesture **96** on the rising edge **100** of the finger state **68**, if and only if the energy state **76** is high. There is only one type of finger gesture **96**, the STOP-START **98**, represented by a two-headed arrow.

There are six possible energy gesture **82** and finger gesture **96** sequence combinations, as shown in FIG. **4**. When the energy state **76** changes, the player gesture **36** of the temporal masking processor **80** is the corresponding energy gesture **82**, as in the case of **102**, **104**, and **106**. If finger gestures **96** occur within the masking time **108** they are ignored. The masking time **108** can be fixed, variable, a fraction of a note duration, or based on the tempo of the song. In a preferred embodiment the masking time **108** is a fraction of the duration, of the next note to be played by the scheduler **28**. In this way, short quick notes produce small masking times **108**, allowing many energy gesture **82** and finger gestures **96** to pass through as player gestures **36**, while slow long notes are not accidentally stopped or started by multiple gestures intended as one.

When the temporal masking processor **80** detects a rising edge **100** of the finger state. **68**, the corresponding player gesture **36** is player gesture **36**, as in case **112**, and **114**. If an energy gesture **82** occurs within the masking time **108** it is ignored unless it is a STOP energy gesture **82**, as in case **116**, in which case the temporal masking processor **80** outputs an UNDO command **118** (represented as X). Upon receiving the UNDO **118** command, the scheduler **28** stops all the notes currently on (as is always done by a STOP gesture), and "takes-back" the erroneous STOP-START gesture **114**. Typically in a software implementation of a scheduler **28**, this means moving internal pointers of the scheduler **28** back to the notes started by the erroneous STOPSTART gesture **114**, preparing to start them again on the next START gesture.

EXPRESSION PROCESSOR 120

Referring back to the block diagram of the controller **6** shown in FIG. **3**, the expression processor **120** receives the continuous energy magnitude **78** and the continuous finger pressure **70**, and produces expression commands **44** which are sent to the music synthesizer **18** to effect the volume and timbre of the sound assigned to the controller **6**. In a preferred embodiment, the expression processor **120** outputs vibrato depth expression commands **44** in proportion to finger pressure fluctuations **70**, and outputs volume expression commands **44** in proportion to energy magnitude **78**.

SCHEDULER 28

The scheduler **28** receives the finger gestures **96** from the controller **6**, consults the score **4**, sends tempo change commands **40** to the accompaniment sequencer **42**, and note output commands **38** to the music synthesizer **18**. These tasks are performed by three processors: the simultaneous margin processor **122**, the pending notes processor **124**, and the rubato processor **126**.

The simultaneous margin processor **122** fetches note events from the score **4** and sends them to the pending notes processor **124**, where they are stored as pending note events. The pending notes processor **124** receives player gestures **36** from the controller **6**, checks them against the pending note events, and sends note output commands **38** to the music synthesizer **18**. The rubato processor **126** calculates tempo changes by comparing the timing of player gestures **36** to pending note events, and sends tempo change commands **40** to the accompaniment sequencer **42**.

FIG. **5**A is a pictorial timing diagram showing the operation of the scheduler **28**.

SIMULTANEOUS MARGIN PROCESSOR 122

Scored notes **128** are stored in the score **4** in chronological order. Each scored note **128** is stored as two commands: a NOTE ON **130** which indicates the pitch, starting time, and volume of a note, and NOTE OFF **132** which indicates the pitch and stopping time of a note. To describe the operation of the simultaneous margin processor **122**, a section of a score containing eight notes **134**i a–**134***h*, designated for one controller **6**, is considered in FIG. **5**A. The simultaneous margin processor **122** fetches all the next note events in the score **4** that occur within a time margin, called the simultaneous margin **150**, and send them to the pending notes processor **124**, where they are referred to as pending events. In a preferred embodiment, the simultaneous margin **150** is calculated as a percentage (e.g. 10%) of the duration of the longest note in the last pending events group, and is reapplied to each note event that occurs within the simultaneous margin **150**.

The simultaneous margin **150***c* for the stop of scored note **128***c* is calculated as 10% of the duration of scored note **128***b* (the longest, and only, note duration of the last pending events). The stop of scored note **128***c* is the only event occurring inside the simultaneous margin **150***c*, so one STOP pending event **164***cc*, is contained in the pending notes processor **124**.

FIG. **5**B is a detailed view of a section of FIG. **5**A, examining how the simultaneous margin processor **122** deals with the concatenation of simultaneous margins. The simultaneous margin **150***d* for the start of scored note **128***d* is 10% of the duration of scored note **128***c*. The stop of scored note **128***d* falls within the simultaneous margin **150***d*, so the event STOP note **128***d* is also sent to the pending notes processor **124**. The start of scored note **128***e* falls within the simultaneous margin **150***d*, so the start of scored note **128***e* is sent to the pending notes processor **124**, and the simultaneous margin **150***dd* (still 10% of the duration of note **128***c*) is applied at the start of scored note **128***e*. By the same process, the stop of scored note **128***e* and the start of scored note **128***f* are sent to the pending notes processor **124**. The pending events for the collection of note events falling within the concatenated simultaneous margins **150***d*, **150***dd*, and **150***ddd* are; START note **128***d*, STOP note **128***d*, START note **128***e*, STOP note **128***e*, and START note **128***f*.

Concatenating simultaneous margins **150** can lead to an undesirable situation when a string of quick :notes (e.g. sixteenth notes) are grouped together as one pending events group. To prevent this from occurring, a limitation on concatenation may be imposed. Limitations include a fixed maximum simultaneous margin length, a relative length based on a fraction of the duration of the longest note in a simultaneous margin, or a variable length set in the score **4** or by the player. In a preferred embodiment, the maximum concatenated simultaneous margin length is a fraction of the duration of the longest note in a simultaneous margin, with the fraction determined by con, hands in the score **4**. This embodiment allows the fraction to be optimized for different sections and passages of the score **4**, for example slow passages would have large fractions, and fast section with a series of quick notes would have a smaller fraction.

In alternate embodiments, the simultaneous margin **150** may be a fixed time, for example set by the player; variable time, for example percentages of other parameters including tempo or other note durations; arbitrary times edited into the score **4** by the creator of the score; or iteratively updated, based on the errors of a player each time the score **4** is performed. In the last case, if the player misses gesturing a particular pending event, the system successively increases the simultaneous margin **150** each re-performance. Eventually the simultaneous margin **150** for the missed pending event will be large enough to incorporate the previous pending event.

PENDING NOTES PROCESSOR 124

Referring back to FIG. 3, the pending notes processor **124** matches pending events to player gestures **36** from the controller **6**, and sends note output commands **38** to the music synthesizer **18**.

Referring again to FIG. 5A, the pending notes processor **124** determines the type of gesture, called a pending event **164**, expected by the pending events. If the pending events will turn off all the notes currently on, a STOP **164***a* gesture is required. If currently there are no notes are on and the pending events will start one or more notes, a START gesture **164***b* is required. If at least one note is on and the pending events will leave at least one note on, a STOP-START **164***c* is required.

If the player gesture **36** received by the pending events processor **124** matches the pending event **164**, all the note events in the pending events processor **124** are output to the music synthesizer **18** in the order and timing specified by the score **4**, preserving the integrity of the music. This is most apparent in FIG. **5**B where note output commands **38***d*, **38***e*, and **38***f* are started with one START player gesture **36***d*, and are output in the same order and in the same relative timing as scored notes **128***d*, **128***e*, and **128***f*.

When the pending event **164** does not match the player gesture **36**, the preferred actions are a) if the player gesture **36** is a STOP, all sound .stops or b) if the player gesture **36** is a START and there is no pending NOTE ON event, the last notes on are turned on again (REATTACHED) The logic of the pending events processor **124** is summarized in Table 4.

TABLE 4

PENDING EVENTS PROCESSOR LOGIC

Case No.	Pending Events 164	Player Gesture 36	Pending Note Action
1.	STOP	STOP	STOP all notes that are on
2.	STOP	START	Not Possible
3.	STOP	STOP-START	REATTACK current notes on
4.	START	STOP	Not Possible
5.	START	START	START pending NOTE ON events
6.	START	STOP-START	Not Possible
7.	STOP-START	STOP	STOP all notes that are on
8.	STOP-START	START	START pending NOTE ON events
9.	STOP-START	STOP-START	STOP-START

In case 3, REATTACK means STOP then START all the notes that were on, without advancing to the next pending events group. Cases 2, 4, and 6 are not possible due to the principles that only a START can come after a STOP and that all the pending events in a pending events group must be processed before a new pending events group is collected and processed. Case 2 is not possible since a START player gesture **36** can only follow a STOP which would not have satisfied the previous pending gesture **164** which could only have been a START or STOP-START, since the current pending gesture **164** is a STOP. Case 4 is not possible for the previous pending gesture **164** could only have been a STOP, satisfiable only by a STOP player gesture **36**, and it is impossible to have two sequential STOP player gestures **36**. In case 6, the previous pending gesture **164** could only have been a STOP (case 3), causing a REATTACK without advancement to the next pending events group. If case 7 occurs, it will always be followed by case 8, completing the pending events in the pending events group.

RUBATO PROCESSOR 126

Referring back to the detailed block diagram of the scheduler **28** in FIG. **3**, the rubato processor **126** compares the time of the first pending note event in the pending notes processor **124** to the player gesture **36**, and sends a tempo change command **40** to the accompaniment sequencer **42**. Referring to FIG. 5A, in a preferred embodiment, the rubato processor **126** generates a time margin, called a rubato window **170**, for all START and STOP-START pending event gestures **164**. The rubato window **170** can be used to limit how much tempo change a player gesture **36** can cause, and determine when pending events in the pending notes processor **124** will be sent automatically to the music synthesizer **18**.

The rubato window **170** is centered about the time of the first pending event, with a duration equal to a percentage (e.g. 20%) of the duration of the longest note in the pending events. If a player gesture **36** occurs within a rubato window **170** a tempo change command **40** is calculated and sent to the accompaniment sequencer **42**. The tempo change is calculates as follows;

tempo change=first pending event time−player gesture time

In a preferred embodiment, tempo is changed when a player gesture **36** occurs outside of a rubato window **170** but is limited to a maximum (clipped) value. Tempo is not updated on a STOP player gesture **36** since the start of a note is more musically significant. In an alternate embodiment, tempo is not updated when a player gesture **36** occurs outside of a rubato window **170**.

If no player gesture **36** is received by the end of a rubato window **170** and both a START and a STOP pending event is present in the pending notes processor **124**, the pending events are processed as if a player gesture **36** was received at the end of the rubato window **170**. This is called a forced output. This feature of the invention covers for lapse of attention by the player, preventing the player from getting too far behind the other players or the accompaniment sequencer **42**.

If a START and STOP pending event is not present, an output is not forced since it would be unmusical to stop all notes while a player is playing or start a note when the player is not playing.

To protect against the player gesturing too early and starting note events prematurely, a time point **178** is set between the current rubato window **170***g* and the previous rubato window **170***d*. In one embodiment the time point **178** can be set at 50%. In a preferred embodiment the time point **178** is varied by commands placed in the score. If a START or STOP-START player gesture **36** is received before the time point **178**, all the current notes on are REATTACHED and the pending events are unaffected. If a player gesture **36** of any type is received after the time point **178**, or a player gesture **36** of STOP type is received at any time, the player gesture **36** is applied to the current pending events. If the player gesture **36** occurs before the rubato window **170**, the value of the tempo change command. **40** is limited to the maximum positive (i.e. speed up tempo) value.

The rubato window **170** can be set by the player as a percentage ("the rubato tolerance") of the duration of the longest note occurring in the pending event. In a preferred embodiment the rubato window **170** is set by commands placed in the score **4**. A large rubato tolerance will allow a player to take great liberty with the timing and tempo of the piece. A rubato tolerance of zero will reduce the invention to that of a player piano, where the note events are played at exactly the times specified in score **4**, and the player and controller **6** will have no effect on the timing of the piece of music. A student may use this feature to hear how a piece is intended to be performed.

EXAMINATION OF NOTE SCHEDULER TIMING

Referring to FIG. **5**A, the scored notes **128** shall now be examined in detail to review the actions of the scheduler **28**. The START play gesture **36***a* arrives slightly early but within the rubato window **170***a* so note output command **38***a* is started, with a positive tempo change **40***a*. The STOP player gesture **36***aa* stops note output command **38***a*, much earlier than specified by the score **4**. Tempo is never updated on a STOP event. Note output command **38***b* is started by a START player gesture **36***b* before the rubato window **170***b* so the tempo change **40***b* is limited to the maximum positive value. In an alternate embodiment, which only allows pending events to be processed inside rubato windows **170**, the start of note output command **38***b* would have been postponed until the beginning of the rubato window **170***b*.

By the end of the rubato window **170***c* no player gesture **36** has been received so the start of note output command **38***c* has been forced and, in the time interval specified by the score **4**, note output command **38***b* has ended. The STOP-START player gesture **36***c*, corresponding to case 3 of Table 4, generates a REATTACK of note output command **38***cc*, which the STOP player gesture **36***cc* ends. The scored notes **128***d*, **128***e*, and **128***f*, are started by the START player gesture **36***d*, within the rubato window **170***d*, and slightly early, so a positive tempo command **40***d* is issued. The STOP-START player gesture **36***dd* falls before the 50% time point **178**, so note output command **38***f* is REATTACHED as note output command **38***ff*. Without the time point **178** feature, note output command **38***f* would have stopped abruptly and note output command **38***g* would have started very early. No player gesture **36** was detected within the next rubato window **170***g* so note output command **38***g* was forced to start at the end of the rubato window **170***g* and the maximum negative tempo change **40***g* sent. The STOP-START player gesture **36***f* stopped note output command **38***ff*. The next STOP-START player gesture **36***h* started note output command **38***h*, and the last STOP player gesture **36***hh* stopped note output command **38***h*. Notice that note output command **38***g* stops after note output command **38***h* stops, as specified by the score **4**.

FIG. **6**A and **6**B illustrates by means of a flow chart the preferred operation of the scheduler previously described and illustrated in FIG. **5**A and FIG. **5**B. The pending events processing logic case numbers listed in Table 4 are referred to in the flow chart by encircled numbers.

ACCOMPANIMENT SEQUENCER 42

Referring back to the detailed block diagram of FIG. **3**, the accompaniment sequencer **42** contains a conventional sequencer **226** whose tempo can be set by external control **228**. The function of the sequencer **226** is to select notes, and in a preferred embodiment expression commands, from the accompaniment channel(s) of the score **4** and send accompaniment note and expression commands **227** to the music synthesizer **18** at the times designated in the score **4**, and at a pace determined by the tempo clock **230**. In a preferred embodiment, time in the score **4** is not an absolute measurement (e.g. seconds) but a relative measurements ticks or beats). The tempo determines the absolute value of these relative time measurements. Expressions for tempo include ticks per second and beats per minute.

The tempo clock **230** can manually be changed by the player, for example by a knob (not shown), or automatically changed by tempo commands in the score, or changed by tempo change commands **40** from a scheduler **28**. If the tempo is to be changed by a scheduler **28**, the tempo selector **232** selects one of the schedulers **28**,**30**,**32**,**34** as the source of tempo change commands **40**. For the case of the preferred embodiment of FIG. **1**, the tempo selector **232** is a one-pole-four-throw switch, set by a tempo selector command **233** in the score **4**.

In string quartet music, for example, it is common for tempo control to pass among several players. The first violinist may start controlling the tempo, then pass tempo control to the cellist during a cello solo. In this case, it would be preferred for the score **4** to contain tempo selector commands each time tempo control changes hand. Typically the controller playing a lead or solo role in the music is given control over the tempo.

In a preferred embodiment, the time base for the invention is based on a clock whose frequency is regulated by tempo. The faster the tempo, the faster the clock frequency. In this way all time calculations and measurements (e.g. simultaneous margins **150**, rubato window **170**, note durations, time between notes) do not have to change as tempo changes, saving a good deal of calculation and making the software easier to implement.

MUSIC RE-PERFORMANCE EDITOR

A re-performance of the score **4** can be recycled by recording the output of the music re-performance system and using the recorded output as the score **4** in another re-performance. The recording can be implemented by replacing the music synthesizer **18** with a conventional sequencer program. In a preferred embodiment, two copies of the score **4** are kept, one is read as the other one is written. If the player is happy with a particular re-performance, the scores **4** are switched and the particular re-performance is used as the one being read. Recycling the score **4** produces a cumulative effect on note timing changes, allowing note timing over several re-performance generations to exceed the note timing restrictions imposed by the rubato window **170** for a single re-performance.

To edit expression commands of a score **4** without effecting the timing of the piece, the rubato window **170** is set to zero and the output of the re-performance is stored. To selectively edit expression commands stored in the score **4**, the expression processor **120** blocks all non-selected expression commands **44** from leaving the controller **6**. To change only note timing information, all expression commands **44** are blocked. In a similar manner, any combination of note timing and expression commands can selectively be edited.

POLYGESTURAL SCHEDULER 34

FIG. **7** illustrates how schedulers **28** can be combined to create a polygestural scheduler **34** capable of handling polyphonic instruments that produce multiple gestures. Some controllers are intrinsically monophonic, that is can only produce one note at a time, like a clarinet or flute. For these controllers, the monogestural scheduler **28** shown in the detailed block diagram FIG. **3** is sufficient. Others instruments, like a violin and guitar, are polyphonic and require a scheduler capable of processing multiple simultaneous gestures. Referring to FIG. **7**, a polygestural controller **12**, for example a guitar controller, with six independent gesture outputs **50** is connected to a polygestural scheduler **34** which contains six schedulers **28***a–f*. The scheduler allocator **54** receives the gestures **50** from the polygestural controller **12** and determines how many schedulers **28** to allocate to the polygestural controller **12**.

In a preferred embodiment of a polygestural scheduler **34** for guitar, the score **4** contains seven channels of guitar music. One channel of the score **4** contains melody notes. The other six channels contain chord arrangement, one channel of notes for each string of the guitar. Various allocation algorithms can be used to determine the routing of controller gesture outputs **50** to schedulers **28**. In a preferred embodiment one of two modes is established; LEAD or RHYTHM. In LEAD mode all gesture inputs **50** are combined and routed to one scheduler **28***a* that is assigned to the lead channel. In RHYTHM mode each gesture input **50** is routed to an individual scheduler **28**, and each scheduler **28** is assigned to individual score **4** channels.

In order to show the operation of the preferred embodiment of the ploygestural scheduler **34** for guitar using the preferred scheduler allocation **54** algorithm, in the context of the embodiment of the music re-performance system **2** illustrated in FIG. **1**, Score **2** MIDI channels must be assigned to each controller **6**, **8**, **10**, **12**. A typical channel assignment is presented in Table 5.

TABLE 5

CONTROLLER CHANNEL ASSIGNMENT			
Controller		Score	
Name	Number	Channel	Timbre
Violin	#1	1	Violin
Cello	#2	2	Cello
Flute	#3	3	Flute
Guitar	#4	4	Lead Guitar
		5	Rhythm Guitar String #1
		6	Rhythm Guitar String #2
		7	Rhythm Guitar String #3
		8	Rhythm Guitar String #4
		9	Rhythm Guitar String #5
		10	Rhythm Guitar String #6
Accompaniment		11	Bass guitar
		12	Piano
		13	Clarinet
		14	Snare drum
		15	High-hat drum
		16	Bass drum

Table 6 illustrates the operation of the scheduler allocator **54**, in LEAD and RHYTHM mode, which assigns gesture inputs **50** to schedulers **28**, and assigns schedulers **28** to score **4** MIDI channels.

TABLE 6

SCHEDULER ASSIGNMENT				
Gesture	LEAD MODE		RHYTHM MODE	
50	Scheduler 28	Score 4 Ch.	Schedule 28	Score 4 Ch
50a	28a	4	28 a	5
50b	28a	4	28b	6
50c	28a	4	28c	7
50d	28a	4	28d	8
50e	28a	4	28e	9
50f	28a	4	28f	10

Various methods can be used to determine the mode of the scheduler allocator **54**. In one embodiment a simple switch (not shown) mounted on the controller **12**, having two positions labeled LEAD and RHYTHM, allows the player to manually set the mode. In another embodiment, the scheduler allocator **54** automatically selects the mode by determining if a single string or multiple strings are being played. In one implementation of this embodiment, a short history of string activity (i.e. gesture outputs **50**) is analyzed. If a single string is plucked several times in succession (e.g. three, for example the string sequence **2,2,2** or **5,5,5**), LEAD mode is selected. If an ascending or descending sequence of a number of strings (e.g. three, for example the sequence of strings **2,3,4** or **6,5,4**) is plucked, RHYTHM mode is selected. If neither condition is met, the mode is not changed.

In a preferred embodiment (not shown) the controller **12** sets the mode of the scheduler allocator **54** by determining the location of the player's hand on the finger board. If the player's hand is high on the neck (towards the bridge), the controller **12** sets the scheduler allocator **54** mode to LEAD. If the player's hand is low on the neck (towards the nut),, the controller **12** sets the scheduler allocator **54** mode to RHYTHM. These gestures of playing lead high up on the neck and playing rhythm low down on the neck are part of the natural guitar gestural language most familiar to non-musicians.

A polygestural scheduler **34** can contain any number of schedulers **28**. Typically the number of schedulers **28** in a polygestural scheduler **34** is equal to the number of sound producing elements on the instrument (e.g. bass guitar and violin=4, banjo=5, guitar=6).

STRING CONTROLLER 236

FIG. **8** shows a string controller **236** capable of detecting energy and finger manipulation with an energy transducer **60** preferred for bowing. In one embodiment of the invention four controllers are used to play string quartets, consisting of a two violins, a viola, and a cello. In an alternate embodiment of the invention guitar and bass guitar controllers are used to play rock music. MIDI controllers exist for these instrument but are very costly since they are designed to generate pitch of acoustic quality, and typically employ pitch trackers, both of which are unnecessary and not used in the present invention.

A preferred embodiment of the music re-performance system **2** includes a string controller **236** which can be bowed and plucked, like a violin, or picked and strummed, like a guitar. The string controller **236** allows the ,use of common inexpensive sensor and signal processing techniques to reduce the cost of string controllers and allow interface to many hardware platforms. The string controller **236** is based on the controller model presented in the block diagram of FIG. 3. Two finger transducers **58** and four energy transducers **60** are examined, along with the signal processing required for them.

PREFERRED FINGER TRANSDUCER 58

Referring to FIG. **8**, the preferred finger transducer **58** consists of one or more metallic strings **240** suspended above a finger board **242** covered with a semiconductive material **244**, such as a semiconductive polymer, manufactured by Emerson-Cumings, Inc. (Canton, Mass.) as ECCOSHIELD (R) CLV (resistivity less than 10 ohm-cm), or by Interlink Electronics (Santa Barbara, Calif.). Use of a string **240** as part of the finger transducer **64** gives a realistic tactile experience and its purpose is instantly recognizable to the player. The string **240** terminates at one end in a rigid block **246**, taking the place of a bridge. The other end of the string **240** terminates in a tuning peg **248** at the head **250** of the neck **252**. Tension in the string **240** is required to keep the string **240** from touching the semiconductive material **244**. A spring can be used (not shown) as an alternative to the tuning peg **248** to provide tension in the string **240**. Electrical contacts are made at each end of the semiconductive material **244**, at the top finger board contact **254** and bottom finger board contact **256**, and at one end Of the string **240**, the string contact **258**. When a finger presses the string **240** onto the semi-conductive material **244**, an electric circuit is made between the string **240** and the semiconductive material **244**. The position of string **240** contact to the semiconductive material **244** is determined by the relative resistance between the string contact **258** to the top finger board contact **254**, and the string contact **258** to the bottom finger board contact **256**.

As finger pressure is applied to the string **240**, the contact resistance between the string **240** and the semiconductive material **244** decreases. Finger pressure is determined by measuring the resistance between the string **240** and the semiconductor material **244**.

For blown instruments the preferred finger transducers **58** are switches (not shown) which are electronically OR'ed together, so that a finger gesture **96** is produced whenever any switch is pressed or lifted. Force sensing resistors are preferred switches for they can measure finger contact and pressure. A force sensing resistor, manufactured by Interlink Electronics, is a semiconductive polymer deposit sandwiched between two insulator sheets, one of which includes conductive interdigiting fingers which are shunted by the semiconductive polymer when pressure is applied. The semiconductive polymer can also be used as the semiconductive material **244**.

ALTERNATE FINGER TRANSDUCER

An alternate finger transducer (not shown) is electrically equivalent to the preferred finger transducer **58** and is commercially available as the FSR Linear Potentiometer (FSR-LP) from Interlink. One version of the FSR-LP is 4" long and ¾" wide, suitable for a violin neck. Larger sizes can be made for other controllers, including violas, cellos, basses, and guitars. The force sensing resistor sensors are prefabricated and hermetically sealed so the internal contacts never get dirty, the surface is waterproof and can be wiped clean of sweat and other contaminants, the operation is stable and repeatable over time, and the sensors are very durable. The force sensing resistor sensor is under 1 mm. thick and has negligible compression and provides no tactile feedback. To compensate, a compressible material such as rubber or foam can be place over or under the force sensing resistor to give some tactile response.

PREFERRED ENERGY TRANSDUCER 60

The energy transducer **60** of the preferred embodiment consists of a textured rod **260** attached to a floating plate **262** suspended by four pressure sensors **264**. The four pressure sensors **264** are mounted to a flat rigid platform **268**. The body **269** of the string controller **236** can substitute for the flat rigid platform **268**. As a bow (not shown) is dragged across the textured rod **260**, forces are applied to the pressure sensors **264**.

FIG. **9**A and **9**B show a detailed top and side view, respectively, of the energy transducer **60** preferred for bowing. The function of the textured rod **260** is to simulate the feel of a string, particularly when bowed. An embodiment of the textured rod **260** is a threaded ¼ diameter steel rod with 20 threads per inch. The grooves give a good grabbing feeling as the bow is dragged across, though the pitch from the threads tends to force the bow off the normal to the rod. This ,can be corrected by sequentially scoring a rod (i.e. non-threaded). Other materials that grip the bow can be used including plastic, rubber, wood, wool, and rosin. Other shapes include a wedge, channel, and rectangle. In a preferred embodiment, the textured rod **260** is fastened with glue **270** to the floating plate **262**, as shown in FIG. **9**B.

When a bow is drawn across the textured rod **260**, the grabbing of the bow on the textured rod **260** generates forces on the floating platform **262**, transmitting pressures to the pressure sensors **264***a*, **264***b*, **264***c*, and **264***d*. These four pressures are analyzed to determine the placement of bow on the textured rod **260**, the bow pressure, and the bowing direction.

Pressure sensors **264** can include strain gauges, capacitance-effect pressure transducers, and piezo-ceramic transducers. A preferred embodiment uses force sensing resistors. The force sensing resistors are under 1 mm. thick and do not appreciably compress. Pads (e.g. foam) (not shown) can be added between the floating plate **262** and the platform **268** to give the sensation of a pliable string.

ALTERNATE ENERGY TRANSDUCERS 60

FIG. **10**A shows a string controller **236** using an optical beam **282** to measure string vibrations. A string **240** is placed between an upper block **272** and a lower block **274**. The blocks **272** and **274** are preferably made of an acoustic damping material like rubber to prevent string **240** vibrations from reaching the sound board (not shown) of the string controller **236**. An optical interrupter **280** (e.g. Motorola H21A1) is placed near the lower block **274**, such that the string **240** at rest is obscuring nominally half of the light beam **282** of the optical interrupter **280**, as illustrated in the cross section view of the optical interrupter **280** shown in FIG. **10**B. When the string **240** is bowed, plucked, picked, or strummed, string **240** vibrations modulate the light beam **282** of the optical interrupter **280**, producing an oscillating. electrical output **72***a* indicating string energy. If the string **240** is made stiff enough, like a solid metal rod, one block **274** can be used, allowing the other end of the string **240** to vibrate freely. This is particularly useful for a guitar controller, since the string **240** would have a naturally long decay which the player could modify for greater expressive control. For example a common guitar gesture is to muffle the strings with the side of the plucking hand. The expression processor **120** could detect this condition by monitoring the decay time, and generate appropriate expression commands **44** accordingly. The optointerrupter **280** does not contact the string **240**, measures string position, has a very fast response time (>10 Khz), is electrically isolated from the string, and produces an electric signal with a large signal-to-noise ratio.

FIG. **11** shows a detail of another method of measuring string vibration, using a piezo ceramic assembly **284**. The piezo-ceramic assembly **284**, mounted in a location similar to the optointerrupter **280** of FIG. **10**A, consists of a piezo-ceramic element **286** attached to a brass disk **290**. The brass disk **290** is placed in contact with the string **240**, so that vibrations in the string **240** are mechanically transmitted to the piezo-ceramic assembly **284**, producing an oscillating electrical output **72***b*, indicating string energy. In a preferred embodiment glue **270** is used to adhere the string **240** to the brass disk **290**. The piezo-ceramic assembly **284** is very low cost, generates it's own electric signal, is an a.c. device so it does not need to be decoupled, generates a large signal, and has a very thin profile.

FIG. **12** shows a tachometer **296** used to measure bow velocity and direction. A spindle **294** is mounted on a shaft **295** that connects at one end to a tachometer **296**, and at the other end to a bearing **298**. When a bow is drawn across the spindle **294**, the spindle **294** rotates, driving the tachometer **296** which produces an electric signal **72***c*, proportional to bow velocity. The side-to-side motion of the bearing **298** is constrained by a cradle **300**, but is free to pass pressure applied from the bow to the spindle **294**, to a bow pressure sensor **299**, which measures bow pressure **301**. A preferred bow pressure sensor **299** is a force sensing resistor.

In one embodiment the spindle **294** surface is covered with cloth thread to provide a texture for the bow to grab. The surface needs to grab the bow, as with the textured rod **260**. Most material can be treated to make the surface rough enough to grab the bow. Some surface treatments and materials include knurled wood, sandpaper, textured rubber, and rough finished plastic. Examples of tachometers **296** include an optical encoder, such as those used in mice pointing devices, a permanent magnetic motor operated as a generator, a stepper motor operated as a generator, or any other device that responds to rotation. An embodiment of the string controller **236** uses a stepper motor (not shown) to allow previously recorded bow motions to be played back, much like a player piano. An alternate embodiment uses a motor as a brake, providing resistance to bow movement, simulating the friction and grabbing of a bow on a string.

PREFERRED FINGER SIGNAL PROCESSING **64**

FIG. **13** show a schematic of an electronic circuit to perform all the signal processing necessary to implement a controller **6** using the preferred energy transducers **60** and finger transducers **58** of the string controller **236**. Most of the signal processing required is performed in software in the microcomputer **302** (MCU) to minimize hardware. A 68HC11 manufactured by Motorola is used as the MCU **302** in the preferred embodiment since it is highly integrated containing a plurality analog-to-digital converts (ADC), digital inputs (DIN) and digital outputs (DOUT), and a serial interface (SOUT), as well as RAM, ROM, interrupt controllers, and timers. Alternate embodiments of the signal processing using simple electronic circuits are presented, eliminating the need for the MCU **302**, and providing an inexpensive means of interfacing finger transducers **58** and energy transducers **60** to multi-media platforms.

The preferred finger transducer **58** is modeled as resistors R3, and R4. The semiconductive material **244** is modeled as two resistors **R2** and **R3** connected in series. The top finger board contact **254** connects to SWX **306**, the bottom finger board contact **256** connects to SWY **308**, and the string contact **258** connects to SWZ **310**. The connection point **304** between **R2**, **R3** and **R4** represents the contact point between the semiconductive material **244** and the string **240**. The .contact resistance between the string **240** and the semiconductive material **244** is represented by R4. The location of finger position along the length of the semiconductive material **244** is the ratio of R2 to R3. For example, when R2 equals R3 the finger is in the middle of the finger board **242**. Finger pressure is inversely proportional to R4.

Switches SWX **306**, SWY **308**, and SWZ **310** (e.g. CMOS switch **4052**), controlled by digital outputs DOUTX **312**, DOUTY **314**, and DOUTZ **316** of the MCU **302**, respectively, arrange the finger transducer contacts **254**, **256**, **258** to make the resistance measurements listed in Table 7. Switch **306**, **308**, **310** configurations place the unknown resistances (R2, R3, or R4) in series with known resistor R6, producing a voltage, buffered by a voltage follower **318** (e.g. National Semiconductor LM324), which is digitized by ADC5 **320**. The unknown resistances are determined by the voltage divider equation;

voltage measured=supply voltage×(R unknown/R6)

TABLE 7

SWITCH SETTINGS FOR RESISTANCE MEASUREMENT

SWX 306	SWY 308	SWZ 310	Resistance Measured
A	B	B	R2 + R4
B	A	B	R3 + R4
A	C	A	R2 + R3

These equations are sufficient to determine the values of R2, R3, and R4. It is important that the resistance measurements be done within a short period of time (.e.g. 20 msec) from each other, since the resistance of the semiconductive material **244** (R2+R3) can decrease when a several fingers hold down a length of the string **240**, electically shorting a portion of the semiconductive material **244**.

PREFERRED ENERGY SIGNAL PROCESSING **74**

Resistors **264*a***, **264*b***, **264*c***, and **264*d*** form voltage divider networks with resistors **R20**, **R22**, **R24**, and **R26**, respectively, producing pressure voltages **338**, **340**, **342**, and **344**, respectively, proportional to pressure, since the resistance of force sensing resistors decrease with pressure. The pressure voltages **338**, **340**, **342**, and **344** are buffered and filtered **346**, to remove high frequency noise caused by the scratching action of the bow across the textured rod **260**, and applied to the analog to digital converters ADC1 **348**, ADC2 **350**, ADC3 **352**, and ADC4 **354** of the MCU **302**. The voltage follower **355** provides the buffering and the combination of **R28** and **C10** provides the low-pass filtering.

Software inside the MPU **302** converts the low-passed pressure voltages **348**, **350**, **352**, and **354**: into bow pressure (BP), bow direction (BD), and the location of bow contact along the textured rod **260** (BC). The relationship between the pressure voltages **338**, **340**, **342**, and **344** and BP, BC, and BD are complicated by the bow orientation angles and torques (twisting actions) introduced by bowing but can be simplified to a first order approximation by the following relationships:

Let	A = the pressure of force sensing resistor 264a	
	B = the pressure of force sensing resistor 264b	
	C = the pressure of force sensing resistor 264c	
	D = the pressure of force sensing resistor 264d	
	Bow Pressure	BP = A + B + C + D
	Bow Contact Position	BC = (A + B) − (C + D)
	Bow Direction	BD = (A + D) − (B + C)

The platform **262** and the textured rod **260** have some weight, producing small pressure that can be compensated for by subtracting off the minimum pressure detected. Bow contact position is measured along length of textured rod **260**, and is a signed value with zero equal to the center of textured rod **260**. Bow direction is a signed value that is positive when the bow is moving towards the A and D force sensing resistors **264*a*** and **264*d*** and negative when moving towards the B and C force sensing resistors **264*b*** and **264*c***.

A property of the preferred energy transducer **60** is the bow does not have to be moving to maintain an energy state **76**, since a valid bow direction can be generated by statically bearing down on the textured rod **260**. This can be advantageous for a player who runs out of bow during a long sustained note. Since changing directions will cause a STOP-START event and likely REATTACK or change the note, the player can pause the bow while maintaining pressure on the textured rod **260** to infinitely sustain a note.

If this attribute is undesirable, the low-pass filters (R28-C10) can be removed, and the unfiltered pressure signals **338**, **340**, **342**, and **344** analyzed for scratching noise to determine bow movement. A preferred method of scratching noise analysis is to count the number of minor slope changes. The slope of a noisy signal changes frequently with small (minor) amplitude differences between slope changes. If the count of the minor slope changes exceeds a count threshold, the bow is moving. The value for the count and amplitude thresholds depend on a multitude of factors including the response characteristics of the pressure sensors **264*a–d***, the material of the textured rod **260**, and the material of the bow. The count and amplitude threshold are typically determined empirically.

FIG. **14** illustrates with wave form and timing diagrams the finger signal processing **64** necessary to determine finger state **68**. Once the finger resistances are determined and digitized, the MCU **302** calculates finger position as R2/R3 and finger pressure as R4. To determine the finger state **68**, the finger position **322** is differentiated, producing a slope signal **324** centered about zero **326**. If the slope **324** exceeds a fixed positive **328** or negative **330** reference, a finger state **68** pulse is produced. The positive threshold **328** is equal in magnitude to the negative threshold **330**. The magnitude of the thresholds **328**, **330** determine the distance the fingers must move (or the trombone valve must slide) in order to generate a finger state **68** pulse. If the magnitude is set too small, wiggling fingers **322***a* will produce a finger state **68** pulse. If the magnitude is set too large, large finger spans will be necessary to generate finger state **68** pulses. The magnitude can be fixed or set by the player for their comfort and playing style, and in the preferred embodiment is set by a sensitivity knob (not shown) on the string controller **236**. Player gesture **36** and expression commands **44** generated by the controller **6** hardware are sent through the serial output **261** (SOUT) to either the midi interface **16** or directly to the computer **14**.

The history of the finger activity presented in FIG. **14** will now reviewed. The finger position signal **322** at time **322***b* indicates a finger is pressing the string **240** onto the semiconductive material **244**. At time **322***c* the finger has released the string **240**. At time **322***d* a finger presses the string **240** onto the semiconductive material **244**, and at time **322***d* uses a second finger to place a higher portion of the string **240** onto the semiconductive material **244**, which is released at **322***f*. At time **322***g* the string **240** is pressed to the semiconductive material **244** and slowly slid up semiconductive material **244** up through time **322***h*. Since this was a slow slide, the slope **324***a* was too small to cause a finger state **68** pulse. At time **322***a*, finger wiggling, probably intended as vibrato, is ignored since the slope signal **324***b* it produces is smaller than the thresholds **328** and **330**.

FIG. **15** is a schematic representation of an electronic circuit to perform the finger signal processing **64** just discussed. A voltage proportional to finger position **322** is differentiated by capacitor C4 and applied to two comparators **332** and **334** that tests for the presence of the differentiated signal **324** above a positive threshold **328** set by the voltage divider R7 and R8, or below a negative threshold **330**, set by R9 and R10.

The finger state **68** output is a pulse generated by a monostable **336**, triggered by the output of true from either comparators **332** and **334**, which are logically ORed by the OR gate **335**.

TACHOMETER 296 AS AN ENERGY TRANSDUCER 60

FIG. **16** shows the wave forms of energy signal processing **74** for a tachometer **296**. A permanent magnetic motor, operating as a generator, is chosen as the preferred tachometer **296** due to its low cost. The motor produces an energy signal **72***c* with magnitude proportional to bow velocity, and sign determined by bow direction.

The energy signal **72***c* is displayed for several back-and-forth bowing motions. The direction of bowing determines the sign of the energy signal **72***c*. The energy state **76** is high when the absolute energy signal **356** exceeds a threshold **358**, representing the smallest acceptable bow velocity. The absolute energy signal **356** can be used as the energy magnitude **78**, but will usually be unacceptable as it drops to zero with every change of bow direction (e.g. at time **356***a*). A more realistic and preferred representation of energy magnitude **78** is an energy model that gives the feeling of energy attack (build-up) and decay, as happens in acoustically resonant instruments. In a preferred embodiment the energy magnitude **78** is expressed as the low-passed filtered product of the bow pressure (BP) and the absolute energy signal **356** (BV), and implemented by the following computational algorithm that is performed each time the energy magnitude **78** is updated (e.g. 60 times per second);

```
Let    Enew =     energy magnitude 78
       Eold =     Enew from last update
       BV =       absolute energy signal 356
       BP =       bow pressure
       Attack =   attack constant (0 to 1)
       Decay =    decay constant (0 to 1)
If (BV * BP > Eold)
       THEN
            Enew = Attack * ((BV * BP) – Eold) + Eold
       ELSE
            Enew = Release * ((BV * BP) – Eold) + Eold
Eold = Enew
```

For clarity, the energy magnitude **78** displayed in FIG. **16** is calculated with constant bow pressure. If bow pressure is not available, BP is set equal to 1. In a preferred embodiment, the expression processor **120** converts bow pressure and bow energy magnitude **78** into timbre brightness and volume expression commands **44**, respectively. With this scheme, slow and hard bowing (small BV, large BP) produces a bright and bold timbre, and fast and light bowing (large BV, small BP) produces a light and muted timbre, yet both at the same volume since volume is the product of bow pressure and absolute energy signal **356** (BV×BP).

FIG. **17** shows an electronic circuit to convert the output of the tachometer **296** into a binary energy event **76** and continuous energy magnitude **78**. A full wave rectifier **360** converts the tachometers output **72***c* into an absolute energy signal **356** which charges, through D20 and R36, or discharges, through D22 and R38, capacitor C20, whose voltage **364** is buffered by a voltage follower **365** and presented as the energy magnitude **78**. R36 determines the attack rate, R38 the decay rate.

PIEZO-CERAMIC 284 AND OPTOINTERRUPTOR 280 AS ENERGY TRANSDUCERS 60

FIG. **18** shows the wave forms of transducers that measure string vibration. The piezo-ceramic assembly **284** shown in FIG. **11** and optointerruptor **280** shown in FIG. **10***a* both measure string **240** vibration and so will be treated together as interchangeable energy transducers **60**. The energy transducer **60** produces an energy signal **72***a* that is a composite of the string vibration frequency **368** and a slower energy envelope **370**. Signal processing is used to extract the energy envelope **370** from the energy signal **72***a*, to produce an energy magnitude signal **382**. The energy signal **382** is similar to the absolute energy signal **356** of the tachometer **296** and can be processed by the energy signal processor circuit **74**, shown in FIG. **17**, to produce desired energy state **76** events and an energy magnitude signal **78**.

FIG. **19** shows an electronic circuit **383** to perform signal processing to convert string **240** vibrations from an energy transducer (e.g. **280** or **284**) into an energy signal **382**. The piezo ceramic crystal **286** generates an oscillating electrical output **72***b* in response to string **240** vibrations. The optointerrupter **280** consists of a light emitter (not shown) and a photo transistor Q1. String **240** vibrations modulate the light received by the photo transistor Q1, which passes a current through resistor R39, producing a corresponding oscillating electrical output **72***a*. The electric circuit **383** can process either oscillating electrical output **72***a* or **72***b*, so just electrical output **72***a* need be considered. The capacitor C40 removes any D.C. bias that might exist (of particular importance in the case of the optointerruptor **280**) in the energy transducer signal **72***a*. The decoupled signal **374** is buffered by a voltage follower **376** and a raw energy envelope **377** is extracted by a envelope follower **378** composed of diode D**10**, capacitor C**42**, and resistor R**44**, and buffered by a voltage follower **379**. A low-pass filter **380** made from resistor R**46** and C**44**, smoothes the raw energy envelope **377** to produce an energy signal **382** that can be applied to the energy signal processor **74**, shown in FIG. **17**, to produce an energy state **76** and energy magnitude **78** signal. R**44** and C**42** can be adjusted to change the decay time of the energy signal **382**. This is particularly useful on instrument controllers such as guitar and bass where the strings are picked and some sustain is desired. As the value of R**44** and C**42** increase, so does the decay time.

PLATFORMS

Many entertainment, multimedia computers, and audiovisual systems can be used as a hardware platform for the invention. The function of many of the system components of the invention can be implemented using the resources of the target machine. Entertainment systems include the NES by Nintendo, the Genesis machine by Sega, the CDI machine by Panasonic, and the 3D0 machine by 3D0. Some of these units have their own sound synthesizers which can be used in place of the music synthesizer **18**. Signal processing circuits have been shown that can be used and adapted, by one skilled in the art of electronics and computer programming, to many of the multimedia computers, video games, and entertainment systems commercially available, some of which have been listed here.

SUMMARY

The controller model **6** has been designed to accommodate a wide variety of musical instruments using low-cost transducers and simple signal processing, while maintaining a high degree of expression and control. The scheduler **28** is flexible enough to cover mistakes of beginners and allow great tempo and rubato control for proficient players. The simultaneous margin processor **122** can process conventional MIDI song files automatically, without player intervention, providing the player access to a large library of commercially available song files. The ability to selectively edit note timing and expression commands by re-performance and score **4** recycling allows a person to add life to song files.

The ability of the simultaneous margins **150** to adjust themselves to compensate for repeated mistakes by the player over several rehearsals, allows the music re-performance system **2** to learn, producing a better performance each time through.

The ability of the scheduler **23** to reattack notes allows the player room to improvise. Musicians often reattack notes for ornamentations. The polygestural scheduler **34** provides a guitarist with the ability to strum any sequence of stings with any rhythm, and the scheduler allocator **54** provides a smooth intuitive method to switch between rhythm and lead lines. The polygestural scheduler **34** also allows a player to select alternate musical lines from the score. A violinist could play one string for melody, another for harmony, and both for a duet. A bass player could use one string for the root of the chord, another for the fifth interval, a third for a sequence of notes comprising a walking bass line, and a forth string for the melody line, and effortlessly switch among them by plucking the appropriate string.

The modularity of the schedulers **28** permits each to have their own simultaneous margin **150** and rubato window **170**, allowing several people of different skill levels to play together, for example as a string quartet, rock or jazz band. The integration of the controllers **6**, schedulers **28**, score **4**, display **24**, and accompaniment sequencer **42**, provides a robust music education system that can grow with the developing skills of the player.

Although the present invention has been shown and described with respect to preferred embodiments, various changes and modifications which are obvious to a person skilled in the art to which the invention pertains are deemed to lie within the spirit and scope of the invention.

What we claim as our invention is:

1. A music re-performance system to generate music in response to musical gestures of a player comprising;

(a) storage means for storing information defining at least note pitch and note timing in at least one preprogrammed musical channel;

(b) finger transducer means for receiving finger manipulations from a player and for generating and for outputting a finger signal in response to said finger manipulations;

(c) energy transducer means for receiving energy applied by a player and for generating and outputting an energy signal in response to said energy applied to said energy transducer means by the player;

(d) signal processing means connected to said finger transducer means and to said energy transducer means for receiving said finger signal and said energy signal and for generating at least one gesture signal in response to said finger signal and to said energy signal;

(e) scheduling means connected to said storage means and to said signal processing means, for sequentially selecting at least one note from said storage means and for transmitting the selected note in response to said gesture signal; and

(f) sound generator means connected to said scheduling means for receiving the transmitted selected note and for producing sound in response to said selected notes.

2. A music re-performance system as set forth in claim **1**, further comprising at least one additional preprogrammed musical channel storing at least note and note timing information thus defining a musical accompaniment, and an accompaniment sequence means for reproducing said additional preprogrammed musical channel.

3. A music re-performance system as set forth in claim **2**, further comprising accompaniment tempo regulation means to regulate the tempo of the reproduction of said additional preprogrammed musical channel by the temporal relationship between said gesture signal and said note timing information.

4. A music re-performance system as set forth in claim **3**, wherein the tempo of reproduction increases when said gesture signal temporally leads said note timing information, and said tempo decreases when said gesture signal temporally lags said note timing information, resulting in the tempo of reproduction following the tempo of the player.

5. A music re-performance system as set forth in claim **1**, wherein said signal processing means further includes temporal masking means for generating a single gesture signal in response to a combination of finger and energy signals occurring within a temporal masking margin, thereby allowing finger and energy signals intended by the player to be simultaneous to generate a single gesture signal.

6. A music re-performance system as set forth in claim **5**, wherein said temporal masking margin lasts for a fraction of the duration of the note selected by said scheduling means.

7. A music re-performance system as set forth in claim **1**, further comprising expressive processing means for receiving said energy signal and for converting said energy signal into at least one control signal and for affecting change in at least one expressive parameter selected from the group consisting of volume, timbre, vibrato, and tremolo, whereby a player can control said expressive parameter through the energy applied to said energy transducer means.

8. A music re-performance system as set forth in claim **7**, wherein the said finger transducer means comprises a conductive wire suspended over a fingerboard whose surface is at least partially covered by a semi-conductive material, across the length of which a voltage potential is applied, whereby an electric signal proportional to the contact position along said fingerboard is produced in the wire when said wire is depressed thus contacting said semi-conductive material.

9. A music re-performance system as set forth in claim **1**, wherein said energy transducer means comprises at least one elongated member set into motion by a player energy gesture, whereby said energy transducer means produces an electric signal in response to the energy applied to said energy transducer means by said player energy gesture.

10. A music re-performance system as set forth in claim **9**, further comprising;

(a) a structure resembling a guitar wherein said finger transducer means is disposed along the neck of said structure and said energy transducer is disposed on the body of said structure;

(b) two preprogrammed musical channels, one defining a lead melody and the other defining chords;

(c) a scheduler allocator means connected to the two preprogrammed musical channels and to said scheduling means, said scheduler allocator means selecting said lead melody if said finger manipulations are applied to said finger transducer at a location substantially near the body of said structure, and otherwise said scheduler allocator means selecting said preprogrammed musical channel defining chords if said finger manipulations are applied to said finger transducer means at a location substantially far from the body of said structure, whereby said finger manipulations and said player energy gestures resemble the gestures of playing a guitar

11. A music re-performance system as set forth in claim **9**, wherein said energy transducer means further includes an optical interrupter means allowing at least some motion of said elongated member eclipsing at least some of the optical path of said optical interrupter means, said optical interrupter means producing an electric signal in response to the motion of said elongated member.

12. A music re-performance system as set forth in claim **9**, wherein said energy transducer means further includes a piezoelectric device in intimate contact with said elongated member, said piezoelectric device converting said motion into an electric signal in response to the motion of said elongated member.

13. A music re-performance system as set forth in claim **1**, wherein said energy transducer means comprises a rotating cylinder means allowing rotation by bowing actions of the player, further including rotational measurement means for producing an electric signal indicating rotation speed and direction, thus producing an electric signal indicating bow speed and direction.

14. A music re-performance system as set forth in claim **2**, further comprising a structure resembling a violin wherein said energy transducer means is disposed on the body of the structure and said finger transducer means is disposed along the neck of said structure, whereby said finger manipulations and said energy applied resembles the gestures of playing a violin.

15. A music re-performance system as set forth in claim **1**, wherein said energy transducer means further includes;

(a) an articulated member allowing a change in physical state, selected from the group consisting of position, compression, and tension, by the actions of the player;

(b) sensing means to convert said change in physical state into electric signals; and

(c) signal processing means to convert said electric signals into processed signals in response to the magnitude of said actions.

16. A music re-performance system as set forth in claim **1**, wherein said scheduling means further comprises means for selecting a plurality of notes from said storage means in response to a single gesture signal.

17. A music re-performance system as set forth in claim **16**, wherein the selection of said plurality of notes is determined by a temporal simultaneous margin, said temporal simultaneous margin chosen from among the following; a constant value, a percentage of the duration of a selected note, a value set by the player, a value stored in said storage means, or a sequence of values stored in said storage means.

18. A music re-performance system as set forth in claim **1**, wherein said scheduling means further comprises, a rubato tolerance means for limiting the magnitude of the temporal difference between said note timing as specified in said storage means and the transmission of said selected note.

19. A music re-performance system as set forth in claim **1**, further comprising;

(a) a plurality of said finger transducers, outputting at least one finger signal in response to said finger manipulations of said finger transducer means;

(b) a plurality of said energy transducer means, for outputting at least one energy signal in response to energy applied to said energy transducer means;

(c) a plurality of said preprogrammed musical channels;

(d) signal processing means for receiving said finger signal and said energy signal and for generating at least one gesture signal in response to said finger signal and to said energy signal;

(e) polygestural scheduling means, connected to said storage means and said signal processing means, for selecting a plurality of notes from a plurality of said preprogrammed musical channels, whereby a temporal sequence of polyphonic music can be regulated by a combination of finger manipulations applied to said finger transducer means and energy applied to said energy transducer means.

20. A music re-performance system as set forth in claim **1**, further comprising computing means connected to said storage means for generating a visual representation of information contained in said preprogrammed musical channel.

21. A music editing system to edit selected note parameters of a musical score by dynamically changing the note parameters comprising;

(a) an information storage means for storing at least one preprogrammed musical channel defining at least one note parameter selected from the group consisting of pitch, start time, stop time, duration, volume, timbre, vibrato, and tremolo, where said musical channel represents the musical score to be edited;

(b) energy transducer means for receiving energy applied by a player and for generating and for outputting an energy signal in response to said energy applied to said energy transducer means;

(c) signal processing means connected to said energy transducer means for receiving said energy signal and for generating at least one energy control signal in response to said energy signal;

(d) scheduling means connected to said storage means and to said signal processing means for sequentially selecting at least one note parameter and for altering said note parameter in response to said energy control signal, whereby said altering represents an edited version of said note parameter; and

(e) sound generator means connected to said scheduling means for receiving said altered note parameter and producing sound in response to said altered note parameter.

22. A music editing system as set forth in claim **21**, further comprising at least one additional preprogrammed musical channel for storing at least note pitch and note timing information thus defining a musical accompaniment, and an accompaniment sequence means for reproducing said additional preprogrammed musical channel.

23. A music editing system as set forth in claim **22**, further comprising accompaniment tempo regulation means to regulate the tempo of the reproduction of said additional preprogrammed musical channel by the temporal relationship between said energy control signal and note timing information stored in said preprogrammed musical channel, whereby the tempo of said accompaniment responds to the timing of said energy signal.

24. A music editing system as set forth in claim **21**, further comprising finger transducer means connected to said signal processing means to receive finger manipulations from a player and for generating and for outputting a finger signal in response to said finger manipulations, said signal processing means receiving said finger signal and generating at least one finger control signal in response to said finger signal and said scheduling means altering said note parameter in response to said finger control signal.

25. A music editing system as set forth in claim **24**, wherein said signal processing means further includes temporal masking means for generating a single gesture signal in response to a combination of said finger signal and said energy signal received within a temporal masking margin, thereby using said gesture signal for altering the timing of notes in said preprogrammed musical channel.

26. A music editing system as set forth in claim **25**, wherein said temporal masking margin lasts for a fraction of the duration of the note selected by said scheduling means.

27. A music editing system as set forth in claim **21**, wherein said scheduling means further comprises a rubato tolerance means for limiting the magnitude of temporal alterations of note parameters.

28. A music editing system as set forth in claim **21**, further comprising computing means connected to said storage means for generating a visual representation of information contained in said preprogrammed musical channel.

29. A music re-performance system to generate music in response to musical gestures of a player comprising;

(a) storage means for storing information defining at least note and note timing in at least one preprogrammed musical channel;

(b) an energy transducer means for receiving player gestures and generating at least one energy signal in response to at least one said player gesture performed on said energy transducer means;

(c) signal processing means connected to said energy transducer means for receiving said energy signal and for generating a gesture signal in response to said energy applied to said energy transducer means;

(d) scheduler means connected to said storage means and to said energy transducer means, for sequentially selecting notes from said storage means that occur within a temporal simultaneous margin, and for transmitting the selected notes in response to said gesture signal, whereby a single player gesture may result in a plurality of transmitted notes; and

(e) sound generator means connected to said scheduler means, for receiving the transmitted selected notes and for producing sound in response to said selected notes.

30. A music re-performance system as set forth in claim **29**, wherein said temporal simultaneous margin is chosen from among the following; a constant value, a percentage of the duration of a selected notes, a value set by the player, a value stored in said storage means, or a sequence of values stored in said storage means.

31. A music re-performance system as set forth in claim **30** wherein said scheduling means further comprises rubato tolerance processing means for limiting the magnitude of the temporal difference between said note timing as specified in said storage means and the transmission of said selected note.

32. A music re-performance system as set forth in claim **29**, further comprising at least one additional preprogrammed musical channel for storing at least note and note timing information thus defining a musical accompaniment, and an accompaniment sequence means for reproducing said additional preprogrammed musical channel.

33. A music re-performance system as set forth in claim **32**, further comprising accompaniment tempo regulation means to regulate the tempo of the reproduction of said additional preprogrammed musical channel by the temporal relationship between said gesture signal and said note timing information, resulting in the tempo of said accompaniment responding to the timing of musical gestures of the player.

34. A music re-performance system as set forth in claim **29**, further comprising expressive processing means to receive said energy signal and for converting said energy signal into at least one control signal for effecting change in at least one expressive parameter selected from the group consisting of volume, timbre, vibrato, and tremolo, for controlling said expressive parameter through the energy applied to said energy transducer.

* * * * *

EXHIBIT 3

EXHIBIT 3

US005491297A

United States Patent [19]

Johnson et al.

[11] **Patent Number: 5,491,297**

[45] **Date of Patent: Feb. 13, 1996**

[54] **MUSIC INSTRUMENT WHICH GENERATES A RHYTHM EKG**

[75] Inventors: **Charles L. Johnson**, Cambridge, Mass.; **Allan A. Miller**, Hollis, N.H.; **Herbert P. Snow**, Somerville, Mass.; **Vernon A. Miller**, Mount Vernon, N.H.

[73] Assignee: **Ahead, Inc.**, Netwon, Mass.

[21] Appl. No.: **177,741**

[22] Filed: **Jan. 5, 1994**

Related U.S. Application Data

[63] Continuation-in-part of Ser. No. 73,128, Jun. 7, 1993, Pat. No. 5,393,926.

[51] **Int. Cl.**^{6} **G09B 15/04**; G10H 1/26

[52] **U.S. Cl.** .... **84/609**; 84/477 R

[58] **Field of Search** .... 84/609–614, 634–638, 84/645, DIG. 22, 477 R, 478

[56] **References Cited**

U.S. PATENT DOCUMENTS

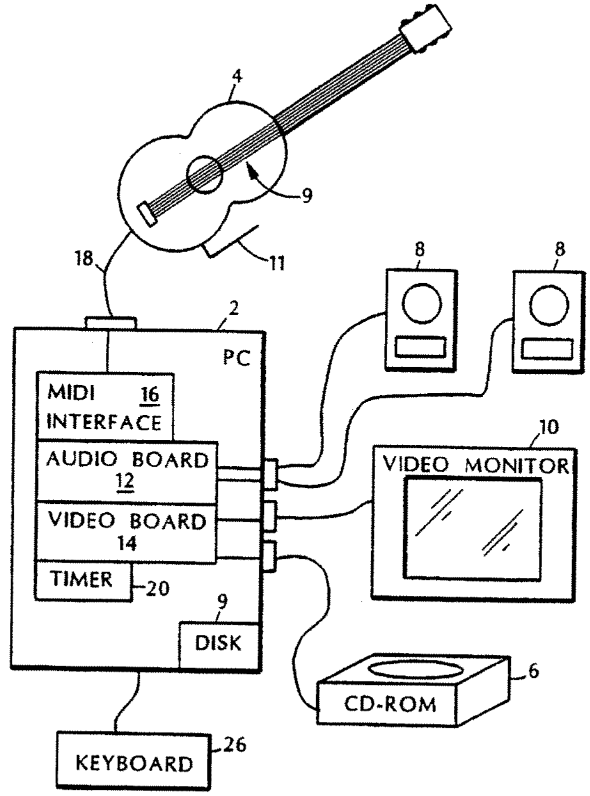
4,960,031	10/1990	Farrand .	
5,074,182	12/1991	Capps et al. .	
5,099,738	3/1992	Hotz .	
5,146,833	9/1992	Lui .	
5,270,475	12/1993	Weiss et al.	84/645 X
5,287,789	2/1994	Zimmerman	84/477 R

Primary Examiner—Stanley J. Witkowski
Attorney, Agent, or Firm—Fish & Richardson

[57] **ABSTRACT**

A musical instrument including an actuator which generates a plurality of signals in response to being played by a user; an audio synthesizer which generates audio tones in response to control signals; a memory storing a musical score for the actuator; a video display unit; and a digital processing means controlling the audio synthesizer and the video display unit. The stored musical score includes a sequence of lead notes each of which has an associated time stamp to identify a time at which it is supposed to be played in the musical score. The digital processing means is programmed to map the plurality of signals to a corresponding subsequence of lead notes from among the sequence of lead notes; it is programmed to produce a sequence of control signals from the subsequence of lead notes for causing the synthesizer to generate sounds representing the subsequence of lead notes; it is programmed to display on the video display unit a trace indicating when the lead notes of the sequence of lead notes are supposed to be played by the user as a function of time; and it is programmed to display relative to that trace an indicator marking where the user is supposed to be within the musical score as a function of an elapsed real time.

9 Claims, 7 Drawing Sheets



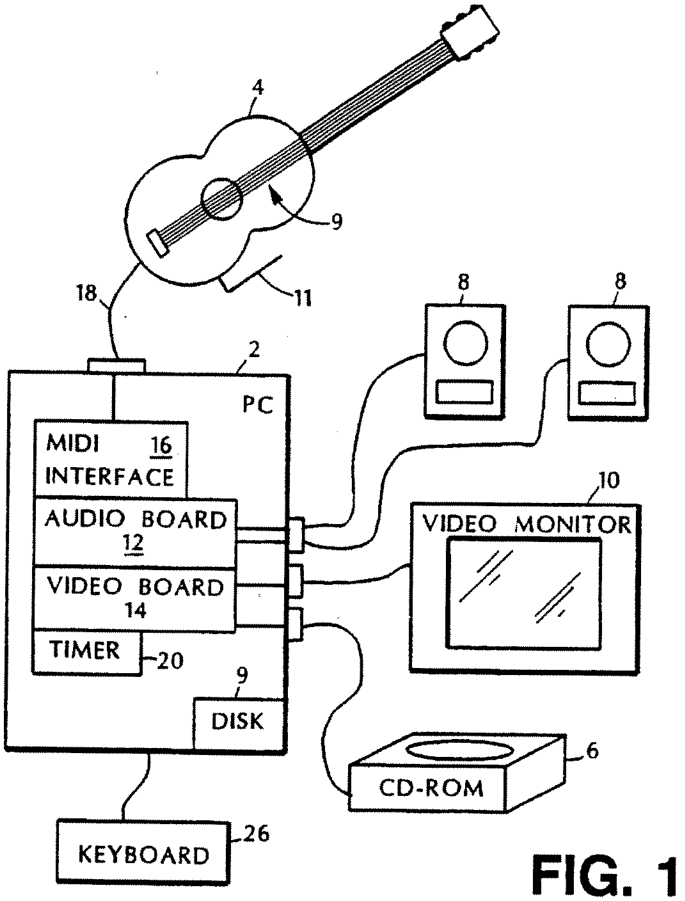

4
9
18
11
8
8
2
PC
MIDI INTERFACE 16
AUDIO BOARD 12
VIDEO BOARD 14
TIMER
20
9
DISK
10
VIDEO MONITOR
6
CD-ROM
KEYBOARD
26

FIG. 1

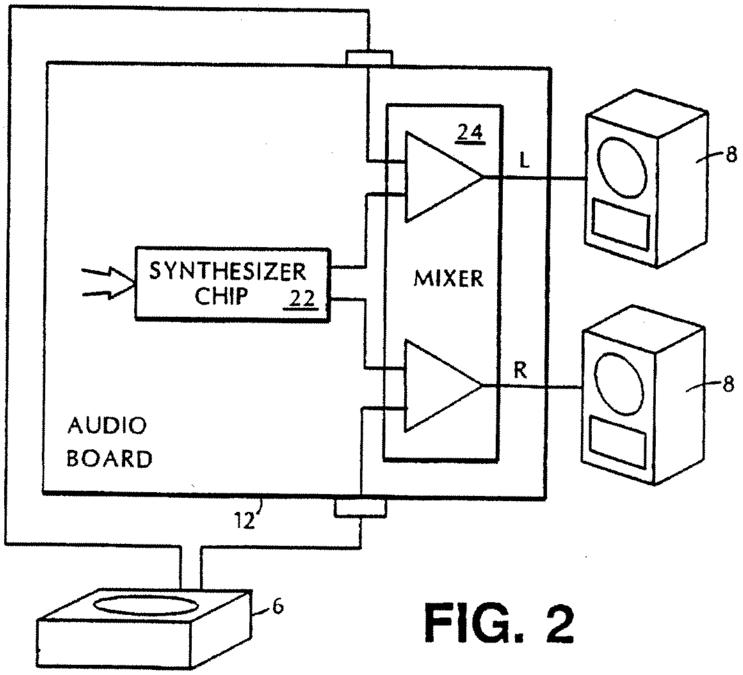

24
L
8
SYNTHESIZER CHIP 22
MIXER
R
8
AUDIO BOARD
12
6

FIG. 2

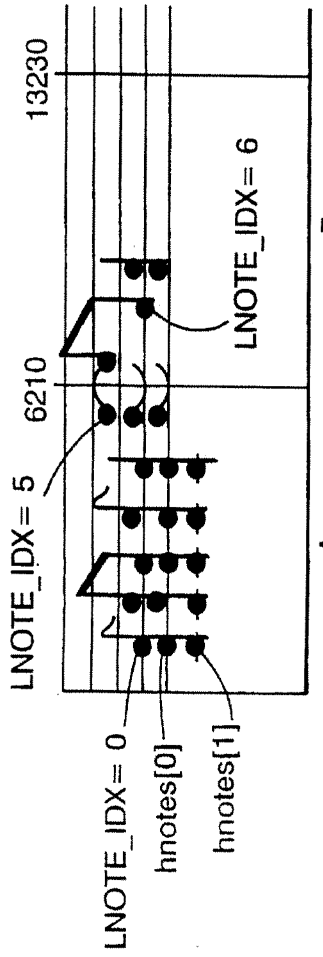

LNOTE_IDX= 5
6210
13230
LNOTE_IDX= 6
F
Am
LNOTE_IDX= 0
hnotes[0]
hnotes[1]

FIG. 3

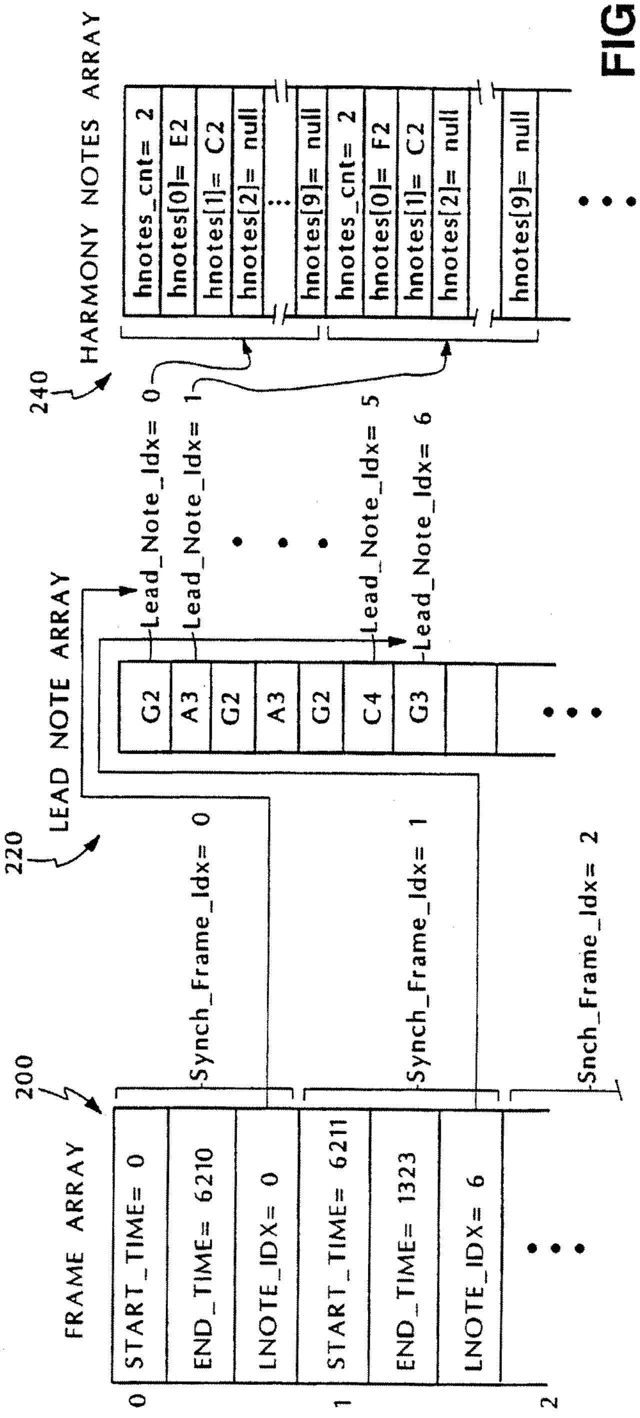

HARMONY NOTES ARRAY
240
hnotes_cnt= 2
hnotes[0]= E2
hnotes[1]= C2
hnotes[2]= null
hnotes[9]= null
hnotes_cnt= 2
hnotes[0]= F2
hnotes[1]= C2
hnotes[2]= null
hnotes[9]= null
Lead_Note_Idx= 0
Lead_Note_Idx= 1
Lead_Note_Idx= 5
Lead_Note_Idx= 6
LEAD NOTE ARRAY
220
G2
A3
G2
A3
G2
C4
G3
FRAME ARRAY
200
Synch_Frame_Idx= 0
Synch_Frame_Idx= 1
Snch_Frame_Idx= 2
0
START_TIME= 0
END_TIME= 6210
LNOTE_IDX= 0
1
START_TIME= 6211
END_TIME= 1323
LNOTE_IDX= 6
2

FIG.4

```
main ()
        {
100     system_initialization()
102     register_midi_callback(virtual_guitar_callback);
104     while (continue)
        {
106         get_song_id_from_user();
108         set_up_data_structures(song_id);
110         initialize_data_structures(song_id);
112         play_song(song_id);
        }
        }
```

FIG. 5

```
play_song(song_id)
        {
130     announce_song_to_user();
132     wait_for_user_start_signal();
134     start_interleaved_audio_video(song_id);
        }
```

FIG. 6

```
virtual_guitar_callback()
        {
200 current_time = get_current_time();

202 event_type = get_guitar_string_event(&string_id, &string_velocity);

204 switch (event_type)

         case STRING_ON :

    210 if (current_frame_idx != get_hframe(current_time)
        then
           {
      212 current_frame_idx = get_hframe(current_time);

      214 start_tone_gen(string_velocity, string_id,
                    lnotes_array[sframes[current_frame_idx].lnote_idx]);

      215 current_lead_note_idx = lnotes_array[sframes[
                    current_frame_idx].lnote_idx]

      216 hnotes_played = 0;
           }
    218 else
           {
      220 diff_time = current_time-last_time;

      222 if (diff_time< SIMULTAN_THRESHOLD)
          then

        224 start_tone_gen(string_velocity, string_id,
            hnotes_array[current_lead_note_idx]
            .hnotes[hnotes_played++]);
```

FIG. 7A

```
                else
                    {
       226 ——— start_tone_gen(string_velocity, string_id,
                                    lnotes_array[++current_lead_note_idx]);
       228 ——— hnotes_played = 0;
          case STRING_OFF :
   230 ——— unsound_note(string_id);
          case TREMELO :
   232 ——— pass_tremelo_control_data();
      }
}
```

FIG. 7B

```
struct sync_frame {
     TIMESTAMP_VALUE     frame_start_time;
     TIMESTAMP_VALUE     frame_end_time;
     int                 lnote_idx;
     }
```

FIG. 8

```
struct lead_note {
     int                 lead_note;
     TIMESTAMP_VALUE     time;
     }
```

FIG. 9

```
struct harmony_notes {
     int hnote_cnt;
     int hnotes[10];
     };
```

FIG. 10

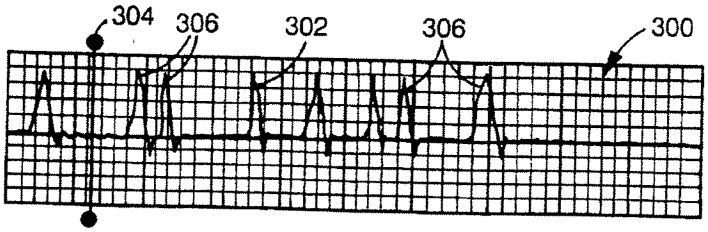


FIG. 11

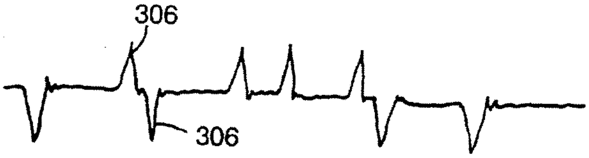


FIG. 12

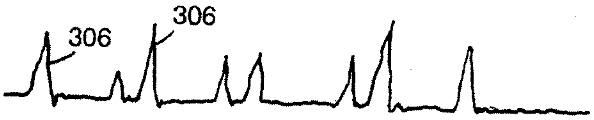


FIG. 13

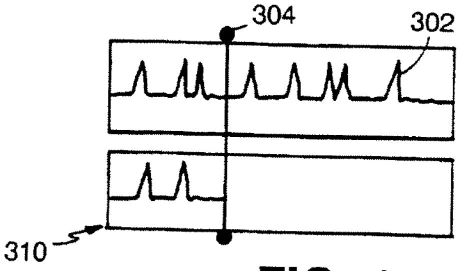


FIG. 14

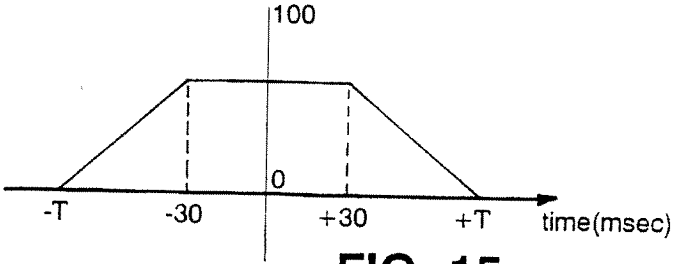


FIG. 15

MUSIC INSTRUMENT WHICH GENERATES A RHYTHM EKG

CROSS-REFERENCE TO RELATED APPLICATION

This application is a continuation-in-part of U.S. patent application Ser. No. 08/073,128, filed on Jun. 7, 1993 and now U.S. Pat. No. 5,393,926.

BACKGROUND OF THE INVENTION

The invention relates to microprocessor-assisted musical instruments.

As microprocessors penetrate further into the marketplace, more products are appearing that enable people who have no formal training in music to actually produce music like a trained musician. Some instruments and devices that are appearing store the musical score in digital form and play it back in response to input signals generated by the user when the instrument is played. Since the music is stored in the instrument, the user need not have the ability to create the required notes of the melody but need only have the ability to recreate the rhythm of the particular song or music being played. These instruments and devices are making music much more accessible to everybody.

Among the instruments that are available, there are a number of mechanical and electrical toy products that allow the player to step through the single tones of a melody. The simplest forms of this are little piano shaped toys that have one or a couple of keys which when depressed advance a melody by one note and sound the next tone in the melody which is encoded on a mechanical drum. The electrical version of this ability can be seen in some electronic keyboards that have a mode called "single key" play whereby a sequence of notes that the player has played and recorded on the keyboard can be "played" back by pushing the "single key play" button (on/off switch) sequentially with the rhythm of the single note melody. Each time the key is pressed, the next note in the melody is played.

There was an instrument called a "sequential drum" that behaved in a similar fashion. When the drum was struck a piezoelectric pickup created an on/off event which a computer registered and then used as a trigger to sound the next tone in a melodic note sequence.

There are also recordings that are made for a variety of music types where a single instrument or, more commonly, the vocal part of a song is omitted from the audio mix of an ensemble recording such as a rock band or orchestra. These recordings available on vinyl records, magnetic tape, and CDs have been the basis for the commercial products known as MusicMinusOne and for the very popular karoeke that originated in Japan.

SUMMARY OF THE INVENTION

In general, in one aspect, the invention features a virtual musical instrument including a multi-element actuator which generates a plurality of signals in response to being played by a user; an audio synthesizer which generates audio tones in response to control signals; a memory storing a musical score for the multi-element actuator; and a digital processor receiving the plurality of signals from the multi-element actuator and generating a first set of control signals therefrom. The musical score includes a sequence of lead notes and an associated sequence of harmony note arrays, each harmony note array of the sequence corresponding to a different one of the lead notes and containing zero, one or more harmony notes. The digital processor is programmed to identify from among the sequence of lead notes in the stored musical score a lead note which corresponds to a first one of the plurality of signals. It is programmed to map a set of the remainder of the plurality of signals to whatever harmony notes are associated with the selected lead note, if any. And it is programmed to produce the first set of control signals from the identified lead note and the harmony notes to which the signals of the plurality of signals are mapped, the first set of control signals causing the synthesizer to generate sounds representing the identified lead note and the mapped harmony notes.

Preferred embodiments include the following features. The multi-element actuator is an electronic musical instrument, namely, a MIDI guitar, and the plurality of multi-element actuators includes strings on the guitar. The virtual musical instrument further includes a timer resource which generates a measure of elapsed time, wherein the stored musical score contains time information indicating when notes of the musical score can be played and wherein the digital processor identifies the lead note by using the timer resource to measure a time at which the first one of the plurality of signals occurred and then locating a lead note within the sequence of lead notes that corresponds to the measured time. The digital processor is further programmed to identify a member of the set of the remainder of the plurality of signals by using the timer resource to measure a time that has elapsed since a preceding signal of the plurality of signals occurred, by comparing the elapsed time to a preselected threshold, and if the elapsed time is less than the preselected threshold, by mapping the member of the set of the remainder of the plurality of signals to a note in the harmony array associated with the identified lead note. The digital processor is also programmed to map the member of the remainder of the plurality of signals to a next lead note if the elapsed time is greater than the preselected threshold.

In general, in another aspect, the invention features a virtual musical instrument including an actuator generating a signal in response to being activated by a user; an audio synthesizer; a memory storing a musical score for the actuator; a timer; and a digital processor receiving the signal from the actuator and generating a control signal therefrom. The stored musical score includes a sequence of notes partitioned into a sequence of frames, each frame of the sequence of frames containing a corresponding group of notes of the sequence of notes and wherein each frame of the sequence of frames has a time stamp identifying its time location within the musical score. The digital processor is programmed to use the timer to measure a time at which the signal is generated; it is programmed to identify a frame in the sequence of frames that corresponds to that measured time; it is programmed to select one member of the group of notes for the identified frame; and it is programmed to generate the control signal, wherein the control signal causes the synthesizer to generate a sound representing the selected member of the group of notes for the identified frame.

In preferred embodiments, the virtual musical instrument further includes an audio playback component for storing and playing back an audio track associated with the stored musical score. In addition, the digital processor is programmed to start both the timer and the audio playback component at the same time so that the identified frame is synchronized with the playback of the audio track. The audio track omits a music track, the omitted music track being the musical score for the actuator. The virtual musical instrument also includes a video playback component for storing and playing back a video track associated with the stored musical score. The digital processor starts both the timer and the video playback component at the same time so that the identified frame is synchronized with the playback of the video track.

In general, in yet another aspect, the invention features a control device including a medium containing stored digital information, the stored digital information including a musical score for the virtual instrument previously described and wherein the musical score is partitioned into a sequence of frames.

In general, in still another aspect, the invention features a method for producing a digital data file for a musical score. The method includes the steps of generating a digital data sequence corresponding to the notes in the musical score; partitioning the data sequence into a sequence of frames, some of which contain more than one note of the musical score; assigning a time stamp to each of the frames, the time stamp for any given frame representing a time at which that frame occurs in the musical score; and storing the sequence of frames along with the associated time stamps on a machine readable medium.

In preferred embodiments, the time stamp for each of the frames includes a start time for that frame and an end time for that frame. The musical score includes chords and the step of generating a digital data sequence includes producing a sequence of lead notes and a corresponding sequence of harmony note arrays, each of the harmony note arrays corresponding to a different one of the lead notes in the sequence of lead notes and each of the harmony note arrays containing the other notes of any chord to which that lead note belongs.

In general, in still another aspect, the invention is a musical instrument including an actuator which generates a plurality of signals in response to being played by a user; an audio synthesizer which generates audio tones in response to control signals; a memory storing a musical score for the actuator; a video display unit; and a digital processing means controlling the audio synthesizer and the video display unit. The stored musical score includes a sequence of lead notes each of which has an associated time stamp to identify a time at which it is supposed to be played in the musical score. The digital processing means is programmed to map the plurality of signals to a corresponding subsequence of lead notes from among the sequence of lead notes; it is programmed to produce a sequence of control signals from the subsequence of lead notes for causing the synthesizer to generate sounds representing the subsequence of lead notes; and it is programmed to display a song EKG on the video display unit. The song EKG is a trace indicating when the lead notes of the sequence of lead notes are supposed to be played by the user as a function of time and it includes an indicator relative marking where the user is supposed to be within the musical score as a function of an elapsed real time.

One advantage of the invention is that, since the melody notes are stored in a data file, the player of the virtual instrument need not know how to create the notes of the song. The player can produce the required sounds simply by generating activation signals with the instrument. The invention has the further advantage that it assures that the player of the virtual instrument will keep up with the song but yet gives the player substantial latitude in generating the music within predefined frames of the musical score. In addition, the invention enables user to produce one or more notes of a chord based on the number of strings (in the case of a guitar) that he strikes or strums. Thus, even though the actual musical core may call for a chord at a particular place in the song, the player of the musical instrument can decide to generate less than all of the notes of that chord.

The rhythm EKG provides an effective tool for helping novices to learn how to play the musical instrument.

Other advantages and features will become apparent from the following description of the preferred embodiment, and from the claims.

BRIEF DESCRIPTION OF THE DRAWING

FIG. **1** is a block diagram of the virtual music system;

FIG. **2** is a block diagram of the audio processing plug-in board shown in FIG. **1**;

FIG. **3** illustrates the partitioning of a hypothetical musical score into frames;

FIG. **4** shows the sframes[], lnote$_{13}$array[], and hnotes$_{13}$array[]data structures and their relationship to one another;

FIG. **5** shows a pseudocode representation of the main program loop;

FIG. **6** shows a pseudocode representation of the play$_{13}$song() routine that is called by the main program lop;

FIGS. **7**A and **7**B show a pseudocode representation of the virtual$_{13}$guitar$_{13}$callback() interrupt routine that is installed during initialization of the system;

FIG. **8** shows the sync$_{13}$frame data structure;

FIG. **9** shows the lead$_{13}$note data structure;

FIG. **10** shows the harmony$_{13}$notes data structure;

FIG. **11** shows a song EKG as displayed to a user;

FIG. **12** shows a song EKG in which the displayed signal exhibits polarity to indicate direction of strumming;

FIG. **13** shows a song EKG in which the amplitude of the peaks indicates the vigor with which the player should be strumming;

FIG. **14** shows a song EKG and a player EKG; and

FIG. **15** shows a sample scoring algorithm for color coding the player EKG.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to FIG. **1**, a virtual music system constructed in accordance with the invention includes among its basic components a Personal Computer (PC) **2**; a virtual instrument, which in the described embodiment is a MIDI guitar **4**; and a CD-ROM player **6**. Under control of PC **2**, CD-ROM player **6** plays back an interleaved digital audio and video recording of a song that a user has selected as the music that he also wishes to play on guitar **4**. Stored in PC **2** is a song data file (not shown in FIG. **1**) that contains a musical score that is to be played by MIDI guitar **4**. It is, of course, for the guitar track of the same song that is being played on CD-ROM player **6**.

MIDI guitar **4** is a commercially available instrument that includes a multi-element actuator, referred to more commonly as a set of strings **9**, and a tremelo bar **11**. Musical Instrument Digital Interface (MIDI) refers to a well known standard of operational codes for the real time interchange of music data. It is a serial protocol that is a superset of RS-232. When an element of the multi-element actuator (i.e., a string) is struck, guitar **4** generates a set of digital opcodes describing that event. Similarly, when tremelo bar **11** is used, guitar **4** generates an opcode describing that event. As the user plays guitar **4**, it generates a serial data stream of such "events" (i.e., string activations and tremelo events) that are sent to PC **2** which uses them to access and thereby play back the relevant portions of the stored song in PC **2**. PC **2** mixes the guitar music with the audio track from CD-ROM player and plays the resulting music through a set of stereo speakers **8** while at the same time displaying the accompanying video image on a video monitor **10** that is connected to PC **2**.

PC **2**, which includes a 80486 processor, 16 megabytes of RAM, and 1 gigabyte of hard disk storage **9**, uses a Microsoft™ Window 3.1 Operating System. It is equipped with several plug-in boards. There is an audio processing plug-in board **12** (also shown in FIG. **2**) which has a built in programmable MIDI synthesizer **22** (e.g. a Proteus synthesis chip) and a digitally programmable analog **2** channel mixer **24**. There is also a video decompression/accelerator board **14** running under Microsoft's VideoForWindows™ product for creating full-screen, full motion video from the video signal coming from CD-ROM player **6**. And there is a MIDI interface card **16** to which MIDI guitar **4** is connected through a MIDI cable **18**. PC **2** also includes a programmable timer chip **20** that updates a clock register every millisecond.

On audio processing plug-in board **12**, Proteus synthesis chip **22** synthesizes tones of specified pitch and timbre in response to a serial data stream that is generated by MIDI guitar **4** when it is played. The synthesis chip includes a digital command interface that is programmable from an application program running under Windows 3.1. The digital command interface receives MIDI formatted data that indicate what notes to play at what velocity (i.e., volume). It interprets the data that it receives and causes the synthesizer to generate the appropriate notes having the appropriate volume. Analog mixer **24** mixes audio inputs from CD-ROM player **9** with the Proteus chip generated waveforms to create a mixed stereo output signal that is sent to speakers **8**. Video decompression/accelerator board **14** handles the accessing and display of the video image that is stored on a CD-ROM disc along with a synchronized audio track. MIDI interface card **16** processes the signal from MIDI guitar **4**.

When MIDI guitar **4** is played, it generates a serial stream of data that identifies what string was struck and with what force. This serial stream of data passes over cable **18** to MIDI interface card **16**, which registers the data chunks and creates interrupts to the 80486. The MIDI Interface card's device driver code which is called as part of the 80486's interrupt service, reads the MIDI Interface card's registers and puts the MIDI data in an application program accessible buffer.

MIDI guitar **4** generates the following type of data. When a string is struck after being motionless for some time, a processor within MIDI guitar **4** generates a packet of MIDI formatted data containing the following opcodes:

MIDI$_{13}$STATUS=On
MIDI$_{13}$NOTE=<note number)
MIDI$_{13}$VELOCITY=<amplitude>

The <note number> identifies which string was activated and the <amplitude> is a measure of the force with which the string was struck. When the plucked string's vibration decays to a certain minimum, then MIDI guitar **4** sends another MIDI data packet:

MIDI$_{13}$STATUS=Off
MIDI$_{13}$NOTE=<note number)
MIDI$_{13}$VELOCITY=0

This indicates that the tone that is being generated for the string identified by <note number> should be turned off.

If the string is struck before its vibration has decayed to the certain minimum, MIDI guitar **4** generates two packets, the first turning off the previous note for that string and the second turning on a new note for the string.

The CD-ROM disc that is played on player **6** contains an interleaved and synchronized video and audio file of music which the guitar player wishes to play. The video track could, for example, show a band playing the music, and the audio track would then contain the audio mix for that band with the guitar track omitted. The VideoForWindows product that runs under Windows 3.1 has an API (Application Program Interface) that enables the user to initiate and control the running of these Video-audio files from a C program.

The pseudocode for the main loop of the control program is shown in FIG. **5**. The main program begins execution by first performing system initialization (step **100**) and then calling a register$_{13}$midi$_{13}$callback() routine that installs a new interrupt service routine for the MIDI interface card (step **102**). The installed interrupt service effectively "creates" the virtual guitar. The program then enters a while-loop (step **104**) in which it first asks the user to identify the song which will be played (step **106**). It does this by calling a get$_{13}$song$_{13}$id$_{13}$from$_{13}$user() routine. After the user makes his selection using for example a keyboard **26** (see FIG. 1) to select among a set of choices that are displayed on video monitor **10**, the user's selection is stored in a song$_{13}$id variable that will be used as the argument of the next three routines which the main loop calls. Prior to beginning the song, the program calls a set$_{13}$up$_{13}$data$_{13}$structures() routine that sets up the data structures to hold the contents of the song data file that was selected (step **108**). The three data structures that will hod the song data are sframes[], lnote$_{13}$array[], and hnotes$_{13}$array[].

During this phase of operation, the program also sets up a timer resource on the PC that maintains a clock variable that is incremented every millisecond and it resets the millisecond clock variable to 0. As will become more apparent in the following description, the clock variable serves to determine the user's general location within the song and thereby identify which notes the user will be permitted to activate through his instrument. The program also sets both a current$_{13}$frame$_{13}$idx variable and a current$_{13}$lead$_{13}$note$_{13}$idx variable to 0. The current$_{13}$frame$_{13}$idx variable, which is used by the installed interrupt routine, identifies the frame of the song that is currently being played. The current$_{13}$lead$_{13}$note$_{13}$idx variable identifies the particular note within the lead$_{13}$note array that is played in response to a next activation signal from the user.

Next, the program calls another routine, namely, initialize$_{13}$data$_{13}$structures(), that retrieves a stored file image of the Virtual Guitar data for the chosen song from the hard disk and loads that data into the three previously mentioned arrays (step **110**). After the data structures have been initialized, the program calls a play$_{13}$song() routine that causes PC **2** to play the selected song (step **112**).

Referring to FIG. **6**, when play$_{13}$song() is called, it first instructs the user graphically that it is about to start the song (optional) (step **130**). Next, it calls another routine, namely, wait$_{13}$for$_{13}$user$_{13}$start$_{13}$signal(), which forces a pause until the user supplies a command which starts the song (step **132**). As soon as the user supplies the start command, the play$_{13}$song routine starts the simultaneous playback of the stored accompaniment, i.e., the synchronized audio and video tracks on CD-ROM player **6** (step **134**). In the described embodiment, this is an interleaved audio/video (.avi) file that is stored on a CD-ROM. It could, of course, be available in a number of different forms including, for example, a .WAV digitized audio file or a Red Book Audio track on the CD-ROM peripheral.

Since the routines are "synchronous" (i.e. do not return until playback is complete), the program waits for the return of the Windows Operating System call to initiate these playbacks. Once the playback has been started, every time a MIDI event occurs on the MIDI guitar (i.e., each time a string is struck), the installed MIDI interrupt service routine processes that event. In general, the interrupt service routine calculates what virtual guitar action the real MIDI guitar event maps to.

Before examining in greater detail the data structures that are set up during initialization, it is useful first to describe the song data file and how it is organized. The song data file contains all of the notes of the guitar track in the sequence in which they are to be played. As illustrated by FIG. **3**, which shows a short segment of a hypothetical score, the song data is partitioned into a sequence of frames **200**, each one typically containing more than one and frequently many notes or chords of the song. Each frame has a start time and an end time, which locate the frame within the music that will be played. The start time of any given frame is equal to the end time of the previous frame plus 1 millisecond. In FIG. **3**, the first frame extends from time 0 to time 6210 (i.e., 0 to 6.21 seconds) and the next frame extends from 6211 to 13230 (i.e., 6.211 to 13.23 seconds). The remainder of the song data file is organized in a similar manner.

In accordance with the invention, the guitar player is able to "play" or generate only those notes that are within the "current" frame. The current frame is that frame whose start time and end time brackets the current time, i.e., the time that has elapsed since the song began. Within the current frame, the guitar player can play any number of the notes that are present but only in the order in which they appear in the frame. The pace at which they are played or generated within the time period associated with the current frame is completely determined by the user. In addition, the user by controlling the number of string activations also controls both the number of notes of a chord that are generated and the number of notes within the frame that actually get generated. Thus, for example, the player can play any desired number of notes of a chord in a frame by activating only that number of strings, i.e., by strumming the guitar. If the player does not play the guitar during a period associated with a given frame, then none of the music within that frame will be generated. The next time the user strikes or activates a string, then the notes of a later frame, i.e., the new current frame, will be generated.

Note that the pitch of the sound that is generated is determined solely by information that is stored the data structures containing the song data. The guitar player needs only activate the strings. The frequency at which the string vibrates has no effect on the sound generated by the virtual music system. That is, the player need not fret the strings while paying in order to produce the appropriate sounds.

It should be noted that the decision about where to place the frame boundaries within the song image is a somewhat subjective decision, which depends upon the desired sound effect and flexibility that is given to the user. There are undoubtedly many ways to make these decisions. Chord changes could, for example, be used as a guide for where to place frame boundaries. Much of the choice should be left to the discretion of the music arranger who builds the database. As a rule of thumb, however, the frames should probably not be so long that the music when played with the virtual instrument can get far out of alignment with the accompaniment and they should not be so short that the performer has no real flexibility to modify or experiment with the music within a frame.

For the described embodiment, an ASCI editor was used to create a text based file containing the song data. Generation of the song data file can, of course, be done in many other ways. For example, one could produce the song data file by first capturing the song information off of a MIDI instrument that is being played and later add frame delimiters in to that set of data.

With this overview in mind, we now turn to a description of the previously mentioned data structures, which are shown in FIG. **4**. The sframes[]array **200**, which represents the sequence of frames for the entire song, is an array of synch$_{13}$frame data structures, one of which is shown in FIG. **8**. Each synch$_{13}$frame data structure contains a frame$_{13}$start$_{13}$time variable that identifies the start time for the frame, a frame—end—time variable that identifies the send time of the frame and a lnote$_{13}$idx variable that provides an index into both a lnote$_{13}$array[] data structure **220** and an hnotes$_{13}$array[] data structure **240**.

The lnote$_{13}$array[] **220** is an array of lead$_{13}$note data structures, one of which is shown in FIG. **9**. The lnote$_{13}$array[] **220** represents a sequence of single notes (referred to as "lead notes") for the entire song in the order in which they are played. Each lead$_{13}$note data structure represents a singly lead note and contains two entries, namely, a lead—note variable that identifies the pitch of the corresponding lead note, and a time variable, which precisely locates the time at which the note is supposed to be played in the song. If a single note is to be played at some given time, then that note is the lead note. If a chord is to be played at some given time, then the lead note is one of the notes of that chord and hnote$_{13}$array[] data structure **240** identifies the other notes of the chord. Any convention can be used to select which note of the chord will be the lead note. In the described embodiment, the lead note is the chord note with the highest pitch.

The hnote$_{13}$array[] data structure **240** is an array of harmony$_{13}$note data structures, one of which is shown in FIG. **10**. The lnote$_{13}$idx variable is an index into this array. Each harmony$_{13}$note data structure contains an hnote$_{13}$cnt variable and an hnotes[] array of size **10**. The hnotes[]array specifies the other notes that are to be played with the corresponding lead note, i.e., the other notes in the chord. If the lead note is not part of a chord, the hnotes[] array is empty (i.e., its entries are all set to NULL). The hnote$_{13}$cnt variable identifies the number of non-null entries in the associated hnotes[] array. Thus, for example, if a single note is to be played (i.e., it s not part of a chord), the hnote$_{13}$cnt variable in the harmony$_{13}$note data structure for that lead note will be set equal to zero and all of the entries of the associated hnotes[] array will be set to NULL.

As the player hits strings on the virtual guitar, the Callback routine which will be described in greater detail in next section is called for each event. After computing the harmonic frame, chord index and sub-chord index, this callback routine instructs the Proteus Synthesis chip in PC * to create a tone of the pitch that corresponds to the given frame, chord, sub-chord index. The volume of that tone will be based on the MIDI velocity parameter received with the note data from the MIDI guitar.

Virtual Instrument Mapping

FIGS. 7A and 7B show pseudocode for the MIDI interrupt callback routine, i.e., virtual$_{13}$guitar$_{13}$callback(). When invoked the routine invokes a get$_{13}$current$_{13}$time() routine which uses the timer resource to obtain the current time (step **200**). It also calls another routine, i.e., get$_{13}$guitar$_{13}$string$_{13}$event(&string$_{13}$id, &string$_{13}$velocity), to identify the event that was generated by the MIDI guitar (step **202**). This returns the following information: (1) the type of event (i.e., ON, OFF, or TREMELO control); (2) on which string the event occurred (i.e. string$_{13}$id); and (3) if an ON event, with what velocity the string was struck (i.e. string$_{13}$velocity).

The interrupt routine contains a switch instruction which runs the code that is appropriate for the event that was generated (step **204**). In general, the interrupt handler maps the MIDI guitar events to the tone generation of the Proteus Synthesis chip. Generally, the logic can be summarized as follows:

If an ON STRING EVENT has occurred, the program checks whether the current time matches the current frame (**210**). This is done by checking the timer resource to determine how much time on the millisecond clock has elapsed since the start of the playback of the Video/Audio file. As noted above, each frame is defined as having a start time and an end time. If the elapsed time since the start of playback falls between these two times for a particular frame then that frame is the correct frame for the given time (i.e., it is the current frame). If the elapsed time falls outside of the time period of a selected frame, then it is not the current frame but some later frame is.

If the current time does not match the current frame, then the routine moves to the correct frame by setting a frame variable i.e., current$_{13}$frame$_{13}$idx, to the number of the frame whose start and end times bracket the current time (step **212**). The current$_{13}$frame$_{13}$idx variable serves as an index into the sframe$_{13}$array. Since no notes of the new frame have yet been generated, the event which is being processed maps to the first lead note in the new frame. Thus, the routine gets the first lead note of that new frame and instructs the synthesizer chip to generate the corresponding sound (step **214**). The routine which performs this function is start$_{13}$tone$_{13}$gen() in FIG. **7**A and its arguments include the string$_{13}$velocity and string$_{13}$id from the MIDI formatted data as well as the identity of the note from the lnotes$_{13}$array. Before exiting the switch statement, the program sets the current$_{13}$lead$_{13}$note$_{13}$idx to identify the current lead note (step **215**) and it initializes an hnotes$_{13}$played variable to zero (step **216**). The hnotes$_{13}$played variable determines which note of a chord is to be generated in response to a next event that occurs sufficiently close in time to the last event to qualify as being part of a chord.

In the case that the frame identified by the current$_{13}$frame$_{13}$idx variable is not the current frame (step **218**), then the interrupt routine checks whether a computed difference between the current time and the time of the last ON event, as recorded in a last time variable, is greater than a preselected threshold as specified by a SIMULTAN$_{13}$THRESHOLD variable (steps **220** and **222**). In the described embodiment, the preselected time is set to be of sufficient length (e.g on the order of about **20** milliseconds) so as to distinguish between events within a chord (i.e., approximately simultaneous events) and events that are part of different chords.

If the computed time difference is shorter than the preselected threshold, the string ON event is treated as part of a "strum" or "simultaneous" grouping that includes the last lead note that was used. In this case, the interrupt routine, using the lnote$_{13}$idx index, finds the appropriate block in the harmony$_{13}$notes array and, using the value of the

hnotes$_{13}$played variable, finds the relevant entry in h$_{13}$notes array of that block. It then passes the following information to the synthesizer (step **224**):

```
string₁₃velocity
string₁₃id
hnotes₁₃array[current₁₃lead₁₃note₁₃idx]        .hnotes
    [hnotes₁₃played++]
```

which causes the synthesizer to generate the appropriate sound for that harmony note. Note that the hnotes$_{13}$played variable is also incremented so that the next ON event, assuming it occurs within a preselected time of the last ON event, accesses the next note in the hnote[] array.

If the computed time difference is longer than the preselected threshold, the string event is not treated as part of a chord which contained the previous ON event; rather it is mapped to the next lead note in the lead$_{13}$note array. The interrupt routine sets the current$_{13}$lead$_{13}$note$_{13}$idx index to the next lead note in the lead—note array and starts the generation of that tone (step **226**). It also resets the hnotes$_{13}$played variable to 0 in preparation for accessing the harmony notes associated with that lead note, if any (step **228**).

If the MIDI guitar event is an OFF STRING EVENT, then the interrupt routine calls an unsound$_{13}$note() routine which turns off the sound generation for that string (step **230**). It obtains the string$_{13}$id from the MIDI event packet reporting the OFF event and passes this to the unsound$_{13}$note() routine. The unsound$_{13}$note routine then looks up what tone is being generated for the ON Event that must have preceded this OFF event on the identified string and turns off the tone generation for that string.

If the MIDI guitar event is a TREMELO event, the tremelo information from the MIDI guitar gets passed directly to synthesizer chip which produces the appropriate tremelo (step **232**).

In an alternative embodiment which implements what will be referred to as "rhythm EKG", the computer is programmed to display visual feedback to the user on video monitor **10**. In general, the display of the rhythm EKG includes two components, namely, a trace of the beat that is supposed to be generated by the player (i.e., the "song EKG") and a trace of the beat that is actually generated by the player (i.e., the "player EKG"). The traces, which can be turned on and off at the option of the player, are designed to teach the player how to play the song, without having the threatening appearance of a "teaching machine". As a teaching tool, the rhythm EKG is applicable to both rhythm and lead guitar playing.

Referring to FIG. **11**, the main display of the "song EKG" which is meant to evoke the feeling of a monitored signal from a patient. The displayed image includes a grid **300**, a rhythm or song trace **302** and a cursor **304**. On grid **300**, the horizontal axis corresponds to a time axis and the vertical axis corresponds to an event axis (e.g. the playing of a note or chord) but has no units of measure. The song trace **302** includes pulses **306** (i.e., a series of beats) which identify the times at which the player is supposed to generate notes or strums with the instrument. The program causes cursor **304** to move from left to right as the music plays thereby marking the real time that has elapsed since the beginning of the song, i.e., indicating where the player is supposed to be within the song. Cursor **304** passes the start of each beat just as the player is supposed to be starting the chord associated with that beat and it passes the peak of each beat just as the player is supposed to be finishing the chord.

To implement this feature, the program can use the time stamp that is supplied for each of the lead notes of the song (see FIG. 9). The time stamp for each lead note identifies the time at which the note is supposed to be played in the song. Alternatively, one can reduce the frame size to one note and use the beginning and ending time of each frame as the indicator of when to generate a pulse.

The program also includes two display modes, namely, a directionality mode and a volume mode, which are independent of each other so the player can turn on either or both of them.

Referring to FIG. 12, if the player optionally turns on the directionality mode, the beats are displayed in the negative direction when the player is supposed to be strumming down and in the positive direction when the player is supposed to be strumming up. The directionality information can be supplied in any of a number of ways. For example, it can be extracted from the direction of frequency change between the lead note and its associated harmony notes or it can be supplied by information added to the lead note data structure.

Referring to FIG. 13, if the player optionally turns on the volume mode, the size of the beats on the display indicates the vigor with which the player should be strumming. A real "power chord" could be indicated by a pulse that goes offscale, i.e. the top of the pulse gets flattened. To implement this feature, volume information must be added to the data structure for either the lead notes or the harmony notes.

The player EKG, which is shown as trace 310 in FIG. 14, looks identical to the song EKG, and when it is turned on, cursor 304 extends down to cover both traces. The player EKG shows what the player is actually doing. Like the song EKG it too has optional directionality and volume modes.

In the described embodiment, the program color codes the trace of the player EKG to indicate how close the player is to the song EKG. Each pulse is color coded to score the players performance. A green trace indicates that the player is pretty close; a red trace indicates that the player is pretty far off; and a yellow trace indicates values in between. A simple algorithm for implementing this color coded feedback uses a scoring algorithm based upon the function shown in FIG. 15. If the player generates the note or chord within ±30 msec of when it is supposed to be generated, a score of 100 is generated. The score for delays beyond that decreases linearly from 100 to zero at ±T, where T is about 100 msec . The value of T can be adjusted to set the difficulty level.

The algorithm for color coding the trace also implements a low pass filter to slow down the rate at which the colors are permitted to change and thereby produce a more visually pleasing result. Without the low pass filter, the color can change as frequently as the pulses appear.

It should be understood that the rhythm EKG can be used as part of the embodiment which also includes the previously described frame synchronization technique or by itself. In either event, it provides very effective visual feedback which assists the user in learning how to play the instrument.

Having thus described illustrative embodiments of the invention, it will be apparent that various alterations, modifications and improvements will readily occur to those skilled in the art. Such obvious alterations, modifications and improvements, though not expressly described above, are nonetheless intended to be implied and are within the spirit and scope of the invention. Accordingly, the foregoing discussion is intended to be illustrative only, and not limiting; the invention is limited and defined only by the following claims and equivalents thereto.

What is claimed is:

1. A musical instrument comprising:

an actuator which generates a plurality of actuation signals in response to being played by a user;

an audio synthesizer which generates audio tones in response to control signals;

a memory storing a musical score, said stored musical score comprising a sequence of lead notes each of which has an associated time stamp to identify a time at which it is supposed to be played by said user in said musical score;

a video display unit;

a digital processing means controlling said audio synthesizer and said video display unit,

said digital processing means receiving said plurality of actuation signals from said actuator and generating a sequence of control signals therefrom,

said digital processing means programmed to map the plurality of actuation signals from said actuator to a corresponding sub-sequence of lead notes from among said sequence of lead notes,

said digital processing means programmed to produce the sequence of control signals from the sub-sequence of lead notes, said sequence of control signals causing said synthesizer to generate sounds representing the sub-sequence of lead notes,

said digital processing means programmed to display on said video display unit a trace of markers as a function of time, wherein each of the markers within said trace of markers indicates a time at which the user is supposed to cause said actuator to generate one of the actuation signals of said plurality of actuation signals in order to cause the audio synthesizer to play a corresponding one of the sequence of lead notes of said musical score, said trace of markers representing a period of time extending from before an actual elapsed time until after the actual elapsed time, the actual elapsed time being measured from a start of the musical score, and

said digital processing means programmed to display on said video display unit an indicator marking a location of the actual elapsed time within said trace of markers and thereby indicating where the user is presently supposed to be within the musical score.

2. The musical instrument of claim 1 wherein said digital processing means is also programmed to generate on said video display a second trace next to said trace of markers indicating when the user actually caused said actuator to generate each of the actuation signals of said plurality of actuation signals and thereby indicating when the lead notes of said sub-sequence of lead notes are actually played by said synthesizer relative to when they are supposed to be played as indicated by said trace of markers.

3. The musical instrument of claim 1 wherein said trace of markers is a sequence of pulses each of which corresponds in time to when the user is supposed to cause said actuator to generate one of the actuation signals of said plurality of actuation signals so as to cause said synthesizer to play an associated lead note.

4. The musical instrument of claim 3 wherein the pulses of said sequence of pulses vary in amplitude and wherein the amplitude of any given pulse indicates a relative intensity with which the user should play an associated lead note on said actuator.

5. The musical instrument of claim 3 wherein said actuator is a multi-element actuator and said sequence of pulses

includes pulses having positive polarity and pulses having negative polarity, the polarity indicating a direction in which a chord is to be played on said multi-element actuator.

6. The musical instrument of claim **2** wherein said second trace is a sequence of pulses each of which corresponds in time to when the user actually caused said actuator to generate the actuation signals of said plurality of actuation signals.

7. The musical instrument of claim **2** wherein said trace of markers is a sequence of pulses each of which corresponds in time to when the user is supposed to cause said actuator to generate one of the actuation signals of said plurality of actuation signals so as to cause said synthesizer to play an associated lead note.

8. The musical instrument of claim **7** wherein the pulses of said sequence of pulses vary in amplitude and wherein the amplitude of any given pulse indicates a relative intensity with which the user should play an associated lead note on said actuator.

9. The musical instrument of claim **7** wherein said actuator is a multi-element actuator and said sequence of pulses includes pulses having positive polarity and pulses having negative polarity, the polarity indicating a direction in which a chord is to be played on said multi-element actuator.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297 Page 1 of 9

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Cover Page [73] Assignee, delete "Netwon" and insert --Newton--

Col. 4, line 2, replace "core" with --score-- (our error).

Col. 4, line 19, replace "lnote$_{13}$array[]" with --lnote_array[]--

Col. 4, line 20, replace "hnotes$_{13}$array[]" with --hnotes_array[]--

Col. 4, line 25, replace "play$_{13}$song()" with --play_song()---

Col. 4, line 27, replace "virtual$_{13}$guitar$_{13}$callback()" with --virtual_guitar_callback()---

Col. 4, line 29, replace "sync$_{13}$frame" with --sync_frame--

Col. 4, line 30, replace "lead$_{13}$note" with --lead_note--

Col. 4, line 31, replace "harmony$_{13}$notes" with --harmony_notes--

Col. 5, line 57, replace "MIDI$_{13}$STATUS=On" with --MIDI_STATUS=On--

Col. 5, line 58, replace "MIDI$_{13}$NOTE=<note number)" with --MIDI_NOTE=<note number)--

Col. 5, line 59, replace "MIDI$_{13}$VELOCITY=<amplitude>" with --MIDI_VELOCITY=<amplitude>--

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297 Page 2 of 9

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 5, line 65, replace "MIDI$_{13}$STATUS=Off" with --MIDI_STATUS=Off--

Col. 5, line 66, replace "MIDI$_{13}$NOTE=<note number)" with --MIDI_NOTE=<note number)--

Col. 5, line 67, replace "MIDI$_{13}$VELOCITY=0" with --MIDI_VELOCITY=0--

Col. 6, line 20, replace "register$_{13}$midi$_{13}$callback()" with --register_midi_callback()--

Col. 6, line 26, replace "get$_{13}$song$_{13}$id$_{13}$from$_{13}$user()" with --get_song_id_from_user()--

Col. 6, line 32, replace "set$_{13}$up$_{13}$data$_{13}$structures()" with --set_up_data_structures()--

Col. 6, line 35, replace "hod" with --hold-- (our error)

Col. 6, line 36, replace "lnote$_{13}$array[]" with ---lnote_array[]--

Col. 6, line 36, replace "hnotes$_{13}$array[]" with ---hnotes_array[]--

Col. 6, line 45, replace "current$_{13}$frame$_{13}$idx" with --current_frame_idx--

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

DATED : February 13, 1996

Page 3 of 9

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 6, line 46, replace "current$_{13}$lead$_{13}$not$_{13}$idx" with --current_lead_not_idx--

Col. 6, line 47, replace "current$_{13}$frame$_{13}$idx" with --current_frame_idx--

Col. 6, line 49, replace "current$_{13}$lead$_{13}$note$_{13}$idx" with --current_lead_note_idx--

Col. 6, line 50, replace "lead$_{13}$note" with --lead_note--

Col. 6, line 54, replace "initialize$_{13}$data$_{13}$structures() with --initialize_data_structures()--

Col. 6, line 58, replace "play$_{13}$song()" with --play_song()--

Col. 6, line 60, replace "play$_{13}$song" with --play_song--

Col. 6, line 63, replace "wait$_{13}$for$_{13}$user$_{13}$start$_{13}$signal()" with --wait_for_user_start_signal()--

Col. 6, line 66, replace "play$_{13}$song" with --play_song--

Col. 7, line 59, replace "paying" with --playing-- (our error)

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

DATED : February 13, 1996

Page 4 of 9

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 8, line 17, replace "synch$_{13}$frame" with --synch_frame--

Col. 8, line 18, replace "synch$_{13}$frame" with --synch_frame--

Col. 8, line 19, replace "frame$_{13}$start$_{13}$time" with --frame_start_time--

Col. 8, line 21, replace "lnote$_{13}$idx" with --lnote_idx--

Col. 8, line 22, replace "lnote$_{13}$array[]" with --lnote_array[]--

Col. 8, line 23, replace "hnotes$_{13}$array[]" with --hnotes_array[]--

Col. 8, line 24, replace "lnote$_{13}$array" with --lnote_array--

Col. 8, line 24, replace "lead$_{13}$note" with --lead_note--

Col. 8, line 26, replace "lnote$_{13}$array[]" with --lnote_array[]--

Col. 8, line 28, replace "lead$_{13}$note" with --lead_note--

Col. 8, line 29, replace "singly" with --single-- (our error)

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

Page 5 of 9

DATED : February 23, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 8, line 36, replace "hnote$_{13}$array[]" with --hnote_array[]--

Col. 8, line 41, replace "hnote$_{13}$array[]" with --hnote_array[]--

Col. 8, line 42, replace "harmony$_{13}$note" with --harmony_note--

Col. 8, line 43, replace "lnote$_{13}$idx" with --lnote_idx--

Col. 8, line 44, replace "harmony$_{13}$note" with --harmony_note--

Col. 8, line 44, replace "hnote$_{13}$cnt" with --hnote_cnt--

Col. 8, line 49, replace "hnote$_{13}$cnt" with --hnote_cnt--

Col. 8, line 52, replace "hnote$_{13}$cnt" with --hnote_cnt--

Col. 8, line 52, replace "s" with --is-- (our error)

Col. 8, line 53, replace "harmony$_{13}$note" with --harmony_note--

Col. 8, line 67, replace "virtual$_{13}$guitar$_{13}$callback()" with --virtual_guitar_callback()--

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 9, line 1 replace "get$_{13}$current$_{13}$time()" with --get_current_time()--

Col. 9, line 4, replace "get$_{13}$guitar$_{13}$string$_{13}$event(&string$_{13}$id, &string$_{13}$velocity)" with --get_guitar_string_event(&string_id, &string_velocity-- (both occurrences)

Col. 9, line 8 replace "string$_{13}$id" with --string_id--

Col. 9, line 10, replace "string$_{13}$velocity" with --string_velocity--

Col. 9, line 31, replace "current$_{13}$frame$_{13}$idx" with --current_frame_idx--

Col. 9, line 33, replace "current$_{13}$frame$_{13}$idx" with --current_frame_idx--

Col. 9, line 34, replace "sframe$_{13}$array" with --sframe_array--

Col. 9, line 40, replace "start$_{13}$tone$_{13}$gen()" with --start_tone_gen()-- (both occurrences)

Col. 9, line 41, replace "string$_{13}$velocity" with --string_velocity--

Col. 9, line 41, replace "string$_{13}$id" with --string_id--

Col. 9, line 42, replace "lnotes$_{13}$array" with --lnotes_array--

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297 Page 7 of 9

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 9, line 44, replace "current$_{13}$lead$_{13}$note$_{13}$idx" with --current_lead_note_idx--

Col. 9, line 45, replace "hnotes$_{13}$played" with --hnotes_played--

Col. 9, line 46, replace "hnotes$_{13}$played" with --hnotes_played--

Col. 9, line 51, replace "current$_{13}$frame$_{13}$idx" with --current_frame_idx--

Col. 9, line 54, replace "last time" with --last_time--

Col. 9, line 56, replace "SIMULTAN$_{13}$THRESHOLD" should read --SIMULTAN___ THRESHOLD--.

Col. 9, line 66, replace "lnote$_{13}$idx" with --lnote_idx--

Col. 9, line 67, replace "harmony$_{13}$notes" with --harmony_notes--

Col. 10, line 1, replace "hnotes$_{13}$played" with --hnotes_played---

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

Page 8 of 9

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 10, line 4, replace "string$_{13}$velocity" with --string_velocity--

Col. 10, line 5, replace "string$_{13}$id" with --string_id--

Col. 10, line 6, replace "hnotes$_{13}$array[current$_{13}$lead$_{13}$note$_{13}$idx] with --"hnotes_array[current_lead_note_idx]--

Col. 10, line 7, replace [hnotes$_{13}$played++] with --[hnotes_played++]--

Col. 10, line 9, replace "hnotes$_{13}$played" with --hnotes_played--

Col. 10, line 16, replace "lead$_{13}$note" with --lead_note--

Col. 10, line 17, replace "current$_{13}$lead$_{13}$note$_{13}$idx" with --current_lead_note_idx--

Col. 10, line 20, replace "hnotes$_{13}$played" with --hnotes_played--

Col. 10, line 24, replace "unsound$_{13}$note()" with --unsound_note--

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,491,297

Page 9 of 9

DATED : February 13, 1996

INVENTOR(S) : Johnson, et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 10, line 26, replace "string$_{13}$id" with --string_id--

Col. 10, line 27, replace "unsound$_{13}$note()" with --unsound_note--

Col. 10, line 28, replace "unsound$_{13}$note()" with --unsound_note--

Signed and Sealed this

Twenty-second Day of July, 1997

Attest:

Bruce Lehman

BRUCE LEHMAN

Attesting Officer

Commissioner of Patents and Trademarks

EXHIBIT 4

EXHIBIT 4

US005739457A

United States Patent [19]

Devecka

[11] **Patent Number: 5,739,457**

[45] **Date of Patent: Apr. 14, 1998**

[54] **METHOD AND APPARATUS FOR SIMULATING A JAM SESSION AND INSTRUCTING A USER IN HOW TO PLAY THE DRUMS**

[76] Inventor: **John R. Devecka**, 286 Pershing Rd., Clifton, N.J. 07013

[21] Appl. No.: **720,295**

[22] Filed: **Sep. 26, 1996**

[51] **Int. Cl.**^{6} **G10H 1/32**; G10H 3/00

[52] **U.S. Cl.** .... **84/743**; 84/600; 84/477 R; 434/307 A

[58] **Field of Search** .... 84/600, 651, 667, 84/743, 464 R, 464 A, 477 R; 434/307 A

[56] **References Cited**

U.S. PATENT DOCUMENTS

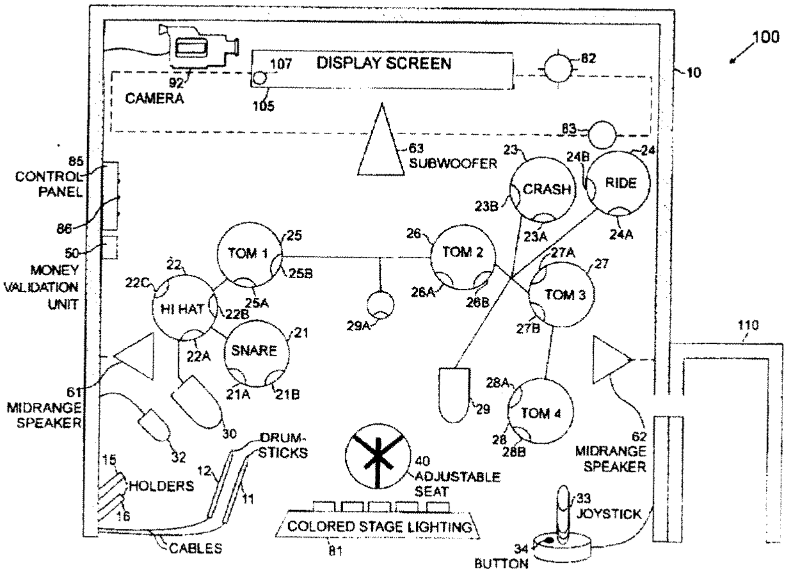
1,613,400	1/1927	McAlleavey .	
3,420,135	1/1969	Wolf	83/478
3,731,582	5/1973	Gullickson	84/478
3,958,487	5/1976	Goldman	84/478
3,990,710	11/1976	Hughes	274/1 R
4,012,979	3/1977	Wemekamp	84/1.01
4,080,867	3/1978	Ratanangsu	84/477 R
4,108,365	8/1978	Hughes	235/419
4,286,495	9/1981	Roof	84/485
4,583,443	4/1986	Senghaas	84/484
4,694,723	9/1987	Shinohara et al.	84/1.03
4,695,903	9/1987	Serap et al.	358/335
4,781,097	11/1988	Uchiyama et al.	84/1.13
4,791,848	12/1988	Blum, Jr.	84/453
4,915,005	4/1990	Shaffer et al.	84/314
4,919,030	4/1990	Perron, III	84/470
4,932,303	6/1990	Kimpara	84/621
4,947,725	8/1990	Nomura	84/723
4,965,673	10/1990	Bozzo et al.	358/335
5,009,146	4/1991	Manabe et al.	84/615
5,027,687	7/1991	Iwamatsu	84/600
5,036,742	8/1991	Youakim	84/411
5,056,403	10/1991	Yamashita	84/723
5,063,821	11/1991	Battle	84/743 X
5,107,743	4/1992	Decker	84/478
5,140,889	8/1992	Segan et al. .	
5,177,313	1/1993	Miyamoto	84/611
5,183,398	2/1993	Monte et al.	434/277
5,206,842	4/1993	Spector	369/4
5,214,231	5/1993	Ernst et al.	84/652
5,218,580	6/1993	Okamura et al. .	
5,223,658	6/1993	Suzuki	84/663
5,266,732	11/1993	Suzuki	84/464 R X
5,286,909	2/1994	Shibukawa	84/609
5,394,784	3/1995	Pierce et al.	84/464 A
5,408,914	4/1995	Breitweiser, Jr. et al.	84/477 R
5,464,946	11/1995	Lewis	84/609
5,481,509	1/1996	Knowles .	
5,484,291	1/1996	Nakai et al.	434/307 A
5,495,786	3/1996	Choi	84/470 R

Primary Examiner—William M. Shoop, Jr.
Assistant Examiner—Jeffrey W. Donels
Attorney, Agent, or Firm—Law Offices of Peter H. Priest

[57] **ABSTRACT**

An interactive electronic drum system and training techniques suitable for use in a coin-operated environment such as an arcade are described. Electronic drum pads, audio speakers, a visual display, training lights and an overall control system are combined to simulate the excitement of a live drum or inactive musical jam session for a user. Positive feedback and, as necessary, instructive aid are provided to make the experience a positive one for both the novice and the expert player. Learning and playing a musical instrument becomes an intuitive, exciting experience and not a boring chore to be endured. Players can simulate the experience of playing in a rock band before a live and appreciative audience. In short, this interactive electronic drum system makes drums and the jamming experience widely accessible to the public.

36 Claims, 9 Drawing Sheets



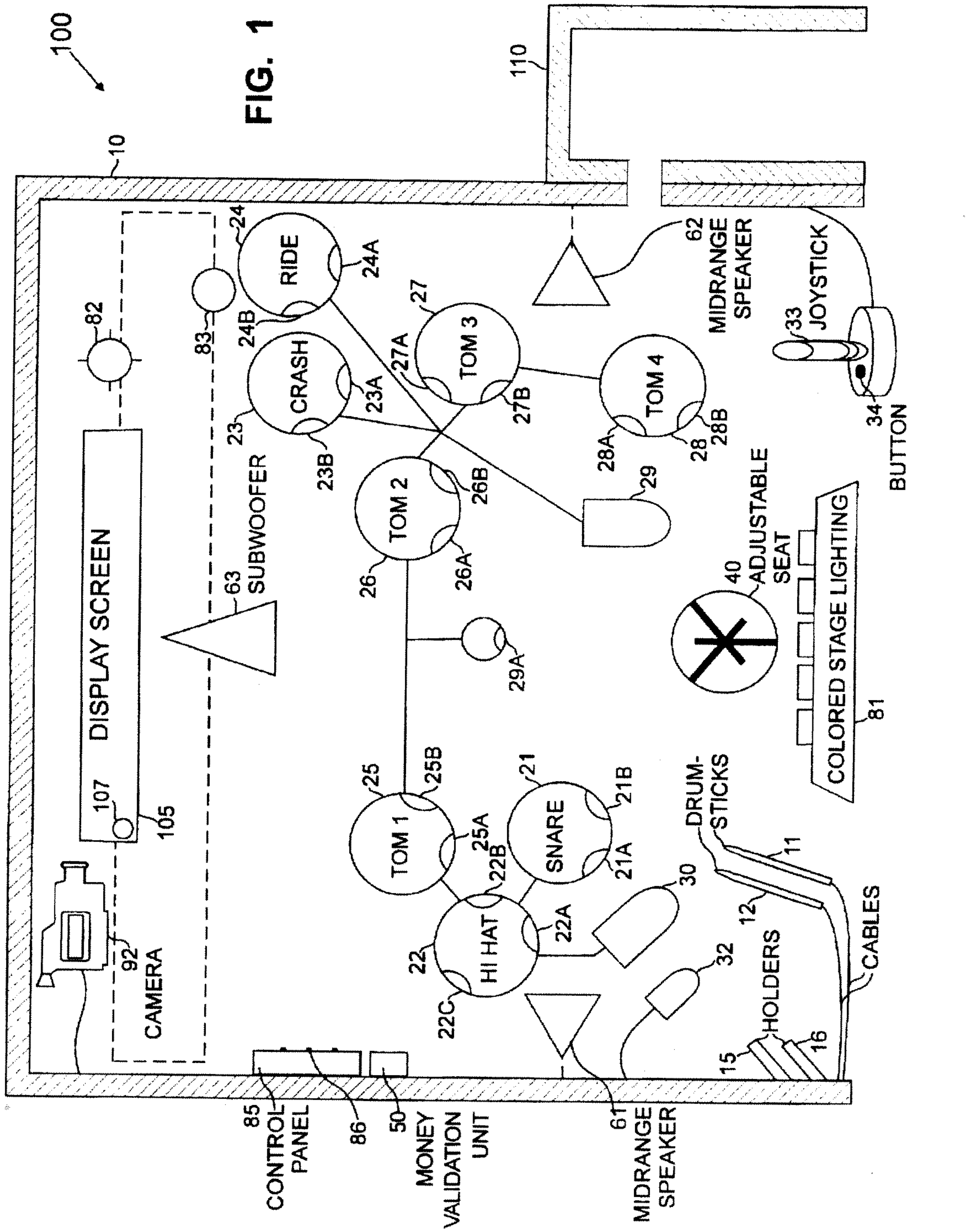

100
10
110
DISPLAY SCREEN
107
105
82
83
CAMERA
92
SUBWOOFER
63
RIDE
24
24A
24B
CRASH
23
23A
23B
TOM 3
27
27A
27B
TOM 4
28
28A
28B
TOM 2
26
26A
26B
29
29A
TOM 1
25
25A
25B
SNARE
21
21A
21B
HI HAT
22
22A
22B
22C
30
32
62
MIDRANGE SPEAKER
JOYSTICK
33
34
BUTTON
40
ADJUSTABLE SEAT
COLORED STAGE LIGHTING
81
DRUM-STICKS
11
12
CABLES
HOLDERS
15
16
85
CONTROL PANEL
86
50
MONEY VALIDATION UNIT
61
MIDRANGE SPEAKER

FIG. 1

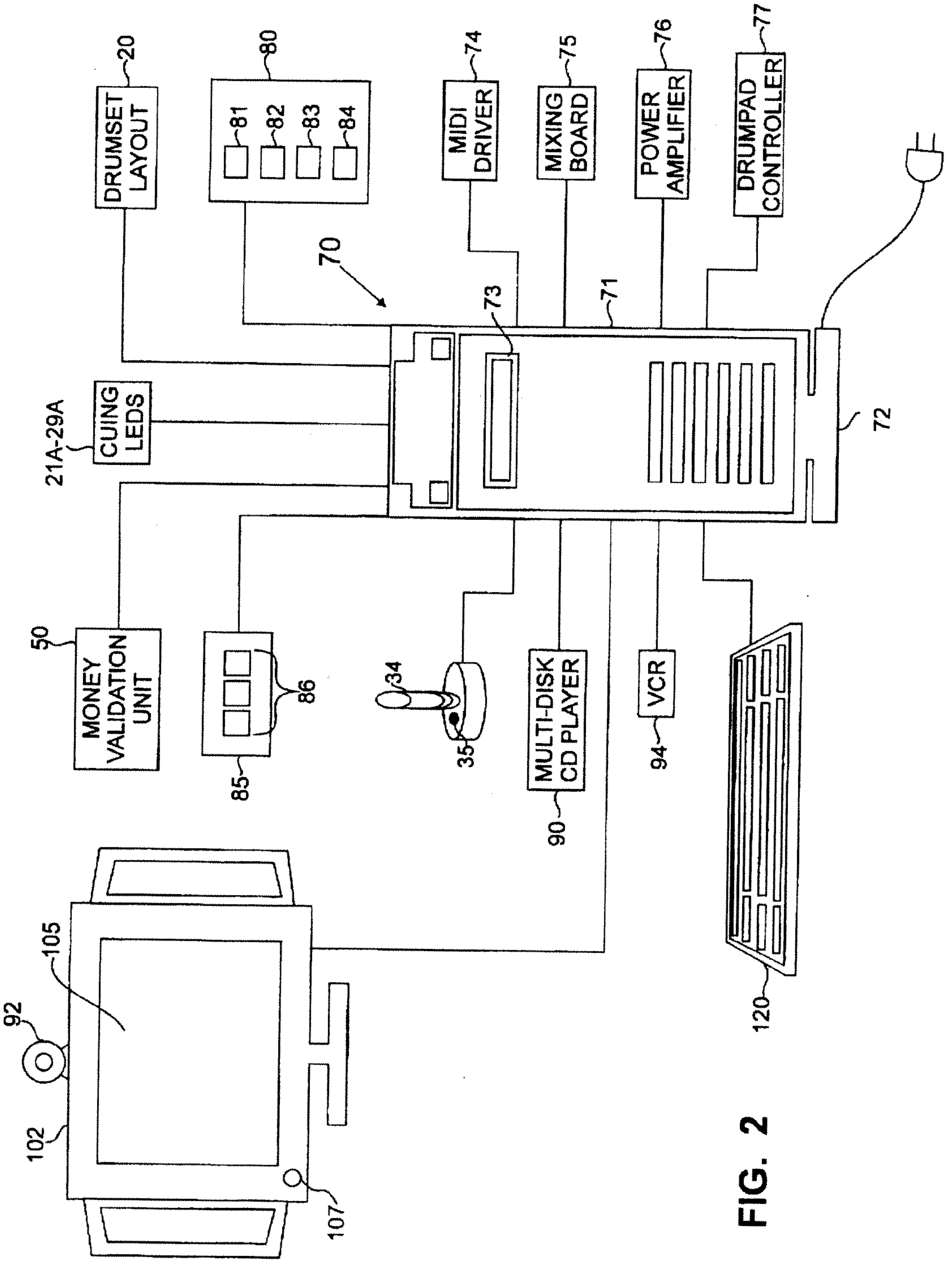

DRUMSET LAYOUT
20
80
81
82
83
84
MIDI DRIVER
74
MIXING BOARD
75
POWER AMPLIFIER
76
DRUMPAD CONTROLLER
77
70
73
71
72
CUING LEDS
21A-29A
MONEY VALIDATION UNIT
50
85
86
34
35
MULTI-DISK CD PLAYER
90
VCR
94
120
92
105
102
107

FIG. 2

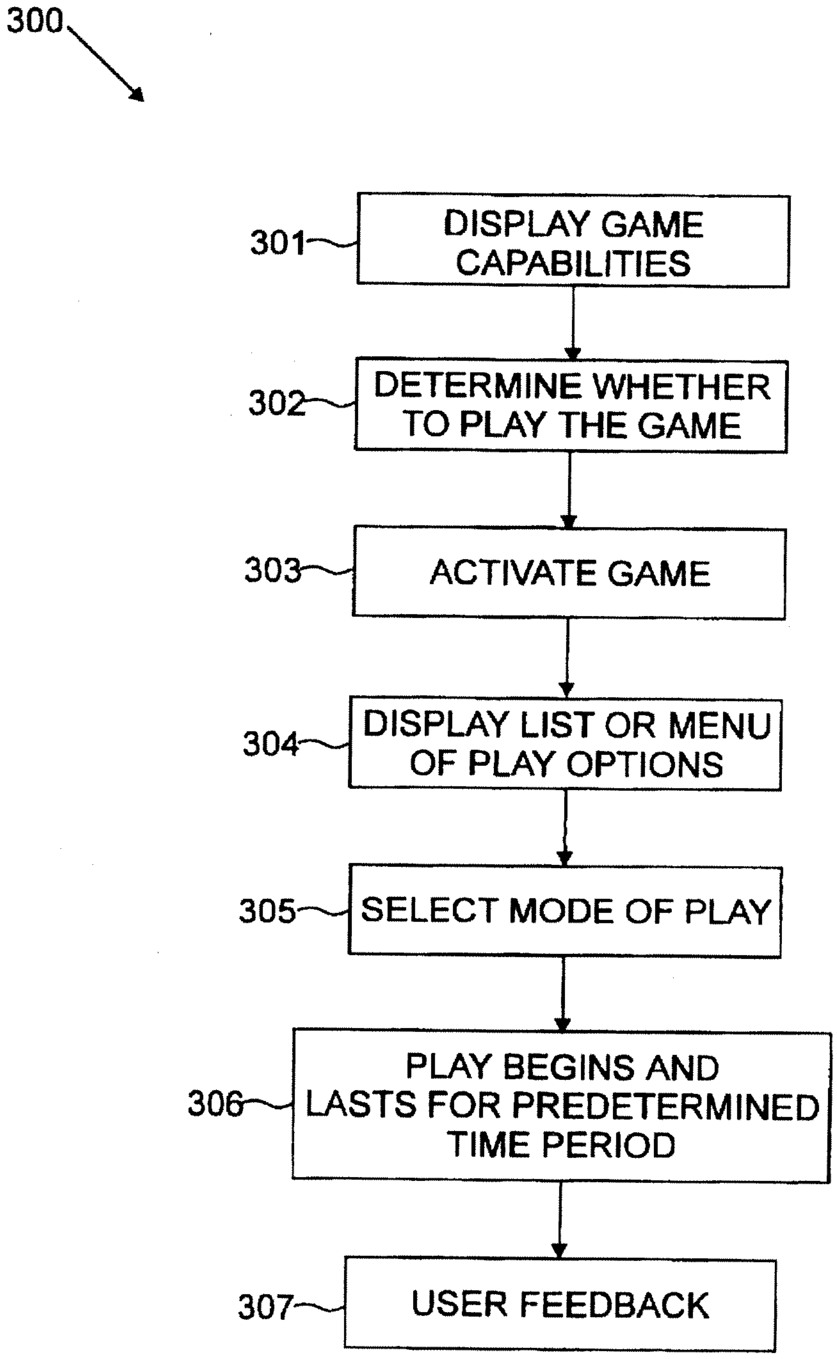

300
301
DISPLAY GAME CAPABILITIES
302
DETERMINE WHETHER TO PLAY THE GAME
303
ACTIVATE GAME
304
DISPLAY LIST OR MENU OF PLAY OPTIONS
305
SELECT MODE OF PLAY
306
PLAY BEGINS AND LASTS FOR PREDETERMINED TIME PERIOD
307
USER FEEDBACK

FIG. 3

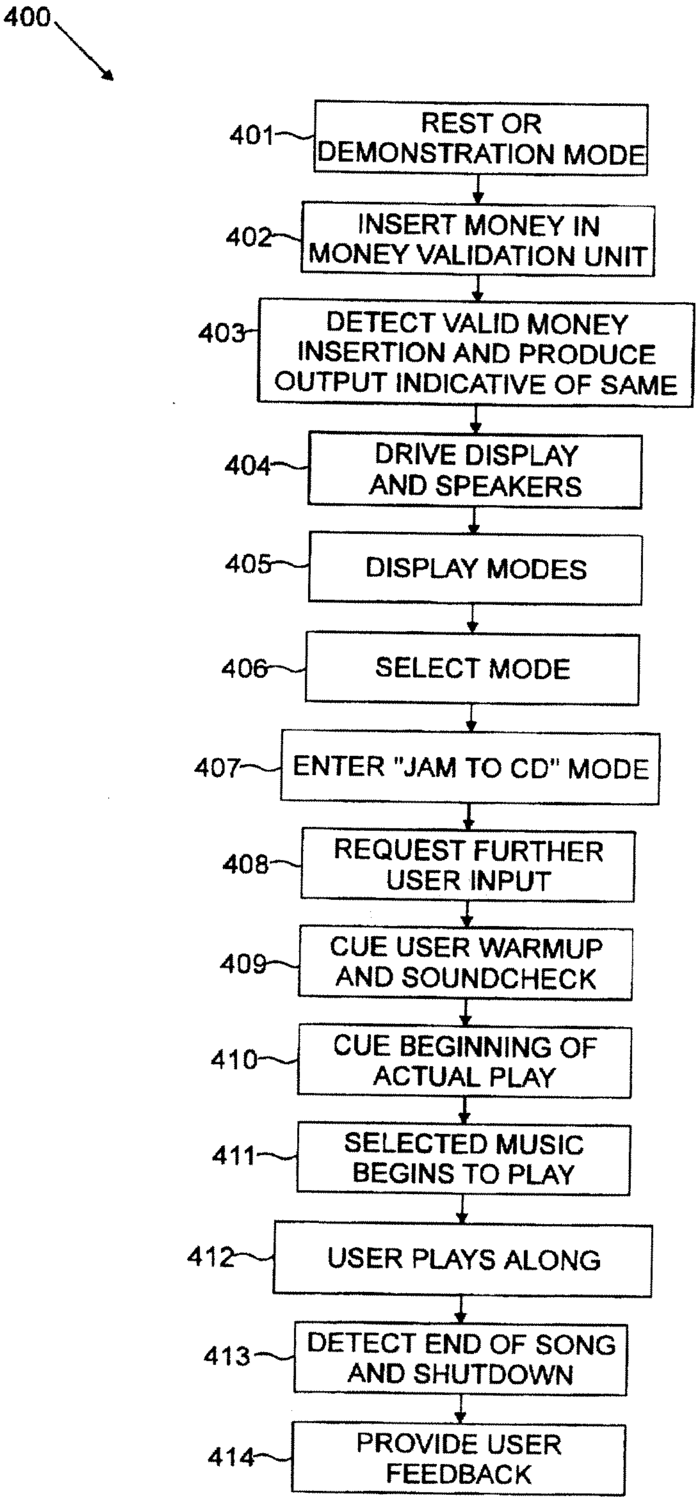

400
401
REST OR DEMONSTRATION MODE
402
INSERT MONEY IN MONEY VALIDATION UNIT
403
DETECT VALID MONEY INSERTION AND PRODUCE OUTPUT INDICATIVE OF SAME
404
DRIVE DISPLAY AND SPEAKERS
405
DISPLAY MODES
406
SELECT MODE
407
ENTER "JAM TO CD" MODE
408
REQUEST FURTHER USER INPUT
409
CUE USER WARMUP AND SOUNDCHECK
410
CUE BEGINNING OF ACTUAL PLAY
411
SELECTED MUSIC BEGINS TO PLAY
412
USER PLAYS ALONG
413
DETECT END OF SONG AND SHUTDOWN
414
PROVIDE USER FEEDBACK

FIG. 4

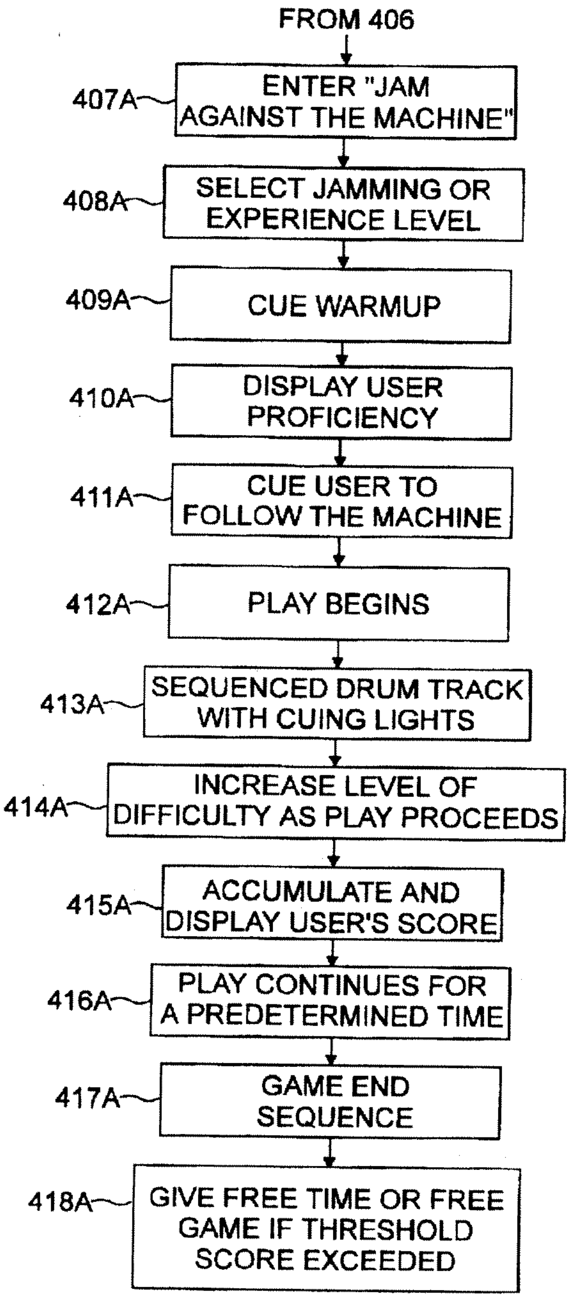

FROM 406
407A
ENTER "JAM AGAINST THE MACHINE"
408A
SELECT JAMMING OR EXPERIENCE LEVEL
409A
CUE WARMUP
410A
DISPLAY USER PROFICIENCY
411A
CUE USER TO FOLLOW THE MACHINE
412A
PLAY BEGINS
413A
SEQUENCED DRUM TRACK WITH CUING LIGHTS
414A
INCREASE LEVEL OF DIFFICULTY AS PLAY PROCEEDS
415A
ACCUMULATE AND DISPLAY USER'S SCORE
416A
PLAY CONTINUES FOR A PREDETERMINED TIME
417A
GAME END SEQUENCE
418A
GIVE FREE TIME OR FREE GAME IF THRESHOLD SCORE EXCEEDED

FIG. 4A

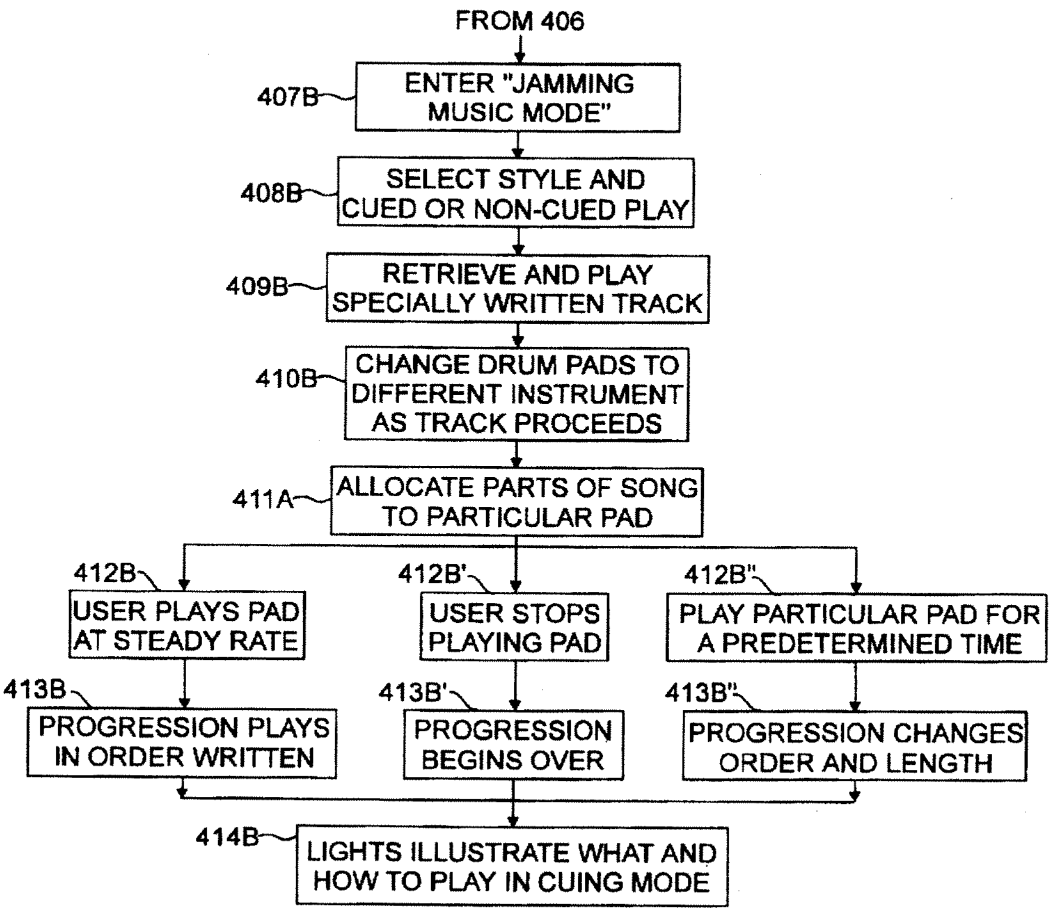

FROM 406
407B
ENTER "JAMMING MUSIC MODE"
408B
SELECT STYLE AND CUED OR NON-CUED PLAY
409B
RETRIEVE AND PLAY SPECIALLY WRITTEN TRACK
410B
CHANGE DRUM PADS TO DIFFERENT INSTRUMENT AS TRACK PROCEEDS
411A
ALLOCATE PARTS OF SONG TO PARTICULAR PAD
412B
USER PLAYS PAD AT STEADY RATE
412B'
USER STOPS PLAYING PAD
412B"
PLAY PARTICULAR PAD FOR A PREDETERMINED TIME
413B
PROGRESSION PLAYS IN ORDER WRITTEN
413B'
PROGRESSION BEGINS OVER
413B"
PROGRESSION CHANGES ORDER AND LENGTH
414B
LIGHTS ILLUSTRATE WHAT AND HOW TO PLAY IN CUING MODE

FIG. 4B

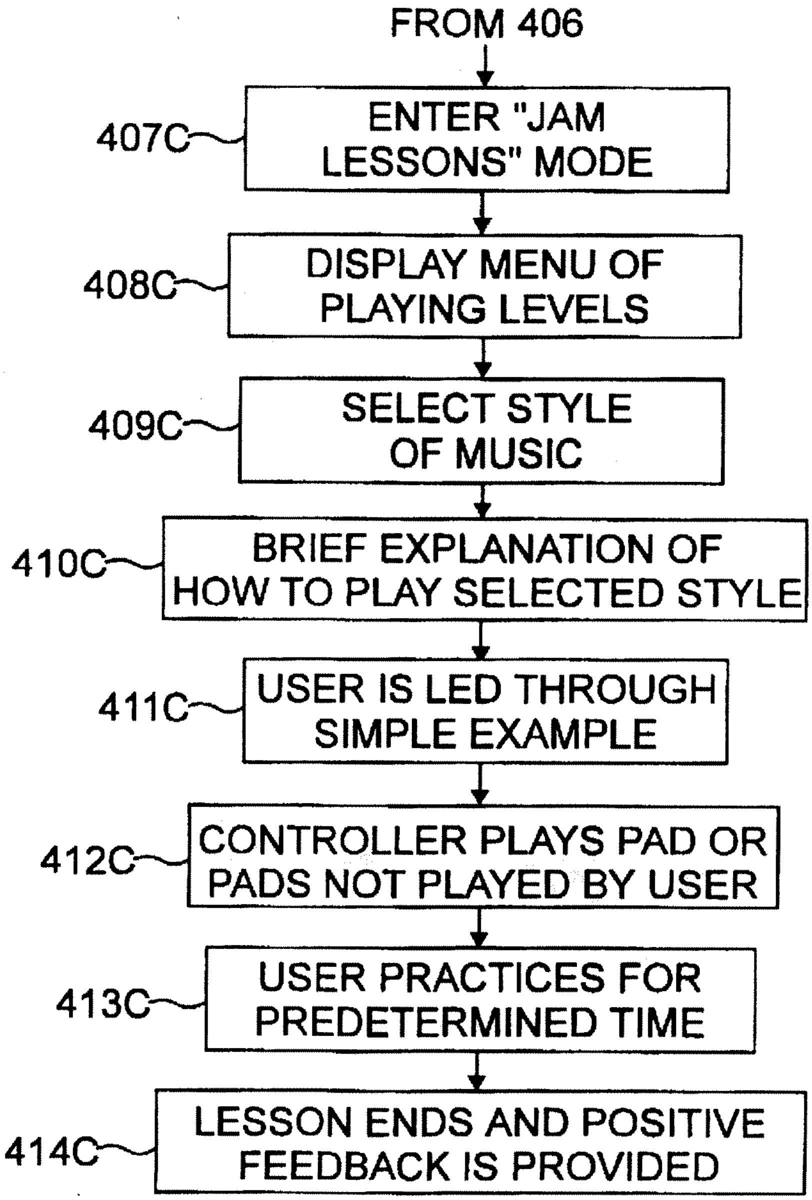
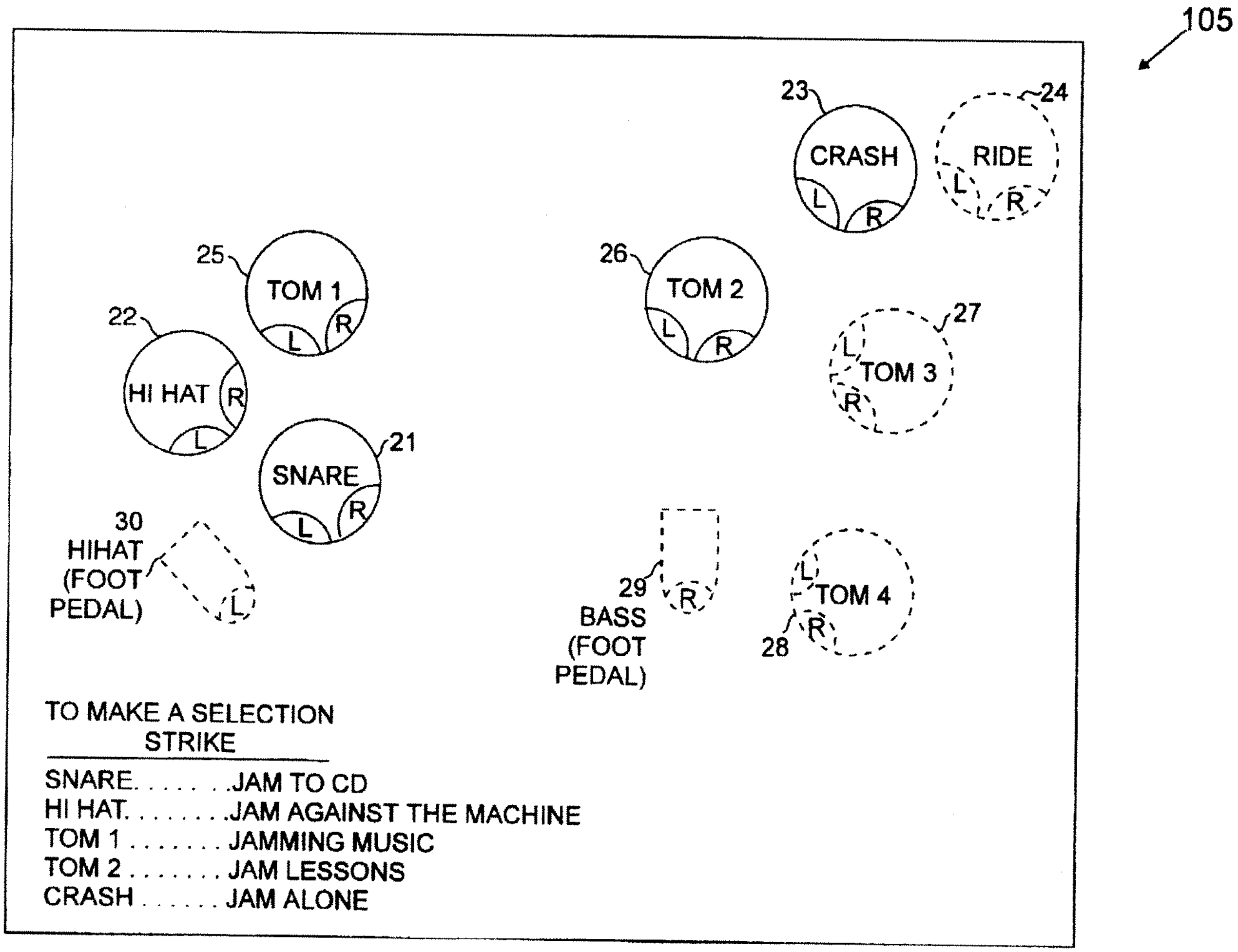

FROM 406
407C
ENTER "JAM LESSONS" MODE
408C
DISPLAY MENU OF PLAYING LEVELS
409C
SELECT STYLE OF MUSIC
410C
BRIEF EXPLANATION OF HOW TO PLAY SELECTED STYLE
411C
USER IS LED THROUGH SIMPLE EXAMPLE
412C
CONTROLLER PLAYS PAD OR PADS NOT PLAYED BY USER
413C
USER PRACTICES FOR PREDETERMINED TIME
414C
LESSON ENDS AND POSITIVE FEEDBACK IS PROVIDED

FIG. 4C



TO MAKE A SELECTION
STRIKE

SNARE. . . . . . .JAM TO CD
HI HAT. . . . . . . .JAM AGAINST THE MACHINE
TOM 1 . . . . JAMMING MUSIC
TOM 2 . . . . JAM LESSONS
CRASH . . . . . . JAM ALONE

FIG. 5

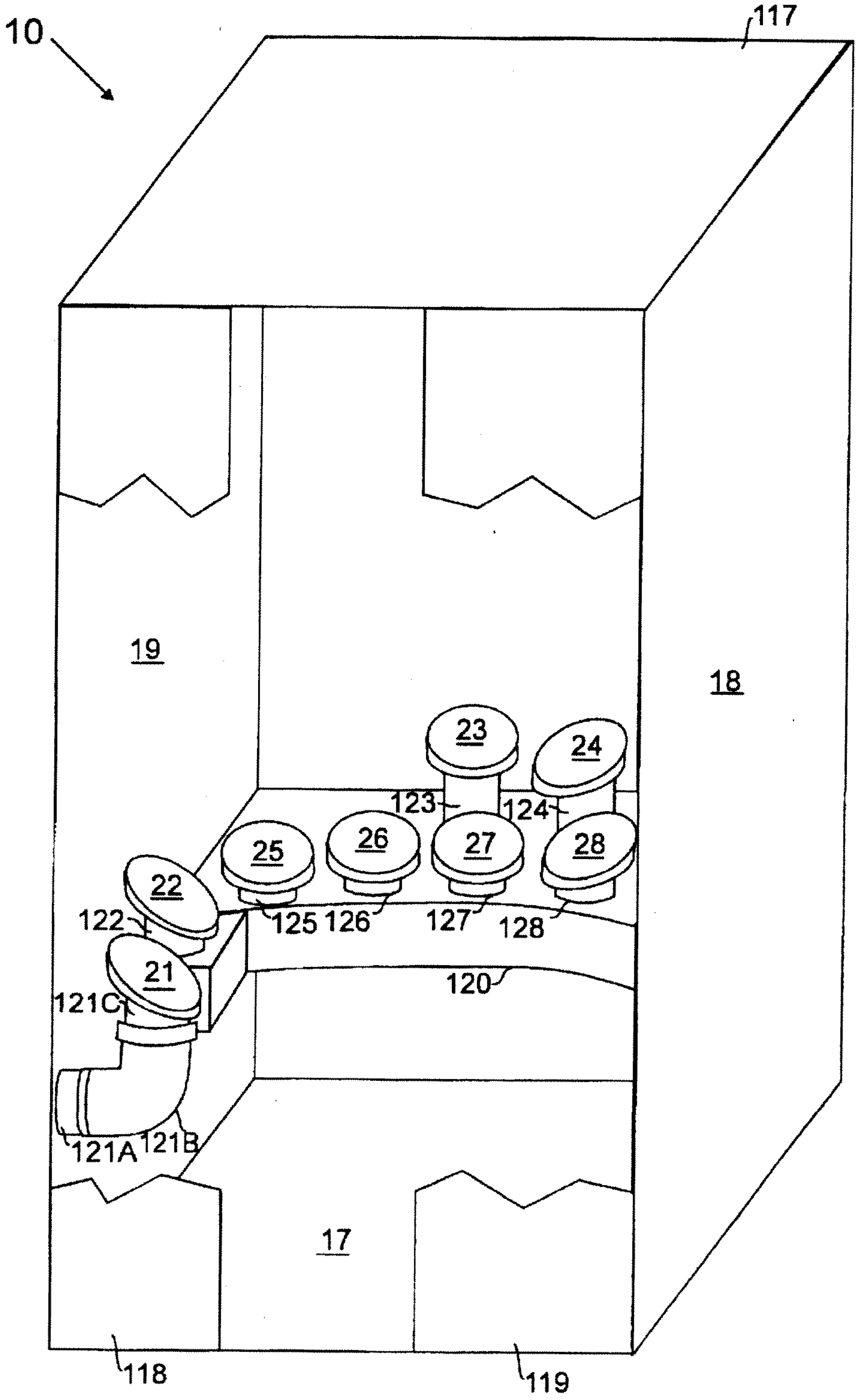

10
117
19
18
23
24
123
124
25
26
27
28
22
125
126
127
128
122
21
121C
120
121A
121B
17
118
119

FIG. 6

METHOD AND APPARATUS FOR SIMULATING A JAM SESSION AND INSTRUCTING A USER IN HOW TO PLAY THE DRUMS

The present invention relates generally to methods and apparatus for providing an exciting interactive audio visual musical experience which may readily be enjoyed by a wide range of users having a range of experience from little or none to those having extensive musical training. More particularly, the present invention relates to an interactive electronic drum and music training method which are suitable for use in a home video game or a coin-operated environment such as an arcade to simulate the excitement of a live jam session for a user.

THE BACKGROUND OF THE INVENTION

A wide variety of learning and teaching aids for musical instruments have been known for some time. See, for example, U.S. Pat. No. 4,919,030 "Visual Indication of Temporal Accuracy of Compared Percussive Transmit Signals," U.S. Pat. No. 5,036,742 "Tempo Monitoring Device and Associated Method," U.S. Pat. No. 5,945,786 "Method and Apparatus for Controlling Scale Practice of Electronic Musical Instrument," and U.S. Pat. No. 5,214,231 "Apparatus for Electronic Teaching Accompaniment and Practice of Music, Which is Independent of a Played Musical Instrument." These and like approaches may suffer from a variety of drawbacks. By way of example, they may only be used by experienced musicians or those with a significant knowledge of music theory. Alternatively, they may not be readily intuitive in their operation, or they are not designed to be fun and provide a positive experience which helps keep the user eager to learn.

For years, electronic drum pads have been available through music stores. Many patents describe a variety of aspects of such drum pads. See, for example, U.S. Pat. Nos. 4,781,097, 4,947,725, 4,932,303, 5,009,146, 5,177,313 and 5,233,658, all of which are incorporated by reference herein in their entirety. A first time user would have to purchase these relatively expensive pads and determine at a later date if he or she enjoyed playing drum pads, what is involved, and what the final outcome of their efforts would produce. After buying the pads, the user was typically left without guidance as to how to play them. Thus, the novice user did not have any formatted learning structure or any positive feedback to encourage continued learning. Further, typical musical instruction exercises are typically renowned for being dull and repetitious.

Further, a variety of coin-operated audio visual musical devices are presently known. See, for example, U.S. Pat. Nos. 3,990,710 "Coin-Operated Recording Machine," U.S. Pat. No. 4,695,903 "Audio Video Entertainment Module," and U.S. Pat. No. 4,965,673 "Apparatus for a Video Recording Booth." Additionally, a wide variety of action and adventure type video arcade games, such as auto race and various war or battle games in which the player fights against the machine or another player are known. None of these approaches provides a musical learning experience that is like a game or adventure encouraging a positive learning experience of a musical skill.

Standard video arcade games are very popular but often are violent in nature and involve a complex combination of button activations and joystick movements that may be daunting to the novice. As a result, a need exists for a non-violent game which provides a positive experience for the novice and the expert alike, and which is exciting enough to replace the standard beat-em-up, shoot-em-up fare that many adults find unacceptable.

SUMMARY OF THE PRESENT INVENTION

The present invention addresses needs, such as those outlined above, by providing a unique combination of features and suitably packaging them so they may be appropriately enjoyed in an arcade setting. Home use can also be envisioned.

In one embodiment of the present invention, a money operated electronic drum system is provided in conjunction with audio-visual inputs to both help a user learn to jam on or play the drums and to enjoy the jamming experience. In this context, "jamming" may suitably be defined as free playing over music. That is to say not reading music, but rather intuitively and naturally playing and reacting to create music as the music proceeds. For example, a drum player may create suitable drum beats to match accompanying instruments such as a guitar or keyboard.

In one aspect, the present invention provides an interactive series of menus to guide a user to select a desired mode of operation. A series of cuing LEDs or other light indicia on or associated with the drum pads, or alternatively a video representation on a display may guide the user in the correct sequence and striking of the drum pads, and a control system controlling audio and video devices will provide appropriate feedback to both encourage the user and to make the experience enjoyable.

In one mode of the present invention, the user chooses to play along with his or her favorite type of music with the system including a source of music, such as a compact disk ("CD") player which may be of a jukebox format, a tape player, a radio or the like. In another mode of operation, the user tries to play along with drum progression which increase in difficulty. The control system monitors and scores the user's play, and also provides feedback to encourage the user.

Other features and advantages of the present invention are described further below and will be readily apparent by reference to the following detailed description and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** illustrates one embodiment of drum apparatus in accordance with the present invention suitable for use in a money-operated arcade embodiment;

FIG. **2** illustrates in block diagram form further details of control and processing circuitry suitable for use in conjunction with the apparatus of FIG. **1**;

FIG. **3** illustrates one overall flowchart of the operation of the drum apparatus of FIG. **1**;

FIGS. **4**–**4C** are more detailed flowcharts illustrating details of various possible play modes in accordance with the present invention;

FIG. **5** illustrates a menu selection technique in accordance with the present invention which assists a user in learning the various drum pads and how to strike those pads; and

FIG. **6** illustrates an embodiment of a suitable mounting arrangement for the mounting of drum pads of a drum apparatus, as in FIG. **1**, in a small footprint cabinet suitable for use in an arcade environment.

DETAILED DESCRIPTION

FIG. **1** shows an overall view of an audio-visual interactive drum studio system **100** in accordance with the present invention. System **100** preferably includes a cabinet **10** for housing, enclosing and mounting the various components of the system. The cabinet **10** will preferably consist of a sound and fire resistant frame which may be painted to emphasize the effectiveness of the system's lighting as further described below. The system **100** also includes a drum set layout **20** which may preferably include electronic drum pads for snare **21**, hi-hat cymbal **22**, crash cymbal **23**, ride cymbal **24**, and toms one through four **25–28**, as well as a bass drum foot activator or pedal **29**, a foot pedal or activator **30** for controlling the opening and closing of the hi-hat **22**, and an additional foot pedal or activator **32** which may be included to add a variety of additional sound effects, such as a "wa-wa" or the like. While it is presently preferred to utilize a standard electronic drum pad employing sensors, such as piezoelectric sensors, to produce an output indicative of the occurrence of a pad strike, as well as the force of the strike, it will be recognized that a simpler and less expensive arrangement may employ a user input sensor which merely senses a touch, strike, or any switch closure causing event initiated by the user. It will also be recognized that while a device looking like a drum pad is presently preferred, other form factors such as a device looking like a guitar, a keyboard or a simple switch arrangement may also be employed. By way of example, a joystick or other activator **33** and a button or buttons **34** may be employed to allow a second user or a younger, less coordinated user to enjoy the system. For example, a left to right or back and forth movement of the joystick **33** may allow a second user to make other sounds such as hi-hat, maraca, tambourine, or other sounds. The button or buttons **34** may allow the second user or a less advanced user to activate a sequence of drum rhythms or the like.

Each of the drum pads or pedals preferably has one or more associated LEDs or other indicia **21A** and **21B**, **22A–C**, **23A** and **23B**, **24A** and **24B**, **25A** and **25B**, **26A** and **26B**, **27A** and **27B**, **28A** and **28B**, and **29A**, respectively, to guide the beginning user in learning how to use the system by cuing the user to strike the appropriate pads or pedals of the drum set layout at the appropriate time as more fully described below.

The system **100** further includes an adjustable stool or seat **40** for the user to sit comfortably in front of the drum set layout **20**, a money validation unit **50**, such as a coin changer, bill validator or the like, and a series of audio speakers **60**, which may suitably include left and right midrange speakers **61** and **62**, and a subwoofer **63**. Additional speakers may be added as desired to improve the audio quality of the system.

The system **100** as further illustrated in FIG. **2** also preferably includes a central processing unit or control system **70**, suitable lighting **80** which may include colored stage lights **81**, overhead white and black lighting **82** and **83** respectively which will preferably be variably controlled by the control system **70** as described further below. A strobe light **84** which will preferably be mounted in the ceiling of the cabinet **10** may also be provided.

The user may enter data and select modes of play using a control panel **85** with selection buttons **86**. A multi-disk CD player **90**, preferably of the jukebox type, or other music source will also preferably be employed to allow a user to select musical accompaniment allowing the user to play along with a favorite musical selection as more fully described below. A camera **92** and a VCR **94** may also be provided. The camera **92** may be used to record a part or parts of the user's session to be displayed on a screen or display **105**. By way of example, the user who achieves a high score may be allowed to choose to have his or her play sequence run on the screen **105** when the system **100** is not in use. As another alternative, the user's performance could be cut and pasted into a video of a live band playing the music that the user is playing. Many exciting possibilities exist.

The display screen **105** may suitably be a CRT display, and this display will preferably be utilized to provide user cues and instructional information. As discussed further below, screen **105** may also be employed to show the user video images, such as video of a rock concert audience positively reacting to the music being played. Such video may be provided from memory in the control system **70** or alternatively from a videotape in the VCR **94**. Such a videotape may include, by way of example, instructional video of how to play the system with a variety of lessons which can be selected by the user or video of a professional drummer utilizing the system **100**.

Finally, the control system **70** and other components such as the multi-disk CD player, the VCR **94** and the like will preferably be housed in a satellite cabinet **110** as shown in FIG. 1. This cabinet will be accessible by lock and key to the operator or service personnel.

While the above described components are shown in the drawings and discussed in the context of a presently preferred embodiment of the invention, it will be recognized that similar and other components may be added to enhance the system or that certain of these components may be subtracted to reduce the cost of the system. As one example, while a money-operated arcade system is presently preferred, it will be recognized that the present invention may be readily adapted to the home environment in a system in which electronic drum pads would be suitably interfaced with a home computer or a video game controller and a television without the need for a money validation unit, such as a coin or bill validator, or special cabinets. Additionally, a CD or radio might be connected in such a home system to provide player accompaniment. Further, while the present specific disclosure is made in the context of electronic drum pads and pedals which are presently preferred, it will be recognized that the present invention may be adapted to other formats in which button presses or switch closures are used by a user to perform or play. For example, a guitar-like device having a number of buttons could be utilized as the user's input device. Alternatively, a simple keypad or keyboard could be employed. Preferably, the aspects of the inventive arrangement described further below would be employed therewith to continue to make the unit accessible to the novice user.

FIG. **2** illustrates further details of one suitable control system **70** for use in conjunction with the various devices to be monitored and controlled in system **100** of FIG. **1**. While the control system **70** is presently preferred because it can be readily implemented with off the shelf components, it will be recognized that custom designed and different components may be readily used to achieve the desired functionality of the present invention.

As shown in FIG. **2**, the control system **70** may comprise a suitably programmed PC computer having a central processing unit **71** in a cabinet **72** which also may include a built in CD ROM drive **73**, which may be employed as an alternative to or in addition to the player **90**. The control system **70** will also include a standard or custom MIDI ("Musical Instrument Digital Interface") driver **74** which may be an internal card or an external component. Driver **74** allows the system **70** to be readily programmed to drive sound speakers, such as the speakers **61–63** to provide appropriate sounds.

The control system **70** will also typically be connected to various additional components such as the pads and pedals of drum set layout **20**, cuing LEDs **21A–29A**, joystick **34**, sound button or buttons **35**, a mixing board **75**, a power amplifier **76**, drum pad controller **77**, lighting system **80**, control panel **85** and control buttons **86**, multi-disk CD player **90**, camera **92**, VCR **94**, a monitor **102** whose CRT screen may be suitably employed as the display **105** and a keyboard **120**. The monitor **102** may be part of an integrated unit also including a microphone **107** for audio pickup and speakers **111** and **112**. These speakers may be utilized as the previously mentioned speakers **61** and **62**, or as speakers supplementing the audio outputs of the speakers **61–63**, or to provide software driven instructions to the user. It will be recognized, however, that larger and higher quality microphones or audio speakers may be desired to provide a superior experience for the user. This will be particularly true in the arcade environment. For example, a better microphone may be desired to provide a karaoke mode or to record the user's singing along with the user's playing.

The keyboard **120** will preferably be a standard keyboard, and may be employed to allow a user to make selections or preferably to allow a service person to diagnose any malfunctions of and to perform routine maintenance on the system **100**. For example, keyboard **120** may be mounted on the cabinet **10** within easy reach of a user sitting on the stool **40**. It may have a protective cover that covers most of the keypad allowing user access to only a limited number of keys to make program selections as described below. A service person could be given a key to open the cover and gain access to the keyboard **120** to service the system **100**.

While shown in FIG. **2** without any protective casing other than the standard cabinet **72** for ease of illustration, the control system **70** will preferably be housed in a protective casing such as the satellite cabinet **110** of FIG. **1** to prevent any damage to the unit in the somewhat rough and tumble environment of the typical arcade. This casing will limit system access to the owner or operator and authorized service personnel. Similarly, drumsticks **11** and **12** will be preferably connected to the cabinet **10** by strong, light and flexible cables **13** and **14**, such as steel stranded cable. When not in use, drumsticks **11** and **12** will preferably be placed in holders **15** and **16** mounted so that the drumsticks **11** and **12** will be readily seen and reached by users. In a home environment, such a protective casing and drumstick cable mounts should not be necessary.

Turning to the operation of the system **100**, the system utilizes the cuing LEDs or other indicia **21A–29A** which are associated with their drum pads and pedals to provide visual guidance on which drum pad or pedal to play, which hand or foot to play it with, and when to play it. The speakers **60** may also be driven to provide audio cues to correct play. As described further below, this cuing or instruction is subject to suitable program control by the control system **70** which drives both the cuing LEDs and the speakers. Data on the users performance which is collected by the control system **70** may be fedback and displayed on the screen **105** to the user. Similarly, the correct order or rhythms of striking the drums and graduated steps of rhythmical sequences may also be displayed on the screen **105**. The user will thus be guided by visual and audio stimuli to use his or her hands and feet for certain repetitive drum sequences generated by the CPU of system **70**. Preferably, left and right lights, such as light **21A** (Left) and light **21B** (Right) for snare **21** indicate when the left and right hands should be playing. Similarly, the light **22C** indicates when the user's foot should activate foot pedal **30** to raise hi-hat **22**. As addressed further below, different sequences can be selected by the user via computer menu selection. Thus, as further described below, the inventive system provides an enjoyable opportunity to learn how to play an electronic drum set.

A first level of play is to master certain basic skills, the user will then be able to move on to several more advanced playing scenarios. By way of example, one level or mode of play of the present invention is to play the electronic drums with visual indications of the rhythmical sequence provided to guide the user. In this option, the visual LEDs or indicia are illuminated to cue the appropriate rhythm and teach the desired audio sounds of the various pads. A second level of play is playing the electronic drum pads with the visual indicators along with an actual sequence of music, such as a prewritten song of the user's favorite type of music. A further level of play is to play against the visual indicators where the performance, may be rated. For example, the accuracy of the user's rhythm of play may be measured, and feedback in the form of scores or otherwise may be given. Another level of play is playing along with music generated from compact disks or the like where the user can try to play along with his or her preferred music. Another option is free play, giving the user the opportunity to hit the electronic drum pads **20** freely to get accustomed to the sounds and feel of the system. This mode also allows the player the chance to test the abilities that were learned via the illuminated displays without any guidance. Free play without any feedback is the normal and only mode of play with standard acoustic drums. As discussed in greater detail below, a variety of feedbacks are possible to enhance the learning or game experience to make it truly enjoyable.

FIG. **3** shows a first overall flowchart of one method or process **300** in accordance with the present invention. FIGS. **4–4C** illustrate in greater detail various presently preferred modes of play. In step **301** of FIG. **3**, a user may watch a display which displays the capabilities of the game, such as the display **105** of FIG. **1**, to determine if the game is one which he or she chooses to play out of the many in an arcade. To this end, a control system such as the control system **70** should preferably be programmed to cause display **105** to engagingly display the capabilities of the system. Additionally, the control system **70** may also drive the speakers **60** to audibly explain the system capabilities, play music, drum jams, or otherwise attract potential users' attention.

In step **302**, a potential user decides whether to play the game. In step **303**, the user decides to try the game and activates the system. For example, the user inserts the necessary amount of money. To this end, the user may insert a dollar in coins or currency ($1) into a coin mechanism or bill validator **50** as shown in FIG. **1**.

Next, in step **304**, the display **105** displays a list or menu of play mode options. At the same time, audio instructions may be provided by the speakers **60**. In step **305**, a user selects a mode of play. The user preferably selects from a menu of options displayed on the display **105**. As addressed further below, a wide variety of other play type selections may also be made. For example, the user may enter a level of skill, select a choice of lights and sequenced music, the level of difficulty, the tempo and the like.

In step **306**, play begins and lasts for a predetermined time period, such as three minutes. Alternatively, play time may vary. For example, if the user chooses to play along with a favorite song, the play time may last for the duration of the selected song. During this period, the user plays the drums in accordance with the mode selected as described in greater detail below. As the drumming session ends, feedback is provided to the user in step **307**. For example, a crowd may cheer, a score may be displayed, or the like.

FIGS. 4–4C illustrate further details of presently preferred modes of operation in accordance with the present invention. In step **401** of FIG. **4**, the system, such as system **100**, is in a rest or demonstration mode waiting for a user to choose to use the system. In one presently preferred rest mode, a demonstration song is played. In this demonstration song, the drums are playing. As the drums play, the learning or cuing lights, such as the LEDs **21A–29A** associated with the drum pads **21–29** corresponding to the drum sounds heard by the user, are flashing in accordance with what is being played. The demonstration song may switch themes from a drum based demonstration to an instrument based demonstration, such as for example, keyboard sounds, orchestra sounds, or various other instruments. The demonstration mode illustrates the flexibility of the playing modes of the machine. Also, it may help to show how the instruments are played on the pads, and is intended to attract the interest of people wanting to play with the system **100**. A videotape of a professional drummer playing the system **100** or giving a quick lesson on how to use the system may also be played. To this end, a suitable videotape may be played on the VCR **94** and displayed on the display screen **105**.

In step **402**, a person has decided that he or she wants to play. As a result, the person sits down, picks up the drum sticks and inserts money into a money validation unit, such as the coin or bill validator **50** of FIG. **1**. A credit card, debit card, smart card or token reader might also be employed to provide the user with additional flexibility in making the required payment.

In the FIG. **4** embodiment, each insertion of money preferably triggers a sound type, for example, a bass drum sound. Alternatively or additionally, a crowd scene may be displayed on a video display, such as the display **105**, and the sound of the crowd starting to clap and cheer just as it occurs before the beginning of an actual concert may be produced by audio speakers, such as speakers **60**.

To this end, the money validation unit **50** detects each insertion of money and produces an output indicative of the recognition of valid money in step **403**. In step **404**, the control system **70** receives the output from the money validation unit **50** and drives the display **105** and speakers **60** as described above.

In step **405**, a variety of playing modes may be displayed on a display such as the display **105**. For example, the following five modes may be displayed: "Jam to CD", "Jam Against the Machine", "Jamming Music", "Jam Lessons" and "Jam Alone" for selection by the user. The user may make a selection in a variety of ways. For example, the display **105** may be a touch screen and the user may make a selection by touching the appropriate part of the screen. Alternatively, the display **105** may display five of the drum pads **20** as shown in FIG. **5** as well as the instruction to strike the appropriate pad to make a selection. Upon a pad being struck, the control system **70** will then drive the speakers to audibly give the same instruction. As shown in FIG. **5**, pads **21**, **22**, **23**, **25** and **26** are shown in solid lines and pads **24**, **27** and **28** and foot pedals **29** and **30** are shown in dotted lines as only five pads are needed to select from the five menu items. A similar approach may be employed to respond "yes" or "no" or to enter other data. The user makes the selection by striking the appropriate pad. This approach has the advantage of immediately starting the learning process of the pad names, their function and their feel. Alternatively, the series of selection buttons **86** provided on the control panel **85** may be employed to make this selection.

If in step **406** the user selects the Jam to CD mode, the system **70** enters that mode in step **407**. Both the display and speakers may be utilized to request further user input in step **408**. For example, the speakers **61–63** may be used to say: "Enter your choice of CD" or "Enter CD Jam Number" which corresponds to the CD and music which the user wishes to play along with. By way example, the CD selection information may be provided by a simple selection list or displayed on the display **105**. After that choice is made, the user may next be asked "Would you like a quick lesson?" with the user being given the option to select "Yes" or "No". The same information will preferably be displayed on the display **105**. Next, the user may be cued through a warm-up in step **409**. For example, the user is told to "Hold your sticks like this" and a picture illustrating the proper way to hold the drumsticks is displayed on the screen **105**. Next, the unit may say "Let's do a sound check." At this point, the lights **21A–29A** indicate and a voice says through speakers **61–63**: "Hit the snare, hi-hat, hi-hat pedal, bass pedal, tom **1**, tom **2**, tom **3**, tom **4**, ride cymbal, and crash cymbal." On the display, as shown in FIG. **5**, both the name of the pad or pedal and the shape and location of the drum pad or pedal with relation to the other pads may be displayed in the correct sequence to illustrate the desired warm up sequence. If during the sound check **409** the user is hitting the drum too lightly, a signal measuring device in drum pad controller **77** will produce an output signal which is then provided as an input to the system control **70** which will drive the speakers to say "Strike the drum harder." Next, a standard 4/4 beat may be played by the system **100** with lights **21A–29A** flashing on and off in the correct sequence to illustrate the correct timing and the correct striking hand or foot to the user. A voice may also say "Most beats are played using the snare, hi-hat and kick pedal. Basic fill is played. Fills are usually played on the toms and end with a cymbal crash. Relax, listen to the music and try to play along or bang around and just have fun. You can't make a mistake. If you want more lessons choose the 'jam lessons' mode from the menu next jam. It's time now to begin." As such a warmup may take a moderate amount of time, an operator may choose to leave out the quick lesson to increase the turnover of the system. Leaving out the warm up may be particularly appealing to an operator where potential users are often lined up waiting to play the game.

At this point, the beginning of actual play is cued in step **410**. For example, a screaming, foot-stomping, hand-clapping crowd may be heard on the speakers **61–63**. Normal room white lights **82** may be lowered. Colored stage lights **81** may be turned on to appropriate levels and begin syncopating to simulate a live concert atmosphere. Also, black lights **83** may be pulsed on and off. Also, strobe **84** may begin flashing with all the above lights preferably controlled by the controller **70**. The display **105** may show an enthusiastic crowd. It will be recognized that the lights **81**, **82**, **83** and **84** may be controlled in a variety of manners. One presently preferred control option is to control them in response to the user's activity. For example, if a user plays faster, the strobe may flash more quickly.

In step **411** the selected CD song is cued by the control system **70**. In step **412**, the user plays along to the song. As the song plays, the user can change to any one of several different pad sounds during the song that are preprogrammed when the CD is programmed. For this mode, the operator will code the music, Rock 01, Rap 02, Jazz 03 and the like, and the control system 70 will generate the drum sets that match the music style.

The user can also preferably change the volume levels of the drums or music to be louder or softer. For example, song volume may be varied via a volume knob on the control panel 85. Alternatively, the unit 100 may ask the user "Do you want to play louder?" and if the user shouts above a certain threshold as detected by microphone 107, the system control 70 will crank up the volume. Alternatively, the operator may preset a fixed level or the user may select the level as part of a setup menu.

When the song is over, the control system 70 detects that the song is over by sensing the lack of DC signals coming from the output of the CD unit 90 in step 413 and shuts the game down. In step 414, positive user feedback is provided. For example, a long crowd roar may be triggered and a voice may announce "great jam" loudly!

If the user selects the "jam against the machine" mode in step 406 as illustrated in FIG. 4 then that mode is entered in step 407A as further illustrated in FIG. 4A. A second menu is preferably displayed in step 408A to allow the user to select a jamming level matched to his or her level of skill, for example, beginner 1, advanced beginner 2, intermediate 3 or expert 4. Next, a warmup is cued in step 409A. For example, a voice may say "let's warm up, give me a drum roll, come on speed it up". Simultaneously, on screen 105, the user's speed of striking the pads, for example, a speed level of 0–100 beats per a predetermined time interval, such as fifteen seconds, shows the user the speed of the drum roll. Next, the voice may say, "not bad, not bad at all." It will be recognized that different warmup exercises may be employed, as well as, different measures of the user's proficiency. For example, the user's proficiency in correctly following the correct rhythm or tempo may be measured.

In step 411A, the user is cued to follow the machine. For example, the machine may start to play a steady bass drum beat indicative of the proper tempo. Next, the user may be told "Now play what I play." For example, two hand drum riffs or rhythms. In this context, a "riff" may be defined as a set of notes or rhythms in a pattern. In step 412A, play begins with an announcement, "We're ready to go", for example. In step 413A, a sequenced drum track starts with cuing lights indicating what the user should play over the music. Alternatively or additionally, a videotape of an instruction lesson of similar content may be displayed. The user then may initiate the actions of a videotape instructor. A display such as that discussed in the context of FIG. 5 may also be used to highlight the correct pad or pedal and striking had or foot. As time passes, the level of difficulty increases in step 414A. At the lower difficulty level, the unit desirably has more flexibility in scoring. In the present example, the user's score is preferably based on repetition, accuracy and speed. Scoring may be based on a point system with a value for each correct strike and increasing points for each advancement in difficulty. The display 105 may show a constantly updated jam score as the jam proceeds. In step 416A, a timer allows play to proceed for a predetermined time, such as 3 minutes, and a final score is given at the end of play. In step 417A, the game end sequence begins. The song ends. The game is over, the crowd cheers, and the unit says "great jam."

If the "jamming music" mode is selected as illustrated in FIG. 4B, a specially written track of background music, written for drum and instrument soloing is preprogrammed in the control system 70. After entering the jamming music mode, the user is provided the options of selecting a preferred style of music and selecting a light indicia, display or videotape cued mode of play or a free play mode in step 409B. Then the specially written track is retrieved and begun in step 409B. When the cuing lights have been chosen and activated, the lights will illustrate to the player what, when and how to play the pads. If jamming with lights, a slow stage light progression of the stage lights 81 is activated to dim the lights 81 prior to the beginning of the specially written track to allow better clarity of viewing of the cuing lights. Crowd cheers also accompany the start of the special jam music track. An enhanced drum set is activated and basic fill progressions synchronized with the cuing lights are started. "Enhanced drum set" as used herein may be defined as a specially programmed drum set to provide special effects, such as echoing or accenting by combining bass drum activation with cymbal activation, or the like for the purpose of improving the sound of inexperienced players. The first part of the song ends, and part two begins. The pads are now activated to become music pads, and cuing lights are activated. Finally, the song ends, and the crowd cheers.

If no visual indicia is selected in step 408B, the user intuitively plays with the unit. In both non-light cued and light cued modes, drum pads and pedals 20 will change to different instruments as the song proceeds in step 410B. First, the control system 70 will activate the standard enhanced drum set 20. As the song theme proceeds and changes, instrument sounds will be sent to the pads by the control system 70. Riffs, progressions, chords and other sounds are contained on a particular pad, step 411B. A variety of programmable variations of the progression or alternate progressions may be employed. For example, a five note bass progression in the key of the music may be assigned to a particular pad. By playing this particular pad at a particular rate, step 412B, the progression is played in the order it was written, with the player's strikes of pads and pedals determining the output or outcome in step 413B. For example, if the player stops at any time during the sequence, step 412B' the progression may begin over in step 413B'. If the particular pad is played for a predetermined time in step 412B", the progression of sound may change order and length in step 413B". Other methods may be employed to keep the pad sounds appealing or unpredictable to the user. To this end, the control system can readily be set up to vary the pad set up.

If the "jam lesson" mode is selected in step 406 as illustrated in step 407C of FIG. 4C, a menu is displayed cuing the user to enter a playing level ranging from Beginner 1 to Expert 4 in step 408C. The user next enters a style of music, for example, 1 Rock, 2 Dance/Rap, 3 Country, 4 Heavy Metal, 5 Jazz in step 409C. Next, in step 410C, a voice gives a short explanation of the style of music and the basics of how it is played. In step 411C, if the user selected 1 Rock, then he or she is encouraged to "Try a simple example, this is a basic rock beat." The light indicia 22A–C, 21A and 21B, and 29A cue the proper activation for the hi-hat 22, snare 21 and base drum, respectively. The control system 70 next switches the hi-hat rhythm 24 to the ride cymbal and then snare 21: "Beats can be played on the ride cymbal 24 and snare drum 21 also. The base drum 29 will be added later, for now we will play it for you. The system 70 then activates the bass drum rhythm. System 70 then activates a short background rock style music piece saying "Notice how the beat keeps the music in time." The user practices for a predetermined time, step 413C. "Practice these beats over and over and it will be automatic." When the lesson is over, positive feedback is provided in step 414C, for example, the voice says, "Good job". Subsequent levels may provide more advanced lesson plans building on previous lessons. While the above discussion describes an exemplary lesson, it will be understood many other lessons may be readily programmed.

Finally, in the "jam alone" mode, the user can play the drums with no instruction or cuing. This mode allows the user to test his or ability to play what he or she has learned in previous lessons or on his or her own.

Turning to FIG. 6, this figure shows a presently preferred embodiment for constructing cabinet 10 and mounting drum pads 21–28 in a compact manner particularly well-suited to the tight space requirements of the typical arcade. The presently preferred depth of cabinet 10 is 31 inches, the preferred width is 42 inches and the preferred height is 72 inches. Cabinet 10 as shown in FIG. 6 comprises a base 17, sidewalls 18 and 19, top 117 and left and right front side panels 118 and 119. The front side panels 118 and 119 are shown cutaway in FIG. 6 to better illustrate the presently preferred mounting of the drum pads 21–28. The open space between the side panels 118 and 119 may be filled with a sliding, sound-dampening curtain which may be drawn closed when the unit 100 is in use and opened when the unit is not in use. The curtain may be simply manually opened or shut, or an automatic control arrangement may be employed to sense whether a user is sitting in seat 40 or not, or to sense the begin or end of play, and then to automatically open and shut the curtain.

The cabinet 10 also includes a desk or counter top 120 on which drum pads 22–28 are mounted. In the presently preferred mounting technique these pads 22–28 are mounted on top of varying lengths of hollow plastic pipe 122–128 respectively. The tops of the plastic pipes 122–128 are preferably beveled at an angle to allow their corresponding pads to be readily mounted at a desired angle for playing. The opposite ends of pipes 122–128 extend through cutouts in the counter top 120 and are secured to the counter top 120. The pad 21 is preferably mounted using three pieces 121A–C, two straight lengths 121A and 121C, and an elbow joint 121B. One end of the piece 121A is secured to side 19 of cabinet 10. One end of piece 121C is beveled at an angle to properly support the pad 21.

While the above discussion has been made in the context of presently preferred constructions, components, modes of operation and the like, it will be recognized that the teachings of the present invention may be widely modified to include larger or smaller physical embodiments, greater or smaller numbers of components, different components, and different modes subject only to the limitations of the claims. By way of example, while specific user lessons and specific user feedbacks are described, it will be recognized that such lessons and feedback may vary widely consistent with the present teachings which provide the impetus to provide such functions in the context of a highly user friendly drum jamming system.

I claim:

1. An audio-visual interactive drum apparatus comprising:

a money validation unit to accept and validate a user's money;

a plurality of electronic drum pads;

a lighting system;

an audio speaker system;

a video display system; and

a control system for receiving an input from the money validation unit; for receiving inputs from the electronic drum pads; and for controllably driving the lighting system, the audio speaker system and the video display system to simulate a live jam session.

2. The apparatus of claim 1 further comprising a musical source connected to the audio speaker and the control system so as to provide musical accompaniment for the user of said drum apparatus.

3. The apparatus of claim 1 wherein the control apparatus is suitably programmed to analyze inputs received from the electronic drum pads during a user's playing of the electronic drum pad to score the user's playing, and to drive the audio speaker system and the video display system to provide feedback to the user.

4. The apparatus of claim 1 further comprising a plurality of user cuing light indicia of which at least one of said plurality of light indicia is associated with each one of said plurality of electronic drum pads.

5. The apparatus of claim 4 wherein the control apparatus is connected to said plurality of light indicia and is suitably programmed to drive said plurality of light indicia to guide the user to play the plurality of electronic drum pads in the correct sequence and with the correct rhythm.

6. The apparatus of claim 1 wherein the video display system includes a touch-screen CRT display which is utilized by the user to provide mode selection data to the control system.

7. The apparatus of claim 1 wherein the control system is suitably programmed to drive the video display system to display a visual representation of the plurality of electronic drum pads and mode selection prompts to the user and the user enters mode selection data by striking an appropriate one of the plurality of electronic drum pads corresponding to an appropriate mode selection prompt.

8. The apparatus of claim 1 further comprising at least two user cuing light indicia, at least a first of said at least two for a left hand cue and at least a second of said at least two for a right hand cue, associated with at least one of said plurality of electronic drum pads.

9. The apparatus of claim 8 wherein the plurality of electronic drum pads include a hi-hat pad, and the apparatus further comprises at least a third cuing light indicia for a raise hi-hat cue.

10. The apparatus of claim 1 further comprising a sound resistant cabinet enclosing at least the audio speaker system.

11. The apparatus of claim 1 wherein said lighting system further comprises one or more lighting components selected from the group comprising: an overhead white light, a black light, a strobe light and colored stage lights.

12. The apparatus of claim 1 in which the lighting system is controlled in response to a user's activation of one or more of the plurality of electronic drum pads.

13. The apparatus of claim 1 further comprising two drumsticks flexibly cabled to a part of a cabinet supporting the plurality of drum pads.

14. The apparatus of claim 1 wherein the plurality of drum pads include a snare, a hi-hat, crash and ride cymbals, a plurality of toms and a bass foot pedal.

15. The apparatus of claim 1 wherein the video display system is controllably driven to display the plurality of electronic drum pads.

16. The apparatus of claim 1 wherein the control system is programmed to controllably drive the audio speaker system and the video display system to provide user feedback.

17. The apparatus of claim 1 wherein the control system is programmed to controllably drive the audio speaker system and the video display system in a rest mode to demonstrate the capability of the apparatus to potential users.

18. An audio-visual interactive drum play method comprising the steps of:

determining that a predetermined amount of money has been inserted into a money validation unit;

activating a plurality of electronic drum pads, a lighting system, an audio speaker system a video display system upon determining that the predetermined amount of money;

controllably driving the lighting system, the audio speaker system, and the video display system to simulate a live jam session;

controllably driving a cuing system to provide a user with cues as to how to play the plurality of electronic drum pads; and

monitoring the plurality of electronic drum pads as they are played by a user in response to said cues.

19. The method of claim 18 further comprising the step of:

providing musical accompaniment as the electronic drum pads are played by the user.

20. The method of claim 18 further comprising the steps of:

analyzing inputs received from the electronic drum pads during said playing;

scoring the playing; and

providing feedback concerning the playing.

21. The method of claim 18 further comprising the step of:

lighting a plurality of light indicia associated with the electronic drum pads to guide said playing.

22. The method of claim 18 wherein said video display system includes a touch screen and the method further comprises the steps of:

displaying a menu of selectable modes of operation to a user on the touch screen; and

selecting a preferred mode of operation in response to a selection from the menu.

23. The method of claim 18 further comprising the steps of:

displaying a visual representation of the plurality of electronic drum pads on the video display system; and

sensing a strike of an appropriate one of the plurality of electronic drum pads and entering user data in response.

24. An audio-visual interactive music apparatus comprising:

a plurality of user input sensors;

an audio speaker system;

a video display system; and

a control system for receiving inputs from the user input sensors; and for controllably driving the audio speaker system and the video display system; said control system being programmed to drive the audio speaker system and the video display system to simulate a live jam session and to provide user feedback as a user is actuating the plurality of user input sensors.

25. The apparatus of claim 24 wherein at least one of said user input sensors is incorporated in an electronic drum pad.

26. The apparatus of claim 24 further comprising a musical source to provide musical accompaniment for the user of said apparatus.

27. The apparatus of claim 26 wherein the musical source is a multi-disk CD player controlled by the control system.

28. The apparatus of claim 24 further comprising a VCR controlled by the control system.

29. The apparatus of claim 24 wherein the control apparatus is suitably programmed to analyze inputs received from the plurality of user input sensors during a user's playing of the music apparatus to score the user's playing, and to drive the audio speaker system and the video display system to provide feedback to the user.

30. The apparatus of claim 29 wherein said feedback comprises playback of previously recorded crowd clapping and crowd cheering by the audio speaker system.

31. The apparatus of claim 24 further comprising a plurality of user cuing light indicia of which at least one of said plurality of light indicia is associated with each one of said plurality of user input sensors.

32. The apparatus of claim 31 wherein at least two of said plurality of user cuing light indicia cue a user to strike a particular user input sensor with either the left or the right hand.

33. An audio-visual interactive arcade apparatus comprising:

a cabinet having an entrance;

a plurality of electronic drum pads;

an audio speaker system;

a video display system; and

a control system for receiving inputs from the electronic drum pads and for controllably driving the audio speaker system and the video display system to simulate a live jam session, whereby said cabinet encloses and supports the plurality of electronic drum pads, the audio speaker system, and the video display system.

34. The apparatus of claim 33 wherein at least one of the plurality of electronic drum pads is mounted on a length of hollow plastic pipe one end of which is beveled at an angle.

35. The apparatus of claim 33 further comprising a strobe light controlled by the control system.

36. The apparatus of claim 33 further comprising a black light mounted within the cabinet.

* * * * *