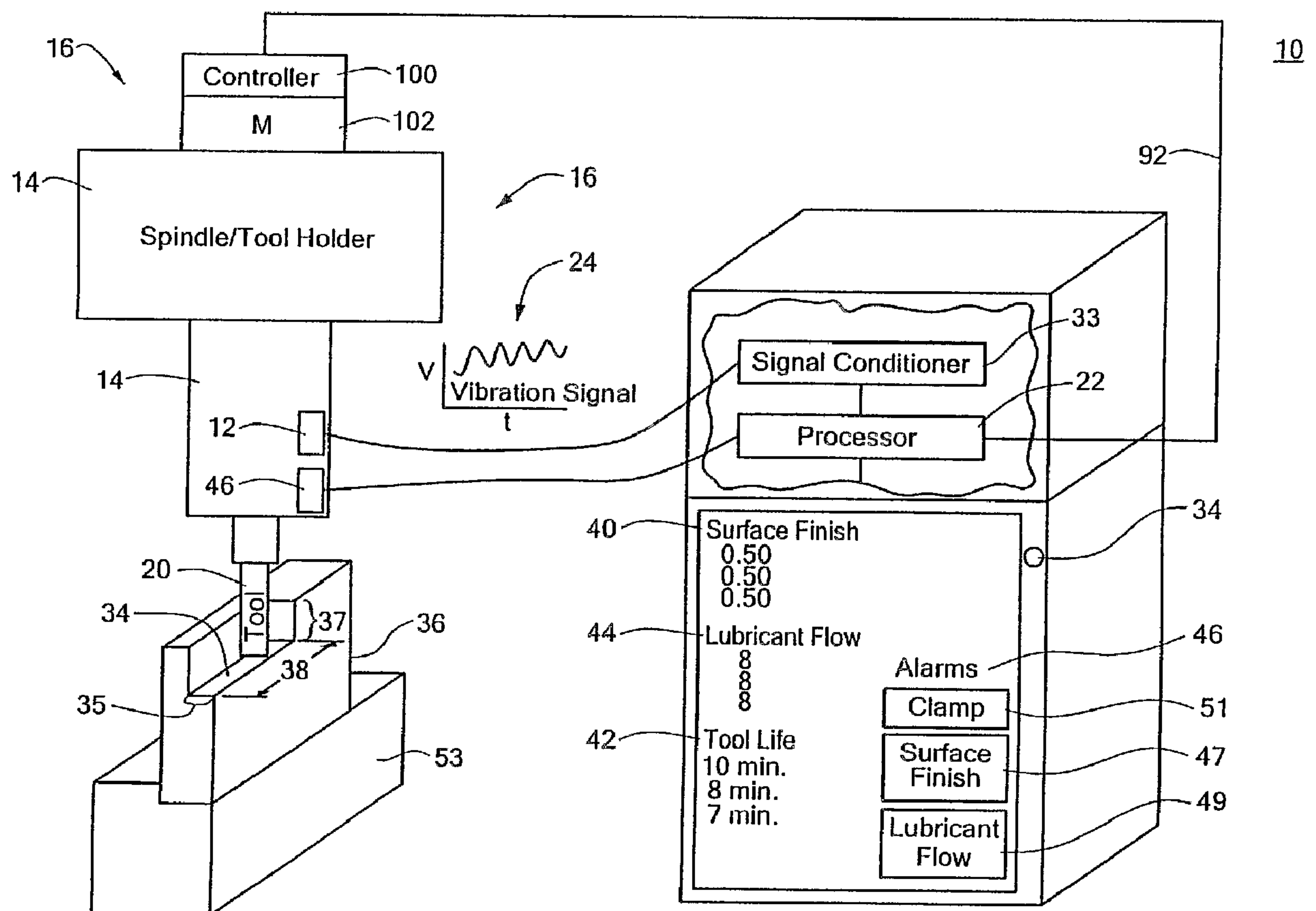




(86) Date de dépôt PCT/PCT Filing Date: 2004/03/05
(87) Date publication PCT/PCT Publication Date: 2004/09/23
(45) Date de délivrance/Issue Date: 2009/09/15
(85) Entrée phase nationale/National Entry: 2005/09/09
(86) N° demande PCT/PCT Application No.: US 2004/006741
(87) N° publication PCT/PCT Publication No.: 2004/080647
(30) Priorité/Priority: 2003/03/10 (US10/385,398)

(51) Cl.Int./Int.Cl. *B23C 1/00* (2006.01),
B23Q 15/00 (2006.01), *G01N 29/24* (2006.01)
(72) Inventeurs/Inventors:
HILL, WAYNE S., US;
BARCK, BRUCE N., US
(73) Propriétaire/Owner:
FOSTER-MILLER, INC., US
(74) Agent: MACRAE & CO.

(54) Titre : INSTRUMENT DYNAMIQUE D'USINAGE
(54) Title: DYNAMICAL INSTRUMENT FOR MACHINING



(57) Abrégé/Abstract:

A dynamical instrument for machining comprising a sensor responsive to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal, and a processor responsive to the sensor output configured to calculate a plurality of



(57) **Abrégé(suite)/Abstract(continued):**

signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
23 September 2004 (23.09.2004)

PCT

(10) International Publication Number
WO 2004/080647 A3

(51) International Patent Classification⁷: **B23C 01/00**,
B23Q 15/00, G01N 29/24

(74) Agents: **COLEMAN, Roy, J.** et al.; Iandiorio & Teska,
260 Bear Hill Road, Waltham, MA 02451-1018 (US).

(21) International Application Number:
PCT/US2004/006741

(22) International Filing Date: 5 March 2004 (05.03.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/385,398 10 March 2003 (10.03.2003) US

(71) Applicant (for all designated States except US): **FOS-
TER-MILLER, INC.** [US/US]; 350 Second Avenue,
Waltham, MA 02451-1196 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **HILL, Wayne, S.**
[US/US]; 26 O'Neil Drive, Westborough, MA 01581 (US).
BARCK, Bruce, N. [US/US]; 652 West Central Street,
Franklin, MA 02038 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN,
CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI,
GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE,
KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD,
MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG,
PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM,
TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM,
ZW.

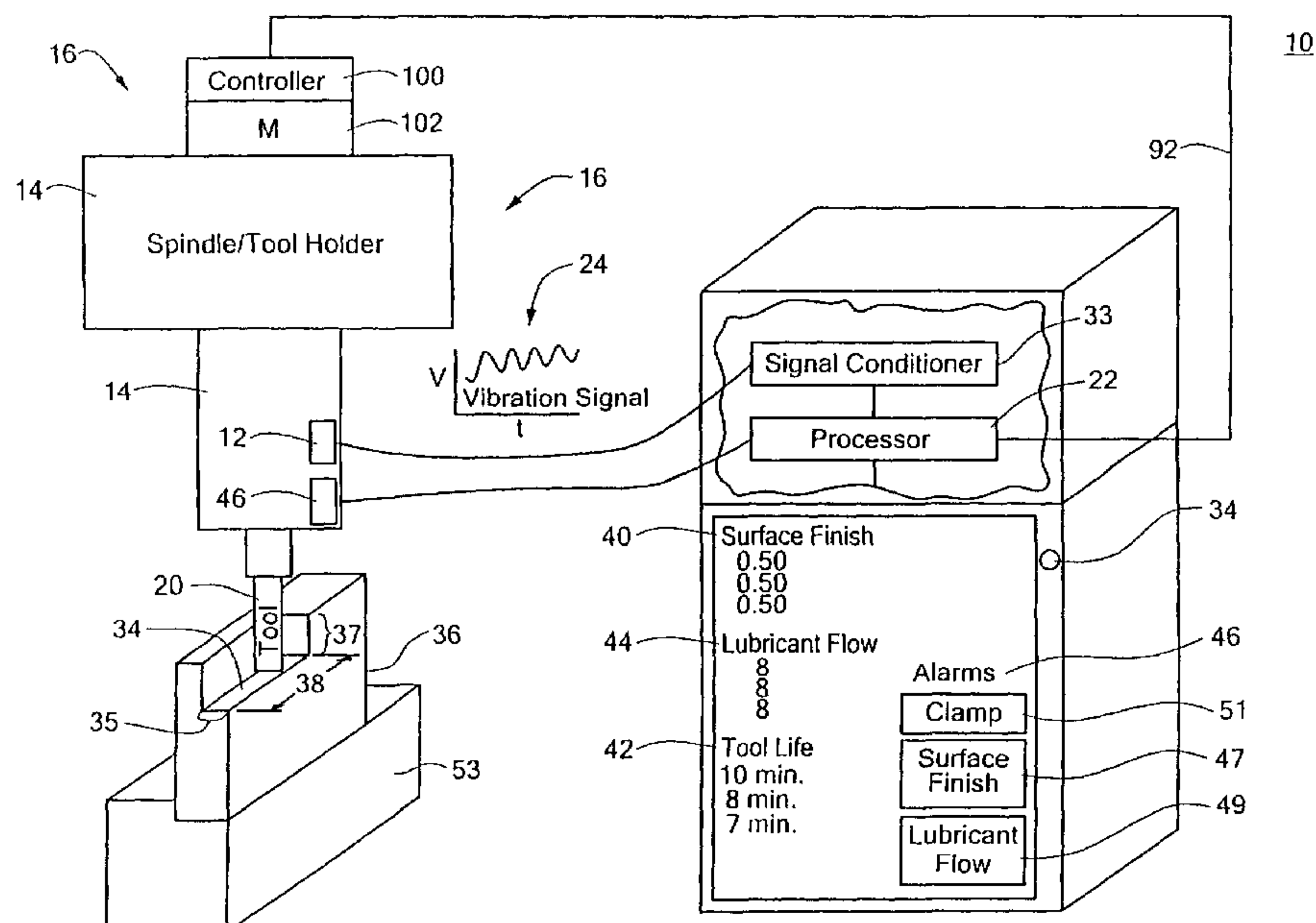
(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),
Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), Euro-
pean (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR,
GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK,
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
ML, MR, NE, SN, TD, TG).

Published:

— with international search report

[Continued on next page]

(54) Title: DYNAMICAL INSTRUMENT FOR MACHINING



(57) Abstract: A dynamical instrument for machining comprising a sensor responsive to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal, and a processor responsive to the sensor output configured to calculate a plurality of signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine.

WO 2004/080647 A3

WO 2004/080647 A3



— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(88) Date of publication of the international search report:

16 December 2004

DYNAMICAL INSTRUMENT FOR MACHINING

FIELD OF THE INVENTION

This invention relates generally to machine tools.

RELATED APPLICATIONS

This invention was made with U.S. Government support under Contract No. 50-DKNB-6-90107 and awarded by the Department of Commerce National Institute of Standards and Technology (NIST). The Government may have certain rights in the subject invention.

BACKGROUND OF THE INVENTION

Some machine shops typically operate machines, such as milling machines, lathes, and the like, at reduced rates, typically one-third to one-half the full capacity, in order to avoid breaking the tool when manufacturing an inexpensive part, e.g., the bit may cost fifteen dollars but the part may cost only five dollars.

Other machine shops may manufacture parts which are more expensive, typically costing hundreds or thousands of dollars to manufacture. The parts produced by large machine shops are usually complex and must conform to strict specification requirements provided by the customer. To insure the machined part meets these strict specification requirements, the machine operator typically reduces the rate (e.g., advance speed) of the machine tool in order to increase the accuracy of the cutting bit and to improve the quality of the machined surface of the workpiece. Using conventional machining processes, the operator relies on visual, auditory, and vibration cues to adjust the spindle speed, depth of cut, and rate of advance of the cutting tool to insure a proper surface

finish of the workpiece. Post inspection of the workpiece surface finish is required and any deviation from the specification requirements is discovered only after the workpiece is complete and may result in the expensive machined part being scrapped.

Thus, in both types of machine shops, it would be advantageous if the machine operators were provided with real time information concerning the current state of the machine tool (e.g., the remaining tool life, and whether the tool is worn or damaged) because the state of the machine tool directly affects the quality of the surface finish of the workpiece being manufactured and the operating capacity of the machine. Currently, the state of the machine tool is typically determined subjectively by relying on a trained operator's ear, vibration cues and/or shutting the machine down to visually inspect the machine tool.

Other key factors which affect the surface finish of the workpiece include proper coolant/lubricant flow and how well the workpiece is secured in the machine clamp. Similar to the above, machine operators currently rely on visual, auditory and vibration cues to determine proper lubricant flow and the security of the workpiece in the clamp of the machine.

The result is that both types of machine shops lose productivity because they lack real-time feedback instrumentation that characterizes the machining process.

SUMMARY OF THE INVENTION

It is therefore an object of this invention to provide a dynamical instrument for machining.

It is a further object of this invention to provide such a dynamical instrument for machining which provides a real-time measurement of the surface finish of a workpiece

in real time as the workpiece is being machined.

It is a further object of this invention to provide such a dynamical instrument for machining which provides a real-time status of the state of a bit as the workpiece is being machined.

It is a further object of this invention to provide such a dynamical instrument for machining which provides a real-time status of the coolant/lubricant.

It is a further object of this invention to provide such a dynamical instrument for machining which increases the operating capacity of the machine while maintaining an accurate surface finish of the workpiece.

It is a further object of this invention to provide such a dynamical instrument for machining which indicates in real-time if the workpiece is properly secured in the machine clamp.

It is a further object of this invention to provide such a dynamical instrument for machining which indicates in real-time if the machine tool is damaged or worn as the workpiece is being machined.

It is a further object of this invention to provide such a dynamical instrument for machining which indicates in real-time the remaining life of a machine tool.

It is a further object of this invention to provide such a dynamical instrument for machining which eliminates the need for post inspection of the workpiece.

This invention results from the realization that a truly effective dynamical instrument for machining can be achieved by the combination of: 1) a sensor responsive to a non-rotating part of the machine, such as a spindle or tool holder, which is located near the bit or tool of the machine and outputs a vibration signal, and 2) a processor, responsive to the sensor output, which calculates a plurality of unique signature quantities to fully

characterize the dynamics of the vibration signal and which correlates those signature quantities to detect parameters associated with the operation of the machine, such as the measurement of the surface finish of a workpiece, the state of the machine tool bit, and the status of the cooling/lubrication of the machine.

This invention features a dynamical instrument for machining including a sensor responsive to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal, and a processor responsive to the sensor output and configured to calculate a plurality of signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine.

In a preferred embodiment the machine may be a lathe. The machine may be a milling machine. The non-rotating part of the machine may be a tool holder. The non-rotating part of the machine may be a spindle assembly. The tool may be a machine bit.

The dynamical instrument for machining may further include a computer subsystem with a device for outputting the detected parameters associated with the operation of the machine. The device for outputting the detected parameters may include an electronic console. The device for outputting the detected parameters may include an auditory device.

The dynamical instrument for machining may further include a signal conditioner circuit for converting and digitizing the vibration signal output by the sensor to obtain a computer code representation of the sensor time series. The detected parameters may include a measurement of the surface finish of a workpiece as the workpiece is machined. The measurement of the surface finish of a workpiece may include the average absolute value deviation in the height from a mean line of the surface finish of the workpiece.

The measurement of the surface finish of a workpiece may include the difference in height between the highest peak above a mean line of the surface finish of the workpiece and the lowest valley below the mean line of the workpiece over a sampling length of the workpiece. The measurement of the surface finish of a workpiece may include the average peak-to-valley height above the mean line of the surface of the workpiece over the sampling length of the workpiece.

The detected parameters may include an indication of the status of lubricant flow in the machine. The detected parameters may include an indication if the machine tool is worn or damaged. The detected parameters may include an indication of the remaining life of the machine tool. The detected parameters may include an indication how secure a workpiece is in a clamp of the machine as the workpiece is being machined.

An electronic console may output a visual display of the detected parameters associated with the operation of the machine. The console may output a visual alarm of the detected parameters associated with the operation of the machine. An auditory device may output an auditory alarm of the detected parameters associated with the operation of the machine.

The sensor may be an accelerometer. The sensor may include a charge amplifier for outputting a signal with reduced influence of external noise. The sensor may be an acoustic sensor. The acoustic sensor may include an amplifier and a narrow band pass filter for outputting a carrier wave with varying amplitude.

Calculating the number of signature quantities may include determining variations in the amplitude of the vibration signal. The processor may further include a neural network trained to correlate the signature quantities to detect parameters associated with the operation of the machine.

A signature quantity may include the average value of the amplitude of the vibration signal. A signature quantity may be the standard deviation of the amplitude of the vibration signal. A signature quantity may include the characteristic auto-correlation time of the vibration signal. A signature quantity may be the average absolute value difference of the amplitude of the vibration signal. A signature quantity may be the average crossing waves variable of the vibration signal. A signature quantity may be the root mean square difference of the average crossing waves variable of the vibration signal. A signature quantity may be the duration of the average crossing waves variable of the vibration signal. A signature quantity may be the root mean square value of the duration of the average crossing waves variable of the vibration signal. A signature quantity may be the period of the standard deviation crossing waves of the vibration signal. A signature quantity may be the duration of crossing waves of the vibration signal. A signature quantity may be the root mean square duration of standard deviation crossing waves of the vibration signal. A signature quantity may be both the average value of the amplitude of the envelope and the standard deviation of the average value of the amplitude of the envelope of the vibration signal.

This invention further features a dynamical instrument for machining including a sensor responsive to a non-rotating part of the machine proximate the tool of the machine for outputting a vibration signal, and a processor responsive to the sensor output for calculating a number of signature quantities which characterize the dynamics of the vibration signal and correlates the signature quantities to detect parameters associated with the operation of the machine, the processor including an output connected to the machine for varying a machine operating parameters.

In one embodiment, the dynamical instrument for machining may further include a

controller connected to the output of the processor and the machine for varying the machine operating parameters. The controller may be connected to a motor of the machine.

This invention also features a dynamical method for machining including coupling a sensor to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal, and calculating, in response to the vibration signal, a plurality of signature quantities which characterize the dynamics of the vibration signal and correlating the signature quantities to detect parameters associated with the operation of the machine.

In one embodiment the detection of parameters associated with the operation of the machine includes using a neural network trained to calculate the parameters in accordance with the plurality of signature quantities representative of a known parameters associated with the operation of the machine. The parameters may include determining variations in the amplitude of the amplitude envelope of the vibration signal. A signature quantity may be the average value of the amplitude of the vibration signal. A signature quantity may be the standard deviation of the amplitude of the vibration signal. A signature quantity may be the characteristic auto-correlation time of the vibration signal. A signature quantity may be the average absolute value difference of the amplitude of the vibration signal. A signature quantity may be the average crossing waves variable of the vibration signal. A signature quantity may be the root mean square difference of the average crossing waves variable of the vibration signal. A signature quantity may be the duration of the average crossing waves variable of the vibration signal. A signature quantity may be the root means square value of the duration of average crossing waves variable of the vibration signal. A signature quantity may be the

period of standard deviation crossings of the vibration signal. A signature quantity may be the root mean square duration of standard deviation crossing waves of the vibration signal. A signature quantity may be the root mean square duration of standard deviation crossing waves of the vibration signal. Signature quantities may be both the average value of the amplitude of the envelope and the standard deviation of the average value of the amplitude of the envelope of the vibration signal.

The method may further include displaying numerical indicators on an electronic console which represent the parameters associated with the operation of the machine. The method may further including displaying sounding an audio alarm to indicate if the parameters associated with the operation of the machine are outside a predetermined range.

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects, features and advantages will occur to those skilled in the art from the following description of a preferred embodiment and the accompanying drawings, in which:

Fig. 1 is a three-dimensional view of one embodiment of the dynamical instrument for machining in accordance with the subject invention;

Fig. 2 is a three-dimensional view of the dynamical instrument for machining shown in Fig. 1 employing a milling machine and computer sub-system of this invention;

Figs. 3A-3C are graphs showing examples of various examples of measurements of the surface finish of the workpiece in accordance with this invention;

Fig. 4 is a schematic three-dimensional view showing in greater detail examples of the various sensors shown in Fig. 1 mounted on the milling machine shown in Fig. 2;

Fig. 5 is a graph showing the typical output from a sensor in accordance with this invention;

Fig. 6 is a graph depicting one particular signature quantity used to characterize the parameters associated with the operation of the machine in accordance with this invention;

Fig. 7 is a graph depicting another signature quantity used to characterize a flow in accordance with the subject invention;

Fig. 8 is a graph depicting another signature quantity used to characterize the parameters associated with the operation of the machine in accordance with this invention;

Fig. 9 is a block diagram of the various components of one embodiment of the dynamical instrument for machining of this invention;

Fig. 10 is a schematic view of a neural network used to detect parameters associated with the operation of the machine in accordance with this invention;

Fig. 11 is a graph depicting the predicted surface finish versus the actual surface finish; and

Fig. 12 is a flow chart showing the primary steps of the dynamical method for machining in accordance with the subject invention.

DISCLOSURE OF THE PREFERRED EMBODIMENT

Aside from the preferred embodiment or embodiments disclosed below, this invention is capable of other embodiments and of being practiced or being carried out in various ways. Thus, it is to be understood that the invention is not limited in its application to the details of construction and the arrangements of components set forth in

the following description or illustrated in the drawings.

As explained in the Background section above, conventional methods for machining do not provide a machine operator with an objective dynamic measurement of the surface finish of a workpiece as it is being manufactured. Therefore, operators must rely on subjective visual, auditory, and/or vibration cues to adjust spindle speed, depth of cut, and rate of advance of the cutting tool to insure proper surface finish. Conventional methods also do not provide an operator with dynamic information to determine whether a machine tool is worn or damaged, whether proper cooling/lubricant flow is occurring, or how well the workpiece is secured in the machine clamp. Similarly, operators must rely on visual, auditory and vibration cues for these.

In contrast, dynamical instrument for machining 10, Fig. 1 of the subject invention includes sensor 12 responsive to non-rotating part 14 (e.g., a spindle or tool holder) of machine 16 proximate tool 20 (e.g., a machine bit) for outputting a vibration signal, such as vibration signal 24. Processor 22 is responsive to the output of sensor 12 and is configured to calculate a plurality of signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine, such as the resulting surface finish 34 of workpiece 36, whether machine tool 20 is worn or damaged, and the status of the lubricant/coolant of the machine.

By uniquely calculating a number of signature quantities which characterize the dynamics of the vibration signal and then correlating the signature quantities to detect parameters associated with the operation of the machine, system 10 provides real time detection of surface finish, the state of the machine tool and the status of the lubricant flow. The result is the elimination of a machine operator having to rely on visual,

auditory, and/or vibration cues to adjust the spindle speed, depth of cut, and rate of advance of the cutting tool to insure proper surface finish of the workpiece. Instead, the operator or a machine controller is able to make the adjustment based on real-time information provided by system 10. Post inspection of the workpiece can be eliminated. By dynamically providing information about machine tool 20 (e.g., the remaining tool life), the machine operator knows exactly when to shut the machine down and replace the tool allowing the operator to run machine 16 at its full capacity and machine rate resulting in more efficient production of the workpiece.

In one example of this invention, machine 16 is a lathe. In other designs, machine 16 is a milling machine, such as an Enco Model 1005200 knee mill (Enco Manufacturing Co., Chicago, IL), 47, Fig. 2. In this example, the spindle speed of milling machine 47 can be adjusted from 215 to 3440 rpm (3.58 to 57.3 Hz) using V-belts and may be employed with three spindle speeds of 860, 1720, and 3440 rpm (14.3, 28.7, and 57.3 Hz). To vary the feed rate, a variable speed drive may be utilized from an Enco Model 205-6404, and modified for use on the Enco Model 1005200 knee mill to provide table at speeds up to approximately 0.44 in/s (11.2 mm/s). Typical milling variables used in accordance with this invention typically include spindle speed, table advance, e.g., feed rate, radial depth of cut, as shown at 35, Fig. 1 of workpiece 36, axial depth of cut, shown at 37, and the stiffness of workpiece 36.

Non-rotating part 14 of machine 16 is typically a tool holder or a spindle assembly. Typically, tool 20 of machine 16 is a machine bit, such as the machine bit indicated at 28, Fig. 3. In this example, the machine bit is a high speed steel end mill and is 0.500 in (12.7 mm) in diameter, includes two helical flutes (with a lead of 2.720 in, 69.09 mm), and may be coated with TiN for wear resistance.

System 10, Figs. 1 and 2 may also include computer sub-system 30 with a device, such as electronic console 32 for outputting the detected parameters associated with the operation of machine 16. In one design, the device for outputting the detected parameters may be an auditory device, such as speaker 34, Fig. 1.

Dynamic instrument for machining 10, or system 10, may also include signal conditioner 33 for converting and digitizing the vibration signal (e.g., vibration signal 24) output by sensor 12 into voltage as a function of time. The output from signal conditioner 32 is input to processor 22 which, as shown above, is configured to calculate a plurality of signature quantities which characterize the dynamics of vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine.

As shown above, system 10 calculates a plurality of signature quantities which characterize the dynamics of the vibration signal and correlates the signature quantities to detect parameters associated with the operation of the machine. The detected parameters associated with the operation of machine 10 may include measurement of surface finish 34 of workpiece 36 as workpiece 36 is machined. The measurement of the surface finish 34 of workpiece 36 may include the average absolute value deviation in height from a mean line of the surface finish 34 of workpiece 36, R_a as indicated at 51, Fig. 3A. In other examples, the measurement of surface finish 34, Fig. 1 of workpiece 36 may include a difference in height between the highest peak above the mean line of surface finish 34 of workpiece 36 and the lowest value below the mean line of workpiece 34 over sampling length 38 of the workpiece 36, R_y indicated at 53, Fig. 3B. In other examples, the measurement of surface finish 34, Fig. 1 of workpiece 36 may include the average

peak to valley height above the mean line of the surface of workpiece 36 over sampling length 38 of workpiece 36, R_z . R_z is calculated using the formula:

$$R_z = \frac{R_{z1} + R_{z2} + R_{z3} \dots + R_{zn}}{n} \quad (1)$$

For example, as shown in Fig. 3C, where five surface measurements are taken, e.g., R_{z1} , R_{z2} , R_{z3} , R_{z4} and R_{z5} , indicated at 55, 57, 59, 61 and 63, respectively, R_z is calculated as:

$$R_z = \frac{R_{z1} + R_{z2} + R_{z3} + R_{z4} + R_{z5}}{5} \quad (2)$$

Ideally, system 10 displays the detected surface finish measurements on electronic console 32, as indicated at 40, Fig. 1. Other parameters detected and displayed by dynamical instrument for machining 10 associated with the operation of machine 16 include an indication if the machine tool 20 is worn or damaged, displayed on electronic console 32 as remaining life, as shown at 42. The status of the lubricant flow may also be displayed on electronic console 32, as shown at 44. Dynamical instrument for machining 10 may also provide visual alarms 46 on console 32 which may include status of the surface finish alarm 47, the lubricant flow alarm 49, and clamp alarm 51, which indicates if the workpiece 36 is not well secured in a clamp 53.

Visual alarms which indicate the security of the workpiece in the clamp (e.g., clamp alarm 51), the surface finish (e.g., surface finish alarm 47), and the lubricant flow (e.g., lubricant flow alarm 49), in conjunction with auditory alarms (e.g., speaker 34) quickly alert the machine operator to any deviation in the surface finish or malfunctioning of machine 16.

In one design in accordance with this invention, sensor 12 is an accelerometer, such as an Endevco Model 7259A-25 available from Endevco Corp., San Juan Capistrano, CA, as shown in detail in Fig. 4. In this embodiment, sensor 12 features a very compact body with a height of only 0.5 inches above the mounting surface and is installed using a mounting stud with a number 10-32 screw thread. Ideally, sensor 12 is unidirectional with sensor 12 sensing direction perpendicular to the mounted surface in a 1.3 % maximum off-axis sensitivity. Sensor 12 as employed in this design typically has a high resonant frequency of approximately 93 kHz rendering sensor 12 well suited for dynamic measurement applications. In one example, sensor 12 may include a charge amplifier (not shown) built into the case sensor 12 for reducing external noise. The output from sensor 12 may be amplified to the level required for data acquisition by using a Kistler Model 504E piezoelectric sensor amplifier (Kistler Instrument Corp., Amherst, NY) (not shown) operated in isotron mode. The output range of the sensor amplifier combined with sensor 12 produces an overall output of 0.4734 V/g. The mounting arrangement of sensor 12 mounted on a machine 16 is shown in Fig. 4.

System 10, Fig. 1 may also include acoustic sensor 46, such as a ClampOn 2000 manufactured by ClampOn AS of Norway. Sensor 46 is typically mounted to machine 16, such as the Enco Model 1005200 knee mill indicated at 47, Fig. 2, as shown in Fig. 4. Sensor 46 may employ a piezoelectric sensor with a resonant frequency of approximately 83 kHz and may also include a high gain amplifier which ranges from 10 to 66 dB. The amplifier may also incorporate a narrow band pass filter centered at approximately 61 kHz. In operation, the output of sensor 46 may be a 61 kHz carrier wave 60, Fig. 5 with varying (modulated) amplitude. Ideally, a filter centered on the resonant frequency of sensor 46 is used to filter the transient events that inevitably occur during machining

(e.g., each tooth strike) and other behaviors, such as noise arising from the sensor (e.g., quantum noise), and surrounding sources, such as motors and fluorescent lighting. As shown in Fig. 4, the sensor head of sensor 46 is designed for use in very rugged working environments. In one example, sensor 46 is housed in a very heavy stainless steel housing approximately 1.9 inches in diameter and 3.7 inches long. Sensor 46 is ideally mounted on the opposite side of collar 48, Fig. 5 of sensor 12.

The collection of dynamic data from sensors 12 and 46 is ideally performed by a computer board, such as a CIO-DAS16/330 Data Acquisition Board (Measurement Computing Corp., Middleboro, MA). In one design, the computer board is capable of collecting data at 12-bit resolution for up to 16 single-ended (or 8 double-ended) channels at a total acquisition rate of up to 330 kHz. The computer board is typically operated using a computer program, written in a language such as Visual Basic. In one embodiment, the computer board is operated in a REP INS W (Repeat Instruction Word) mode. Because a high sampling rate is desired in accordance with dynamical instrument for machining 10, a single range is typically selected for all four of the input channels. To collect representative sample of data for each machining condition at the highest possible sampling rate while maintaining a manageable quantity of data to analyze, five seconds of data may be collected for each condition, which corresponds to only 143 tooth strikes at the slowest spindle speed. This results in 1,650,000 samples which when stored in ASCII format would which consume approximately 20 Mb each. In 82 tests over 1.55 Gb of raw data is collected. Even with this high rate of data acquisition, the key dynamic sensors 12 and 46 are recorded at 82.5 kHz, which is insufficient to fully capture the dynamics of the signals. Thus, the subject invention is succesful even with undersampled data.

System 10 operates by analysing the vibration signal (e.g., vibration signal 24) provided by sensor 12 and/or sensor 46 and calculating a set of quantities that characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine. Calculating the signature quantities includes determining variations in the amplitude of the vibration signal. Some of these signature quantities are simple and straight forward. Examples include the signal's average value or its standard deviation. Others are more involved, intended to quantify specific aspects of the signal, such as the characteristic sizes, durations, or sequences of behaviors in the signal dynamics. If the data is collected for a sufficient time period, the "undersampled" data stream will capture a representative set of readings. Even though the standard deviation of the sampled data is not equal that of the actual signal, there is a one-to-one relationship between the values found for the sampled data and for the actual signal.

One such signature quantity, (Avg) is the average value of the vibration signal as shown at 80, Fig. 6. Another signature quantity is the standard deviation of the signal (StDev) which reflects the harmonic power of the signal and is an important variable for any irregularly varying signal that carries significant information. A third signature quantity is the characteristic auto-correlation time (Auto). This is the time scale during which the signal loses linear correlation and characterizes the rate of mixing in a process. The auto-correlation quantity is the integral over all time of the signal multiplied times a delayed version of the signal. For long time delays, the auto-correlation approaches:

$$\frac{1}{n} \sum_{i=1}^n x_i^2 \quad (3)$$

where n is the number of readings in the record. For no time delay, the auto-correlation equals:

$$\sum x^2 \quad (4)$$

measured in sampling intervals. The auto-correlation time is:

$$\delta t_{auto} = \frac{\sum x^2 - n\bar{x}^2}{\sum x^2 - \sum x_i x_{i+1}} \quad (5)$$

A fourth flow indicator quantity is the average difference (Dif), more specifically the average absolute value difference in sequential readings. This flow indicator quantity provides a first order indication of how rapidly the data change on the smallest time scale. It is analogous to the first surface finish parameter, R_a , which is a similar measure applied to a surface profile measurement.

A fifth flow indicator quantity is the root mean square difference (RMSDif) in sequential readings, a measure similar to the average difference, but tends to weigh larger changes in sequential readings more strongly than smaller changes.

The remaining flow indicator quantities are calculated by replacing the raw time series data with the moving standard deviation of raw data. That is, the standard deviation of the first twenty readings of the data record is calculated, the window advanced ten readings, and the process repeated until the moving standard deviation of the entire record is calculated. The resulting standard deviation vector characterizes the harmonic power present in the signal as a function of time. Consequently, flow indicator quantities calculated from this vector characterize the variation of harmonic power with time. Such flow indicator quantities are identified by the prefix "Sd".

Another flow indicator quantity is the period of average crossing waves, (SdaPd), which is reflective of the average number of readings that occur between upward crossings of the average value of the amplitude as shown at 82, Fig. 7, and characterized

as the time length of first order signal variations.

Another flow indicator quantity, (SdaRMSpd), is the root mean square value of the period of average crossing waves variable which emphasizes longer-period waves. A related flow indicator quantity, (SdaDur), is the duration of average crossing waves which is a measure of the average number of readings that occur between an upward and downward crossing of the average amplitude value as shown at 82, Fig. 7. A related flow indicator quantity is the root mean square value of the duration of average crossing waves, (SdaRMSDur).

The period of standard deviation crossing waves flow indicator quantity, (SdsPd), represented in Fig. 8, is a measure similar to the average period of average crossing waves, but in this case the bar is raised somewhat to look at the period of waves that cross the average plus the standard deviation as shown at 84, Fig. 8. A related flow indicator quantity is the root mean square value of the period of standard deviation crossing waves value, (SdsRMSPd). Another flow indicator quantity, (SdsDur), is the duration of standard deviation crossing waves as shown in 86, Fig. 8. Finally, a related flow indicator quantity is the root mean square duration of standard deviation crossing waves, (SdsRMSDur). Other flow indicator quantities, however, are within the scope of this invention.

One version of a Quick Basic 4.5 programming language running under a DOS operating system, operated on computer subsystem 32, Figs. 1 and 2 for calculating the above signature quantities is provided in the computer program listing as shown below,

```
' Program DOC96195:  accepts command line input of a DataFile$ name or the
' name of a file containing a list of data files (FileList$).  It opens
' each DataFile$ and calculates indicator quantities for the first 3.75 sec
' of data in the file (309,375 points, as limited by QuickBasic).  It
' calculates quantities for both the raw data (original time series) and for a
' dataset created by calculating moving standard deviations in overlapping
' 20-point windows.  The results are placed in an output file named using the
' input file name plus the extension .OUn, where n is the number of the column
' that was analyzed (0, 1, or 2).  For a FileList$ input file, the output file
```

```

' contains one line for each input file named in the list.

' Usage:
'       DOC96195.EXE [DataFile$] [-L FileList$] n

' Without arguments, program provides a description of usage syntax.
' With DataFile$ provided, program analyzes the single file and puts results
'   in a new file named the same as DataFile$ but with the extension .OUT.
' With -L FileList$ provided, program analyzes each file listed in FileList$
'   and puts results in a new file with the same name as FileList$ but with
'   the extension .OUn.
' The integer n is the column number to analyze (0, 1, or 2)

' Metaccommand that uses maximum available conventional memory.
' $DYNAMIC

' Default variable type is integer in this module.
DEFINT A-Z

' Declare the Comline subprogram (which retrieves the program command line),
' as well as the number and type of its parameters. Comline is included as an
' example routine with QuickBasic 4.5, is appended to this listing.
DECLARE SUB Comline (N, a$(), Max)

' Dimension arrays
DIM a$(1 TO 15) ' Contains the command line parameters
DIM f$(1 TO 100) ' Contains list of input files
DIM array!(-21000 TO 11001) ' Contains array of windowed standard deviations

' Get what was typed on the command line.
CALL Comline(N, a$(), 6)

COLOR 0, 15 ' Makes the screen look better in a Windows NT DOS box.

' Figure out whether command line is workable:

IF N < 2 GOTO Bomb ' Not enough command line arguments
IF N > 3 GOTO Bomb ' Too many arguments
IF N = 3 AND (a$(1) <> "-L" AND a$(1) <> "-l") GOTO Bomb ' Wrong syntax for
' FileList$ operation
IF N = 3 THEN
    column = VAL(a$(3))
ELSE
    column = VAL(a$(2))
END IF

IF column <> 0 AND column <> 1 AND column <> 2 GOTO Bomb ' Need an integer there
GOTO NoBomb ' No problems so far...

Bomb: ' Give user program usage syntax pointers, then quit.
PRINT "
PRINT "       DOC96195:  Calculates signature quantities for 1 or more
PRINT "                   input data files.
PRINT "
PRINT "       Usage:  DOC96195 [DataFile] [-L ListFile] n
"
PRINT "
PRINT "       1) DataFile - path and name of single input file, must be 4
PRINT "                   columns of normal integers (-32768 to +32767).
PRINT "       2) ListFile - path and name of text file containing a list of
PRINT "                   paths + names of files to be analyzed (4 columns).
PRINT "       3) Command line can contain DataFile_OR_ListFile (not both).
PRINT "       4) The integer n is the column number to analyze
PRINT "       5) Results are placed in an output file named using input file
PRINT "                   name with extension replaced by .OUn, where n is the column
PRINT "                   number.
PRINT "
END

NoBomb: ' OK, so far so good. Next, open the input list file (if there

```

```

        ' is one) and create an output file.

IF N = 2 THEN      ' Only one input file.
    f$(1) = a$(1)
    iFiles = 1

ELSE              ' N=3, FileList$ input, get list of files

    FileList$ = a$(2)
    ON ERROR GOTO Handle1
        flag = 1
        OPEN FileList$ FOR INPUT AS #1
    ON ERROR GOTO 0

    iFiles = 0      ' Number of files in list
    DO WHILE NOT EOF(1) ' Read list of input files
        INPUT #1, f$(iFiles + 1)
        iFiles = iFiles + 1
    LOOP
    CLOSE #1        ' Done with FileList$
    flag = 2
    IF iFiles = 0 THEN GOTO Handle1 ' No file names
END IF

'Next, create output file with name based on FileList$ or DataFile$, as
' appropriate.

IF N = 2 THEN
    file$ = a$(1)
ELSE
    file$ = FileList$
END IF

    ' Find extension in file name
    FOR ichar = 1 TO LEN(file$)
        IF MID$(file$, ichar, 1) = "." THEN
            ' everything to the right of this character must be extension
            IF column = 0 THEN
                oFile$ = LEFT$(file$, ichar) + "OU0"
            END IF
            IF column = 1 THEN
                oFile$ = LEFT$(file$, ichar) + "OU1"
            END IF
            IF column = 2 THEN
                oFile$ = LEFT$(file$, ichar) + "OU2"
            END IF
        END IF
    NEXT ichar

' Plug hole for case where file didn't have an extension initially
IF LEFT$(RIGHT$(oFile$, 4), 3) <> ".OU" THEN
    IF column = 0 THEN
        oFile$ = oFile$ + ".OU0"
    END IF
    IF column = 1 THEN
        oFile$ = oFile$ + ".OU1"
    END IF
    IF column = 2 THEN
        oFile$ = oFile$ + ".OU2"
    END IF
END IF

    ' Open output file
    ON ERROR GOTO Handle1
        flag = 3
        OPEN oFile$ FOR OUTPUT AS #2
    ON ERROR GOTO 0

    ' Write header line in output file

    PRINT #2, "   FileName      Average      StDev      Auto      AvgDif      RMSDif  ";
    PRINT #2, "      sdAvg      sdStdDev      sdAuto      sdAvgDif      sdRMSDif      sdaPd  ";
    PRINT #2, "      sdaRMSPd      sdaDur&      sdaRMSDur      sdsPd      sdsRMSPd      sdsDur  ";

```

```

PRINT #2, " sdsRMSDur"

' Next, analyze each input file in turn and print results in oFile$.
FOR fileNum = 1 TO iFiles ' For each data file
  ON ERROR GOTO Handle2 ' Data file open error
  OPEN f$(fileNum) FOR INPUT AS #1
  ON ERROR GOTO 0
  PRINT "Analyzing "; f$(fileNum); "."

' First, go through data set to calculate preliminary numbers and create an
' array of overlapping windowed standard deviations.

J& = 0 ' Number of values found in file (so far)
FirstCnt = 0 ' Window counter for first std dev window
FirstSum& = 0 ' Accumulator for first window sum
FirstSum2# = 0 ' Accumulator for first window squares
SecondCnt = 0 ' Window counter for second std dev window
SecondSum& = 0 ' Accumulator for second window sum
SecondSum2# = 0 ' Accumulator for second window squares
NArray = 0 ' Number of standard deviation windows generated

Sum& = 0 ' Accumulator of sum of raw values
Sum2# = 0 ' Accumulator of sum of values squared
SumDif# = 0 ' Accumulator for sum of absolute value changes in seq rdgs.
SumDif2# = 0 ' Accumulator for sum of squares of differences in seq rdgs.
Corr# = 0 ' Accumulator for product of current value and preceding value.

DO WHILE NOT EOF(1)
  IF column = 0 THEN
    INPUT #1, value, x, y, z
  END IF
  IF column = 1 THEN
    INPUT #1, x, value, y, z
  END IF
  IF column = 2 THEN
    INPUT #1, x, y, value, z
  END IF
  IF NArray = 30937 GOTO Loopy 'Out of array space: can't use more data
  J& = J& + 1
  FirstCnt = FirstCnt + 1
  FirstSum& = FirstSum& + value
  FirstSum2# = FirstSum2# + value ^ 2
  IF J& > 10 THEN ' OK to work in second window (starts at J&=11)
    SecondCnt = SecondCnt + 1
    SecondSum& = SecondSum& + value
    SecondSum2# = SecondSum2# + value ^ 2
  END IF
  IF FirstCnt = 20 THEN ' Ready to put value in array and reset window
    NArray = NArray + 1
    IF FirstSum2# / 20 > (FirstSum& / 20) ^ 2 THEN
      array!(NArray - 20999) = SQR(FirstSum2# / 20 - (FirstSum& / 20) ^
2)
    ELSE
      array!(NArray - 20999) = 0
    END IF
    FirstCnt = 0
    FirstSum& = 0
    FirstSum2# = 0
  END IF
  IF SecondCnt = 20 THEN ' Similarly for second window
    NArray = NArray + 1
    IF SecondSum2# / 20 > (SecondSum& / 20) ^ 2 THEN
      array!(NArray - 20999) = SQR(SecondSum2# / 20 - (SecondSum& / 20) ^
2)
    ELSE
      array!(NArray - 20999) = 0
    END IF
    SecondCnt = 0
    SecondSum& = 0
    SecondSum2# = 0
  END IF
  IF J& > 1 THEN 'Not first data point
    SumDif# = SumDif# + ABS(value - oldvalue&)

```

22

```

        SumDif2# = SumDif2# + (value - oldvalue&) ^ 2
        Corr# = Corr# + value * oldvalue&
    END IF
    Sum# = Sum# + value
    Sum2# = Sum2# + value ^ 2
    oldvalue& = value
Loopy: LOOP

CLOSE #1      ' Done with raw data, so close file.

PRINT "There are "; NArray; " points in the std-dev array."

'Calculate a few results
Ave! = Sum# / J#      ' Average value (presumably near zero for most sensors)
SDev! = SQR(Sum2# / J# - Ave! ^ 2)  ' Standard deviation
Auto! = (Sum2# - J# * Ave! ^ 2) / (Sum2# - Corr#)  ' Characteristic
                                                    ' autocorrelation time
AvgDif! = SumDif# / (J# - 1)  ' Average absolute value change in seq readings
RmsDif! = SQR(SumDif2# / (J# - 1))  ' RMS change in sequential readings

' Next go through the standard deviation array a couple of times.
' The first time, find average, standard deviation, and autocorrelation times.

Sum! = 0          ' Accumulator of sum of raw values
Sum2# = 0         ' Accumulator of sum of values squared
SumDif# = 0       ' Accumulator for sum of absolute value changes in seq rdgs.
SumDif2# = 0      ' Accumulator for sum of squares of differences in seq rdgs.
Corr# = 0         ' Accumulator for product of current value and last value.

FOR ptNum = 1 TO NArray
    Sum! = Sum! + array!(ptNum - 20999)
    Sum2# = Sum2# + array!(ptNum - 20999) ^ 2
    IF ptNum > 1 THEN
        SumDif# = SumDif# + ABS(array!(ptNum - 20999) - array!(ptNum - 21000))
        SumDif2# = SumDif2# + (array!(ptNum - 20999) - array!(ptNum - 21000)) ^ 2
        Corr# = Corr# + array!(ptNum - 20999) * array!(ptNum - 21000)
    END IF
NEXT ptNum

' Calculate some preliminary results

sdAvg! = Sum! / NArray
sdStdDev! = SQR(Sum2# / NArray - sdAvg! ^ 2)  ' Std dev of std dev's (!)
sdAuto! = (Sum2# - NArray * sdAvg! ^ 2) / (Sum2# - Corr#) * 10
            ' Characteristic autocorrelation time of std dev data,
            ' with time base adjusted to original data rate (*10, rather
            ' than *20, because std dev windows overlap, so one step in
            ' std dev data progresses 10 time steps in original data).
sdAvgDif! = SumDif# / (NArray - 1)  ' Avg abs value change in seq std dev's
sdRmsDif! = SQR(SumDif2# / (NArray - 1))  ' RMS change in seq std devs

' Next, find and characterize large events (crossings of average and
' avg+(std-dev) lines).

' Note: the following is not the most efficient method of performing this
' analysis, but is logically simple and easier to read (and debug). A final
' version could probably be performed in half as many operations.

' Crossings of the average value line:

' First, find the first positive crossing of average line:

x = 1
y = 0
DO UNTIL y = 1
    IF array!(x + 1 - 20999) > sdAvg! AND array!(x - 20999) < sdAvg! THEN y = 1
    x = x + 1
LOOP

'Resulting X is the first positive crossing of the average line

```

```

' Next, start there and find number of positive crossings and duration of
' periods above average and periods between positive crossings

aNum = 0          ' No +ve avg crossings counted yet
avgCross = x      ' Pointer to most recent +ve crossing of avg
aDur& = 0         ' Accumulator for duration above avg
aDur2& = 0        ' Accumulator for duration squared above avg
aPd2& = 0         ' Accumulator for square-length of crossing period

FOR ptNum = x + 1 TO NArray
  IF array!(ptNum - 20999) > sdAvg! AND array!(ptNum - 21000) < sdAvg! THEN '
+ve
    ' crossing
    aNum = aNum + 1 'Count the crossing
    aPd2& = aPd2& + (ptNum - avgCross) ^ 2 'Accumulate square period
    avgCross = ptNum 'Update pointer to +ve crossing
  END IF
  IF array!(ptNum - 20999) < sdAvg! AND array!(ptNum - 21000) > sdAvg! THEN '
-ve
    ' cross
    aDur& = aDur& + ptNum - avgCross ' Accumulate duration above bar
    aDur2& = aDur2& + (ptNum - avgCross) ^ 2 ' Accum dur^2 above bar
    lastMinus = ptNum 'Store pointer to -ve crossing
  END IF
NEXT ptNum

' Calculate statistics for average-crossing events

sdaPd! = (lastMinus - x) / aNum ' Avg period of average-crossing events
sdaRmsPd! = SQR(aPd2& / aNum) ' RMS -----"-----
sdaDur! = aDur& / aNum ' Avg duration of time above avg
sdaRmsDur! = SQR(aDur2& / aNum) ' RMS -----"-----

' Crossings of the average+(standard deviation) line:

' First, find the first positive crossing of avg+(std-dev) line:

x = 1
y = 0
bar! = sdAvg! + sdStdDev!
DO UNTIL y = 1
  IF array!(x + 1 - 20999) > bar! AND array!(x - 20999) < bar! THEN y = 1
  x = x + 1
LOOP

'Resulting X is the first positive crossing of the avg+(std-dev) line

' Next, start there and find number of positive crossings and duration of
' periods above the bar and periods between positive crossings

aNum = 0          ' No +ve avg+(std-dev) crossings counted yet
avgCross = x      ' Pointer to most recent +ve crossing of avg+(std-dev)
aDur& = 0         ' Accumulator for duration above avg+(std-dev)
aDur2& = 0        ' Accumulator for duration squared above avg+(std-dev)
aPd2& = 0         ' Accumulator for square-length of crossing period

FOR ptNum = x + 1 TO NArray
  IF array!(ptNum - 20999) > bar! AND array!(ptNum - 21000) < bar! THEN ' +ve
    ' crossing
    aNum = aNum + 1 'Count the crossing
    aPd2& = aPd2& + (ptNum - avgCross) ^ 2 'Accumulate square period
    avgCross = ptNum 'Update pointer to +ve crossing
  END IF
  IF array!(ptNum - 20999) < bar! AND array!(ptNum - 21000) > bar! THEN ' -ve
    ' cross
    aDur& = aDur& + ptNum - avgCross ' Accumulate duration above bar
    aDur2& = aDur2& + (ptNum - avgCross) ^ 2 ' Accum dur^2 above bar
    lastMinus = ptNum 'Store pointer to -ve crossing
  END IF
NEXT ptNum

```

```

' Calculate statistics for avg+(std-dev) crossing events

sdsPd! = (lastMinus - x) / aNum 'Avg period of events above avg+(std-dev)
sdsRmsPd! = SQR(aPd2& / aNum)   'RMS -----"-----
sdsDur! = aDur& / aNum          'Avg duration of time above bar
sdsRmsDur! = SQR(aDur2& / aNum) 'RMS -----"-----

' Print results in output file:

' First, find 8.3 format name of input file (i.e., strip off path)

fileName$ = "" ' Preset to null in case path isn't present
FOR i = 1 TO LEN(f$(fileNum))
  IF MID$(f$(fileNum), LEN(f$(fileNum)) + 1 - i, 1) = "\" THEN
    fileName$ = RIGHT$(f$(fileNum), i - 1)
  EXIT FOR
END IF
NEXT i

IF fileName$ = "" THEN fileName$ = f$(fileNum) ' No path

SELECT CASE LEN(fileName$):
CASE 12: ' Sheesh, what we won't do for a clean output format!
PRINT #2, fileName$;
CASE 11:
PRINT #2, " "; fileName$;
CASE 10:
PRINT #2, " "; fileName$;
CASE 9:
PRINT #2, " "; fileName$;
CASE 8:
PRINT #2, " "; fileName$;
CASE 7:
PRINT #2, " "; fileName$;
CASE 6:
PRINT #2, " "; fileName$;
CASE 5:
PRINT #2, " "; fileName$;
CASE 4:
PRINT #2, " "; fileName$;
CASE 3:
PRINT #2, " "; fileName$;
CASE 2:
PRINT #2, " "; fileName$;
CASE 1:
PRINT #2, " "; fileName$;
END SELECT

PRINT #2, USING "#####.###"; Ave!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; SDev!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; Auto!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; AvgDif!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; RmsDif!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdAvg!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdStdDev!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdAuto!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdAvgDif!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdRmsDif!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdaPd!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdaRmsPd!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdaDur!;

```

```

PRINT #2, " ";
PRINT #2, USING "#####.###"; sdaRmsDur!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdsPd!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdsRmsPd!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdsDur!;
PRINT #2, " ";
PRINT #2, USING "#####.###"; sdsRmsDur!

Restart:      'Re-entry point for missing input file

NEXT fileNum

CLOSE #2

END

Handle1:      ' Fatal errors that require program termination:
PRINT
IF flag = 1 THEN PRINT "Couldn't open list file "; FileList$; " for
input."
IF flag = 2 THEN PRINT "No file names in "; FileList$; "."
IF flag = 3 THEN PRINT "Couldn't open "; oFile$; " for output."
PRINT
END

Handle2:      ' Non-fatal errors that allow program to continue:
PRINT
PRINT "Can't open file "; f$; " for input."
ON ERROR GOTO 0
GOTO Restart

REM $STATIC
SUB Comline (NumArgs, Args$(), MaxArgs) STATIC
CONST TRUE = -1, FALSE = 0

    NumArgs = 0: in = FALSE
    ' Get the command line using the COMMAND$ function.
    Cl$ = COMMAND$
    L = LEN(Cl$)
    ' Go through the command line a character at a time.
    FOR i = 1 TO L
        C$ = MID$(Cl$, i, 1)
        'Test for character being a blank or a tab.
        IF (C$ <> " " AND C$ <> CHR$(9)) THEN
            ' Neither blank nor tab.
            ' Test to see if you're already
            ' inside an argument.
            IF NOT in THEN
                ' You've found the start of a new argument.
                ' Test for too many arguments.
                IF NumArgs = MaxArgs THEN EXIT FOR
                NumArgs = NumArgs + 1
                in = TRUE
            END IF
            ' Add the character to the current argument.
            Args$(NumArgs) = Args$(NumArgs) + C$
        ELSE
            ' Found a blank or a tab.
            ' Set "Not in an argument" flag to FALSE.
            in = FALSE
        END IF
    NEXT i

END SUB

```

This program accepts a command line in the form: DOC96195[DataFile\$][-1FileList\$] n.

If analysis of a single data file is of interest, then DataFile\$ field above is filled with the

path name and file containing the data. If a list of files is to be analyzed, then the minus 1 switch is used and the path and name of the ASCII file containing a list of files used as the FileList\$ variable. The command line parameter n was the column number to be analyzed (0 for the accelerometer, 1 for the passive acoustic sensor). In either event, single ASCII output files are created from the DataFile\$ or ListFile\$, with the name extension replaced with "Oun", where n is the number of the column containing the data that was analyzed. The file contains one row that is a header describing the names of each of the signature quantities calculated, plus one row for each data file that was analyzed containing values corresponding to signature quantities.

In one embodiment of this invention, the signature quantities are presented to network correlation subsystem 70, Fig. 9 operating on computer subsystem 30, Fig. 1 to provide real time output to electronic console 32. Raw data 71, Fig. 9 from sensor 12 mounted on machine 16 is amplified, digitized and fed into feature extract analysis 73. The set of signature quantities resulting from this analysis, plus other useful settings (such as spindle speed and advance rate, shown at 72), are fed to a neural network correlation 70 network, which determines the surface finish that best matches the dynamics of the data. The surface finish, plus other outputs such as a Tool Wear, Coolant OK, and Clamping OK, as shown at 74, are displayed in real-time to assist the machine operator or machine controller to modify the machining process (e.g., increase or decrease spindle speed, cutting depth, and/or advance rate). Typical inputs from machine 16 may include spindle speed, or table speed, as shown at 72. Neural network 80, Fig. 10 is then used to determine the signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of the machine.

Neural network 80, Fig. 10 is then used to determine the signature quantities which characterize the dynamics of the vibration signature and correlate the signature quantities to detect parameters associated with the operation of the machine. The primary reason neural network 80 is used is that it has the ability to handle arbitrarily non-linear correlations involving abrupt transitions, strongly varying behaviors, or complicated interactions among input variables. In addition, neural network 80 does not require any assumptions about the form or the correlation between a set of inputs and the desired set of outputs. Neural network 80 is programmed by training a set of weighted connections, rather than by explicitly defining a set of operations that must be performed. Once trained, neural network 80 is then implemented in a very fast, efficient software, firmware, or hardware environment rendering it ideally suited for real time applications. In addition, once properly trained, neural network 80 is extraordinarily fault tolerant, and/or fault indicating. Thus, if the data is noisy or occasionally erroneous or missing, a neural network can indicate this and still provide a reasonable estimate of the operating conditions.

In practice, neural network 80 is trained by first inputting known indicator quantities for a broad range of known conditions and training the network to handle those particular cases. Once network 80 is trained, it can be applied to a new set of indicator quantities providing a means of analyzing data in real world applications.

Neural network 80 is best viewed as being made up of a layer of neurons. The most common type of a neural network, called a "back propagation" network, often contains three layers: input layer 82, hidden layer 84, and output layer 86. The input layer receives the input information, in this case the signature quantities discussed above. Each of the inputs is scaled from its full range to a range of (-1, 1) which the network is

very good at handling. No processing is done at the input layer, instead, each neuron simply passes along its input value which is fed to the next layer through a set of weighted connections.

Usually, successive layers are thoroughly interconnected so the weighted connections link each neuron in input layer 82 to each neuron in hidden layer 84. The value that reaches a hidden neuron along a given connection is the product of the value injected into the connection by the input neuron and the value of the connection weight. In hidden layer 84, each neuron sums the values that reach it via the connections and then applies a simple, non-linear transformation to that value. Gaussian (bell-shaped) and sigmoidal (S-shaped) functions are the most commonly used, partly because they are easily calculated.

In a preferred embodiment, neural network 80 uses a sigmoidal function called a "logistic" function. This function maps any input value in an infinite range to an output of $(-1, 1)$. The resulting value is then fed forward along the weighted connections to output layer 86. Output layer 86 functions identically to the hidden layer, summing the weighted values that are presented to it, applying a logistic transformation and outputting the result. The output value in the range $(-1, 1)$ is scaled back to the full range of interest, producing a final result of the network, including, in one example, surface finish 87, chatter proximity 89, tool sharpness 91, sufficient coolant 93, loose workpiece 95, and fault 95.

In the subject invention, the eighteen signature quantity inputs are used for each data set and three outputs corresponding to the machine rates or yes/no answers. The number of hidden neurons can vary considerably, depending upon the difficulty of the required calculation. More neurons permit greater complexity, but slow the processing

speed. The “NeuroShell 2” neural network simulation program available from Ward Systems Group may be used to implement neural network 80, Fig.10.

Training back propagation network 80 requires the preparation of a set of training data sets containing both the inputs and the desired outputs. The weighting values initially are selected at random. During training, training cases are chosen at random, presented to the network, and the error in the network’s output is “back propagated” through the network to determine minimum changes to the weighted values that will eliminate the error. By performing this process over and over, updating the weighted values as needed, the network’s calculation accuracy continually improves until, finally, it is not improved further. If the network’s accuracy is acceptable at that point, it can be used as desired to rapidly produce an output value for the set of input values.

Neural network 80 may be subject to “over-training” unless care is taken to train it properly. Over-training is the ability of the network to recognize individual training sets, rather than the basic variation in the patterns between different training sets. Over-trained networks often perform poorly when used to estimate the state of new input sets. This problem can be avoided by using one-sub-set of the available data sets as training data, and the remainder as test data. The network is trained using the training sets, but a new set of values for the connection weights is saved only when it performs better with the test data (which were not used to train the network). This method usually results in a network with satisfactory ability to “generalize” beyond the training data to the larger range of possible inputs.

Each data file of the raw data is filtered into a signature file using the computer program described above. The signature files are merged with a master spreadsheet that contains machining conditions and resulting surface finishes to produce a database that

serves as input to neural network 80.

In one example, neural network 80, Fig. 10 is trained using fifty-five training sets and twenty-seven test sets. Training with one set of data and testing with another ensures that neural network 82 is not simply memorizing the data.

The weakest of three neural network correlations between the inputs accelerometer, e.g., sensor 12, Fig. 1, plus spindle speed and rate of advance, and the output of surface finish, for R_a , is illustrated in Fig. 11. As shown in Fig. 11, there is a strong overall correlation for the predicted surface finish provided by dynamical instrument for machining 10 of this invention, with the vast majority of the test conditions clustered near the $Y=X$ line. This indicates that a dynamical instrument for machining 10 of this invention accurately predicts the surface finish of a workpiece.

Neural network 80, Fig. 10 may be trained to employ surface finish parameters R_a , R_y , and R_z to predict one another as shown in Table 1 below. As shown in Table 1, there is a 1:1 mapping from any measure of surface finish to any other and that the variations in surface finish parameters are inherently linked. Changing one parameter tends to change the others in a predictable fashion with a narrow band of error.

Network Inputs and Outputs	Correlation Coefficient r^2
R_a predicts R_y	0.9564
R_a predicts R_z	0.9558
R_y predicts R_a	0.9567
R_y predicts R_z	0.9644
R_z predicts R_a	0.9570
R_z predicts R_y	0.9604
$R_a + R_y$ predict R_z	0.9647
$R_a + R_z$ predict R_y	0.9670
$R_y + R_z$ predict R_a	0.9610

Table 1. Relationships between surface finish parameters

In one embodiment of this invention, dynamical instrument for machining 10, Fig. 1 may include controller 100 connected to the output of processor 22 on line 92 for varying machine operating perimeters, such spindle speed, depth of cut, and rate of advance of the cutting tool. In one example, controller 100 is connected to motor 102 for varying the rate of machine 16. In one design, feedback from controller 100 is provided to processor 22.

The dynamical method for machining of this invention includes coupling sensor 12, Fig. 1 to non-rotating part 14 of machine 16 proximate tool 20 for outputting vibration signal 24, step 200, Fig. 15, and calculating in response to the vibration signal a plurality of signature quantities which characterize the dynamics of the vibration signal and correlate the signature quantities to detect parameters associated with the operation of machine 16, step 202, Fig. 15. In one design, the dynamical method for machining may further include the step of using a neural network trained to calculate the parameters associated with the operation of the machine in accordance with a plurality of signature quantities representative of known parameters associated with the operation of the machine. Calculating the plurality of signature quantities typically includes determining variations in the amplitude of the vibration signal, e.g., vibration signal 24, Fig. 1. The signature quantities may include the average value of the amplitude, standard deviation of the amplitude, characteristic auto-correlation time, the average difference, more specifically the average absolute value difference in sequential readings, the root mean square difference (RMSDif) in sequential readings, the period of average crossing waves variable, (SdaPd), the root mean square value of the period of average crossing waves variable which emphasizes longer-period waves (SdRMSpd), the duration of average crossing waves (SdaDur), the root mean square value of the duration of average crossing

waves, (SdaRMSDur), the period of standard deviation crossing waves flow indicator quantity, (SdsPd), the root mean square value of the period of standard deviation crossing waves value, (SdsRMSPd), the duration of standard deviation crossing waves (SdsDur), and the root mean square duration of standard deviation crossing waves, (SdsRMSDur), as discussed in detail above.

Although specific features of the invention are shown in some drawings and not in others, this is for convenience only as each feature may be combined with any or all of the other features in accordance with the invention. The words “including”, “comprising”, “having”, and “with” as used herein are to be interpreted broadly and comprehensively and are not limited to any physical interconnection. Moreover, any embodiments disclosed in the subject application are not to be taken as the only possible embodiments.

Other embodiments will occur to those skilled in the art and are within the following claims:

What is claimed is:

CLAIMS

1. A dynamical instrument for machining comprising:
a sensor responsive to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal; and
a processor including a neural network having an input layer, a hidden layer, and an output layer responsive to output from the sensor and configured to calculate a plurality of signature quantities which characterize the dynamics of the vibration signal, the neural network trained to correlate said signature quantities to detect parameters associated with the operation of the machine.
2. The dynamical instrument for machining of claim 1 in which said machine is a lathe.
3. The dynamical instrument for machining of claim 1 in which said machine is a milling machine.
4. The dynamical instrument for machining of claim 1 in which the non-rotating part of the machine is a tool holder.
5. The dynamical instrument for machining of claim 1 in which the non-rotating part of the machine is a spindle assembly.
6. The dynamical instrument for machining of claim 1 in which said tool is a machine bit.
7. The dynamical instrument for machining of claim 1 further including a computer subsystem with a device for outputting the detected parameters associated with the operation of the machine.
8. The dynamical instrument for machining of claim 7 in which the device for outputting the detected parameters includes an electronic console.

9. The dynamical instrument for machining of claim 7 in which the device for outputting the detected parameters includes an auditory device.

10. The dynamical instrument for machining of claim 1 further including a signal conditioner circuit for converting and digitizing the vibration signal output by the sensor to obtain a computer code representation of the sensor time series.

11. The dynamical instrument for machining of claim 1 in which the detected parameters include a measurement of the surface finish of a workpiece as the workpiece is machined.

12. The dynamical instrument for machining of claim 11 in which the measurement of the surface finish of a workpiece includes the average absolute value deviation in the height from a mean line of the surface finish of the workpiece.

13. The dynamical instrument for machining of claim 11 in which the measurement of the surface finish of a workpiece includes the difference in height between the highest peak above a mean line of the surface finish of the workpiece and the lowest valley below the mean line of the workpiece over a sampling length of the workpiece.

14. The dynamical instrument for machining of claim 13 in which the measurement of the surface finish of a workpiece includes the average peak-to-valley height above the mean line of the surface of the workpiece over the sampling length of the workpiece.

15. The dynamical instrument for machining of claim 1 in which the detected parameters include an indication of the status of lubricant flow in the machine.

16. The dynamical instrument for machining of claim 1 in which the detected parameters include an indication if the machine tool is worn or damaged.

17. The dynamical instrument for machining of claim 1 in which the detected parameters include an indication of the remaining life of the machine tool.

18. The dynamical instrument for machining of claim 1 in which the detected parameters include an indication how secure a workpiece is in a clamp of the machine as the workpiece is being machined.

19. The dynamical instrument for machining of claim 8 in which said electronic console outputs a visual display of the detected parameters associated with the operation of the machine.

20. The dynamical instrument for machining of claim 8 in which a said console outputs a visual alarm of the detected parameters associated with the operation of the machine.

21. The dynamical instrument for machining of claim 9 in which said auditory device outputs an auditory alarm of the detected parameters associated with the operation of the machine.

22. The dynamical instrument for machining of claim 1 in which said sensor is an accelerometer.

23. The dynamical instrument for machining of claim 1 in which said sensor is an acoustic sensor.

24. The dynamical instrument for machining of claim 22 in which the sensor includes a charge amplifier for outputting a signal with reduced influence of external noise.

25. The dynamical instrument for machining of claim 23 in which the acoustic sensor includes an amplifier and a narrow band pass filter for outputting a carrier wave with varying amplitude.

26. The dynamical instrument for machining of claim 25 in which calculating the number of signature quantities includes determining variations in the amplitude of the vibration signal.

27. The dynamic instrument for machining of claim 26 in which a said signature quantity is the average value of the amplitude of the vibration signal.

28. The dynamical instrument for machining of claim 26 in which a said signature quantity is the standard deviation of the amplitude of the vibration signal.

29. The dynamical instrument for machining of claim 26 in which a said signature quantity is the characteristic auto-correlation time of the vibration signal.

30. The dynamical instrument for machining of claim 26 in which a said signature quantity is the average absolute value difference of the amplitude of the vibration signal.

31. The dynamical instrument for machining of claim 26 in which a said signature quantity is the average crossing waves variable of the vibration signal.

32. The dynamical instrument for machining of claim 31 in which a said signature quantity is the root mean square difference of the average crossing waves variable of the vibration signal.

33. The dynamical instrument for machining of claim 31 in which a said signature quantity is the duration of the average crossing waves variable of the vibration signal.

34. The dynamical instrument for machining of claim 31 in which a said signature quantity is the root mean square value of the duration of the average crossing waves variable of the vibration signal.

35. The dynamical instrument for machining of claim 26 in which a said signature quantity is the period of the standard deviation crossing waves of the vibration signal.

36. The dynamical instrument for machining of claim 35 in which a said signature quantity is the duration of crossing waves of the vibration signal.

37. The dynamical instrument for machining of claim 26 in which a said signature quantity is the root mean square duration of standard deviation crossing waves of the vibration signal.

38. The dynamical instrument for machining of claim 26 in which a said signature quantity is both the average value of the amplitude of the envelope and the standard deviation of the average value of the amplitude of the envelope of the vibration signal.

39. A dynamical instrument for machining comprising:
a sensor responsive to a non-rotating part of the machine proximate the tool of the machine for outputting a vibration signal; and
a processor including a neural network having an input layer, a hidden layer, and an output layer responsive to output from the sensor and configured to calculate a plurality of signature quantities which characterize the dynamics of the vibration signal, the neural network trained to correlate said signature quantities to detect parameters associated with the operation of the machine, the processor including an output connected to the machine for varying a machine operating parameters.

40. The dynamical instrument for machining of claim 39 further including a controller connected to the output of the processor and the machine for varying the machine operating parameters.

41. The dynamical instrument for machining of claim 40 in which said controller is connected to a motor of the machine.

42. A dynamical method for machining comprising:
- coupling a sensor to a non-rotating part of a machine proximate the tool of the machine for outputting a vibration signal; and
 - calculating, in response to the vibration signal, and using a neural network having an input layer, a hidden layer, and an output layer, a plurality of signature quantities which characterize the dynamics of the vibration signal, the neural network trained to correlate said signature quantities to detect parameters associated with the operation of the machine; and
 - adjusting an operational parameter of the machine in response to said calculation.
43. The method of claim 42 in which calculating the parameters includes determining variations in the amplitude of the amplitude envelope of the vibration signal.
44. The method of claim 43 in which a said signature quantity is the average value of the amplitude of the vibration signal.
45. The method of claim 42 in which a said signature quantity is the standard deviation of the amplitude of the vibration signal.
46. The method of claim 42 in which a said signature quantity is the characteristic auto-correlation time of the vibration signal.
47. The method of claim 42 in which a said signature quantity is the average absolute value difference of the amplitude of the vibration signal.
48. The method of claim 42 in which a said signature quantity is the average crossing waves variable of the vibration signal.
49. The method of claim 42 in which a said signature quantity is the root mean square difference of the average crossing waves variable of the vibration signal.

50. The method of claim 42 in which a said signature quantity is the duration of the average crossing waves variable of the vibration signal.

51. The method of claim 42 in which a said signature quantity is the root mean square value of the duration of average crossing waves variable of the vibration signal.

52. The method of claim 42 in which a said signature quantity is the period of standard deviation crossings of the vibration signal.

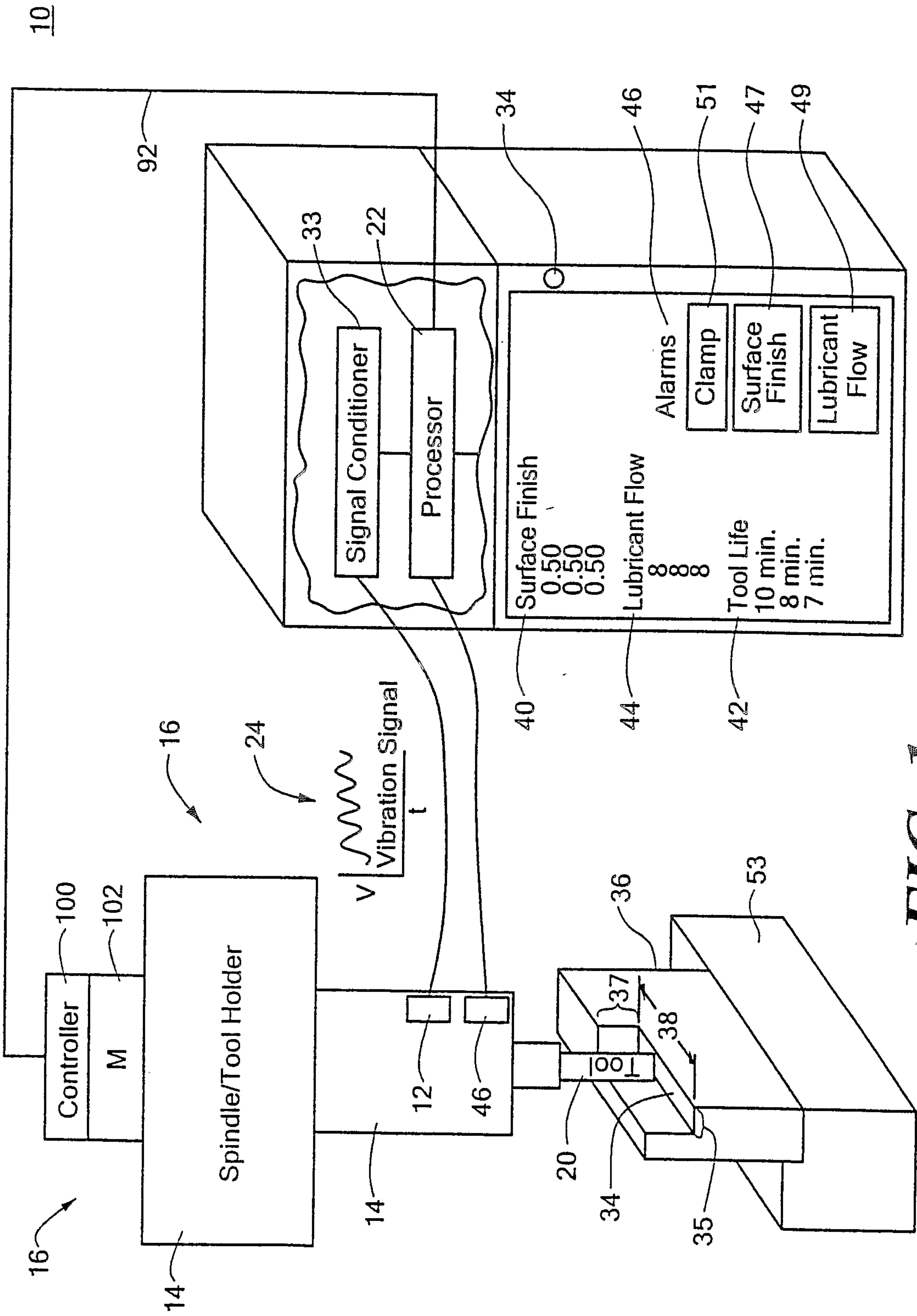
53. The method of claim 42 in which a said signature quantity is the root mean square duration of standard deviation crossing waves of the vibration signal.

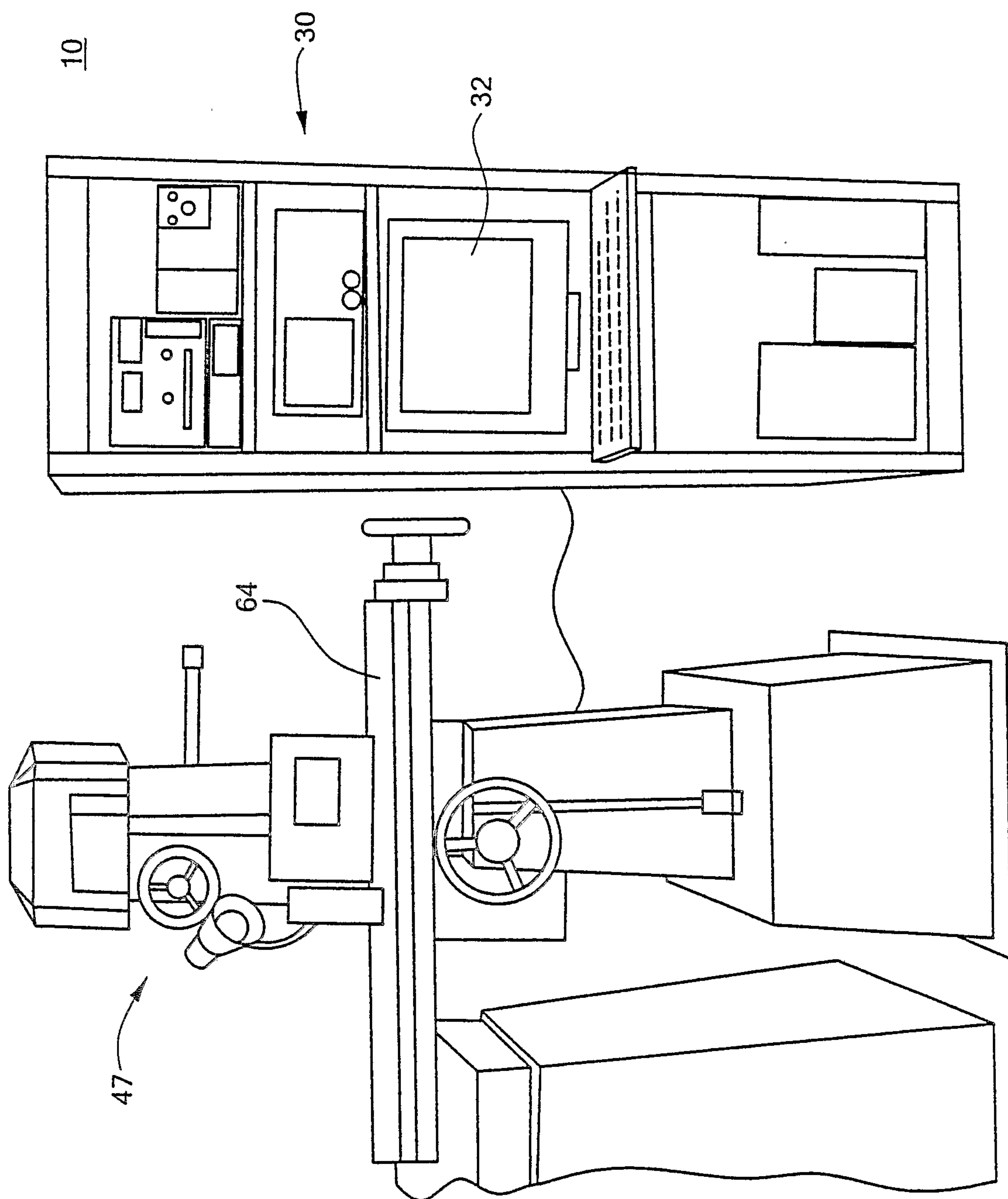
54. The method of claim 42 in which a said signature quantity is the root mean square duration of standard deviation crossing waves of the vibration signal.

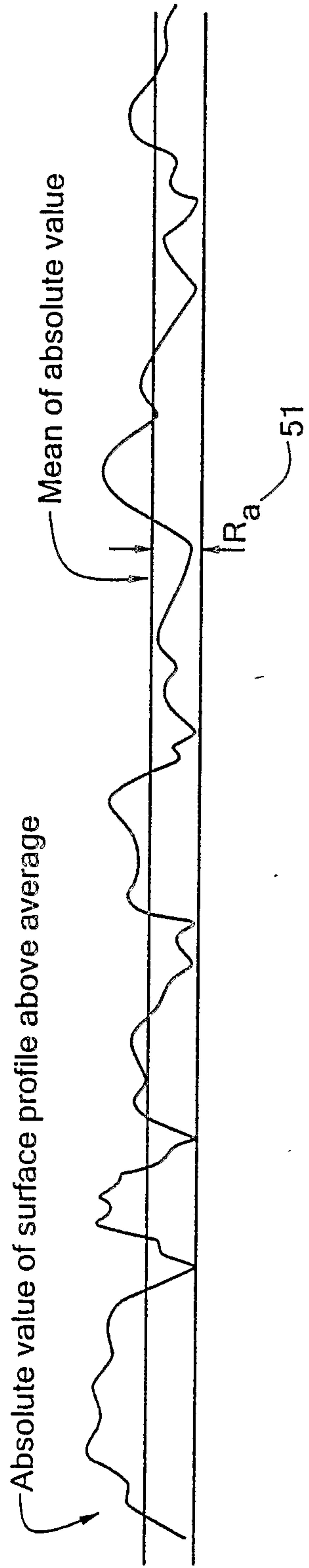
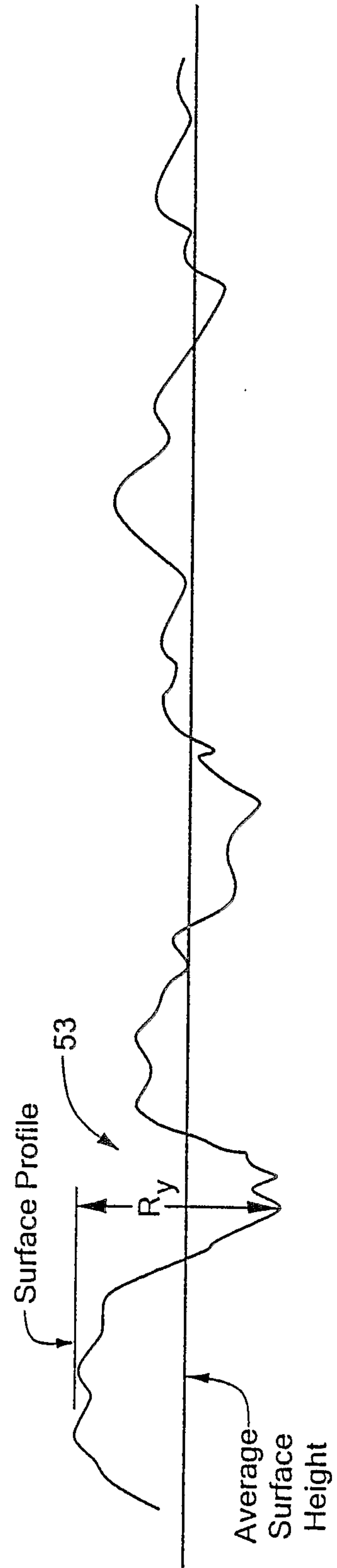
55. The method of claim 42 in which a said signature quantity is both the average value of the amplitude of the envelope and the standard deviation of the average value of the amplitude of the envelope of the vibration signal.

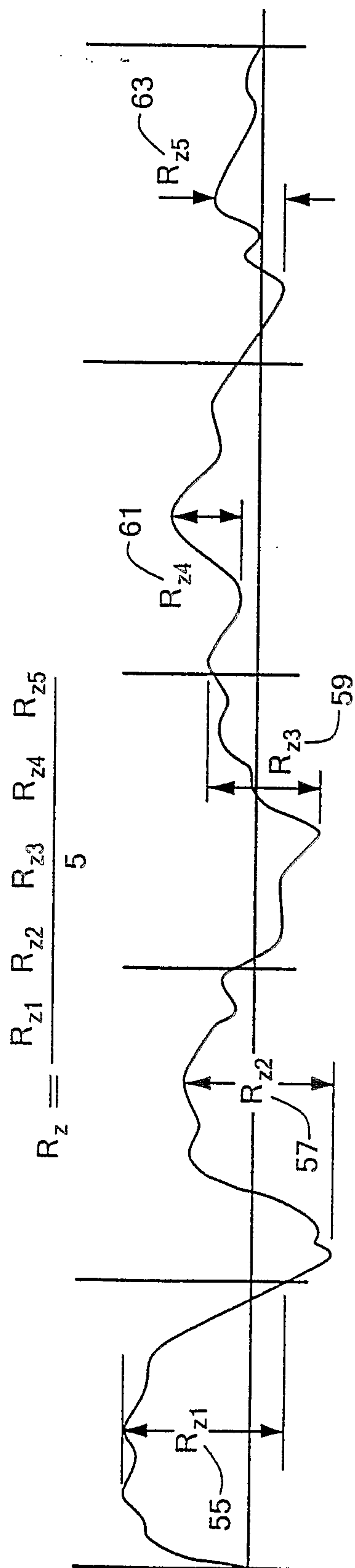
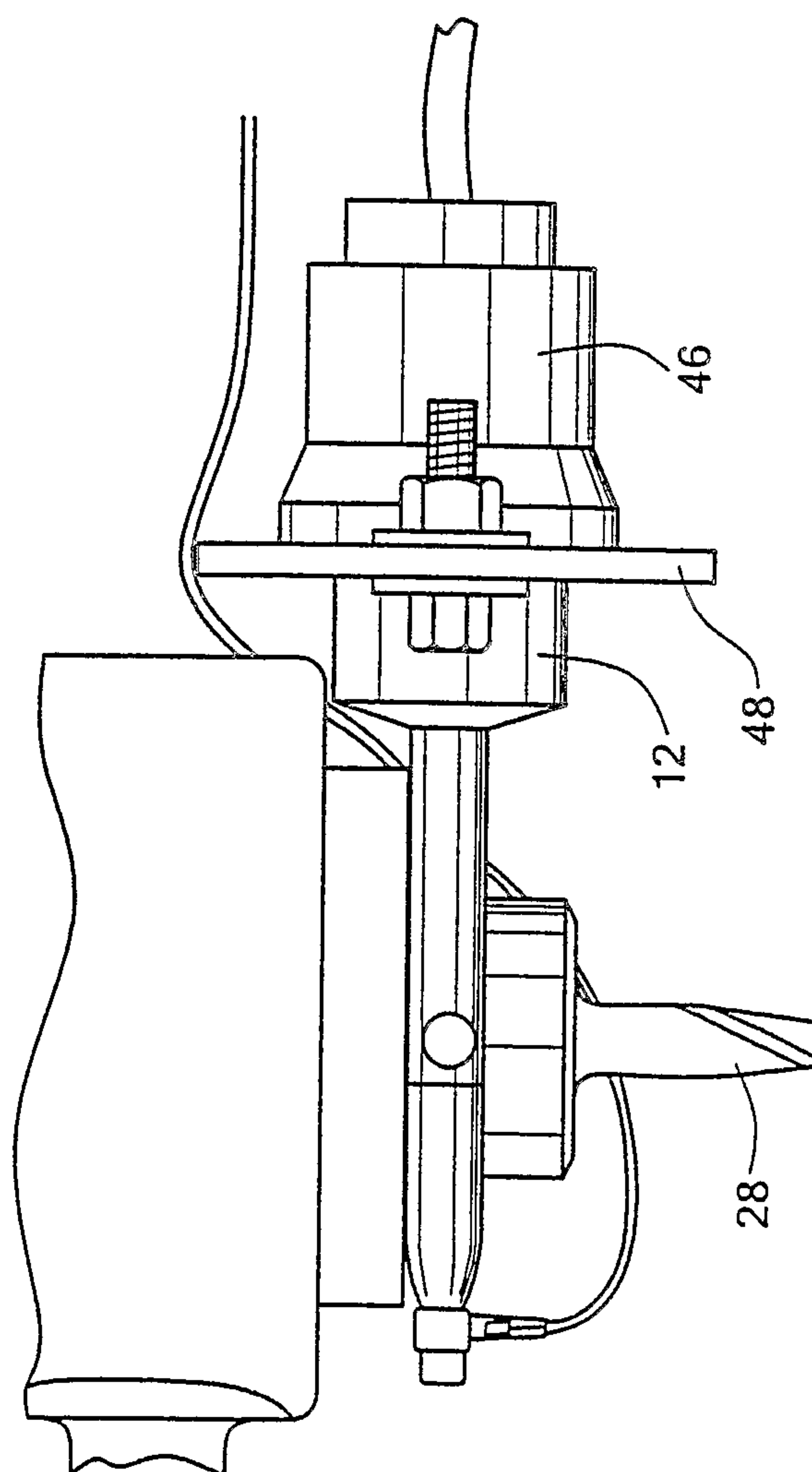
56. The method of claim 42 further including displaying numerical indicators on an electronic console which represent the parameters associated with the operation of the machine.

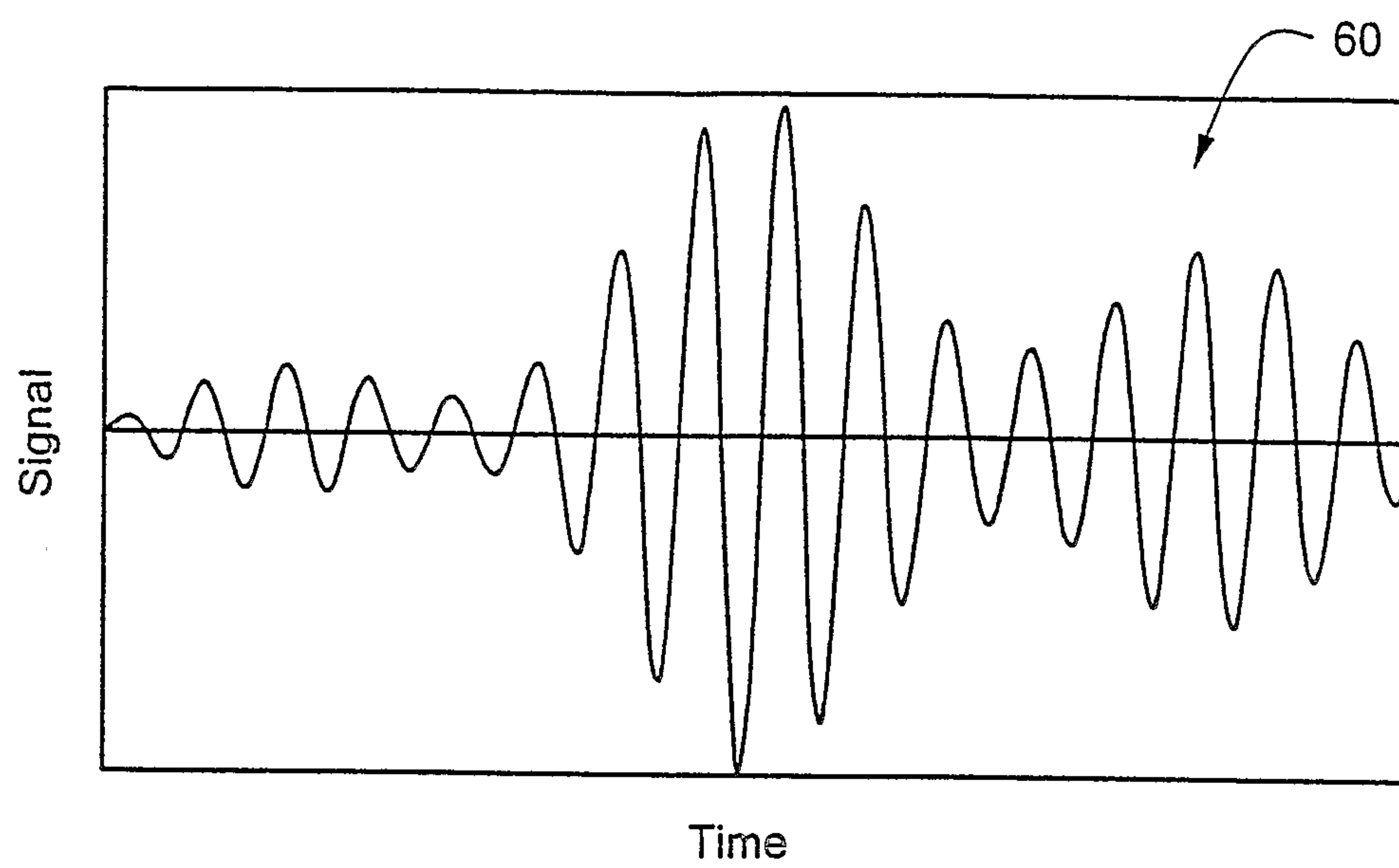
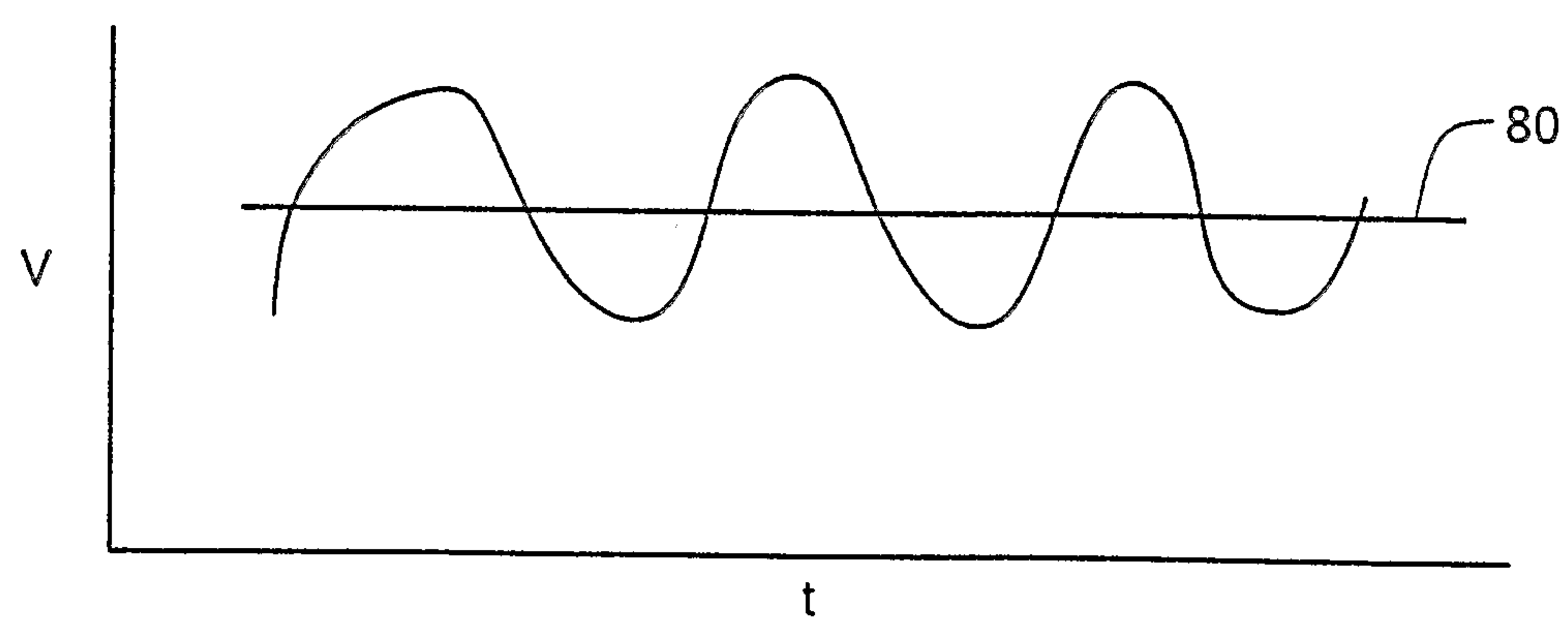
57. The method of claim 42 further including displaying sounding an audio alarm to indicate if the parameters associated with the operation of the machine are outside a predetermined range.

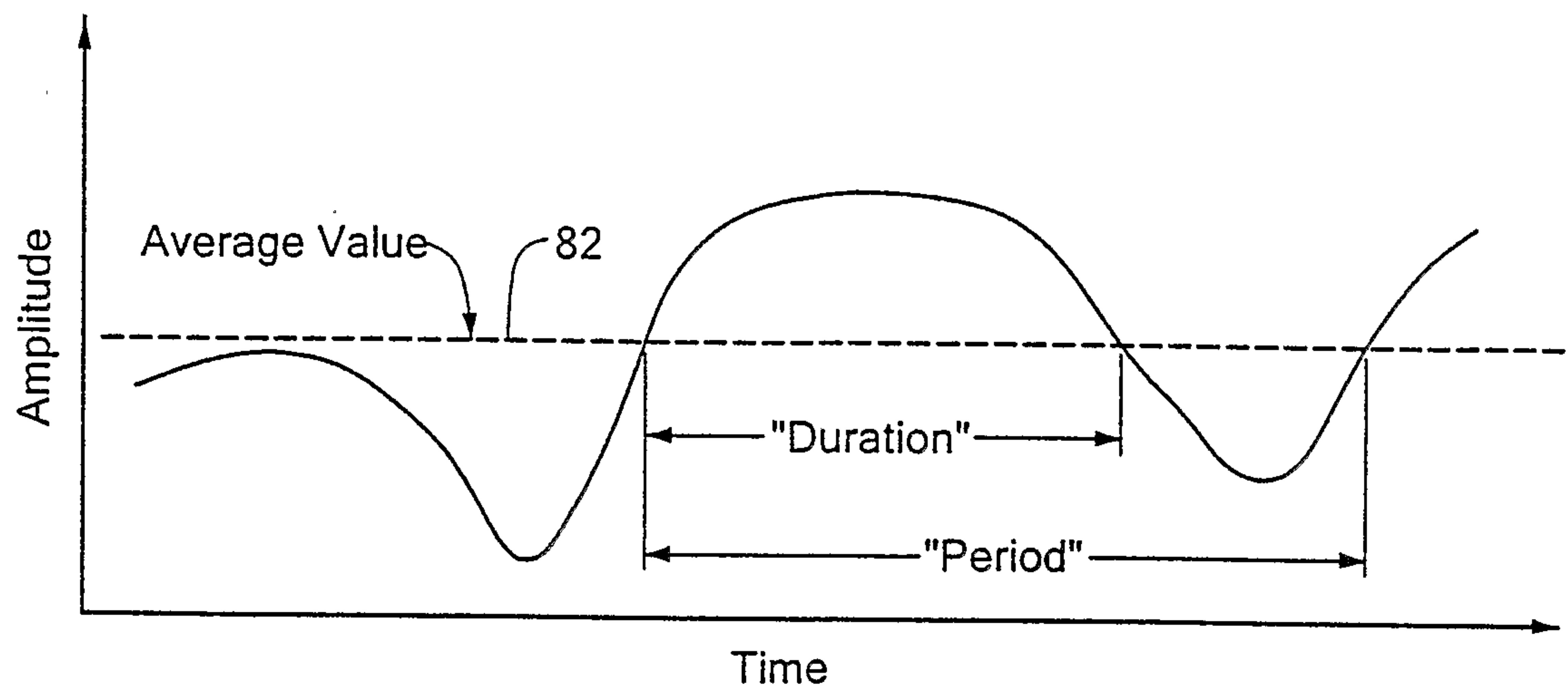
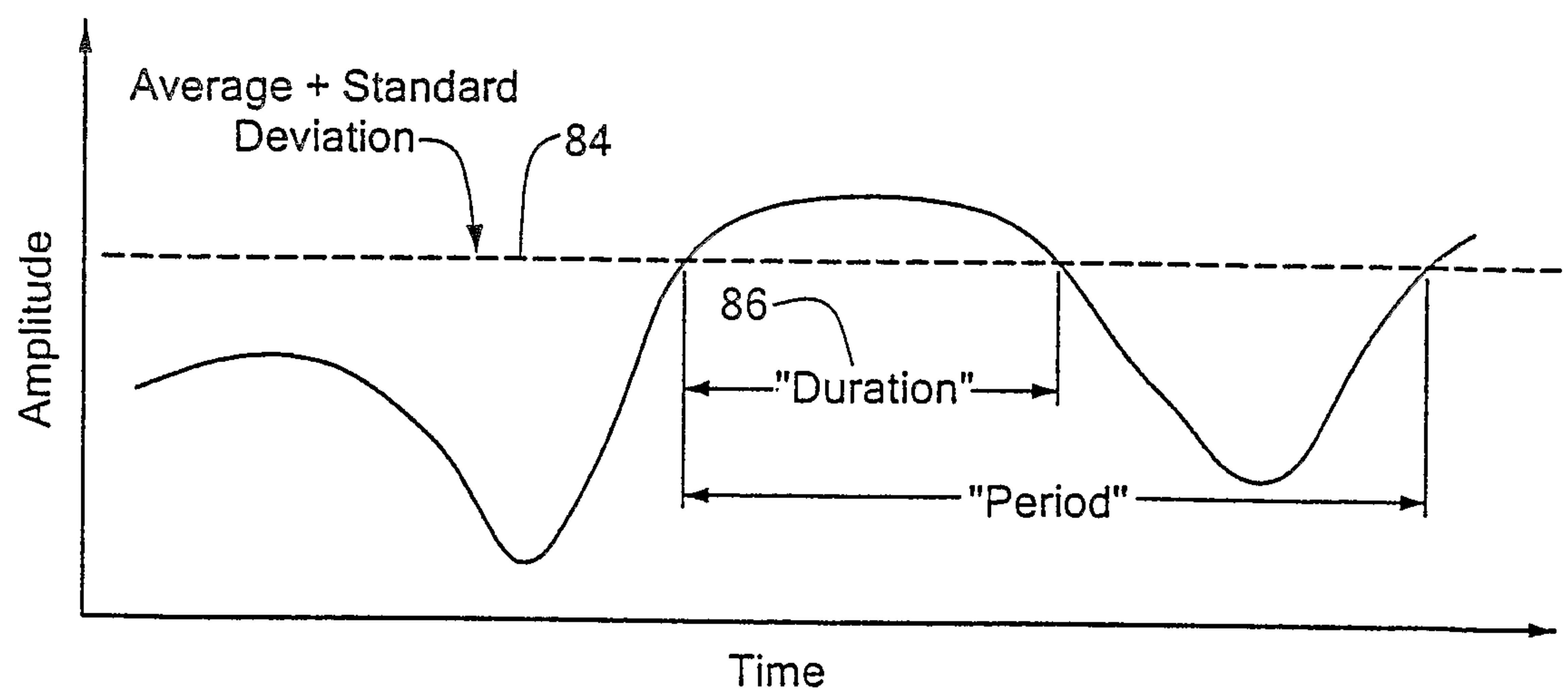


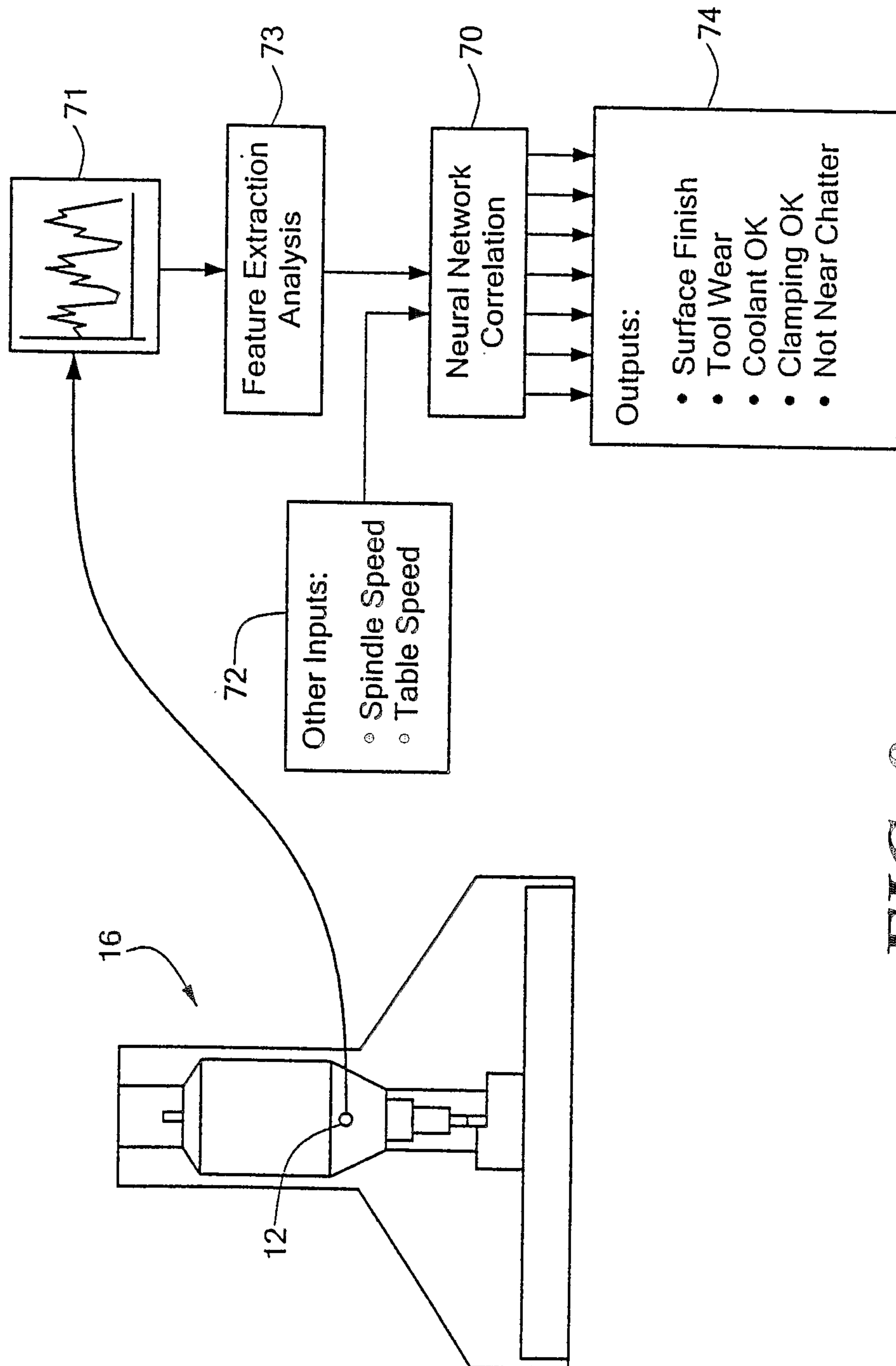
**FIG. 2**

**FIG. 3A****FIG. 3B**

**FIG. 3C****FIG. 4**

**FIG. 5****FIG. 6**

**FIG. 7****FIG. 8**

**FIG. 9**

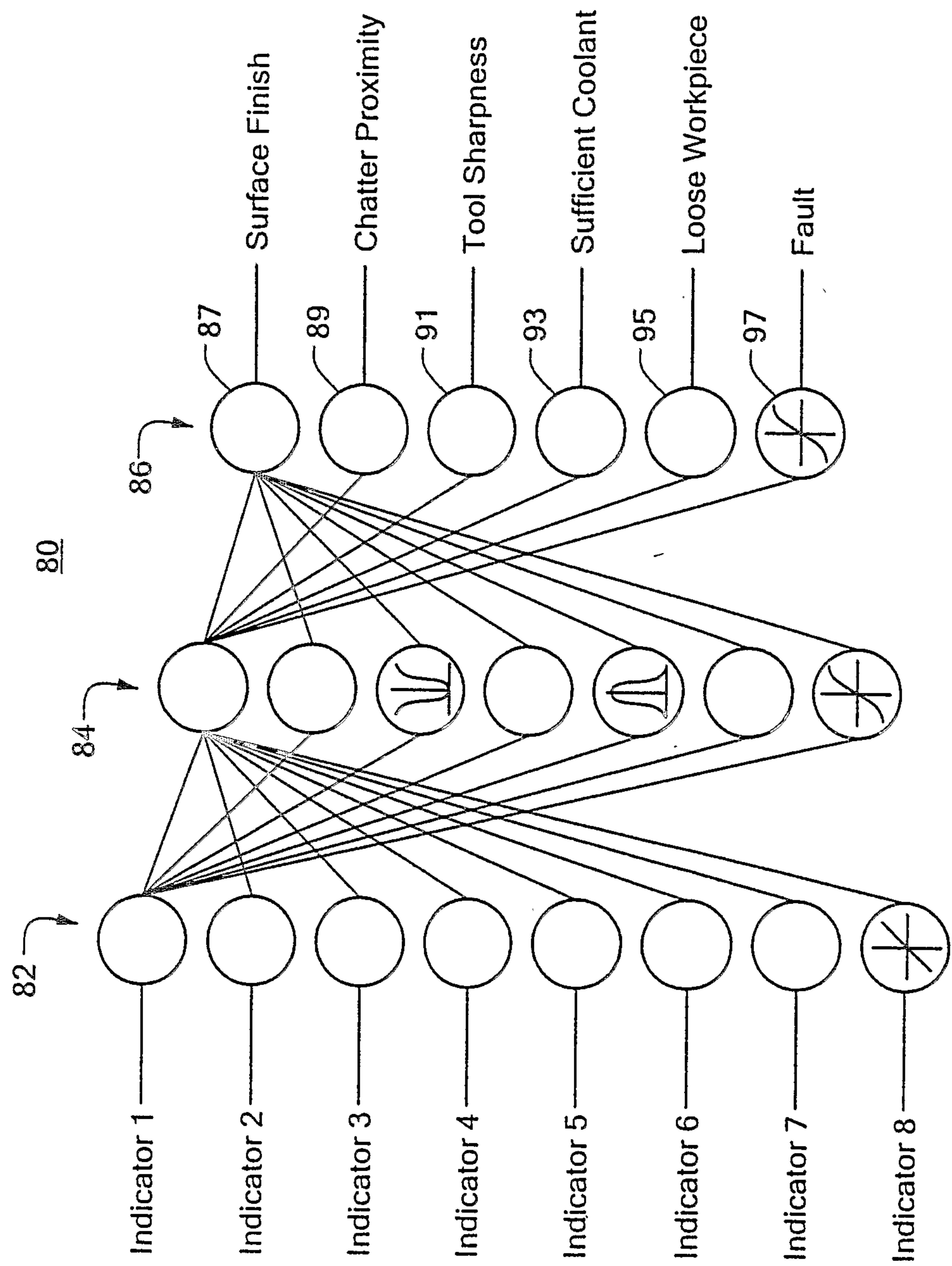
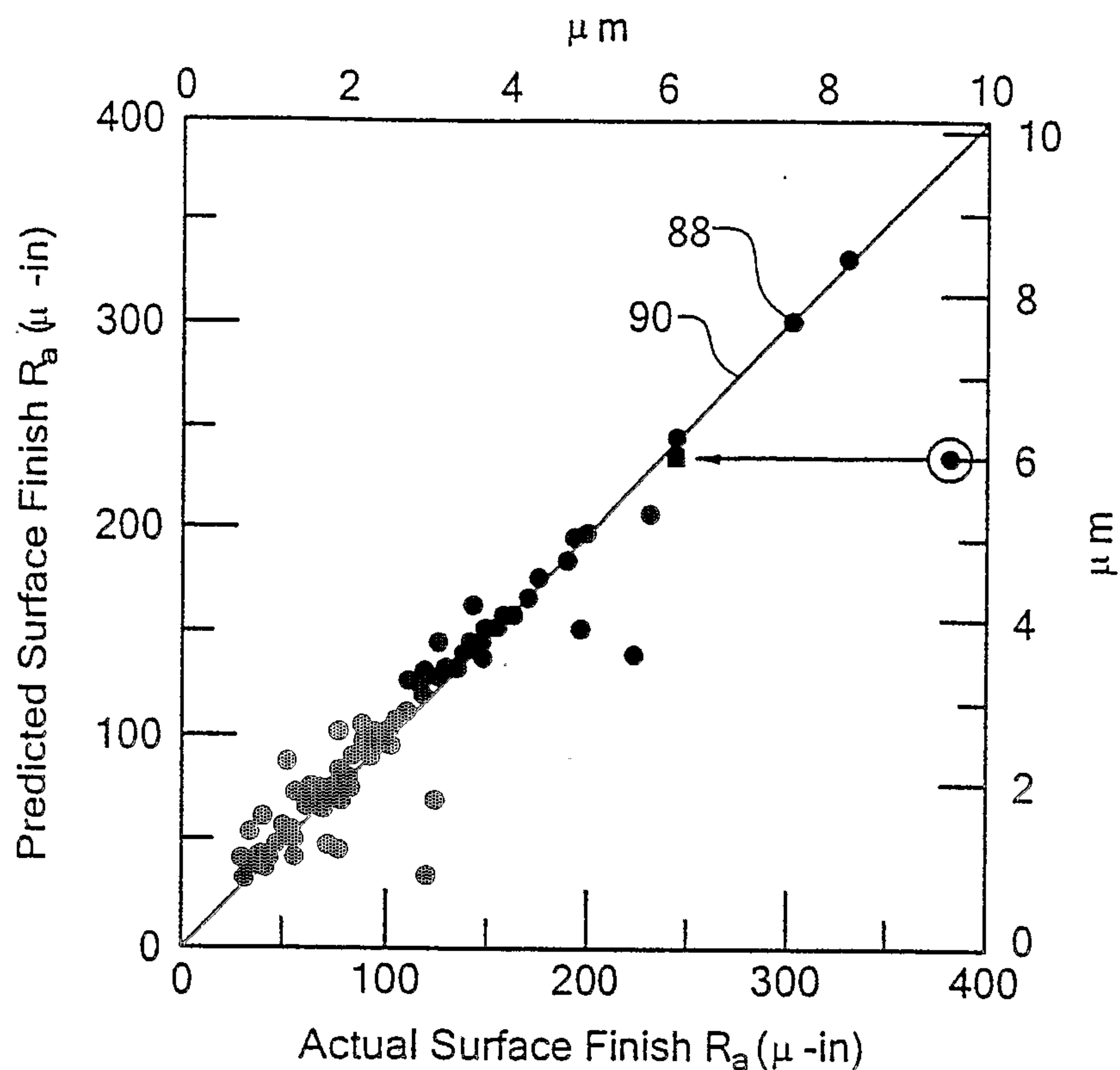
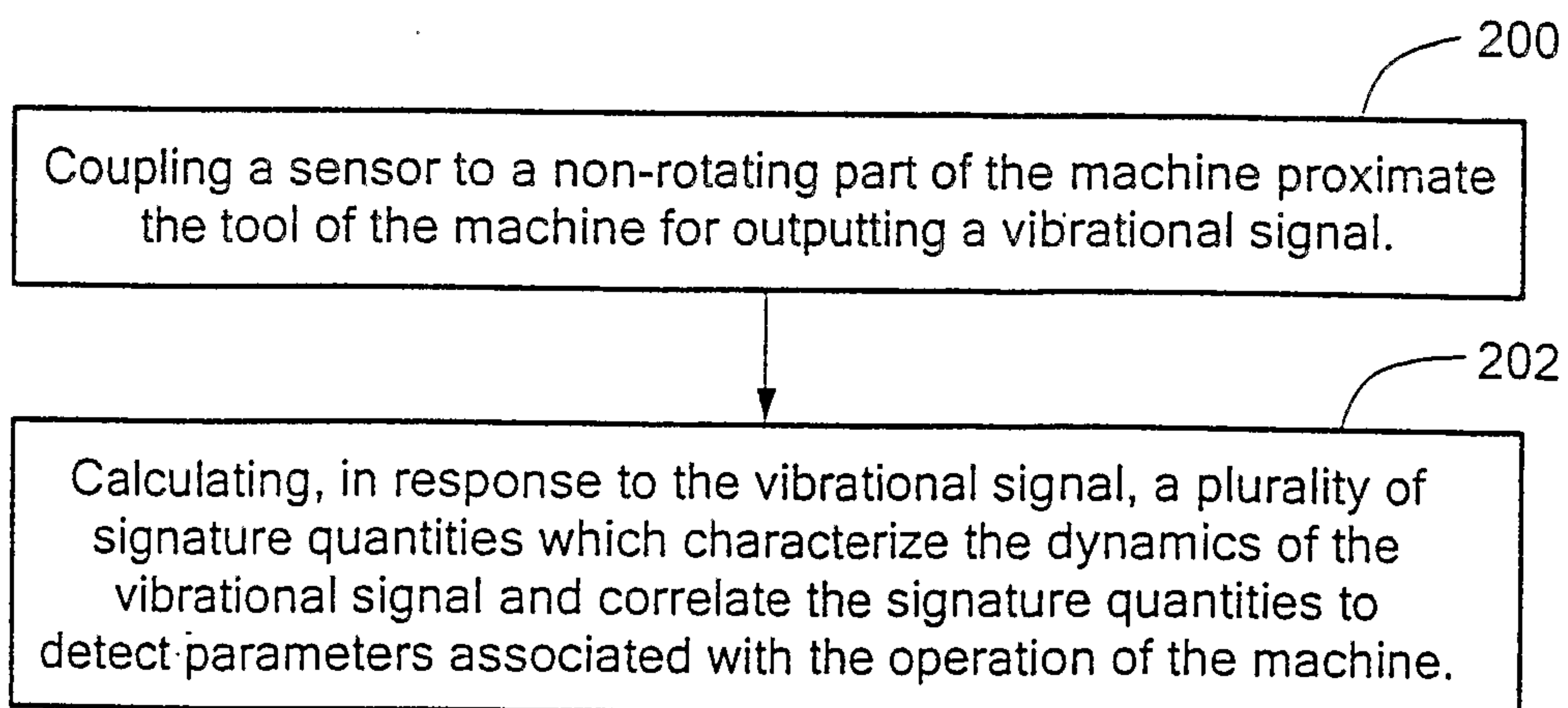


FIG. 10

*FIG. 11**FIG. 12*

