

CERTIFICATE OF TRANSLATION ACCURACY

I am a professional reviewer and coordinator specializing in translating Chinese, Japanese and Korean to English and vice versa.

I served as Chief Examiner of the certified court interpreter test for the State of California and as a contract translator and interpreter for various federal agencies through the U.S. Department of State for more than a decade. I served as an instructor at the University of California at Berkeley and the Middlebury Institute of International Studies at Monterey.

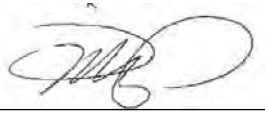
I have more than 30 years of experience translating thousands of technical, legal, and business submitted to, among others, Korean judicial authorities, various U.S. federal courts, the U.S. International Trade Commission (ITC), and the USPTO Patent Trial and Appeal Board (PTAB).

I certify that the following document translated from Japanese into the English language is a true, correct, and complete translation of the corresponding source text to the best of my knowledge and ability.

I certify under penalty of perjury that the foregoing is true and correct.

Executed this 19th day of July 2023 in the Contra Costa County of the State of California.

By:



Alex N. Jo
Member, ATA

(19) Japanese Patent
Office (JP)

(12) Patent Application Gazette (A)

(11) Published Application No.

2005-242456**(P2005-242456A)**

(43) Published: Sept. 8, 2005

(51) Int. Cl.⁷

G 06 T 13/00

FI

G 06 T 13/00

C

Theme code (ref.)

5B050

Request for Examination: No

Number of Claims: 8 OL Total pages: 42

(21) Filed Application No.: 2004-48028 (P2004-48028)	(71) Applicant:	592263425 International Media Research Foundation 1-1 Kagurazaka, Shinjuku-ku, Tokyo
(22) Filed: Feb. 24, 2004 (2004.2.24)	(74) Agent or Attorney:	100090033 ARAFUNE Hiroshi, Patent Attorney
	(72) Inventor:	SUGIHARA Satoshi c/o International Media Research Foundation, Sanko Bldg., 4F, 1-1 Kagurazaka, Shinjuku-ku, Tokyo
		F-term (Reference) 5B050 BA08 BA18 EA12 EA24 FA02

(54) Title of the Invention:

ROTATING BODY IMAGE DISPLAY DEVICE AND ROTATING BODY IMAGE DISPLAY PROGRAM

(57) ABSTRACT

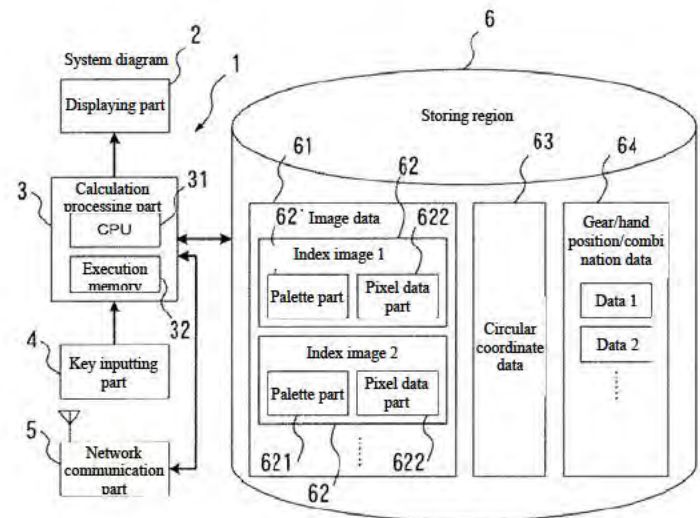
PROBLEM:

To enable a plurality of gears synchronously rotatable at different rotation rates to be arbitrarily arranged and displayed in rotated states even in an environment where arithmetic processing capability is limited.

MEANS FOR SOLUTIONS:

A program for displaying an animation of synchronously rotating gears on a mobile phone 1. An index image 62, which represents an image of a gear at each rotation angle position obtained by dividing the rotation angle of one tooth by the same number of divisions for each gear at substantially equal intervals corresponding to a plurality of gears having different numbers of teeth, is stored in a storing region 6. By sequentially switching and displaying the index images 62, gears that rotate synchronously but at different rotating speeds due to having different numbers of teeth are displayed. Here, because mutually synchronous gears having different numbers of teeth are displayed, the timing for sequentially switching the index image 32 is substantially synchronized in the display of each gear.

[Selected Drawing] FIG. 1



What is claimed is:

[Claim 1]

A rotating body image display device for displaying animation of a plurality of rotating bodies such as gears that can synchronously rotate with each other, have a rotationally symmetrical form in which a form prior to rotation matches a form following rotation whenever rotated a respectively inherent predetermined angle, and in which times for the plurality of rotating bodies to rotate respectively inherent predetermined angles essentially match when synchronously rotated together, the rotating body image display device comprising:

storing means for expressing, as image data for respective animations of the plurality of rotating bodies having mutually different predetermined angles, an image of the rotating bodies at respective rotation angle positions obtained by dividing the respectively inherent predetermined angle for each rotating body by the same number of divisions for each rotating body at substantially equal intervals and for storing the image data, which is equalized for each rotating body by dividing the predetermined angles by the same number of divisions for each rotating body; and

display controlling means for displaying the animation of the rotating bodies by sequentially displaying a plurality of the image data per plurality of rotating bodies stored in the storing means and for substantially synchronizing the timing for switching the image data for the rotating bodies together, which are synchronously displayed in rotated states, when the image data stored in the storing means is sequentially switched and displayed according to given rotation rates of the rotating bodies.

[Claim 2]

The rotating body image display device according to claim 1, further comprising circular coordinate storing means for presetting, per substantially equivalent interval, a number of virtual points of a value obtained by multiplying a number of the image data for displaying an animation of rotating bodies rotating a predetermined angle to a number of the predetermined angle included in one rotation of each rotating body on the circumferences of virtual circles synchronously rotating in mutual contact, and which respectively correspond to each rotating body synchronously rotating adjacent to each other, where each virtual point is made to correspond to one of the plurality of image data for displaying the respective animations, and for storing virtual point coordinate data indicating each virtual point individually, wherein the display controlling means obtains the virtual points, which correspond to the rotation angle according to the rotation rates of the rotating bodies at the time the image data is displayed when displaying an animation of rotating bodies by sequentially displaying the image data, and the image data corresponding to the virtual points obtained from within the plurality of image data is displayed.

[Claim 3]

The rotating body image display device according to claim 2, wherein when, between rotating bodies synchronously rotating together adjacent to each other, times for the respective rotating bodies to rotate a predetermined angle are equalized, and the condition of offsetting the time by half of the time for a phase when rotating a predetermined angle together to rotate the predetermined angle, the display controlling means makes the rotating bodies synchronously rotating together adjacent to each other respectively match, obtains the virtual points closest to mutual contact points where the virtual circles contact each other wherever virtual circles contact each other, and based on a relationship between phases corresponding to the rotating angle of the rotating bodies in which virtual points obtained with respect to one virtual circle correspond to virtual points on another and based on the above condition, and by obtaining the relationship between the phases of the rotating bodies corresponding to the virtual points obtained with respect to the one virtual circle and the virtual points of the other virtual circle, displays the rotating bodies synchronously rotating together while satisfying the condition.

[Claim 4]

The rotating body image display device according to any one of claims 1 to 3, further comprising color changing means for changing the color of an image of a rotating body, wherein the storing means stores index images having a color palette and color scheme image data as index images by overwriting color data on the color palette of an index image stored in the storing means.

[Claim 5]

A rotating body image display program for displaying, on a computer provided with a display device, an animation of a plurality of rotating bodies such as gears that can synchronously rotate with each other, have a rotationally symmetrical form in which a form prior to rotation matches a form following rotation whenever rotated a respectively inherent predetermined angle, and in which times to rotate respectively inherent predetermined angles essentially match when synchronously rotated together, wherein the program implements, on the computer, a function for expressing, as image data for respective animations of the plurality of rotating bodies having mutually different predetermined angles, an image of the rotating bodies at respective rotation angle positions obtained by dividing the respectively inherent predetermined angle for each rotating body together by the same number of divisions at substantially equal intervals and for displaying the animation of the rotating bodies by sequentially displaying the image data, which is equalized for each rotating body by dividing the predetermined angles by the same number of divisions for each rotating body, and a function for substantially synchronizing the timing for switching the image data for the rotating bodies together, which are synchronously displayed in rotated states, when image data is sequentially switched and displayed according to the given rotation rates of the rotating bodies.

[Claim 6]

The rotating body image display program according to claim 5, further comprising: implementing, on the computer, a function of presetting, per substantially equivalent interval, a number of virtual points of a value obtained by multiplying a number of the image data for displaying an animation of rotating bodies rotating a predetermined angle to a number of the predetermined angle included in one rotation of each rotating body on the circumferences of virtual circles synchronously rotating in mutual contact, which respectively correspond to each rotating body synchronously rotating adjacent to each other, where each virtual point is made to correspond to one of the plurality of image data for displaying the respective animations, obtaining the virtual points, which correspond to the rotation angle according to the rotation rates of the rotating bodies at the time the image data is displayed when displaying an animation of rotating bodies by sequentially displaying the image data, and displaying the image data corresponding to the virtual points obtained from within the plurality of image data.

[Claim 7]

The rotating body image display program according to claim 6, further comprising: implementing, on the computer, a function of, when, between rotating bodies synchronously rotating together adjacent to each other, times for the respective rotating bodies to rotate a predetermined angle are equalized, and the condition of offsetting the time by half of the time for a phase when rotating a predetermined angle together to rotate the predetermined angle, making the rotating bodies synchronously rotating together adjacent to each other respectively match, obtaining the virtual points closest to mutual contact points where the virtual circles contact each other wherever virtual circles contact each other, and based on a relationship between the phases corresponding to the rotating angle of the rotating bodies in which virtual points obtained with respect to one virtual circle correspond to virtual points on another virtual circle and based on the above condition, and by obtaining the relationship between the phases of the rotating bodies corresponding to the virtual points obtained with respect to the one virtual circle and the virtual points of the other virtual circle, displaying the rotating bodies synchronously rotating together while satisfying the condition.

[Claim 8]

The rotating body image display program according to any one of claims 5 to 7, further comprising: implementing, on the computer, a function of changing the color of an image of a rotating body, wherein index images having a color palette and color scheme image data are stored in storing means on the computer as index images, by overwriting color data on the color palette of an index image stored in the storing means.

DETAILED DESCRIPTION OF THE INVENTION

TECHNICAL FIELD

[0001]

The present invention relates to a rotating body image display device and a rotating body image display program for displaying rotating bodies such as gears synchronously rotating together.

CONVENTIONAL TECHNOLOGY

[0002]

In recent years, mobile phones have been equipped with color liquid crystal displays whereby not only information in the form of text or marks, but also picture or photographic information is displayed, and moreover, moving images such as videos can also be displayed.

In addition, mobile phone dedicated webpages on the internet and systems composed of a mobile phone equipped with a dedicated browser for viewing these mobile phone dedicate webpages have been developed, enabling for the accumulation of information via the internet as well as the ability to send and receive emails over the internet.

[0003]

In addition, there are also mobile phones that can execute applications using Java (registered trademark) for embedded devices, for example, there are even mobile phones on the market capable of downloading simple videogame programs over the internet as content (Java applet) using Java and executing the simple downloaded videogames.

Before executing a videogame, an animation such as a character must be displayed, but, for example, mobile telephones are being developed that are capable of executing animations using browser software for mobile phones. (For example, see Patent Document 1)

Patent Document 1: [Japanese Unexamined Patent Application 2002-352262](#) (page 5)

DISCLOSURE OF THE INVENTION

PROBLEMS TO BE SOLVED BY THE INVENTION

[0004]

The present applicants set out to develop, as content provided on a mobile phone having an internet connection environment, a program for displaying an animation of interactively operable gears of any number for mobile phone users to arrange and display in a plurality of combinations, wherein rotation is performed by a user operation at separate rotation rates according to the rotation rate of the gears on the drive side of each gear displayed on the display screen and the ratio of the number of teeth on each gear.

In addition, to enhance user interest, the content described above can also display watch hands that rotate corresponding to the rotation of each gear which, by a user successfully arranging gears in combination, enables the rotation of an hour hand, minute hand, and second hand in hour, minute, and second units respectively, as well a display such as of a so-called skeleton clock.

[0005]

However, current mobile phones are in demand that enable long-time usage as a phone using the limited power of a battery and that are compact and lightweight, and there are hardware limits such as memory storage capacity and the processing capability of an arithmetic processing device, and Java for embedded devices used in mobile phones is set such that only limited arithmetic processing can be executed depending on the hardware mentioned above. This Java is limited to integer arithmetic and is incapable of performing floating point arithmetic. Moreover, there are few usable functions, with constraints such as being unable to use trigonometric functions, as an example. Also, the types of image files that can be used with this manner of mobile phones are limited. For example, usable image files are limited to GIFs (service mark), which are index image files, and Jpegs using irreversible compression.

[0006]

Therefore, in order to implement the aforementioned content in mobile phones such as those described above, rather than a preset gear rotating display, an animation is displayed in a state in which a plurality of gears meshed together that a user places arbitrarily are synchronized with each other, which was extremely difficult. That is, the gears meshing together have different rotation rates based on the ratio of the number of their teeth, and because the gears meshed together are synchronized so that the tooth space between the teeth of one gear and the tooth of another gear mesh, displaying an animation of two or more gears having different rotation rates and that are synchronized together was difficult under conditions in which the functions of a mobile phone are limited as described above.

[0007]

In addition, in displays such as animations, where the number of colors is lower than that of a photograph; GIFs, for example, which are index images, are advantageous compared to Jpeg, so index images are used as the image data. Furthermore, in cases where the color of each gear can be changed in response to a user operation, an animation can be incorporated such as performing a change of the color of the index image described above, and image data for a plurality of types of different colors must be prepared in advance in the image data for the animation of a plurality of types of each gear, but high memory capacity is required for the storage of applications in such a case and for the storage of image data for different colors, the memory capacity or processing capability of a mobile phone may have insufficient capacity or capability, and as memory capacity or processing capability approach their limits, ease of use can worsen.

[0008]

In addition, Java applets are downloaded as content, for example, from a server connected to a mobile phone via the internet. Moreover, in cases such as those where a plurality of Java applets are used, mobile phone storage capacity is a problem, so unused Java applets are deleted, and must be redownloaded when a deleted Java applet is used. Also, there may be cases where, due to the problem of being charged a usage fee or the like for a Java applet, the need arises for the Java applet to be downloaded for each use or for each prescribed period of time.

Therefore, even when there are no hardware restrictions for mobile phones such as those described above or software restrictions such as those described above for embedded Java, communication-related fees when content is downloaded impose a large burden on users in cases where there is high content data volume because a lot of processing is conducted as content, so even if a usage fee for content itself is inexpensive, usage of the content becomes difficult.

[0009]

An object of the present invention is to provide a mobile terminal and a rotating body image display program that allow a plurality of rotating bodies in which gears or the like can synchronously rotated together at different rotation rates to be arbitrarily arranged and displayed in rotated states even in an environment where arithmetic processing capability is limited.

MEANS FOR SOLVING THE PROBLEM

[0010]

The invention according to claim 1 is a rotating body image display device for displaying animation of a plurality of rotating bodies such as gears that can synchronously rotate with each other, have a rotationally symmetrical form in which a form prior to rotation matches a form following rotation whenever rotated a respectively inherent predetermined angle, and in which times for the plurality of rotating bodies to rotate respectively inherent predetermined angles essentially match when synchronously rotated together, the rotating body image display device comprising:

storing means for expressing, as image data for respective animations of the plurality of rotating bodies having mutually different predetermined angles, an image of the rotating bodies at respective rotation angle positions obtained by dividing the respectively inherent predetermined angle for each rotating body by the same number of divisions for each rotating body at substantially equal intervals and for storing the image data, which is equalized for each rotating body by dividing the predetermined angles by the same number of divisions for each rotating body; and

display controlling means for displaying the animation of the rotating bodies by sequentially displaying a plurality of the image data per plurality of rotating bodies stored in the storing means and for substantially synchronizing the timing for switching the image data for the rotating bodies together, which are synchronously displayed in rotated states, when the image data stored in the storing means is sequentially switched and displayed according to given rotation rates of the rotating bodies.

[0011]

Here, the rotating body image display device is, for example, a mobile terminal provided with a liquid crystal display device, an EL display device, or the like, and having a limited storage capacity and arithmetic processing capability, being a small computer capable of executing a program. The present invention is particularly suitable for mobile phones.

In addition, the rotating body is a gear, for example, a spur gear in particular (which may include not only an external gear, but an internal gear as well); however, the gears need not be completely displayed, and may be rotating bodies displayed in a gear-like fashion. For example, when a gear-like rotating body is formed into an actual gear via tooth shapes or the like, those for which mutual meshing and rotation is difficult are also acceptable. Moreover, typically, one gear has a plurality of rotationally symmetrical angles, but here, an angle being the rotation angle of one tooth and a minimum angle among the rotationally symmetrical angle serves as a predetermined angle.

[0012]

According to the invention described in claim 1, each rotating body has a rotationally symmetrical shape that is symmetrical at each predetermined angle, and by preparing a plurality of image data for animations rotating the rotating bodies only the predetermined angle and repeatedly displaying this only a number ($360^\circ / \text{predetermined angle}$) of the predetermined angle included in one rotation of the rotating bodies, the number of required image data can be reduced.

[0013]

In gears that form rotating bodies, gears that mesh together have an equal time for the rotation of the rotation angle of one tooth, being a predetermined angle, an image is depicted of the rotating bodies at each rotation angle position obtained by dividing the respectively inherent predetermined angle of each rotating body by the same number of divisions for each rotating body at substantially equal intervals, and when animation image data is sequentially displayed using image data equalized for each rotating body by dividing the predetermined angles by the same number of divisions for each rotating body, by displaying the sequential image data by the same timing for each rotating body, gears (rotating bodies) that mesh together can be displayed in rotated states at rotation rates synchronized based on a ratio of the number of teeth of these rotating bodies.

[0014]

That is, when a 6-tooth gear and a 12-tooth gear mesh, the 6-tooth gear rotates at twice the rotational rate of the 12-tooth gear; however, according to the present invention, when image data stored in the storing means is sequentially switched and displayed according to an arbitrary rotation rate of the rotating bodies, by substantially synchronizing just the timing for switching the image data of the rotating bodies displayed in a synchronized and rotated state, an animation can be displayed in rotated states in which rotation rate is synchronized according to the number teeth on each gear.

In addition, since essentially the meshed gears can be displayed together by a process of simply continuing to display the image data for each rotating body in accordance with the timing for switching thereof, a user arbitrarily arranges and meshes a plurality of gears, and even if the rotation rate is arbitrarily changed, an animation of meshed gears can be easily displayed by a mobile terminal such as a mobile phone having limited storage capacity or arithmetic processing capability. In other words, an animation of rotating bodies synchronously rotating together even without using decimal point values or trigonometric functions.

[0015]

Note that rotation at an arbitrary rotation rate entails, for example, a driving gear connected to a drive source, where a driven gear meshed with the driving gear is rotated at a rotation rate according to the rotation rate of the driving gear and to the ratio of the numbers of teeth of the driving gear and the driven gear. Note that if the rotation rate of the driving gear can be arbitrarily changed, an arbitrary change to the rotation rate of the driving gear also subjects the rotation rate of the driven gear to the arbitrary change.

In addition, image data, when sequentially switched and displayed, corresponds to a number of frames displayed per unit of animation time; however, when the arbitrary elapsed time between frames for rotating bodies rotating at a rotation rate is shorter than the time to rotate an angle of division obtained by dividing a predetermined angle by the same number of division for each gear, for example, after the same image data is displayed across a plurality of frames, subsequent image data is switched to and displayed for frames thereafter, but switching the image data for each rotating body in the same frame allows for clear display in a synchronous state of gears meshed together.

[0016]

In addition, when the elapsed time between frames is longer than the time for a rotating body rotating at an arbitrary rotation rate to rotate just the angle of division, for example, when image data is in the order 1, 2, 3 ..., after the image data for 1 is displayed, the image data between 2, 3, or the like may be skipped and the image data for 3 or 4 may be displayed; however, likewise in this case, by displaying the previous image data then displaying the image data only for the same number that was skipped for each rotating body, an animation can be displayed that rotates in a state in which each rotating body is synchronized. Note that in the present invention, even when the image data for a rotating body is switched by one skip, two skips, or the like according to the relationship between rotation rate and elapsed time between frames, such is displayed by sequential switching.

In addition, in the animation display that is displayed for each frame, it is not always necessary to display a plurality of image data for each rotating body in synchronous rotation by synchronously switching in the same frame, and a number of frames may be off or the like, but synchronization must be carried out in advance so that deviations do not increasingly accumulate.

[0017]

The invention according to claim 2 includes, in the image display device according to claim 1, circular coordinate storing means for presetting, per substantially equivalent interval, a number of virtual points of a value obtained by multiplying a number of the image data for displaying an animation of rotating bodies rotating a predetermined angle to a number of the predetermined angle included in one rotation of each rotating body on the circumferences of virtual circles synchronously rotating in mutual contact, and which respectively correspond to each rotating body synchronously rotating adjacent to each other, where each virtual point is made to correspond to one of the plurality of image data for displaying the respective animations, and for storing virtual point coordinate data indicating each virtual point individually, wherein the display controlling means obtains the virtual points, which correspond to the rotation angle according to the rotation rates of the rotating bodies at the time the image data is displayed when displaying an animation of rotating bodies by sequentially displaying the image data, and the image data corresponding to the virtual points obtained from within the plurality of image data is displayed.

[0018]

Here, the virtual circles are gear-pitch circles in cases where the rotating bodies are gears. When each rotating body is synchronously rotated, that is, when meshed and rotated, the virtual circles are centered around the rotation center of each rotating body, and each of the circles corresponding to a rotating body in mutually synchronous rotation, has a radius in mutual contact. Note that when the virtual circles in mutual contact here are regarded as a pair of friction discs, the rate ratio between both friction discs (virtual circles) is set equivalent to the rate ratio between rotating bodies corresponding to each friction disc (virtual circle), and the radius each friction disc in this case is the radius of the virtual circle.

In addition, each virtual point on the circumference of a virtual circle corresponds to a rotation angle of a rotating body expressed by the plurality of image data within the predetermined angle described above. For example, if the predetermined angle for one tooth of a gear is divided by five and image data for each rotation angle is stored, five virtual points each are arranged for every tooth angle interval and each tooth has five virtual points that correspond one-to-one with five image data, and each image data repeatedly corresponds only to the number of teeth of a gear with respect to virtual points around the circumference.

Moreover, if coordinate data for each virtual point is just for selecting image data displayed simply according to rotation rate, and does not necessarily need to specify a position in relative coordinates such as x, y coordinates, the coordinate data may simply be able to specify each virtual point and be able to specify image data corresponding to each virtual point, for example, sequential numbers may be applied to each virtual point along the circumference from virtual reference points on the circumference.

According to the invention of claim 2, for example, when sequential image data is displayed according to an arbitrary rotation rate, by applying a rotation rate corresponding to the number of virtual points on the circumference of the virtual circle, an animation can be displayed according to the rotation rate, which is easily changed even if the rotation rate changes. In particular, the rotation rate can be changed more easily by expressing the rotation rate in association with a number of frames per unit of time. For example, if an animation is displayed at 10 frames/sec, in the case that the rotation rate of a rotating body is 0.1 rotations/sec, a rotating body has rotated one rotation when 100 frames have been displayed, making the rotation number of the rotating body 1/100 turns per frame. Meanwhile, in the case that the number of teeth on a gear forming a rotating body is 12 and the number of image data for displaying an animation of the rotating body rotating a prescribed angle is 5, the number of image data for displaying one complete rotation of the rotating body becomes $12 \times 5 = 60$, and the number of rotations of the rotating body per 1 image data becomes 1/60. Here, when 1/100 and 1/60 are set over a common denominator, 6/600 and 10/600 are obtained. Therefore, the number of rotations of the rotating body per frame is 6/600, and the number of rotations per one image data is 10/600.

That is, the number of rotations of the rotating body per image data in a first image data is $0(600)/600$ turns, the number of rotations in a second image data is 10/600 turns, the number of rotations from a third image data is 20/600 turns, ... Furthermore, when the number of image data is set to five, a sixth image data becomes the first image data, at which time the number of rotations is 60/600 turns. Furthermore, after image data number five has been reached, while repeating the image data number one, each image data is again rotated every 10/600 turns.

[0019]

In such as case, a reference rotation angle, that is, an image data forming $0(600)/600$, is displayed in a first frame. Next, for example, in a second frame, the number of rotations of the rotating body is $6/600$ turns, so the number of rotations closest to this number of rotations is selected for the second image data for $10/600$ turns and displayed. Then, in a frame 3, the number of rotations of the rotating body is $12/600$ turns, so the number of rotations closest to this number of rotations is again selected for the second image data for $10/600$ turns and displayed. Then, in a frame 4, the number of rotations of the rotating body is $18/600$ turns, so the number of rotations closest to this number of rotations is selected for the third image data for $20/600$ turns and displayed. In this manner, each time a sequential frame is displayed, image data is selected for display, which allows an animation of a rotating body to be displayed according to the arbitrarily set rotation rate.

Note that even when using fractions, in cases such as described above, numerators and denominators are handled separately, by essentially using the remainder when dividing a numerator by a denominator, changes to an arbitrary rotation rate can be easily responded to as described above without handling decimal point values or trigonometric functions. Therefore, even on a terminal such as a mobile phone having a limited storage capacity or arithmetic processing capability, display of rotating bodies synchronized together can be easily performed.

[0020]

For the invention according to claim 3, in the image display device according to claim 2, when, between rotating bodies synchronously rotating together adjacent to each other, times for the respective rotating bodies to rotate a predetermined angle are equalized, and the condition of offsetting the time by half of the time for a phase when rotating a predetermined angle together to rotate the predetermined angle, the display controlling means makes the rotating bodies synchronously rotating together adjacent to each other respectively match, obtains the virtual points closest to mutual contact points where the virtual circles contact each other wherever virtual circles contact each other, and based on a relationship between phases corresponding to the rotating angle of the rotating bodies in which virtual points obtained with respect to one virtual circle correspond to virtual points on another and based on the above condition, and by obtaining the relationship between the phases of the rotating bodies corresponding to the virtual points obtained with respect to the one virtual circle and the virtual points of the other virtual circle, displays the rotating bodies synchronously rotating together while satisfying the condition.

[0021]

According to the invention of claim 3, rotating bodies in synchronous rotation can be displayed in a state in which the teeth of gears are meshed together.

Furthermore, based on a relationship between a virtual point obtained with respect to one virtual circle and the phase corresponding to a rotation angle of a rotating body corresponding to the one virtual circle and based on the above condition, when displaying the rotating bodies synchronously rotating together corresponding to the condition by obtaining the relationship between the virtual point obtained with respect to another virtual circle and a phase of a rotating body corresponding to the other virtual circle, for example, perform the following.

[0022]

For a virtual point closest to a point of contact between virtual circles respective to gears meshing together, each virtual point corresponds to image data for an animation, so a rotation of an angle smaller than a resolution of a rotation angle in a rotating display of rotating bodies cannot be displayed. Therefore, the point becomes that closest to the point of contact in terms of the resolution of the rotation angle. In addition, a remainder when sequential numbers applied along the circumferential direction at each virtual point are divided by a number of divisions (number of image data) of the predetermined angle described above forms one period of a rotation of the predetermined angle, and when a period for a number of teeth is present in one rotation of a gear, a phase is expressed within each period of the one rotation. Furthermore, image data corresponding to each phase within one period is present.

[0023]

Here, the phase of a virtual point closest to the point of contact with the one virtual circle is obtained. As described above, the remainder obtained by dividing the sequential numbers attached to the virtual points in the circumferential direction by the number of image data for animation display of a prescribed angle is the phase of the virtual point closest to the point of contact. In addition, the phase of a virtual point closest to the point of contact with another virtual circle is similarly obtained.

Moreover, it is presumed that the rotation angle of the gear corresponding to the one virtual circle has already been determined when meshing gears together, and the phase of the rotation angle determined for the one gear corresponding to the one virtual circle is obtained. This phase corresponds to image data when displaying the one gear.

[0024]

The difference between the phase of the virtual point closest to the point of contact of the one virtual circle and the phase corresponding to the rotation angle of the one gear corresponding to the one virtual circle is obtained. Furthermore, the rotation angle (phase) of the other gear corresponding to the other virtual circle is determined based on the rotation angle (phase) of the gear already determined for the one gear. In this case, the virtual point closest to the point of contact of the one virtual circle is approximated for maximum points of contact in terms of the resolution of the rotation angle of the corresponding gears, and the virtual point closest to the points of contact of the other virtual circle are approximated for maximum points of contact in terms of resolution of the rotation angle of the corresponding gears, so together, these virtual points also form closest points in terms of the resolution of the rotation angle of the gears and can be regarded as substantially the same point.

[0025]

Here, when a phase for a virtual point closest to the point of contact of the other virtual circle—being out of phase only a difference (amount of deviation) between the phase of the virtual point closest to the point of contact of the one virtual circle and the phase corresponding to the rotation angle of one gear corresponding to one virtual circle—is achieved, the phase corresponding to the rotation angle of the other gear, the phase of the one gear and the phase of the other gear deviate by exactly the same amount of deviation from two virtual points that essentially coincide with each other on two virtual circles in mutual contact, and in this case, the teeth of the one gear and the teeth of the other gear overlap and the tooth space of the one gear and the tooth space of the other gear overlap. In other words, the two gears rotate in substantially the same phase, but that does not mean that they are displayed in a meshed state.

[0026]

Therefore, when determining the phase of the other gear with respect to the phase of the one gear, a phase of exactly half a period of the amount of deviation is deviated in the forward or reverse direction rather than deviating the phase of the virtual point closest to the point of contact of the other virtual circle by exactly the amount of deviation described above. For example, when the number of image data is 5, there are five phases, so the amount of deviation of the half period phase is 2.5, however, when a decimal point is rounded down or rounded up, the amount of deviation of the half period phase becomes 2 or 3. Here, it is presumed that the amount of deviation of the half period phase is 2, and if the amount of deviation between the phase of the virtual point closest to the point of contact of the one virtual circle and the phase corresponding to the rotation angle of the one gear corresponding to the one virtual circle is set to 3, a phase in which the phase of the virtual circle closest to the point of contact for the other virtual circle is deviated by 1 or 5 serves as the phase of the other rotating body, whereby a mutually meshed state can be formed such that a tooth of the other gear meshes with the tooth space of the one gear, and a tooth of the one gear meshes with the tooth space of the other gear. Note that the when the number of image data is 5, there are, for example, phases 0, 1, 2, 3, 4, and 5, and when the value of the phase when phase-deviated is 5 for higher, the remainder when divided by 5 (number of image data) is the value indicating the phase, and in the case that the value of the phase when deviated is negative, 5 (number of image data) is added to the negative value, and this value (value is further divided by 5 (number of image data) when this value is 5 or higher) indicates the phase.

[0027]

In other words, when an animation of mutually meshed gears is displayed as rotating bodies that synchronously rotate with each other, display must be done such that the teeth of the other gear mesh with the tooth spaces between the teeth of the one gear, and likewise in this case, arithmetic processing using a trigonometric function or the like is generally required.

However, by presupposing the virtual points on the virtual circumference, being the pitched circles described above, meshing gear teeth can be portrayed without using a trigonometric function or the like. Note that the coordinate data for the virtual points here must be values that express a position in coordinates such as x, y coordinates so that a virtual point closest to a point of contact between virtual circles can be obtained, however, these values must be obtained as processes of a rotating body image display program, so coordinate data obtained in advance may be used for the data. In addition, in cases such as when decimal point values are required for coordinate data, an integer may be obtained by rounding down, rounding up, or the like. Moreover, in cases where a decimal point value is obtained as an arithmetic processing result, and the portion following the decimal point may be ignored.

Therefore, the values of coordinate data and the like are essentially approximated values unless the actual data are integers.

[0028]

The invention according to claim 4 includes, in the rotating body image display device according to any one of claims 1 to 3, color changing means for changing the color of an image of a rotating body, wherein the storing means stores index images having a color palette and color scheme image data as index images by overwriting color data on the color palette of an index image stored in the storing means.

[0029]

Here, the index images are, for example, GIFs or PNGs.

According to the invention of claim 4, a region of color change is designated in a state in which an index image is decompressed and displayed as an image, where the color in gear image data can be changed such that the color of the region is changed without the use of index image graphic software. In other words, even without decompressing an index image in, for example, the display memory of a VRAM or the like to enable a displayable state, the color of a rotating body can be overwritten in the color table, that is, the color palette of image data in a non-decompressed binary state by, for example, directly overwriting the RGB value of an index used in the color of the region of the rotating body of the image.

[0030]

Note that color palette (when global and local, basically a global color palette) positions in the binary data of an index image are determined, and the data (for example, RGB values) for each color in the color palette are arranged in an index order, so even for index images in a binary data state, the RGB value of an index corresponding to a gear can easily be changed if the index corresponding to the gear is known, which allows for the color of the rotating body to be changed. Therefore, indexes specifying the color of a rotating body must be predetermined. In addition, the rotating body is split into a plurality of regions, for example, by an outline region (border), an inside thereof, and the like, and if different indexes are determined for each region, the color can be changed for each region even when the rotating body is split into a plurality of regions and displayed in a plurality of colors.

[0031]

Moreover, the color palette in an index image is not compressed and portions of the color scheme image data are compressed. Therefore, even if an index image is not decompressed, the color palette can easily be overwritten.

As described above, the color of a rotating body can be changed even without using an application for an index image, which allows the color of the rotating body to be changed by a program having an extremely limited data capacity. Also, no rotating body image data for a plurality of different colors needs to be stored in advance, and the number of changeable colors in the rotating body can be extremely large.

That is, even though the color of the rotating body can be changed, the data capacity of the rotating body image display program is not increased, likewise, the image data capacity that is required can be minimized, therefore allowing for utilization suited to a mobile terminal such as a mobile phone having limited storage capacity or arithmetic processing capability. In addition, communication fees can be reduced when downloading the rotating body image display program and the data therefore from the internet using a mobile phone communication line, or downloading and uploading a file created by the rotating body image display program.

[0032]

The invention according to claim 5 is a rotating body image display program for displaying, on a computer provided with a display device, an animation of a plurality of rotating bodies such as gears that can synchronously rotate with each other, have a rotationally symmetrical form in which a form prior to rotation matches a form following rotation whenever rotated a respectively inherent predetermined angle, and in which times to rotate respectively inherent predetermined angles essentially match when synchronously rotated together, wherein the program implements, on the computer, a function for expressing, as image data for respective animations of the plurality of rotating bodies having mutually different predetermined angles, an image of the rotating bodies at respective rotation angle positions obtained by dividing the respectively inherent predetermined angle for each rotating body together by the same number of divisions at substantially equal intervals and for displaying the animation of the rotating bodies by sequentially displaying the image data, which is equalized for each rotating body by dividing the predetermined angles by the same number of divisions for each rotating body, and a function for substantially synchronizing the timing for switching the image data for the rotating bodies together, which are synchronously displayed in rotated states, when image data is sequentially switched and displayed according to the given rotation rates of the rotating bodies.

[0033]

Here, the computer is a device capable of executing the rotating body image display program, for example, a mobile terminal such as a mobile phone having limited storage capacity or arithmetic processing capability.

According to the invention of claim 5, a function effect similar to that of the invention of claim 1 can be obtained.

[0034]

The invention according to claim 6 includes, in the rotating body image display program according to claim 5, implementing, on the computer, a function of presetting, per substantially equivalent interval, a number of virtual points of a value obtained by multiplying a number of the image data for displaying an animation of rotating bodies rotating a predetermined angle to a number of the predetermined angle included in one rotation of each rotating body on the circumferences of virtual circles synchronously rotating in mutual contact, which respectively correspond to each rotating body synchronously rotating adjacent to each other, where each virtual point is made to correspond to one of the plurality of image data for displaying the respective animations, obtaining the virtual points, which correspond to the rotation angle according to the rotation rates of the rotating bodies at the time the image data is displayed when displaying an animation of rotating bodies by sequentially displaying the image data, and displaying the image data corresponding to the virtual points obtained from within the plurality of image data.

According to the invention of claim 6, a function effect similar to that of the invention of claim 2 can be obtained.

[0035]

The invention according to claim 7 includes, in the rotating body image display program according to claim 6, implementing, on the computer, a function of, when, between rotating bodies synchronously rotating together adjacent to each other, times for the respective rotating bodies to rotate a predetermined angle are equalized, and the condition of offsetting the time by half of the time for a phase when rotating a predetermined angle together to rotate the predetermined angle, making the rotating bodies synchronously rotating together adjacent to each other respectively match, obtaining the virtual points closest to mutual contact points where the virtual circles contact each other wherever virtual circles contact each other, and based on a relationship between the phases corresponding to the rotating angle of the rotating bodies in which virtual points obtained with respect to one virtual circle correspond to virtual points on another virtual circle and based on the above condition, and by obtaining the relationship between the phases of the rotating bodies corresponding to the virtual points obtained with respect to the one virtual circle and the virtual points of the other virtual circle, displaying the rotating bodies synchronously rotating together while satisfying the condition.

According to the invention of claim 7, a function effect similar to that of the invention of claim 3 can be obtained.

[0036]

The invention according to claim 8 includes, in the rotating body image display program according to any one of claims 5 to 7, implementing, on the computer, a function of changing the color of an image of a rotating body, wherein index images having a color palette and color scheme image data are stored in storing means on the computer as index images, by overwriting color data on the color palette of an index image stored in the storing means.

According to the invention of claim 8, a function effect similar to that of the invention of claim 4 can be obtained.

EFFECTS OF THE INVENTION

[0037]

According to the present invention, a user arbitrarily arranges gears in a combination on a mobile terminal such as a mobile phone having limited storage capacity or arithmetic processing capability, creating a gear train and allowing for content displayed in a state in which these gears are rotated at an arbitrary rotation rate without using floating point arithmetic, trigonometric function calculation, or the like.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0038]

The rotating body image display device and rotating body image display program are described below as one Embodiment of the present invention.

The rotating body image display device in this example is a mobile terminal having the communication function of a mobile phone 1 or the like, which can be connected to the internet. Furthermore, the mobile phone 1 has a well-known browser installed as an application for viewing webpages. In addition, the mobile phone 1 can execute, for example, Java applets, and the rotating body image display program of this Embodiment can be downloaded onto the mobile phone 1 as a Java applet (a program that can be executed on multiple platforms and has a relatively low storage capacity) over the internet and can be executed on the mobile phone 1. Moreover, the Java applet executed on the mobile phone 1 is an embedded Java applet with limited function, which is incapable of using floating point arithmetic, trigonometric function calculations, or the like directly.

[0039]

Furthermore, as illustrated in the block diagram of FIG. 1, the mobile phone 1 includes a displaying part 2 for displaying an image, a calculation processing part 3 for executing various programs, a key inputting part 4 provided with various keys used when a user operates the mobile phone 1 and inputs data as well as a tenkey or the like used to move a pointer or the like, a network communication part 5 used to connect to the internet or the like, and a storing region 6 (may be a magnetic or optical storage medium) composed of non-volatile memory.

[0040]

The displaying part 2 includes, for example, a color liquid crystal display, a drive circuit for the liquid crystal display, a display control circuit provided with VRAM, and the like, and performs display of an image on the color liquid circuit display based on control of the calculation processing part 3.

The calculation processing part 3 can execute the browser, the Java applet, or the like as a program. Furthermore, the calculation processing part 3 is provided, for example, with a CPU 31, an execution memory 32 (DRAM), and the like as an MPU chip for embedded equipment.

The calculation processing part 3 executes the rotating body image display program of the present invention and functions as display controlling means and color changing means of the present invention.

[0041]

The key inputting part 4 has the key arrangement of the mobile phone 1 and therefore is capable of inputting hiragana, katakana, alphabetic characters, and the like, is provided with a tenkey and symbol keys by which kanji, symbols, and the like can be input by a conversion function, arrow keys or the like that can be used for menu selection, pointer movement, as a pointing device, and the like, and dedicated keys for calls, internet connection, and the like.

In this example, the network communication part 5 is a wireless communication device for the mobile phone 1, which enables wireless data communication with a base station and connection to the internet using a communication line of the mobile phone 1.

In addition, the storing region 6 is composed, for example, of a non-volatile flash memory, an EEPROM, or the like, and along with storing the Java applet or the like downloaded from a server via the internet, in this example, as described below, image data used by the rotating body image display program, data related to display of a gear generated for display, or the like is stored as a Java applet. Therefore, the storing region 6 functions as the storing means and circular coordinate storing means of the present invention.

[0042]

Here, a summary is given of the rotating body image display program executed by the calculation processing part 3 as a Java applet.

As illustrated in FIG. 2, the rotating body image display program essentially displays an animation of a plurality of gears 7 (spur gears) as rotating bodies, and when the plurality of gears 7 are meshed together, displays a synchronously rotating animation in a state in which the plurality of gears 7 are meshed; however, the number of teeth and diameter of each gear 7, the position of each gear 7, whether each gear 7 is meshed, determination of a drive gear 7 (driving gear) and a driven gear 7 (driven gear), the position of each gear 7, and the like can be set or selected by a user of a mobile terminal. Moreover, hands 9 (illustrated in FIG. 5) can be displayed integrally rotating with arbitrary gears in addition to the gears 7, and by adjusting a rotation rate of a gear serving as a driving gear, a gear ratio of gears meshing together, and the like, movement of an hour hand 9a, a minute hand 9b, and a second hand 9c of a clock such as depicted in FIG. 5(L) can be replicated. That is, design of a skeleton clock in which the gears 7 are exposed is enabled. Note that a clock does not need to be replicated by the rotating body image display program, and by combining the plurality of gears 7 and displaying the hands 9 (illustrated in FIG. 5), which rotate along with the gears 7, an animation of the gears 7 can be artistically portrayed.

[0043]

Note that the gears have a rotationally symmetrical form in which a form prior to rotation matches a form following rotation for every rotation of a predetermined angle. In addition, for gears, for example, a plurality of predetermined angles in which a form prior to rotation matches a form after rotation are included at 30° , 60° , 120° , 180° , and the like in the case that the number of teeth of a gear is 12, however here, the predetermined angles are less than 360° and of an angle forming a minimum rotational symmetry according to a rotating body thereof; in the case described above where the number of teeth of the gear is 12, the predetermined angle is 30° . In other words, the predetermined angle is the rotation angle of one tooth of the gears 7. Here, when viewed from one angular position at which rotation of an outer periphery of the gears 7 is fixed, alternately repeating a tooth part and a tooth space therebetween of the gears 7, the predetermined angles can be viewed as one period. That is, in the course of one rotation of a gear, repeating wave-like periods of the tooth part and tooth space described above repeat only as many times as the number of teeth. Accordingly, a case in which rotation of a prescribed angle of a gear is portrayed as a period is given below.

[0044]

In addition, the gears 7, which are synchronized in a meshed state, have respectively opposite rotation directions, and the tooth spaces of one gear 7 and the tooth parts of another gear 7 mesh where portions of teeth meshing together are seen, so each gear 7 synchronously rotating while meshed together takes an equivalent time to rotate one period (minimum prescribed angle) whether a number of periods (number of teeth) in one rotation of the gears 7 meshing together is the same or different. The rotation angle of each gear is therefore shifted by a half period. In other words, the meshed gears synchronously rotate in reverse together, and a phase thereof is deviated by a half period.

[0045]

Processing of the Java applet as a rotating body image display program in a mobile terminal as a rotating body image display device is described below.

First, the mobile phone 1 is, for example, in an internet connection environment, and by accessing a prescribed webpage via the internet, the rotating body image display program is automatically downloaded as a Java applet via a hyperlink or the like, and is stored in the storing region 6 of the mobile phone 1. Here, data for the rotating image display program and data used by the rotating body image display program are downloaded as data for the Java applet.

[0046]

The data stored in the storing region 5 includes rotating body image data, and the storing region 5 is provided with an image data storing region 61.

In this example, rotating body image data is stored as an index image 62 provided with a palette part 621 having a color palette (color table, color palette), and a pixel data part 622 having image data (color scheme image data) based on the color palette.

Note that the index image 62 utilizes a GIF in this example. In addition, colored forms of the gears 7 are stored in the image data, but the gears 7 here are split into three regions: outline regions forming outer borders, regions inside the outline, and regions outside of the outline, for example, that depicts shadows or side surfaces that make the gears appear 3-dimensional, and a color is set for each region. In other words, an index for the color palette of the palette part 621 is designated for each of the aforementioned regions of the gears.

[0047]

The palette part 621 of the index image 62 has, for example, a maximum of 256 indexes, where an RGB value (for example, 8 bits each for R, G, and B) can be designated for each index, and a maximum of 256 colors can be selected from so-called full color. Note that the number of indexes can be set in a number of stages not exceeding 256. For example, just four colors can be selected for four indexes.

Furthermore, when usable colors in the palette part 621 are set to four, for example, a brightness (if 8-bit, 0 to 255) for each of R, G, and B is designated for an index of 0 to an index of 3 respectively.

[0048]

In addition, in the pixel data part 622, for example, all indexes of pixels (pixels) included in the outline regions of the gears are designated 0, indexes of pixels in the regions inside of gears enclosed by the outlines are designated 1, and indexes of pixels in the regions outside of the outlines (outer shadows or the like of the gears) are designated 2. Furthermore, the color of the outline regions of the gears, the color of the regions inside of the gears, and the color of the regions outside of the gears are determined by each R, G, B value of the designated indexes. Note that a color can also be designated as transparent. For example, when gears overlap frontward and rearward in the case that the regions inside the outline regions of gears are designated as transparent, the gear in the rear is displayed inside of the gear outline. When there is no rear gear, the transparent portion is displayed as the background color.

Here, when the index image 62 is downloaded, an RGB value set by default is stored for each index of the palette part 621.

[0049]

Also, in this example, five types of the gears 7, for example, for which the numbers of teeth are 6, 12, 18, 24, and 30 are set as displayable gears 7. Note that the shapes of the teeth of each of the gears 7 will be substantially the same. Here, a value cog, which forms the basis of the number of teeth of the gears, is set as a positive integer. For example, the value cog is 6 in the present example. Furthermore, types by number of teeth of the gears 7 are obtained by multiplying the cog by a (positive) integer, and with a basis of 6, cog₀ is 6, cog₁ is 12, cog₂ is 18, cog₃ is 24, and cog₄ is 30. These cog₀ to cog₄ are elements of a collection COG of the numbers of teeth of the gears 7. The gear ratio when these gears 7 are meshed is (integer:integer).

Note that here, when cog_m is expressed, m is sequential numbers ordered from a lowest number of teeth of the gears 7 having different numbers of teeth, that is, m = 0, 1, 2, 3, ..., and corresponds to a value 1 less than a value obtained as the integer for multiplying with the number of teeth serving as the basis described above.

[0050]

In addition, pitch circles 8 (virtual circles) are set for the gears 7 as depicted in FIG. 2, a distance between two of the gears 7 that are meshed together is the sum of radii rad of the two pitch circles 8 corresponding to these gears 7, and the pitch circles 8 of the two gears 7 that are meshed together are in a state of contact. Moreover, a gear ratio between the gears 7 and a gear ratio when the pitch circles 8 in mutual contact form friction discs are designed to match.

Note that when radii rad of the gears 7 (rotating bodies) is described, this indicates the radii rad of the pitch circles 8 corresponding to the gears 7.

Furthermore, in the types of the gears 7 based on the number of teeth described above, a radius rad_m (m indicates the type of gear having a different number teeth as described above) of each of the gears 7 is the positive integer to multiply with to obtain the number of teeth for the number of teeth type, so when the radius rad₀ of a cog₀ gear forming the basis is, for example, 1, rad₁ is 2, rad₂ is 3, rad₃ is 4, and rad₄ is 5.

These rad₀ to rad₄ (rad_m) are elements of a collection RAD of the gear radius types.

[0051]

Accordingly, index images have different numbers of teeth, that is, the index images 62 (image data) for the plurality of types (five types) having different predetermined angles (periods) described above are downloaded and stored in the storing region 6.

Also, in order to display a rotating animation of the gears 7, a plurality of the index images 62 having different rotation angles are required, but in this example, a same number of the index images 62 are stored in the storing region 6 in order to portray an animation of each of the gears 7 having differing numbers of teeth, that is different predetermined angles (periods). Note that when a change to the color of the gears 7 or the like is performed, different colored index images 52 are generated, but each of the gears 7 (gears 7 having different numbers of teeth and gears 7 having different colors) all use a same number of index images 62 in the animation.

[0052]

Furthermore, for the index images 62 of the animation, for example, the index image 62 for each rotation angle (angle of division) is used, which is obtained by dividing the predetermined angle (period) described above by a same number of divisions according to each type of the gears 7 at equal intervals along the circumference. Therefore, the number of image data for each type of the gears 7 is the same for each of the gears 7 having differing numbers of teeth. In addition, by setting the rotation of a predetermined angle of one tooth to one period, the index image 62 for each rotation angle, which is divided by the same number of divisions at equal intervals for each gear, indicates each phase position when the one period is divided at equal intervals, and when sequential numbers are applied to the index image 62 at each divided rotation angle, index images 62 at a same number in the gears 7 each having differing numbers of teeth indicate a same phase position. The number of pieces of image data indicates the number of phase positions displayable within one period (a common value according to each gear, which have different phases and numbers of teeth).

[0053]

Furthermore, as depicted in FIG. 3, in the case that the number of pieces of image data for a predetermined angle is 5, for example, in a gear 7 in which the number of teeth is 12, when one tooth 7 (for example, this tooth 7 is a reference) is pointed straight up, because the predetermined angle is 30° , the angle of division when divided into equal parts is 6° , and for the index images 62, image data image_0 serving as a reference for the 0° state described above, then image data rotated 6° each, that is, image data image_1 rotated 6° , image data image_2 rotated 12° , image data image_3 rotated 18° , and image data image_4 rotated 24° are created in advance and stored. Note that image data further rotated 6° , that is, image data rotated 30° , is rotated at a rotationally symmetrical angle, forming the same image data as the first 0° image data image_0 . Moreover, respective image data (index image 62) is assigned by sequential natural numbers (including 0) that correspond to a set number of divisions, such as 0 to 4 or the like, ordered by angle size as described above.

Note that in this example, the stored respective image data is expressed as $\text{image}^{\text{collar}}_{m,n}$, and in this case, color expresses a color set in the color palette of the index image 62 described above, m expresses a type (number of teeth) of gears having differing numbers of teeth, and n expresses the sequential numbers applied to the respective image data described above.

[0054]

Furthermore, when the number of teeth of the gears 7, which are rotationally symmetrical, is 12, a form prior to rotation matches a form following rotation at 30° of rotation, so the image data of image_0 (lists only sequential numbering) to image_4 are displayed in order, then display of the image data image_0 to image_4 in order is repeated, which allows an animation to be displayed of continuing to rotate the gears 7 (rotating body).

[0055]

As such, the image data of image_0 to image_4 for each rotation angle can be thought of as indicating respective phases at equally divided points of one phase in which tooth parts and tooth spaces repeat. Furthermore, in each of the gears 7 having differing numbers of teeth, the respective image data of image_0 to image_4 are drawn in the displaying part 2 in order and displayed on the display screen, which enables rotation in a synchronized state. Note that the rotation directions of the gears meshing together reverse together, and a phase of rotation is deviated by a half period at portions where a tooth part of another gear corresponds to a tooth space of one gear due to the teeth meshing.

[0056]

As described above, the storing region 6 stores the image data (index images 62) of the plurality of gears 7 having differing numbers of teeth respectively, and is storing means for images depicting each rotating body as a rotating body at a rotation angle position (respective phase), obtained by dividing by a respectively inherent predetermined angle (periods) of the gears 7 along with a same number of divisions at equal intervals and in which respectively equalized image data are stored for each rotating body.

In addition, circular coordinate data 63 for each of the plurality of gears 7 and their respective number of teeth described above is stored in the storing region 6. As depicted in FIG. 4, the circular coordinate data 63 sets, for each of the gears 7 and their respective number of teeth, a virtual point K at each of the aforementioned angles of division for each of the gears 7 on the circumference of the pitch circle 8, and indicates the position of each virtual point K. Note that a number res_m of a virtual point K set for each angle of division is obtained by multiplying a number of image data (phase (number of displayable phases)) by the number of teeth cog_m . Here, res_m indicates a total number of the virtual points K for each of the gears 7 having a differing numbers of teeth. FIG. 4 illustrates the pitch circle 8 of the gear 7 in which the number of teeth is 12 and the virtual points K on the circumference of the pitch circle 8. Also, in FIG. 4, the black dots among the virtual points K indicate the center points of the teeth of the gear 7, with the predetermined angle being between the black dots, in other words, forming one period.

[0057]

Furthermore, in the case of the gear 7 in which the number of teeth is 12, when the phase of the virtual points K is set to 5, for example, the predetermined angle is 30° and the angle of division is 6° , so virtual points k are set every 6° on the circumference. Therefore, each of the virtual points K are set to an angular position of $0^\circ, 6^\circ, 12^\circ, 18^\circ, 24^\circ, \dots$, in clockwise rotation, and the last virtual point K is 354° .

Furthermore, in the case that the number of image data (phase) is 5, 30 virtual points K are set at every 12° when the number of teeth is 6, 90 virtual points K are set at every 4° when the number of teeth is 18, 120 virtual points K are set every 3° when the number of teeth is 24, and 150 virtual points K are set every 2.4° when the number of teeth is 30. Each of the virtual points K is set with a sequential number, for example, of 0 to res_m-1 .

In addition, each of the virtual points K also corresponds to the image data $image_0$ to $image_4$. For example, when sequential numbers are assigned to each of the virtual points K, the remainder when divided by 5, being the number of image data (phase) equalized for each of the gears 7 described above, forms a sequential number n, which is assigned to each image data. For example, virtual points K 0 to 4 correspond in order as-is to image data from $image_0$ to $image_4$, virtual points K 5 to 9 correspond in order as-is to image data from $image_0$ to $image_4$, and similarly thereafter, each of the virtual points K correspond five at a time to the image data from $image_0$ to $image_5$ [sic]. Therefore, when the virtual points K are determined, the image data $image_0$ to $image_4$ are set. In addition, each of the image data $image_0$ to $image_4$ correspond to a phase within one period as described above. Moreover, a total number res_m of the virtual points K for each type of each of the gears 7 corresponds to a chronological resolution of an animation because each of the virtual points K corresponds to the image data from $image_0$ to $image_4$ for each angle of division as described above. An animation more detailed than res_m cannot be displayed for each of the gears 7. That is, there is no image data of an angular position more minute than an angle of division obtained by dividing 360° by res_m , so the value of the res_m indicates resolution of the animation of rotating bodies.

[0058]

In this example, the coordinate position of each of the virtual points K indicates coordinates on the circumference of the radius rad_m , with the x, y coordinates at the center of the pitch circle 8 being 0, 0.

Here, a collection CPOS_m (center position) of coordinates of virtual points K having a number of teeth m is expressed by $((x_{m,0}, y_{m,0}), (x_{m,1}, y_{m,1}), \dots, (x_{m,res_m-1}, y_{m,res_m-1}))$.

Note that each x, y coordinate utilizes integers and do not use numerical values containing a decimal point. As such, each x, y coordinate value is an integer approximated value of coordinates. Note that x_m and y_m indicate x, y values for each of the gears 7 having differing numbers of teeth as described above, where 0, 1, 2, ..., res_m-1 is i, and sequential numbers are applied to the virtual points K on the gears. In addition, each coordinate is simply associated with the sequential numbers applied to the virtual points K, and may be expressed as CPOS_m = { $X_{m,0}, X_{m,1}, \dots, X_{m,res_m-1}$

y_{m,res_m-1} .

[0059]

Here, the x, y coordinates are given as coordinates on the circumference of the radius rad_m with the x, y coordinates in the center of the pitch circle 8 being 0, 0 as described above, and having $(x_{m,0}, y_{m,0}) = (0, rad_m \text{ (radius)})$ as a starting point, rotation commences clockwise. When $i = 0, 1, 2, \dots, res_m-1$, the following relationship is established to enable the reduction of information required to obtain $CPOS_m$.

That is, having $(x_{m,0}, y_{m,0}) = (0, rad_m \text{ (radius)})$ as a starting point, when the virtual points K are arranged on the circumference in equal intervals, the positions of each of the virtual points K are linearly symmetrical with the y-axis, so the relationship $(x_{m,i}, y_{m,i}) = (-x_{m,res_m-1-i}, y_{m,res_m-1-i})$ is established.

Moreover, in the case that the remainder is 0 when the total number of virtual points K res_m is multiplied by 2, the relationship $(x_{m,i}, y_{m,i}) = (x_{m,(res_m-1)/2-i}, -y_{m,(res_m-1)/2-i})$ is established.

[0060]

In addition, in the case that the remainder is 0 when the total number of virtual points K res_m is multiplied by 4, being also symmetrical with respect to the 45° and -45° line segments passing through the origin of the x, y coordinates, the relationship $(x_{m,i}, y_{m,i}) = (x_{m,(res_m-1)/4-i}, y_{m,(res_m-1)/4-i})$ is established.

Furthermore, the coordinate data for each virtual point is a coordinate system corresponding to a display region of the display screen; however, gears 7 that partially protrude from the screen can also be displayed, and the coordinate system may be wider than the display region. When the image data for the gears 7 are actually arranged on the coordinate system, the positions of the gears 7 are expressed by a center of rotation (center of the pitch circle 8) of the gears 7, and the coordinates of each of the virtual points K on the coordinate system are obtained by adding the x, y coordinate values at the centers of rotation of the gears 7 arranged on the coordinate system to the x, y coordinate values of the virtual points K for which the x, y coordinates at the centers of pitch circles 8 stored in the storing region 6 are 0, 0.

Therefore, while corresponding to synchronously rotating adjacent rotating bodies (gears 7), the storing region 6 functions as the circular coordinate storing means that stores coordinate data of virtual points K indicating each of the virtual points K in the case where the number of virtual points K, a value obtained by multiplying the number (phases) of image data for animation displaying a rotating body (gear 7) rotating a prescribed angle by the number of prescribed angles (number of teeth) included in one rotation of each rotating body (gear 7), are set at approximately equivalent intervals on the circumferences of virtual circles (pitch circle 8) mutually in contact and synchronously rotating, and each virtual point K corresponds to one of the plurality of image data for the respective animation displays.

[0061]

In this example, the rotation rate is set as a fraction. For example, a numerator and denominator of a rotation rate are input by a user, for example, a numerator of 1 and a denominator of 600 or the like are input. Furthermore, in this example, the value of the numerator is incremented for every 100 msec, that is, every 1/10 seconds; thus increasing the numerator. At this time, the rotating body gears rotate, for example, 1/600 of a rotation. Note that in this example, the animation is displayed by displaying 10 frames per second, or 1 frame every 100 msec.

Gears 7 in which the number of teeth is 6 require 6 periods to go around when considered in terms of the period described above, and because there are 5 image data in 1 period, one round of the gears 7 is expressed by 30 (the total number res_m of the virtual points K set as described above) image data. That is, one rotation (one round of rotation) of the gears 7 is divided into 30, being the number of virtual points K set as described above, and the animation is performed using images that are 1/30th of one round each. Presuming rotation at the rotation rate expressed as a fraction above, with a fraction where the number of the virtual points K set on the circumferences of the pitch circles 8 is set to the denominator and the numerator is set to 1, and the fraction of the rotation rate, the denominators, are unified to form a common denominator. In other words, a common multiple of the denominators is obtained. Here, the common multiple of the denominator is 600.

Therefore, the numerical value $1/30$ (res_m), being the resolution described above, is $20/600$. Furthermore, each of the virtual points K described above are assigned as sequential numbers obtained by multiplying the denominator of the rotation rate and a value f, which is obtained by multiplying by the number of the virtual points K when the common multiple of the numbers of virtual points is taken.

Therefore, when the virtual points K are set to $K_0, K_1, K_2, \dots, K_{\text{res}_m-1}$, $0(600)$ is assigned to the virtual point K_0 , 20 is assigned to the virtual point K_1 , 40 is assigned to the virtual point K_2 , and 580 is assigned to the virtual point K_{res_m-1} .

Note that when a common denominator is found, the value (20) multiplied by res_m , being the denominator, becomes f in this example, which is described below. In addition, a common multiple of the denominator of the rotation rate and the denominator of the resolution becomes b, which is described below, and the numerator after a common denominator is found becomes s, which expresses a rate described below (circumferential rate per frame update time).

[0062]

When an animation is performed at a rotation rate of $1/600$ from the image data image_0 corresponding to the virtual point K_0 set as described above, the denominator, starting at 0 , increases by 1 every 100 msec. Here, after 100 msec, the numerator of the fraction expressing the rotation rate moves from $0(600)$ to 1 , becoming 2 after 200 msec, however, these are both smaller than the $20/600$ corresponding to a second virtual point K, so the image data image_0 is displayed at the respective timing of 100 msec each.

Furthermore, when $2,000$ msec have elapsed, the numerator of the fraction of the rotation rate becomes $20/600$. In this case, the fraction $(20/600)$ of the rotation rate is at or above $20/600$, which corresponds to the image data image_1 , so until the timing of $1,900$ msec, the image data image_0 is displayed, and at or after the timing of $2,000$ msec, the image data image_1 is displayed.

[0063]

Moreover, at the timing of $4,000$ msec, the numerator of the rotation rate is 40 , and as the numerator corresponding to the image data image_2 is 40 or above, the image data image_2 is displayed. Also, at the timing of $6,000$ msec, the numerator of the rotation rate is 60 , and as the numerator of the fraction corresponding to the image data image_3 is 60 or above, the image data image_3 is displayed. That is, every $2,000$ msec thereafter, the image data image_0 to image_4 switch sequentially in ascending order and are displayed, and until switched, the same image data image_0 to image_4 will be displayed every 100 msec.

[0064]

Note that when the numerator of fractions sharing a common denominator with the rotation rate exceed the denominator, the fraction is reduced by exactly the value of the denominator and so becomes $600(0)/600$ in the case described above, when $60,000$ msec have elapsed, and $1/600$ when $60,100$ msec have elapsed. In addition, although a process is performed to display 1 frame every 100 msec, in actuality, a deviation may occur whenever a frame is displayed. For example, displaying a previous frame then displaying a subsequent frame may take 120 msec or the like. Therefore, when the numerator of the fraction of the rotation rate counts up each time the process of displaying the one frame is performed, a deviation arises between the clock provided as a function of the mobile phone 1 and the clock formed by combining the gears 7 using the rotating body image display program. Therefore, the up count of the numerator of the fraction of the rotation rate is unrelated to the display timing per frame, and corresponds to a clock having a clock function based on an oscillator in the mobile phone 1 and is performed every 100 msec. Therefore, when the display timing per frame cumulatively exceeds 100 msec or more, for example, the numerator of the rotation rate counts up twice between the display timing of the previous frame and the display timing of the subsequent frame.

[0065]

Moreover, with the fraction of the rotation rate at $1/13$, for example, when the rotational display of the gears 7 in which the number of teeth is 6 is performed, the number of virtual points K res_m set as described above for the gears 7 in which the number of teeth is 6 becomes 30 , and the common multiple of the denominator of the rotation rate and the set number of virtual points K becomes 390 . Therefore, the rotation rate fraction becomes $30/390$, and the numerator of the rotation rate increases by 30 every 100 msec (time for 1 frame). Here, when the image data image_0 is displayed first, after 100 msec, the rotation rate fraction moves from $0(390)/390$ to $30/390$. In this case, the numerator of the resolution assigned to the virtual point K_0 becomes $0(390)$, the numerator assigned to the virtual point K_1 becomes 13 , the numerator assigned to the virtual point K_2 becomes 26 , and the numerator assigned to the virtual point K_3 becomes 39 , so after 100 msec have elapsed, the image data image_2 corresponding to the virtual point K_2 closest to the numerator of the rotation rate fraction is displayed.

That is, when a rapid rotation speed is set, one or more of the image data image₀ to image₄ may be skipped and displayed without the image data image₀ to image₄ being displayed in order of the sequential numbers.

[0066]

From the above, a rotating animation of gears rotating can be displayed at an arbitrary rotation rate via addition, subtraction, multiplication, and division arithmetic of integers and natural numbers without the use of floating point arithmetic, trigonometric functions, or the like. Note that although fractions are used in the arithmetic, the calculations mainly deal with the handling of numerators, and, for example, when a numerator increased along with an elapsed time exceeds a denominator, the denominator is used when multiplying the numerator and the denominator to obtain a remainder, and calculations are performed using integers.

[0067]

Processing based on execution of the rotating image display program in the calculation processing part 3 of the mobile phone 1 is described below.

Here, when the rotating image display program is run, an execution mode is entered, setting up a state capable of displaying gears, but when run initially, no settings for the gears 7 and the hands 9 have been performed, so the gears 7 will not be displayed.

In this state, editing mode can be selected in a menu. When the editing mode is selected by an operation of a user, a menu for the editing mode depicted in FIG. 5(A) is displayed on the display screen of the mobile phone 1. Within the menu, the following pop-up menus are displayed: a Create (generation) Menu for generating (selecting) the gears 7 and the hands 9, an Edit (change properties) Menu for performing positional changes to the gears 7 already generated (the hands 9 move along with the set gears 7) and color changes to the gears 7 and the hands 9, that is, for changing attributes of the gears 7 and the hands 9, a View (display menu) Menu for selecting whether to display lines connecting the centers of the meshed gears 7 together as topology connecting the meshed gears 7 together, a File (file) Menu for performing storage, deletion, or the like of data files such as the generated gears 7, a Run (execution mode) Menu for transitioning to the execution mode, and a Cancel (cancel) Menu for cancelling the program. Note that selection of menu items is performed by operating the tenkey, and selection of menu items is performed using a set key, which are prescribed keys that are predetermined for the key inputting part 4.

[0068]

As depicted in FIG. 5(B), the Generation Menu is used to select whether to generate either the gears 7 (gears) or the hands 9 (hands). As depicted in FIG. 5(C), when the gears 7 are selected, selectable numbers of teeth for the gears 7 are displayed as a menu, and by selecting a number of teeth for the gears 7, gears 7 to be generated are set. Here, as described above, gears 7 in which the number of teeth are the multiples of 6: 6, 12, 18, 24, and 30, can be selected.

[0069]

Furthermore, when a type (number of teeth) of the gears 7 is selected, as depicted in FIG. 5(D), one image data is selected from the plurality of image data for animation of the gears 7 for which a number of teeth was selected and is displayed on the display screen. In this case, coordinate positions at the centers of the gears 7 are determined. For a gear 7 generated first, for example, the coordinate position at the center of the display screen is determined by default.

Note that as depicted in FIG. 5(E), when operations for continuing to generate a plurality of gears 7 is performed, for example, here, the user of the mobile phone 1 generates a first gear 7 in which the number of teeth is 6 then generates a gear 7 in which the number of teeth is 18, in which case, as depicted in FIG. 1(F), the subsequent gear 7 is generated in a meshed state with the previously generated gear 7.

In addition to meshing with another of the gears 7, as depicted in FIG. 5(G), generated gears 7 can be set so as to be fixed coaxially to one of the gears 7, and as depicted in FIG. 5(H), gears 7 can be set coaxially front and back and each meshed with separate gears 7. Moreover, as depicted in FIG. 5(I), three or more gears can be set coaxially.

[0070]

As depicted in FIG. 5(J), when selecting the hands 9 (hands), a size (length) of the selectable hands 9 can be displayed as a menu. Note that display of the hands 9 is performed corresponding to the gears 7, and in generating the hands 9, one or more of the gears 7 must be generated, and when a plurality of the gears 7 are generated, gears 7 to which the hands 9 are attached are selected. In other words, the hands 9 are combined with one of the gears 7, and the centers of rotation of the combined gears 7 are displayed such that the centers of rotation (base end) rotate at the same rotation rate as the gears 7.

Here, the sizes of the hands 9 are set to exactly a same type as a selectable type by number of teeth of the gears 7. For example, sizes of the hands 9 corresponding to the 5 types of the numbers of teeth "6, 12, 18, 24, and 30", are the 5 types SS, S, M, L, LL. Furthermore, a length of each of the hands 9 is set to correspond to lengths of radii of the pitch circles 8 of the gears 7 having corresponding numbers of teeth, and are set to correspond to lengths from a center of the gears 7 to tip ends of the teeth.

Here, the length of a hand 9 substantially equivalent to the radius of the gear 7 in which the number of teeth is 6 is designated SS, the length of a hand 9 substantially equivalent to the radius of the gear 7 in which the number of teeth is 12 is designated S, the length of a hand 9 substantially equivalent to the radius of the gear 7 in which the number of teeth is 18 is designated M, the length of a hand 9 substantially equivalent to the radius of the gear 7 in which the number of teeth is 24 is designated L, and the length of a hand 9 substantially equivalent to the radius of the gear 7 in which the number of teeth is 30 is designated LL. Note that there are no limits to the lengths of selected hands 9 or the numbers of teeth of the gears 7 combined with the selected hands 9. For example, the gear 7 in which the number of teeth is 6 may be combined with the hand 9 of size LL.

[0071]

Furthermore, attributes of generated gears 7 and hands 9 can be changed in a Change Attributes Menu.

Attributes of the gears 7 include position (x, y coordinates of center positions of displayed gears 7), color, frontward and rearward arrangement of the displayed gears 7, rotation rate, and the like.

Note that when the gears 7 are generated, a gear 7 generated first is positioned at the center of the screen, and those thereafter are positioned to mesh with a gear 7 previously generated, the colors are set by default, and the rotation rate is, for example, 0. In addition, when the gears 7 and the hands 9 are generated, gears 7 and hands 9 that are generated later in a generation order are more forward than gears 7 and hands 9 generated previously. Note that a frontward and rearward relationship between the gears 7 and the hands 9 is a display issue, and in the case that a plurality of gears 7 and hands 9 are displayed in at least one overlapping state, which of the gears 7 and the hands 9 to display as hidden is determined. Furthermore, the frontward and rearward relationship of the gears 7 is ignored when the gears 7 are combined together, and whether the gears 7 are meshed is determined based on planar distance.

[0072]

Moreover, a rotation rate of a driven gear meshed and driven by another gear 7 is determined based on a rotation rate of a driving gear forming a drive gear and on a ratio of numbers of teeth; however, before the rotation rate is set by a user, the rotation rate of the driven gear is 0.

Therefore, a gear 7 for which a rotation rate other than 0 is set forms a driving gear. Note that in a gear train composed of a plurality of meshed gears 7, only one gear 7 can serve as a driving gear. Different gear trains can be arranged within one screen, and if the gear trains are different, driving gears respective to the gear trains can be set. In addition, a free gear 7 that does not mesh with another gear 7 can also be a driving gear when having a rotation rate other than 0.

[0073]

Moreover, the hands 9 have attribute data such as color, type, and angle. The three types of hands 7 that correspond to types of clock hands, including: the hour hand 9a, the minute hand 9b, and the second hand 9c, and the widths (thickness) in terms of hand shape are: hour hand 9a > minute hand 9b > second hand 9c.

Although not an attribute of the gears 7 or the hands 9, the background color can also be changed in the Change Attributes Menu.

Furthermore, as depicted in FIG. 5(N), for example, in a state in which gears 7 have already been generated, when the Change Attributes (edit) Menu is selected, Change position (move), Change front/back arrangement (back/front), Change color (collar), Rotation rate (speed), Hand angle (angle), Gear/Hand removal (remove), and Background changes (background) are displayed as menu items.

In changing the position of a gear 7, a process of selecting the gear 7 whose position is to be changed is performed based on a tenkey operation by the user, then, when generated as second or subsequent, the gear 7 is in a meshed state (connected state) with other gears 7, so a process of disconnecting the selected gear 7 is performed. Furthermore, according to a tenkey operation by the user, the gear 7 is moved and a process of changing position is performed. In addition, in the process of moving the gear 7, distances between the center of the gear being moved and the centers of the other gears is measured, and when the distance is a sum of a radius of the pitch circle 8 of the gear 7 being moved and a radius of the pitch circle 8 of another gear 7, or is within a range of values obtained by adding or subtracting the value of a range of error to or from this sum, that is, when a predetermined distance is achieved, the gear 7 being moved and the other gear 7 can be connected, movement of the gear 7 at this state is stopped by a tenkey operation, and, for example, when a set key is operated, the gear 7 whose position was changed and the other gear 7 form a connected state. Note that under predetermined conditions, when two or more driving gears are arranged in one gear train or the like, a connected state will not form even if the distance between the center of a gear 7 being moved and the center of the other gear 7 is set to the predetermined distance.

[0074]

Moreover, one gear 7 can be meshed with a plurality of gears 7, and if a distance between the center of the gear 7 being moved and the centers of the plurality of gears 7 is a sum of the radii of the pitch circles 8 of the respective gears 7 or a value near thereto (within a range of error), positionally, the one gear 7 can be connected to the plurality of gears 7. Here, whether connection is possible in terms of connective topology is determined. That is, for example, in terms of connective topology, connections such as the following cannot be performed: connections in which two gear trains respectively having a driving gear are connected, connections in which two driving gears are arranged on one gear train, connections in which two or more gears 7 belonging to one gear train are connected together, and connections in which respective gears are connected to a driving gear either directly or through another gear of a same gear train. As such, connections in which one gear is subjected to rotational drive from two locations cannot be performed. In addition, a gear train cannot be connected in a ring form.

[0075]

When a moved gear 7 is connected to another gear 7, in the case that the moved gear 7 is not a driving gear and the other connected gear 7 is a driving gear or in the case that the moved gear 7 is included in a gear train to which the other connected gear 7 also belongs, the rotation rate of the moved gear 7 is determined based on rotation rate of the driving gear and gear ratio between the driving gear, the moved gear 7, and a gear 7 between the two. Moreover, as described below, when a gear 7 and a gear 7 are meshed, an angle of the gear 7 that is moved is set so as to form a mutually meshed state with the gear 7.

Note that when changing a position of a gear 7 combined with hands 9, the combined hands 9 also move along with the gear 7.

[0076]

As depicted in the flowchart in FIG. 6, in the change front/back arrangement process for the gears 7 and the hands 9, gears 7 and hands 9 are selected, and one is arranged forward (step S1) and one is arranged rearward based on operations of a user for forward and backward arrangement of the selected gears 7 and hands 9 (step S2). Note that when there are three or more gears, for example, after a gear 7 arranged furthest backward is selected, when an operation to bring forward is performed twice, the selected gear 7 is brought forwardmost. Sequential numbers Z_i indicating a frontward and rearward relationship are applied to each of the gears 7 and the hands 9, where Z_i is 0 for the rearmost gear and Z_i is highest for the forwardmost gear. Therefore, by changing the Z_i set for each of the gears 7 and the hands 9, the forward and backward arrangement can be changed. For example, by modifying a Z_i value for a gear 7 or a hand 9, one frontward or one rearward position change is performed. Furthermore, each changing operation can be repeated, but when a set key is operated to set an arrangement (step S3), the process ends.

[0077]

Note that drawing of the gears 7 and the hands 9 starts from a gear 7 and hands 9 having a lowest Z_i , and in the case of overlapping gears 7 or hands 9, a previously drawn gear 7 and hands 9 having a lowest Z_i are overwritten by a subsequently drawn gear 7 and hands 9 having a higher Z_i . As a result, the gear 7 and hands 9 having the higher Z_i are drawn hiding the gear 7 and hands 9 having the lower Z_i .

[0078]

In Change Color for the gears 7 and the hands 9, a color of a gear 7 can be changed and a color and a type of a hand 9 can be changed. A gear 7 or a hand 9 is selected by a user operation.

Furthermore, as far as menu items, as also depicted in FIG. 5(N), when Change Color (collar) is selected, a Change Color window opens as depicted in FIG. 5(O), and within the Change Color window, for the gears 7, Gear 7 Border (outline region), Gear 7 Border Fill (body), and Gear 7 Background can be selected. In addition, a Transparent (transparent) selection is displayed for Gear 7 Border Fill, allowing for the selection of transparency. Then, as depicted in the flowchart in FIG. 7, in response to selection operations by the user, editing of the color inside of the border (step S4), editing of the border color (step S5), editing of the color outside of the border (step S6), and a process of making the inside of the border transparent (step S7) can be performed. Furthermore, in each editing process apart from the set transparent process, when the user performs an operation of changing an RGB value, RGB values corresponding thereto are also changed (steps S8, 9, and 10). Each change operation can be repeated, but when a set key is operated, a color of a gear 7 is set and the process ends (step S11). Note that the process of making the inside of the border of a gear 7 transparent makes it so that an inside of the border is not drawn when the gear 7 is drawn.

[0079]

As described above, a color at an index 0 of the palette part of an index image is assigned to an outline region (border) of the gear 7, a color at an index 1 of the palette part is assigned to the part of the gear 7 inside the outline region, and a color at an index 2 of the palette part is assigned to the part of the gear 7 outside the outline region.

Furthermore, in the color changing process, an index image 62 corresponding to a gear 7 targeted for a color change and stored in the storing region 6 is written so as to be copied to a region separate from the storing region 6, and in a binary data state without decompression of this index image, RGB values for the index 0 for the outline, the index 1 for the part inside the outline region, and the index 2 for the part outside the outline are directly overwritten in the palette (color table). As depicted in FIG. 5(N), overwriting RGB values are input by changing RGB values in the Change Color window.

[0080]

Compared to cases where a plurality of image data must be stored in advance, changing a gear color in this manner enables for an extremely limited image data storage capacity, the use of a lot of colors, and an extremely limited program storage capacity without requiring an application such as for decompressing an index image and changing an image color. Note that, normally, index images are compressed, but the compression is that of a portion of the image data, that is, the pixel data part 622, and the palette part 621 is not compressed, being easily changeable even index images in a non-decompressed binary state.

Note that first, a same number of the index images 62 are stored as animation image data as the respective gears 7 described above for each type by number of teeth of the gears 7; however, a same number of image data having a different color due to a color change are overwritten as animation image data as the respective gears. Furthermore, although writing the new image data to the storing region 6 increases the storage capacity, the gear 7 having a different color can be displayed while simultaneously displaying a plurality of gears 7 having the same number of teeth.

By changing the color of the gear 7, the color of the gear 7 is changed as depicted in FIG. 5(P), and as described above, the plurality of gears 7 having the same number of teeth can also be displayed.

From the above, the calculation processing part 3 for performing the color changing process overwrites color data on the color table of the index image 62 stored in the storing means (storing region 6) as a function executed by the rotating image display program, which serves as the color changing means for changing the color of the image of a rotating body (gear 7).

[0081]

Moreover, when the hands 9 are selected, the user can select as items to change, the color of the hands 9 and the shape of the hands 9, as depicted in the flowchart in FIG. 8.

When a color is selected and changed for the hands 9, the RGB values change as in the case of the gears 7 (step S12); however, the hands 9 are not stored as image data in the storing region 6, and when an animation of the hands 9 is displayed, the angle changes corresponding to a rotation rate of combined gears 7 and is drawn again every frame (not limited to when the angle changes for each drawing). Here, using the changed RGB values, the color of the hands 9 is changed, and different colors of the hands 9 can be displayed. In addition, during drawing, different line widths and lengths are drawn according to a shape type of the hands 9. Accordingly, in a color change of the hands 9, no color update is performed in the image data of the storing region 6.

[0082]

Moreover, by selecting a shape change for a hand 9 and determining one hand from a plurality of hand types, that is, an hour hand, a minute hand, and a second hand, a user can select a thick line, a standard line, or a thin line for the hand 9 (step S13). Note that the thick line corresponds to the hour hand, the standard line corresponds to the minute hand, and the thin line corresponds to the second hand. By selecting a shape for the hand 9, a length size for the hand 9 described above can also be selected. Furthermore, each change operation can be repeated, but when the user operates the set key, the color and shape of the hand 9 are fixed and the process ends (step S14).

[0083]

As depicted in the flowchart in FIG. 9, in changing a rotation rate of gears, first, either clockwise or counterclockwise is selected based on an operation of a user for the rotation direction of the selected gears (step S15). Next, a rotation rate is input as a fraction as described above, and based on a user operation, a denominator for the rate is changed to an input value (step S16) and a numerator for the rate is changed to an input value (step S17).

A common multiple of the number of virtual points $K \text{ res}_m$ for the selected gears 7 and the denominator is obtained as described above, and the denominator is set to the common multiple (step S18). In addition, the numerator is multiplied by the value multiplied when setting the denominator to the common multiple to form a common denominator state with a fraction formed by setting res_m to the denominator (step S19).

Furthermore, the change operation can be repeated, but by operating the set key, the rotation rate is fixed and the process ends (step S20).

[0084]

Note that essentially, the numerator increases for every value set for each time corresponding to a drawing of one frame (value of a numerator of a common denominator obtained as above). Furthermore, when an increased value of the numerator exceeds the value of the denominator, the gear has rotated one round, and by dividing the set value of the numerator by the value of the denominator, the remainder forms a present numerator. That is, the time required for a gear 7 to make one round is obtained by multiplying a time corresponding to a drawing of one frame and a value obtained by dividing a denominator by a numerator. Note that the time corresponding to one frame is not the actual time between when a previous frame is drawn and a subsequent frame is drawn, but rather a time fixed as the drawing time for one frame, which is set in advance, is used.

Furthermore, one or higher positive integers can be used as denominator values for rotation rates, and zero or higher natural numbers can be used as numerator values. Here, when the numerator value is set to 0, the rotation rate is 0, serving as a driven gear such as described above. On the other hand, when the numerator value is set to 0 or higher, a driving gear such as described above is formed, and in one gear train in which gears are sequentially connected, just one gear can serve as a driving gear as described above, and for the driven gears on the same gear train that are not the driving gear, the numerators of the fractions expressing rotation rates are all set to 0. Note that the driven gears also rotate corresponding to the driving gear, and a rotation rate emerges, so although the set values of the numerators are 0, a rotation rate of the driven gears connected to the gear train having a driving gear is obtained as described below, at which time a numerator and denominator of the rotation rate are determined.

[0085]

Angles of the hands 9 are set in degrees, for example, with one round being 360° . Therefore, as depicted in the flowchart in FIG. 10, for example, the angles of the hands 9 are changed based on an operation by the user (step S21).

When the hands 9 actually rotate, by rotating similarly to the gears 7, which have radii corresponding to length sizes thereof, the hands 9 are displayed so as to rotate each of an angle of division, which is obtained by dividing 360° into the total number of virtual points $K \text{ res}_m$, being a value obtained by multiplying a number of image data by a number of teeth of corresponding gears. Therefore, the angles of the hands 9 are converted to fractions such as $1/\text{res}_m$. For example, in the case of the LL sized hands 9 equivalent to the gear 7 in which the number of teeth is 30, rotation occurs every 2.4° , being the angle of division described above for the gear 7 in which the number of teeth is 30. That is, rotation occurs in 1/150th increments. Furthermore, the rotation rates for the hands 9 are expressed as fractions, similarly to the gears 7. Therefore, angles of the hands 9 are changed as fractions, numerators are changed as natural numbers (including 0), and denominators are changed as positive integers (step S22).

[0086]

Note that a hand 9 rotates integrally with a gear 7 to which it is attached, and similarly rotates with a gear 7 fixed onto the same axis as the rotating gear 7. Here, when a number of teeth of a gear 7 having a radius rad_m corresponding to a length of a hand 9 and a number of teeth of a gear 7 rotating integrally with a hand 9 are different, because the total numbers of virtual points $K \text{ res}_m$ of these gears 7 are different, the resolution for animation (res_m or aforementioned angles of division) for each is different. For example, when a hand 9 corresponding to a radius rad of a gear 7 in which the number of teeth is 30 is attached to a gear 7 in which the number of teeth is 6, while a gear in which the number of teeth is 6 moves one image data amount, the hand 9 moves 5 image data amounts. However ultimately, the number of frames (time) required for one round of the gear 7 and the hand 9, which rotates in a state of attachment thereto, is the same.

[0087]

In addition, in changing the angle of the hands 9, a current time can be obtained based on the clock function of the mobile phone 1, and menu items are prepared for setting the hour hand 9a, the minute hand 9b, and the second hand 9c respectively to angles based on the current time.

That is, based on an operation by the user, the current time can be read from the clock function of the mobile phone 1, and corresponding to the hour, minute, and second that was read, a process is performed in which the hour hand 9a, the minute hand 9b, and the second hand 9c are respectively set to an angle for the current hour, an angle for the current minute, and an angle for the current second (steps S23, 24, and 25). Furthermore, each change operation can be repeated, but by operating the set key, the angles of the hands 9 are fixed and the process ends (step S26).

[0088]

Note that when a time display is enabled using the hand 9a, the minute hand 9b, and the second hand 9c, in editing mode, editing is performed with rotation of the gears 7 in a stopped state (a state upon opening the editing mode in which rotation of the hands 9 is stopped), the execution mode is returned to, rotation of the hands 9 recommences, and when drawing the hands 9, the hands 9 advance to the current time and are redrawn. However, upon opening the editing mode, the clock can actually be seen, and it may be a long time until the angles of the hands 9 change, there may be a big difference between a time when the editing mode is opened and a time on the clock when the hands 9 are adjusted, and even if the angles of the hands 9 when the editing mode is opened while the execution mode is opened advance to angles of the hands 9 based on the current time, angles of the hands 9 when changed will not match the time when the editing mode was opened, and the time cannot be set accurately. Therefore, by making it possible to adjust the angles for the hour hand 9a, the minute hand 9b, and the second hand 9c in the short time described above, an accurate time can be set for the angles of the hands 9.

[0089]

As depicted in the flowchart in FIG. 11, in determining the background color, a window for changing the color of the background is displayed, and a process is performed to change to an RGB value based on a user operation (step S27). Note that prior to a change, a default background color is set. The background color is also drawn first in each frame.

When the Display Menu is selected, a selection menu is displayed for display/non-display of a connective topology of line segments mutually connecting the centers of gears 7 meshed together, and based on a user operation, the display/non-display is selected, and when displayed, a process is performed for displaying the connective topology when in execution mode. Note that even if adjacent gears 7 are positioned meshed together when the gears 7 are displayed, these may be non-meshed, and in a state where the connective topology is not displayed, there may be cases where a meshed state of gears 7 cannot be determined immediately.

[0090]

When the File Menu is selected, Create new file, Open local file, Overwrite and save file, Delete file, Save as file, Upload file, and Download file are displayed as menu items. Note that here, the files are attribute data for a generated gear or hand, data for connecting a gear and a gear and for connecting a gear and a hand, and data such as background color data, and when the color of a gear 7 is changed, if the changed RGB value or the like can be stored separately from the image data, including with the image data of the gear 7 is not absolutely necessary. Furthermore, this type of file can be stored as gear/hand position/combination data 64 in the storing region 6. Note that files are each given an ID (file name) for identifying the file. Note that in a file for a gear 7 and hand 9 set, the gear 7 is indispensable but the hand 9 may be removed.

[0091]

In Create new file, a currently opened file in editing mode, that is, a generated gear 7 and hand 9 set is discarded or overwritten and saved in the storing region 6, an ID for a set of a new gear 7 and hand 9 set (file) is set, and the new file allows for the gear 7 and the hand 9 to be generated.

In Open local file, a local file ID stored in the storing region 6 is selected by a user, the data for a gear 7 and hand 9 set written in a file corresponding to the selected ID is read into the execution memory 32. The selected file ID is also set as the file ID being currently executed. Subsequently moving to execution mode, an animation of the gear 7 and the hand 9 are displayed based on the file data of the set ID. Also, the local here indicates the mobile phone 1 or the storing region 6 thereof. Note that global, in contrast to local, refers to internet storage such as a server on a network.

[0092]

In Overwrite and save file, a file is stored to the storing region 6 using a currently executable file ID. Here, in the case of a file stored using an ID already in use, the previously stored file having the same ID is deleted (updated), and the current executable file is stored.

In File save as, a new ID is set by a user operation, a file is saved under the new ID in the storing region 6, and the now executable file ID is updated to this ID.

For Delete file, a local file ID stored in the storing region 6, which is local, is selected by a user, and a local file having the selected ID is deleted from the storing region 6.

[0093]

In Download file, a file stored in an internet storage is downloaded. The mobile phone 1 is connected to the internet, an internet storage server is accessed, and a network ID, being an ID on a file network, is designated (selected) by a user. A file for a gear 7 and hand 9 set written into the designated file is read into the mobile phone 1. Furthermore, a new ID is set for the current file read in over the internet, downloaded, and the file set using the ID (local ID) is stored in the storing region 6 under the set ID.

In Upload file, under the same circumstances as with downloads, a network ID is set for an executable file, and the file is transmitted using the set ID and stored in a network storage.

[0094]

Note that due to having limited storage capacity, the mobile phone 1 cannot store multiple data files, and cannot store multiple application files simultaneously. Therefore, there may be cases where the rotating body image display program is deleted from the storing region 1 to allow for the execution of another application. In such a case, generated gear 7 and hand 9 files are saved so as not to be deleted, which required the use of internet storage. In addition, if a file created as a sample or the like is stored to a dedicated internet storage in advance, downloading this allows, for example, for a display of gears 7 / hands 9 to be performed without an operation of generating the gears 7 or the like being performed.

[0095]

Next, the process of the rotating body image display program in execution mode is described.

As depicted in FIG. 12 in the main flow of the execution mode, a main process consists of updating a gear 7 / hand 9 set (the hands 9 may be removed) (step S30) and drawing the updated gear 7 / hand 9 set (step S31).

In updating a gear 7 / hand 9 set, a rotation angle is obtained based on an elapsed time of generated gears 7 and hands 9 combined with the generated gears 7, and a phase of the gears 7 to which the plurality of image data for animation described above corresponds is obtained. That is, a phase corresponding to image data for a rotation angle closest to the obtained rotation angle is obtained.

Furthermore, in drawing the gear 7 / hand 9 set, image data for the gears 7 corresponding to the phase are selected and drawn, and hands 9 are drawn at the rotation rate corresponding to a rotation angle of the gears 7.

Furthermore, at a prescribed timing, for example, a number of frames per unit of time of 10/sec, by repeatedly performing the process described above approximately every 100 msec, a drawing of the gears 7 and the hands 9 is performed.

More specifically, as illustrated in the flowchart in FIG. 13, in updating a gear 7 / hand 9, first, an update starting time is acquired based on the clock function of the mobile phone 1, and is stored as t_{now} (msec units) (step S32).

[0096]

Next, a gear g_i , and a hand h_i are selected in ascending order of Z_i , which expresses the frontward and rearward relationship described above, from generated gears g_i , which are elements of a collection G of gear 7 / hand 9 sets, and the following process is executed (step S33). Note that the collection G includes generated gears (7) g_i , (g_0 , g_1 , ...), and hands (9) h_i (h_0 , h_1 , ...) when there is a hand 9 set in the gears g_i , however, hours_i are set to gears_i having a radius of a length thereof, and the following process is executed. Note that in the execution mode, $i = 0, 1, 2, \dots$ expresses each of the generated gears 7 and hands 9, and hereinafter, corresponds to the generated gears 7 and hands 9 for i , s_i , Z_i , b_i , p_i , r_{s_i} , and the like as well.

[0097]

Note that when the following process ends for the gears g_i and the hands h_i , being every element in the collection G, the update process ends (step S34).

Furthermore, angles a_i of each of the gears g_i (including gears 7 of radii corresponding to lengths the hands h_i) at the current time t_{now} described above are obtained based on the formula given in step S35. Here, as described above, a number of image data for an animation of one period of the gears g_i and a number of teeth c_i of the gears 7 to rotate a prescribed angle (one period) as described above, that is, values obtained by multiplying the aforementioned phase numbers and phase values are set to the resolutions r_{s_i} described above (res_m), which expresses a total number of virtual points K set at equal intervals on pitch circles 8 of the gears g_i .

In addition, phases p corresponding to each image data, in which only phase value numbers are stored are set to p_i (0, 1, 2, ..., phase-1).

[0098]

Moreover, a rotation rate is obtained by inputting a numerator and denominator as described above.

That is, in the case that the input denominator of a rotation rate is 600 and the numerator is 1, 1/600th of a rotation occurs every single drawing update time t_f .

In the case that the number of teeth is 6 and the phase is 5, the resolution r_{s_i} is 30. Therefore, the rotation angle for each phase (image data) is 1/30. Here, once a common denominator between the fraction expressing the rotation angle for each phase and the fraction of the input rotation rate is found, the rotation angle for each phase is expressed as 20/600. The denominator when a common denominator is found is set to a denominator b_i . Note that the denominator b_i is multiplied by a positive integer f_i (for example, 20) used in the common denominator with the r_{s_i} (for example 30).

[0099]

In addition, when the numerator over the denominator b_i obtained as a common denominator with the fraction of the input rotation rate is set to a rate S_i as an amount of increase for the fraction (the numerator is b_i) of the rotation angle per single drawing update time, the rate S_i is set to 1.

Furthermore, according to the formula in step S35, an elapsed time is obtained by dividing a reference time t_i^{st} from the update starting time t_{now} , and a number of drawings (displayed number of frames) since the reference time t_i^{st} is obtained by dividing the elapsed time by the drawing update time t_f . Note that the drawing update time t_f is not the actual time taken for a drawing to update, but rather a set time.

[0100]

Furthermore, an amount of change in angle (numerator) from the reference time is obtained by multiplying the number of drawings by the rate S_i as the amount of increase for the numerator of the angle described above for every single drawing, the angle a_i^{st} of the reference time t_i^{st} is added thereto, and this is divided by the denominator b_i to obtain a remainder, whereby the angle a_i according to one rotation of a gear is obtained as a numerator of the fraction expressing the rotation angle (step S35).

For example, the reference time is set to 100 msec, and a numerator a_i^{st} of the reference angle is set to 100, and when the reference time is set to 40,000 msec, the angle a_i becomes 499. That is, the rotation angle when one round is set to 1 is 499/600. Note that here, $[x]$ expresses a Gaussian function for reversing large integers not exceeding x , and a mod b expresses the remainder when a is divided by b . Note that in the case of $a < 0$, $b - (\text{absolute value of } a) \text{ mod } b$ is set.

Here, in the second function of step S35, dividing the obtained angle a_i (499) by the f_i (20) used in the common denominator causes the rotation angle described above correspond to a 24th virtual point K. Furthermore, by dividing 24, which indicates the 24th virtual point K, by a phase (5) and obtaining a remainder, the phase $p_i (= 4)$ can be obtained. Image data (image₃) corresponding to the phase $p_i (= 4)$ can be selected from among the plurality of image data for animation. That is, $[a_i/f_i]$ corresponds to sequential numbers applied to the virtual points K for identifying the virtual points K, and by obtaining $[a_i / f_i]$ for each time a frame is drawn, image data corresponding to the virtual points K can be selected and a gear at a rotation angle corresponding to a time when drawn can be displayed.

[0101]

As illustrated in the flowchart in FIG. 14, each gear g_i , hand h_i set continues to be drawn. Here, as described above, the gears g_i and hands h_i are selected from the collection G in ascending order of the Z_i expressing the frontward and rearward relationships and the following process is executed (step S37). Note that when the following process ends for the gears g_i and the hands h_i , being every element in the collection G, the update process ends (step S38).

Next, it is determined whether the gears g_i or the hands h_i are selected (step S39). In the case of the gears g_i , by image data corresponding to the phases p_i obtained by update processing within the image data for the gears g_i having the number of teeth c_i , image data in which colors collar _{g_i} are changed as attributes of the gears as described above or image data for the default color are drawn.

[0102]

That is, the image data image^{collarg _{i}} _{c_i, p_i} is drawn (step S40). In addition, in the case that the hands h_i are selected, the hands h_i are drawn according to a length size of the hands h_i , colors collar _{h_i} set for the hands h_i , and a set shape (either the hour hand 9a, the minute hand 9b, or the second hand 9c). Furthermore, for the angles of the hands h_i when drawing, the angles a_i/b_i described above are also obtained for the hands h_i similar to the gears g_i at radii corresponding to length sizes of the hands h_i , and the hands h_i are drawn at the angles a_i/b_i from coordinate positions at the centers of the gears g_i to which the hands h_i are connected. Note that here, the hands h_i are drawn from the central coordinates of the gears g_i to which the hands h_i are connected, to the coordinate positions of the virtual points K expressed by X_{c_i} , $[a_i/f_i]$ (step S41). Note that $[a_i/f_i]$ corresponds to the sequential numbers applied to the virtual points K, by which the virtual points K can be identified and the circular coordinate data 63 for virtual points K identified by the storing region 6 can be obtained.

[0103]

By means of processing by the calculation processing part 3 implemented by the rotating body image display program in an execution mode such as described above, the calculation processing part 3 displays an animation of rotating bodies by sequentially displaying a plurality of image data for each of the plurality of rotating bodies (gears 7) stored in the storing means (storing region 6), and functions as display controlling means in which timings are substantially synchronized for switching the image data for the rotating bodies together, which are displayed in a synchronously rotating state, when the image data stored in the storing means is sequentially switched and displayed according to an arbitrary rotation rate of the rotating bodies.

Moreover, when the animation of the rotating bodies is displayed by sequentially displaying the image data, the calculation processing part 3 obtains the virtual points corresponding to rotation angles according to a rotation rate of the rotating bodies when the image data is displayed, and functions as the display controlling means for displaying image data corresponding to the virtual points obtained from among the plurality of image data.

[0104]

The above processing can display a rotating display at a rotation rate set for gears 7 meshed together, a plurality of gears 7 integrally rotating on a same axis, gears 7 and hands 9 integrally rotating, or the like, but cannot display rotation of respective gears 7 or hands 9 in a state in which teeth are accurately meshed. Therefore, in the editing mode, a plurality of processes must be performed after the gears 7 and the hands 9 are generated and the attributes are changed to coordinate phases of meshed gears together.

[0105]

For example, the number of teeth c_i , being variables corresponding to each of the gears 7, the radii r_i , the resolutions rs_i , the positive integers f_i used when forming a common denominator described above, the denominator b_i of the fraction expressing the rotation rate at the common denominator, the sequential numbers z_i expressing the frontward and rearward relationships, and the reference time t_i^{st} are determined as illustrated in the flowchart in FIG. 15 when a gear is generated. That is, in the step S43, the number of teeth c_i is set to a number of teeth cog_m for a gear 7 selected by a user, the radius r_i is set to the radius r_m , which is predetermined corresponding to the selected number of teeth cog_m , and the resolutions (total number of virtual points set on the pitch circles 8) rs_i are also set to the resolutions res_m , which are predetermined corresponding to the number of teeth cog_m .

[0106]

Note that the positive integers f_i used when forming a common denominator are set to 1 until the rotation rate is changed or connection to a gear in a gear train having a driving gear is achieved. Also, the denominators b_i are set to $rs_i \times f_i$ based on the positive integers f_i set to 1 by default.

In addition, Z_i , which express frontward and rearward arrangements, are numbers of elements (a sum of the number of generated gears g_i and the number of hands h_i) in the collection G of the gears g_i when generated. That is, the values are determined in sequential order starting from 0 in the order of Z_i generation.

Moreover, the reference time t_i^{st} is set to a time t_{ed} of when the editing mode is opened (step S43).

The x, y coordinates at the centers of the gears g_i are also determined, and the colors $collar_{g_i}$ for each region of the gears g_i are likewise also set to the default color. Note that the coordinate positions at the centers of the gears g_i may be preset coordinate positions, may be designated by a user, and in the case that separate gears g_j have already been generated, positions meshing with the separate gears g_j , that is, a distance between centers of the separate gears g_j and centers of generated gears g_i may be at a position substantially equivalent to a sum of the respective radii of the pitch circles 8.

Note that whether attributes of the above gears g_i are predetermined values for each gear or are automatically applied during generation is determined by user selection of a number of teeth during gear generation.

Furthermore, gears g_i having each attribute determined are set to elements of the collection G of generated gears g_i and hands h_i (step S44), and data for the attributes of the generated gears g_i are stored as the gear/hand position/combination data 64 of the storing region 6.

[0107]

In addition, when the hands h_i are generated, as illustrated in the flowchart in FIG. 16, the hands h_i having the lengths of the selected sizes of radii rad_m are generated. Here, the hands h_i rotate similarly to the gears 7 having the radii rad_m (gears 7 having pitch circles of radii rad_m), which are substantially equivalent in size, and similar to the gears 7, the number of teeth c_i are set to the numbers of teeth cog_m of the gears having the radii rad_m , the radii r_i are set to r_m , and the resolutions rs_i are set to res_m .

Moreover, as described above, the frontward and rearward arrangements z_i are numbers of elements in the collection G of gears 7 and hands 9 when generated. Also, the reference time t_i^{st} is set to a time t_{ed} of when the editing mode is opened.

[0108]

Here, rates (numerators of the rotation rates) s_i of the hands h_i are obtained by multiplying a rate s_j of gears g_j connected thereto and the number of teeth c_i of the gears corresponding to the lengths of the hands h_i , and common denominator values f_i are obtained by multiplying common denominator values f_j of the connection destination gears g_j and a number of teeth c_j of the connection destination gears g_j .

Furthermore, when a greatest common divisor of a rate s_i set as described above for the generated hands h_i and the common denominator values f_i is set to n (when there are common divisors), s_i and f_i are divided by the greatest common divisor. That is, the denominators f_i of the fractions of the rotation rates obtained by multiplying by the resolutions rs_i , and the numerators s_i of the fractions of the rotation rates are mutually reduced.

Furthermore, the denominators b_i of the rotations rates of the gears g_i corresponding to the radii of the hands h_i are obtained by multiplying the values f_i reduced as described above and the resolutions rs_i .

Here, as described above, the rates s_i/b_i of the hands h_i are set to $s_i = s_j \times c_i$, $f_i = f_j \times c_j$, $b_i = f_i \times rs_i$; therefore, $s_i/b_i = (s_j \times c_i)/(f_i \times rs_i) = (s_j \times c_i)/(f_j \times c_j \times rs_i)$.

In addition, from the definition of the resolution rs described above and the definition of the rate denominator b :

$$rs_i = c_i \times \text{phase}$$

$$rs_j = c_j \times \text{phase}$$

$$b_j = f_j \times rs_j$$

exist as relationships.

Therefore,

$$s_i/b_i$$

$$= (s_j \times c_i)/(f_j \times c_j \times rs_i)$$

$$= (s_j \times c_i)/(f_j \times c_j \times c_i \times \text{phase})$$

$$= s_j/(f_j \times c_j \times \text{phase})$$

$$= s_j/(f_j \times rs_j)$$

$$= s_j/b_j$$

are true.

Here, s_j/b_j is the rotation rates of the gears g_j to which the hands h_i are connected, and the rotation rates s_i/b_i of the hands h_i and the rotation rates s_j of the connection destination gears g_j are equivalent.

Note that the processing described above equalizes the rotation rates of the hands h_i and the connection destination gears g_j , however, when a number of teeth differs between the gears g_i of the radii corresponding to the lengths of the hands h_i and the connection destination gears g_j , the timing for switching the image data may be different.

[0109]

Furthermore, gears P_{gi} to which the hands h_i are connected are set to the gears g_j , and the hands h_i are included in collections CG_j of gears connected to the connection destination gears g_j (step S46). Note that within the hands h_i and the gears g_i , there is one connection destination, that is, one gear g_j as a source for transmission driving force. In addition, the gears g_i and hands h_i to which the driving force from the driving force transmission source gear g_j is transmitted are not limited to one; a plurality of the gears g_i and the like (gears g_i and hands h_i belonging to the collection CG_j) can be connected.

Furthermore, the generated hands h_i for which attributes have been determined are added to the collection G of generated gears g_i and hands h_i (step S47).

[0110]

When the gears g_i or the hands h_i are generated as described above in the editing mode, the reference time t^{st}_i of the generated gears g_i or the generated hands h_i becomes the starting time t_{ed} when opened in the editing mode, and in the case that the reference angle a^{st}_i corresponding to the reference time t^{st}_i is set, when gears g_i or hands h_i already generated previous to entering the editing mode have different reference times t^{st}_i , and the angles, phases, and the like are aligned to connect the gears g_i or hands h_i to these existing gears g_i , because the reference times for the generated gears g_i or hands h_i and the existing gears g_i differ, setting the reference angles a^{st}_i of the gears g_i generated corresponding to the reference angles a^{st}_i of the existing gears g_i cannot be set, which makes the calculation complicated.

[0111]

Therefore, at the time the editing mode is opened, the reference times t_i^{st} of the previously generated gears g_i and hands h_i are set to an editing mode starting time t_{ed} , and by setting the reference angles a_i^{st} to the angles a_i at the starting time t_{ed} , the existing gears g_i and the reference times t_i^{st} and reference angles a_i^{st} of the gears g_i and hands h_i generated in the editing mode are aligned.

As depicted in the flowchart of FIG. 17, in the process of opening the editing mode, a time when the editing mode is opened in the clock function of the mobile terminal is saved in the storing region as t_{ed} (step S49).

All of the gears g_i and hands h_i forming the elements included in the collection G of gear g_i and hand h_i sets for which the following process are generated (step S50). Note that when the following process is performed for all of the gears g_i and the hands h_i included in the collection G, the process ends (step S51).

The angle a_i at the editing mode starting time t_{ed} is obtained. First, an elapsed time is obtained by dividing a previous reference time t_i^{st} from the editing mode starting time t_{ed} , a number of drawing updates up to the present is obtained by dividing the elapsed time by the drawing update time t_f , and an amount of change of a rotation angle (numerator of the rotation rate) from the reference time t_i^{st} to the starting time t_{ed} is obtained by multiplying the number of drawing updates and the rates s_i (numerator of the rate), which express rotation rates per drawing update. The amount of change here is a maximum integer not exceeding the amount of change obtained according to the Gaussian function described above. Furthermore, after the angles a_i^{st} serving as references for the amount of change of the rotation angle are added, a remainder is obtained by dividing the denominator b_i of the rotation angle and the rotation rates, by which the angles a_i are obtained as angles within one rotation of the gears h_i .

The obtained angles a_i are set as the reference angles a_i^{st} .

In addition, the phase p_i at the reference time t_i^{st} obtained by the rotation angles a_i can be obtained in a manner similar to the case above.

[0112]

Furthermore, an elapsed time is obtained by dividing the original reference time t_i^{st} from the editing mode starting time t_{ed} , and a remainder is obtained by dividing this by the drawing update time t_f . The obtained value expresses a deviation from a drawing starting time per drawing start time (frame 0) closest to the starting time t_{ed} prior to the editing mode starting time t_{ed} . By dividing the deviation time obtained from the starting time t_{ed} , a drawing starting time closest to the starting time t_{ed} prior to the starting time t_{ed} is obtained. The obtained time is set as a new reference time t_i^{st} (step S52). This is used to align the reference angles a_i^{st} with the timing of the drawing start time.

Moreover, in the attribute changing process described above, when the gears g_i are moved, whether the moved gears g_i can connect with the other gears g_j is determined, and when connection with the other gears becomes possible, it is displayed that connection is possible, nevertheless, the determination process for gear connection capability is illustrated in the flowchart in FIG. 18.

Note that this determination process is performed when the process for moving a gear 7 into the editing mode is performed.

[0113]

That is, the gear 7 is moved by an operation of the user (step S54). Next, it is determined whether positionally, as described below, the gear 7 can connect to the gear 7 at the position moved to, and the determination result is displayed (may be displayed only when a connection is available) (step S55).

The process described above is performed every time while the position of the gear 7 is being moved. Furthermore, when the gear 7 arrives at a target position (for example, a position meshing with another gear), movement of the gear 7 is halted, and thereby a position of the gear 7 is determined (step S56).

Next, a number of gears 7 that can connect to the gear 7 whose position is moved is determined (step S57). The process ends when there are 0 connectable gears 7, but when there are 1 or more, a user is given a selection of the 1 or more gears 7 to connect the moved gear 7 to, or a selection to not connect the moved gear 7 to the other gears 7 (step S58).

[0114]

A user selection result is determined (step S59), and in the case that the selection to not connect is made, the process ends; in the case that one gear is selected, the selected gear 7 and the moved gear 7 are connected, and according to the angle and rotation rate of the selected gear 7, the angle and rotation rate of the moved gear are automatically set as described below (step S60).

Next, a number of gears 7 that can connect to the gear 7 whose position has moved again is determined (step S61).

The process ends when there are 0 connectable gears 7, but when there are 1 or more, a determination of topologically connectable gears, described below, and determination results are displayed (step S62).

Next, a number of the positionally and topologically connectable gears 7 is determined (step S63).

[0115]

The process ends when there are 0 connectable gears 7, but when there are 1 or more, a user is given a selection of the 1 or more gears 7 to connect the moved gear 7 to, or a selection to not connect the moved gear 7 to the other gears 7 (step S64). A user selection result is determined (step S65), and in the case that the selection to not connect is made, the process ends; in the case that one gear is selected, the selected gear 7 and the moved gear 7 are connected, and the angle and rotation rate of the moved gear are recursively and automatically set according to connective topology as described below (step S60). Note that when one gear 7 can be connected to two or more gears 7, it means that at least one of the connectable gears is not connected to a gear train having a driving gear. In addition, in the case that the one gear 7 can be connected to the two or more gears 7, when an angle and rotation rate of the one gear 7 is aligned with the one connectable gear 7, the need arises of aligning with the one gear 7, which is moved by the angle and rotation rate of other connectable gears 7.

[0116]

Accordingly, it must be determined whether the two or more connectable gears 7 are free gears, gears 7 in a gear train having a driving gear, or the like. When one gear 7 is connected to both a gear 7 in a gear train not having a driving gear and a driving gear or a gear 7 in a gear train having a driving gear, an angle and rotation rate must be recursively set along the gear train in order for each gear 7 in the gear train not having a driving gear via the gear 7 moved from the connectable gear 7 side of the gear 7 in the gear train not having a driving gear.

Furthermore, until there are no more gears 7 that can connect to the moved gear 7, the processes for steps S61 to S66 are repeatedly performed.

[0117]

Next, the determination process as to whether a gear is positionally connectable to the moved gear in the process described above is described with reference to the flowchart in FIG. 19.

Here, the moved gear (gear being edited) is set to g_i , and gear determined capable of connecting to the gear g_i is set to g_j . The pitch circle radius of the gear g_i is set to r_i and the pitch circle radius of the gear g_j is set to r_j . In addition, coordinates of the rotation center of the gear g_i (center of the pitch circle 8) are set to (x_i, y_i) and the coordinates of the rotation center (center of the pitch circle 8) of the gear g_j are set to (x_j, y_j) .

Furthermore, in the case that the gears 7 are meshed together and connected, the pitch circles 8 of the meshed gears 7 must be in mutual contact. Therefore, in this case, the distances between the centers of the gears 7 meshing together are the sums of the respective radii of the pitch circles 8 of two gears 7.

[0118]

Here, in consideration of operability and the like of the key inputting part 4, even if distances between the center of the gear g_i and the center of the gear g_j do not perfectly match a sum of the radius r_i and the radius r_j , connection is possible within the deviation tolerance range: $-d, +d$.

Therefore, as illustrated in step S67 of the flowchart in FIG. 19, a distance between the center of the gear g_i and the center of the gear g_j is equal to or greater than a distance obtained by dividing the tolerance range d from the sum of radius r_i and a radius r_j of the two gears, and in the case that the tolerance range d is added to the sum of the radius r_i and the radius r_j of the two gears, it is determined that the gear g_i and the gear g_j can be connected.

Moreover, in connecting the gear g_i and the gears g_j , in addition to meshing, there are also gears 7 fixed coaxially, and whether or not a coaxial gear 7 is formed is determined by if center positions of the gear g_i and a gear g_j match, that is, when the x, y coordinates of the center of the gear g_i and the x, y coordinates of the center of the gear g_j match, it is determined that a connection can be made (step S67).

[0119]

Note that there may be a plurality of gears that can connect to the gear_i, but in this case, when both of a plurality of connectable gears g_j are either a drive gear or a gear in a gear train that includes a drive gear, it is impossible to connect both at the same times in terms of connective topology. Connection is also impossible with gears that are connected in a closed ring form.

Furthermore, when determination of the connectable gears g_j ends and there are connectable gears g_j , the connectable gears g_j are all displayed as connectable (step S68). Note that the display at this time may display, for example, as illustrated in FIG. 5(F), a frame enclosing the gears 7 being edited and the connectable gears 7, and a dotted line connecting the center of the gear 7 being edited and the center of the connectable gear 7.

[0120]

In addition, when two or more gears 7 are connected and generated or when a position change of a gear 7 is performed in editing mode, connecting the gear 7 to another gear 7, the phases, angles, and rotation angles of the gear 7 being edited (generated or positionally changed gear 7) are automatically set for the gears being connected to so that the connected gears 7 and 7 synchronously rotate together.

As illustrated in the flowchart in FIG. 20, the gear being edited is set as g_i and the connection destination gears are set to g_j (step S70).

That is, connection destination gears Pg_i for the gear g_i are set to the gear g_j , and the gear g_i is included in the collection CG_j for the gears connected to the connection destination gears g_j (step S71).

[0121]

Furthermore, it is determined whether a connection relationship between the gear g_i and the gear g_j is a meshed connection or a coaxial connection. For example, if the central coordinates of the gear g_i and the central coordinates of the gear g_j match, the connection is determined to be coaxial, and if the central coordinates do not match, the connection is determined to be meshed (step S72).

In the case of a meshed connection, the phases of the gear g_i and the gear g_j are set so as to mesh as described below (step S73), then the ratio and angle of the gear g_i are set to mesh with the gears g_j (step S74).

Moreover, when the gear g_i and the gears g_j connected by coaxial fixation, the rate and angle of the gear_i are set to overlap with the gears g_j (step S75).

[0122]

Furthermore, when the phase of the gear g_i and the gears g_j are set to mesh, as illustrated in the flowchart in FIG. 21, a minimum k and l are obtained from the formula given in step S77.

Here, $X_{ci,k}$ indicates, when a number of teeth of the gear g_i is set to c_i and the x coordinate of the center of the gear g_i is set to 0, an x coordinate or an approximated value of a k th point from a reference point within virtual points K formed in equal intervals as described above on the pitch circle 8 of the gear g_i having a number of teeth c_i , and $Y_{ci,k}$ indicates, when the number of teeth of the gear g_i is set to c_i and the y coordinate of the center of the gear g_i is set to 0, a y coordinate or an approximated value of the k th point from a reference point within the virtual points K formed in equal intervals as described above on the pitch circle 8 of the gear g_i having the number of teeth c_i . In addition, $X_{cj,l}$ indicates, when a number of teeth of the gears g_j is set to c_j and the x coordinate of the center of the gear g_j is set to 0, an x coordinate or an approximated value of an l th point from a reference point within virtual points K formed in equal intervals as described above on the circumferences of pitch circle 8 of the gear g_j having the number of teeth c_j , and $Y_{cj,l}$ indicates, when the number of teeth of the gear g_j is set to c_j and the y coordinate of the center of the gear g_j is set to 0, a y coordinate or an approximated value of an l th point from a reference point within the virtual points K formed in equal intervals as described above on the pitch circle 8 of the gear g_j having the number of teeth c_j .

[0123]

Note that the x , y coordinates of each of the virtual points K on the circumference of the pitch circle 8 when the x , y coordinates are set to (0, 0) are stored in the storing region 6 as data for the rotating body image display program obtained for each of a gear g_m having a plurality of different preset number of teeth cog_m . Moreover, in this example, numerical values including decimal point portions are not used, as described above, and because only integers, positive integers, and natural numbers are used, when the actual coordinate values for x , y coordinates each preset point include decimal point portions, for example, approximated value is used to form an integer. That is, decimal points are rounded down or rounded up.

Furthermore, by adding the x coordinate of the center of the pitch circle 8 of the gear g_i (gear g_j) to the x coordinates of each of the virtual points K on the circumference of the pitch circle 8 when the x, y coordinates of the center of the pitch circle 8 of the gear m having the same number of teeth c_i (numbers of teeth c_j) as the gear g_i (gear g_j) are set to 0, 0, an x coordinate (approximated value) of each of the virtual points K on the circumference of the pitch circle 8 of the gear g_i (gear g_j) can be obtained. Similarly, by adding the y coordinate of the center of the pitch circle of the gear g_i (gear g_j) to the y coordinates of each of the virtual points K on the circumference of the pitch circle 8 when the x, y coordinates of the center of the pitch circle of the gear having the same number of teeth c_i (numbers of teeth c_j) as the gear g_i (gear g_j) are set to 0, 0, a y coordinate (approximated value) of each of the virtual points K on the circumference of the pitch circle of the gear g_i (gear g_j) can be obtained.

[0124]

Here, the formula in step S77 obtains a distance between the kth point from the kth point from the virtual point K serving as a reference on the circumference of the pitch circle 8 of the gear g_i and the lth point from the virtual point K serving as a reference on the circumference of the pitch circle 8 of the gear g_j . Therefore, when a combination of the virtual points K having mutually minimum distances is obtained in a one-to-one combination of the plurality of virtual points K on the circumference of the pitch circle 8 of the gear g_i and the plurality of virtual points K on the circumference of the pitch circle 8 of the gear g_j , k becomes a number sequential to the virtual point K serving as a reference in the virtual points K of the obtained gear g_i , and l becomes a number sequential to the virtual point K serving as a reference in the virtual points K of the obtained gear g_j . Note that because the gear g_i and the gear g_j are at a distance where they can mutually mesh, the pitch circle 8 of the gear g_i and the pitch circle 8 of the gear g_j are in contact, and the virtual points of each of the gears g_i and g_j , being mutually nearest as described above, form points nearest to a point of contact 71 of two pitch circles 8 in a state of mutual contact.

[0125]

Furthermore, how to obtain a phase p_i of the gear g_i for meshing with the gears g_j having phases p_j in a subsequent editing mode starting time is obtained in step S78.

Note that here, a position at one period of rotation of a predetermined angle of one tooth of the gear g_i is in a state where, in the gears g_i and g_j meshed together, a tooth space between teeth of the one gear g_i and a tooth of another gear g_j mesh (may be the reverse), so when the gear g_i and the gear g_j are meshed together, the phases deviate by a half period.

Here, the phases at the reference time t^{st}_i of gears x_j (phases at the editing mode starting time based on the process at editing mode starting time described above) are set to p_j , and the phase of the lth virtual point K obtained as described above is obtained as a remainder when the lth sequential number is divided by the number of phases (number of image data for one period) phase. By dividing the phases of the virtual point at the lth sequential number from the phases p_j at the reference time t^{st}_i , a deviation in the obtained virtual point K from the phase at the reference time t^{st}_i can be obtained. Here, when this phase deviation is subtracted from the kth sequential number obtained for the gear g_i and this is divided by the number of phases phase and a remainder obtained, the phases of the gear g_i and the gears g_j match, and in this state a tooth of the gear g_i and a tooth of the gear g_j overlap. Furthermore, by dividing the amount of phase deviation described above from the Kth sequential number prior to dividing by the number of phases phase and adding a resolution of 1/2 (when a remainder emerges, decimal points are rounded down values for the Gaussian function described above), a value at which the phase is deviated by a half period is set as the position p_i of the gear g_i at the reference time.

[0126]

By this, the phase p_i of the gear g_i is determined corresponding to the phases p_j of the gear g_j such that the gear g_i and the gear g_j mesh. In this state, respective predetermined angles of the gear g_i and the gear g_j , that is, movements in rotating one period are performed at a same time, allowing for continuous rotation in a meshed state.

Next, as illustrated in the flowchart in FIG. 22, a process is performed to set the rate and angle of the gear g_i to mesh with the gear g_j such that the gears g_i and g_j , which are meshed as described above, are maintained in a meshed state.

First, a common denominator value f_i for that described above for the gears g_i and a common denominator value f_j of the gears g_j are set to be the same.

In addition, the numerator s_i of the rotation rate of the gear g_i and the numerators s_j of the rotation rates of the gears g_j are equalized and the signs thereof are switched so that the rotation direction reverses.

[0127]

Here, the denominators b_i and b_j of the rotation rate of the gear g_i and the gear g_j are obtained by multiplying the total number of virtual points K rs_i and rs_j , being values obtained by multiplying the numbers of teeth c_i and c_j of the gears g_i and g_j and the number of phases phase, and the common denominator values f_i and f_j , and when $f_i = f_j$, a ratio of the denominator b_i of the rotation rate of the gear g_i and the denominator b_j of the rotation rate of the gears g_j becomes equal to a ratio of the number of teeth c_i of the gear g_i and the number of teeth c_j of the gears g_j , that is the rotation rates become equal.

Moreover, because the numerator s_i of the rotation rate of the gear g_i and the numerator s_j of the rotation rate of the gear g_j are equal (reverse directions), the gear g_i and the gear g_j have equal rate of period rotation, have different times per rotation depending on a gear ratio, and rotate at a rotation rate corresponding to the meshed gears g_i and g_j .

Also, the angle a_i is obtained based on the formula for determining the angle a_i in step S80. That is, the angle a_i of the gear g_i is obtained based on the phase p_i of the gear g_i meshed with the gear g_j obtained as described above. Here, the rotation rate of the gear g_i and the denominator b_i of the fraction expressing the rotation rate are changed as described above, and corresponding thereto, an angle prior to meshing is changed (that is, the original angle a_i is multiplied by the original denominator b_i and divided by the new denominator b_i), such that the phase phase of the gear g_i for the angle becomes 0. Furthermore, a value obtained by multiplying the phase p_i obtained as described above by the common denominator value f_i (the phase p_i corresponds to the denominator b_i) is added to the angle of the obtained phase 0, and as described above, form a phase in which the angle a_i of the gear g_i and the gears g_j mesh. At this stage, resolution levels (res_i) corresponding to the phases phase are positionally aligned. In adjustments up to this stage, the gear g_i and the gear g_j are displayed in a meshed state; however, the likelihood of a deviation arising in the timing for switching the image data corresponding to the phases phase for the gear g_i and the gear g_j , and the image data being switched at the same time and the same frame is low.

Therefore, accurate angle alignment is required for forming the denominator b_i at the rotation rate obtained by multiplying the resolution level (res_i) by the common denominator value f_i , and an amount of deviation from the rotation rate in units of resolution (phase) at a current rotation rate of the rotation angles a_j of the gears g_j is obtained for an angle adjustment of the level of the denominator b_i in the rotation angle a_j of the gear g_j meshed with the gear g_i , against which the angle a_i of the gear g_i can be more finely adjusted. The rotation angle in units of resolution is an angle in which the numerator is deviated per f_i with respect to the denominator b_i expressing the rotation rate. Therefore, the numerators a_j of the fractions expressing the rotation rates of the current gears g_j are divided by f_j , and by obtaining the remainder, an amount of deviation from the rotation rate in units of resolution (phase) of the rotation angles a_j of the gears g_j can be obtained. Note that here, as a condition for the meshing described above, the gear g_i and the gear g_j rotate in opposite directions, so the value of a remainder obtained by dividing the angles a_j of the gear g_j from the value of the numerator of one resolution portion (one phase portion) (that is, f_i) with respect to the denominator b_i , and the fine adjustment portion based on the value from which 1 was further subtracted is further added to a value obtained by adding the value ($p_i \times f_i$) of meshing phase portion in the value of the rotation rate of the gear g_i in which the phase is 0 as described above. Thereby, the angle a_i and the speed s_i of the gear g_i meshed with the gear g_j are determined and are displayed such that the gear g_j and the gear g_i mesh, and in a state where the image data of the gear g_j and the image data of the gear g_i have substantially synchronous switching timing when displayed, that is, a state where the image data for the gear g_i and the gear g_j switches every time in the same frame with high accuracy, is achieved.

In addition, similar to the case in step S52, a drawing starting time closest to the starting time t_{ed} is obtained prior to starting time t_{ed} , and the obtained time is set to the new reference time t^{st}_i . The reference angle a^{st}_i at the reference time t^{st}_i is set to the obtained angle a_i (step S80).

[0128]

Next, the process of setting the speed and angle of the gear g_i when connected by coaxial fixation to overlap with the connection destination gears g_j is described with reference to the flowchart in FIG. 23.

In this case, essentially a process substantially similar to when the hands h_i are attached to the gears g_i is performed. Therefore, the gear g_i rotates one round in a same time corresponding to rotation of the connected gears g_j , and the fraction s_i/b_i of the rate of the gear g_i becomes equivalent to the fraction s_j/b_j indicating the rate of the connection destination gear g_j . That is, the numerator s_i of the fraction of the rate of the gear g_i is obtained by multiplying the rates s_j of the gear g_j by the number of teeth c_i of the gear g_i ; and the common denominator value f_i of the denominator of the rotation rate of the gear g_i is obtained by multiplying the common denominator value f_j of the connection destination gear g_j by the number of teeth c_j of the connection destination gear g_j .

Furthermore, when the maximum common divisor of the rate s_i of the gear g_i during editing and the common denominator f_i is set to n (when there is a common divisor), s_i and f_i form values divided by the maximum common divisor. That is, the denominators f_i of the fractions of the rotation rates obtained by multiplying by the resolutions rs_i , and the numerators s_i of the fractions of the rotation rates are mutually reduced.

[0129]

Moreover, the denominator b_i of the rotations rate of the gear g_i is obtained by multiplying the value f_i changed as described above and the resolution rs_i .

Note that the processing described above equalizes the rotation rate of the gear g_i similarly to the hands h_i and the rotation rate of the connection destination gear g_j , however, when the numbers of teeth c_i and c_j differ between the gear g_i and the connection destination gear g_j , the timing for switching the image data for animation may be different.

In addition, the angle a_i is obtained based on the formula by which the angle a_i is obtained in step S81. Likewise in this case, similar to when the angles of the gears meshing with gears described above are obtained, the rotation rate of the gear g_i and the denominator b_i of the fraction expressing the rotation rate are changed as described above, and corresponding thereto, an angle prior to meshing is changed (that is, the original angle a_i is multiplied by the original denominator b_i and divided by the new denominator b_i), such that the phases phase of the gear g_i for the angle becomes 0. Furthermore, a value obtained by multiplying the original phase p_i by the common denominator value f_i (the phase p_i corresponds to the denominator b_i) is added to the angle of the obtained phase 0, and the angle of the phase level is determined. Furthermore, for accurate angle alignment at the denominator b_i level of the rotation rate obtained by multiplying the resolution level (res_i) and the common denominator value f_i , an amount of deviation from the rotation rate is obtained in units of resolution (phase) at a current angle of the rotation angle a_i of the gear g_i , and the obtained amount of deviation is added to the angle of the phase level described above. Note that two coaxially rotating gears may even match in terms of rotation rate, where the original angle a_i is converted corresponding to the new denominator b_i (that is, f_i) to an integer and may be used.

In addition, similar to the case in step S52, a drawing starting time closest to the starting time t_{ed} is obtained prior to starting time t_{ed} , and the obtained time is set to the new reference time t^{st}_i . The reference angle a^{st}_i at the reference time t^{st}_i is set to the obtained angle a_i (step S81).

[0130]

Moreover, the processes of determining and displaying the connectable gears on the connective topology described above, as illustrated in the flowchart in FIG. 24, can obtain the positionally and topologically connectable gears h_i by removing gear g_j that do not satisfy the conditions given in step S82 from the gear g_j positionally connectable to the gear g_i (step S82). Note that here, the gear g_i being edited and the connected gear g_j are different. Furthermore, $groot_i$ indicates, in the gear train to which the gear g_i being edited belongs, just one gear having no connection destination (gear upstream of a drive source) among gears (including gear and hand combinations) where meshing connection relationships are all traceable to a connection relationship in both directions from the gear g_i .

Also, $groot_i$ indicates, in the gear train to which the gear g_i connected to the gear g_i of the collection belongs, just one gear having no connection destination (gear upstream of a drive source) among gears (including gear and hands combinations) where meshing connection relationships are all traceable to a connection relationship in both direction from the gears g_i .

Furthermore, for a non-connectable gear in terms of connective topology and a gear, in the case that the $groot_i$ and $groot_j$ match, the rate s_i of the gear g_i and the rate s_j of the gear g_j may both be non-0. That is, a connection such as driving forces from two gears being transmitted to one gear is impossible, and a gear train cannot be connected in a closed ring state.

Furthermore, for all of the gears g_j positionally capable of connecting to the gear g_i , the connection capability is displayed (step S83).

[0131]

In addition, the process of recursively and automatically setting the angle and rotation rate according to connective topology described above, as illustrated in the flowchart in FIG. 25, sets a gear being edited to g_i and connection destination gear to g_j (step S85).

Next, whether the rate s_j of the connection destination gear g_j is 0 is determined (step S86).

Here, as illustrated in the flowchart in FIG. 18, the gear g_i being edited is already connected to another gear 7, but the rate s_i of the gear g_i may not be 0, and in the case the rate s_j of the gear g_j is not 0, it is determined that the gear g_i and the gear g_j are connectable in terms of connective topology, so the rate s_i of the gear g_i is set to 0. Therefore, the gear g_i forms an element of the collection CG_j having the gear g_j as a connection source (step S87).

Furthermore, as described below, the gear g_i is recursively connected to the gear g_j (step S88).

[0132]

Moreover, when the rate s_j of the gear g_j is 0, the gear g_j forms an element of the collection CG_i having the gear g_i as a connection source (step S89).

Furthermore, as described below, the gear g_i is recursively connected to the gear g_j (step S90).

Here, a method for recursively connecting the gear g_i to the gear g_j and a method for recursively connecting the gear g_j to the gear g_i are essentially the same, and a method is described below for recursively connecting the gear g_i to the gear g_j with reference to the flowchart in FIG. 26.

First, whether or not setting of a connection destination of the gear g_i is complete and that a connection destination of the gear g_i is not the gear g_j is determined (step S91).

[0133]

If “setting of the connection destination of the gear g_i is complete and the connection destination of the gear g_i is not the gear g_j ” is not true, the connection destination gear P_{gi} of the gear g_i is set to the gear g_j (step S93).

In addition, if “setting of the connection destination of the gear g_i is complete and the connection destination of the gear g_i is not the gear g_j ” is true, the connection destination of the gear g_i is set to the collection CG_j having the gear g_j as a connection source. That is, the gear g_i being edited is already connected to a gear g_k apart from the gears g_j , and the gear g_k connected to the gear g_i forms an element of the collection CG_j having the gears g_j as a connection source (step S92).

Likewise in this case, the connection destination of the gear g_i is set to the gears g_j .

Next, whether the center of the gear g_i and a rotation center of the gear g_j match is determined (step S94).

[0134]

If these do not match, meshed connection is determined, and in the case of a meshed connection, the phase of the gear g_i and the gears g_j are set so as to mesh as described above (step S95), then the ratio and angle of the gear g_i are set to mesh with the gears g_j (step S96). If the rotation center of the gear g_i and the rotation centers of the gears g_j do match, coaxially fixed connection is determined for the gear g_i and the gear g_j , and the rate and angle of the gear g_i and the gear g_j are set to overlap (step S97).

Next, whether the gear g_j belongs to the collection CG_i having the gear g_i as a connection source or not is determined (step S98).

[0135]

Furthermore, when the gear g_j does not belong to the collection CG_i , the following process is executed on all of the gears g_k belonging to the collection CG , and when the gears g_j do belong to the collection CG_i , the gears g_j are deleted from the collection CG_i (step S99), and the following process is executed on all of the gears g_k belonging to the collection CG_i (step S100). Note that when the following process is executed on all of the gears g_k belonging to the collection CG_i , and the process ends (step S101).

The gears g_k included in the collection CG_i are sequentially traced and recursively connected as a gear train of the gear g_i (step S102).

According to a process such as the above, for example, the gear train of the gears g_k previously connected to the gear g_i has no driving gear, and when a driving gear is arranged in a gear train to which the gear g_j belongs and to which the gear g_i was connected following the gears g_k , the gear train of the gear g_k previously connected to the gear g_i and the gear g_i are sequentially traced from the gear g_j of the gear train having a driving gear, and the gear train of the gear g_k , the gear g_i , and the gear train of the gear g_j can be connected as one gear train.

[0136]

According to the rotating body image display device and the rotating body image display program of the present embodiment described above, images of rotating gears are arbitrarily arranged on a display screen of a mobile terminal such as the mobile phone 1 for which arithmetic processing of trigonometric functions, floating point arithmetic, or the like is limited, a plurality of gears can be combined, fixed coaxially or the like, configured into a gear train, and portrayed in color as rotating gears. That is, the mobile phone 1 can be provided with excellent content. In addition, the rotating image display program, data used by the program, and the like are minimal, and can therefore grant the storage capacity of the mobile phone 1 some breathing room, and a communication fee such as when downloading the rotating image display program over the internet as content can be reduced.

BRIEF DESCRIPTION OF THE DRAWINGS

[0137]

FIG. 1 is a block diagram depicting a mobile phone forming the rotating body image display device of an embodiment of the present invention.

FIG. 2 is a drawing for describing the number of teeth, the meshing of gears, and the pitch circles as rotating bodies displayed based on the rotating body image display program on the display screen of the mobile phone.

FIG. 3 depicts image data for an animation of gears stored in the storing region of the mobile phone.

FIG. 4 is a drawing for describing the virtual points arranged on the circumference of the pitch circle of the gear.

FIG. 5 is a drawing for describing display and operation when the rotating image display program is executed on the mobile phone.

FIG. 6 is a flowchart for describing the process of determination operations for frontward and rearward arrangement of gears and hands.

FIG. 7 is a flowchart for describing the process of determination operations for gear color.

FIG. 8 is a flowchart for describing the process of determination operations for hand color/form.

FIG. 9 is a flowchart for describing the process of determination operations for gear rotation rate.

FIG. 10 is a flowchart for describing the process of determination operations for hand angles.

FIG. 11 is a flowchart for describing the process of determination operations for background color.

FIG. 12 is a flowchart for describing the process of execution and display of gears/hands.

FIG. 13 is a flowchart for describing the process of updating gear/hand sets.

FIG. 14 is a flowchart for describing the process of drawing gear/hand sets.

FIG. 15 is a flowchart for describing the process of generating gears.

FIG. 16 is a flowchart for describing the process of generating hands.

FIG. 17 is a flowchart for describing the process of starting the editing mode.

FIG. 18 is a flowchart for describing the process of determining gear connection capability.

FIG. 19 is a flowchart for describing the process of determining and displaying positionally connectable gears.

FIG. 20 is a flowchart for describing the process of automatically setting the angle and rotation rate of a gear.

FIG. 21 is a flowchart for describing the process of aligning phases of mutually meshing gears.

FIG. 22 is a flowchart for describing the process of setting such that the rates and angles of mutually meshing gears align.

FIG. 23 is a flowchart for describing the process of setting such that the rates and angles of coaxially positioned gears align together.

FIG. 24 is a flowchart for describing the process of determining connectable gears in terms of connective topology.

FIG. 25 is a flowchart for describing the process of recursively and automatically setting the angle and rotation rate of a gear according to connective topology.

FIG. 26 is a flowchart for describing the process of recursively connecting gears.

DESCRIPTION OF REFERENTIAL NUMERICALS

[0138]

- 2. Displaying part
- 3. Calculation processing part (display controlling means)
- 4. Key inputting part
- 5. Network communication part
- 6. Storing region (storing means, circular coordinate storing means)
- 7. Gear (rotating body)
- 8. Pitch circle
- K. Virtual point
- rad. Radius

[FIG. 1]

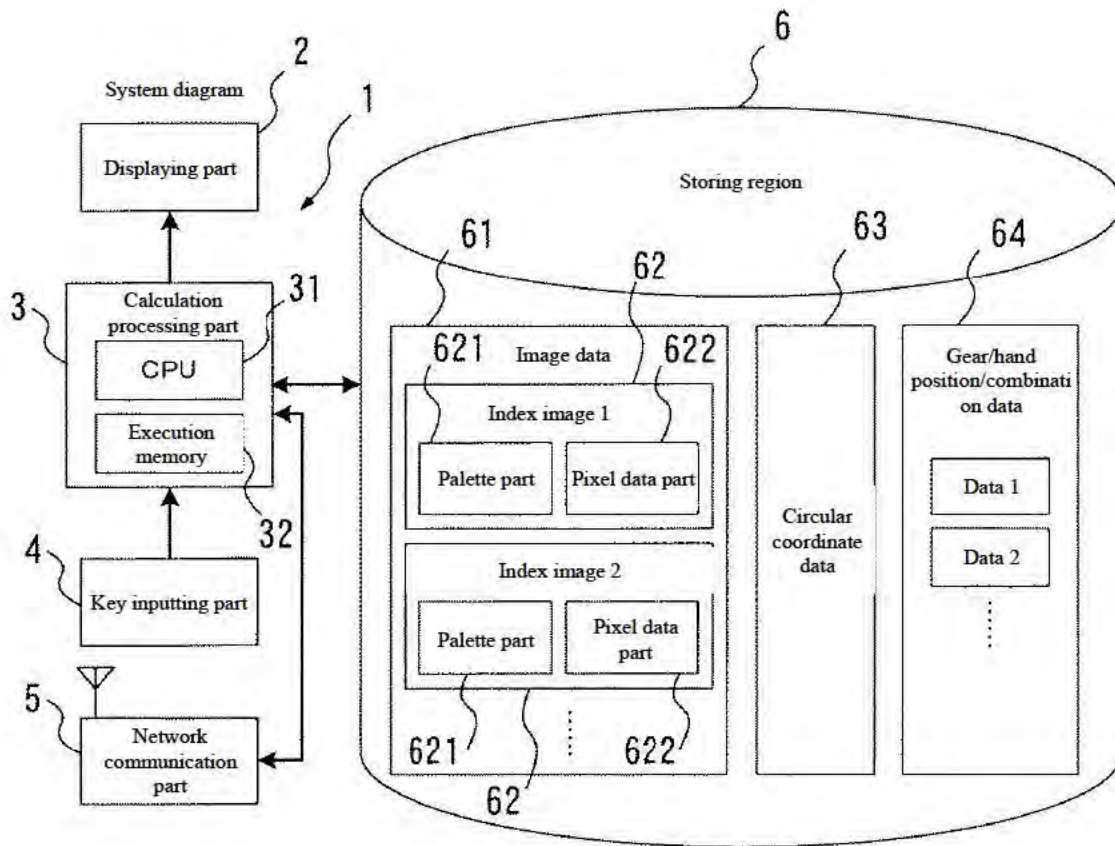


FIG. 2

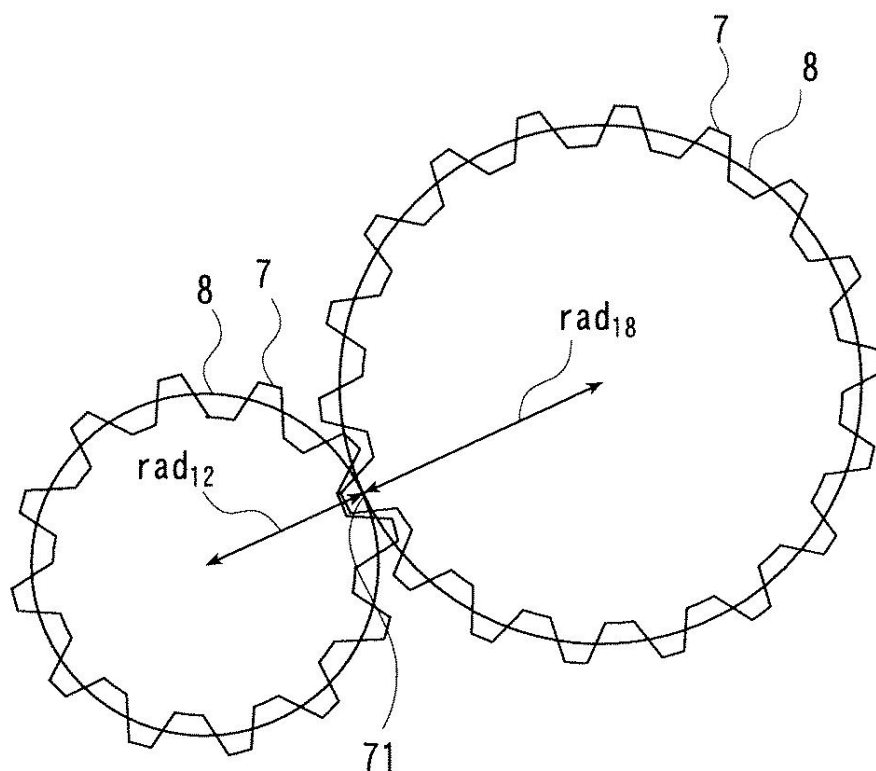


FIG. 3

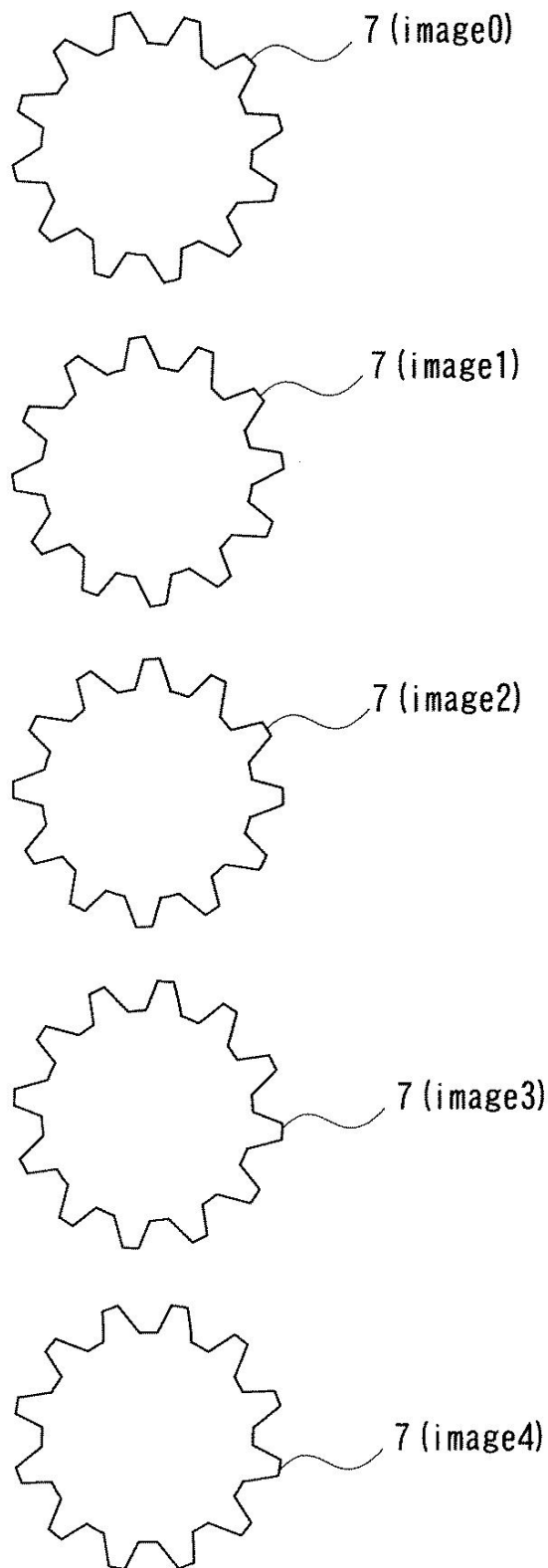


FIG. 4

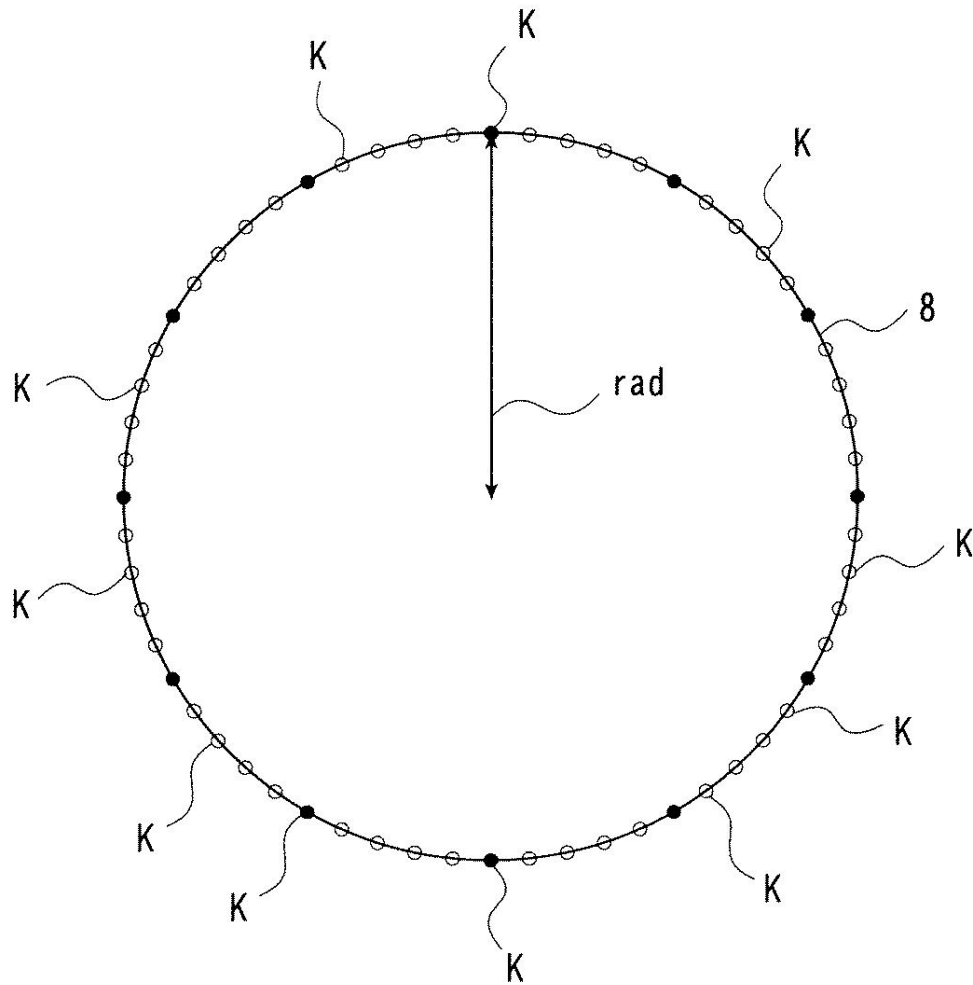


FIG. 5

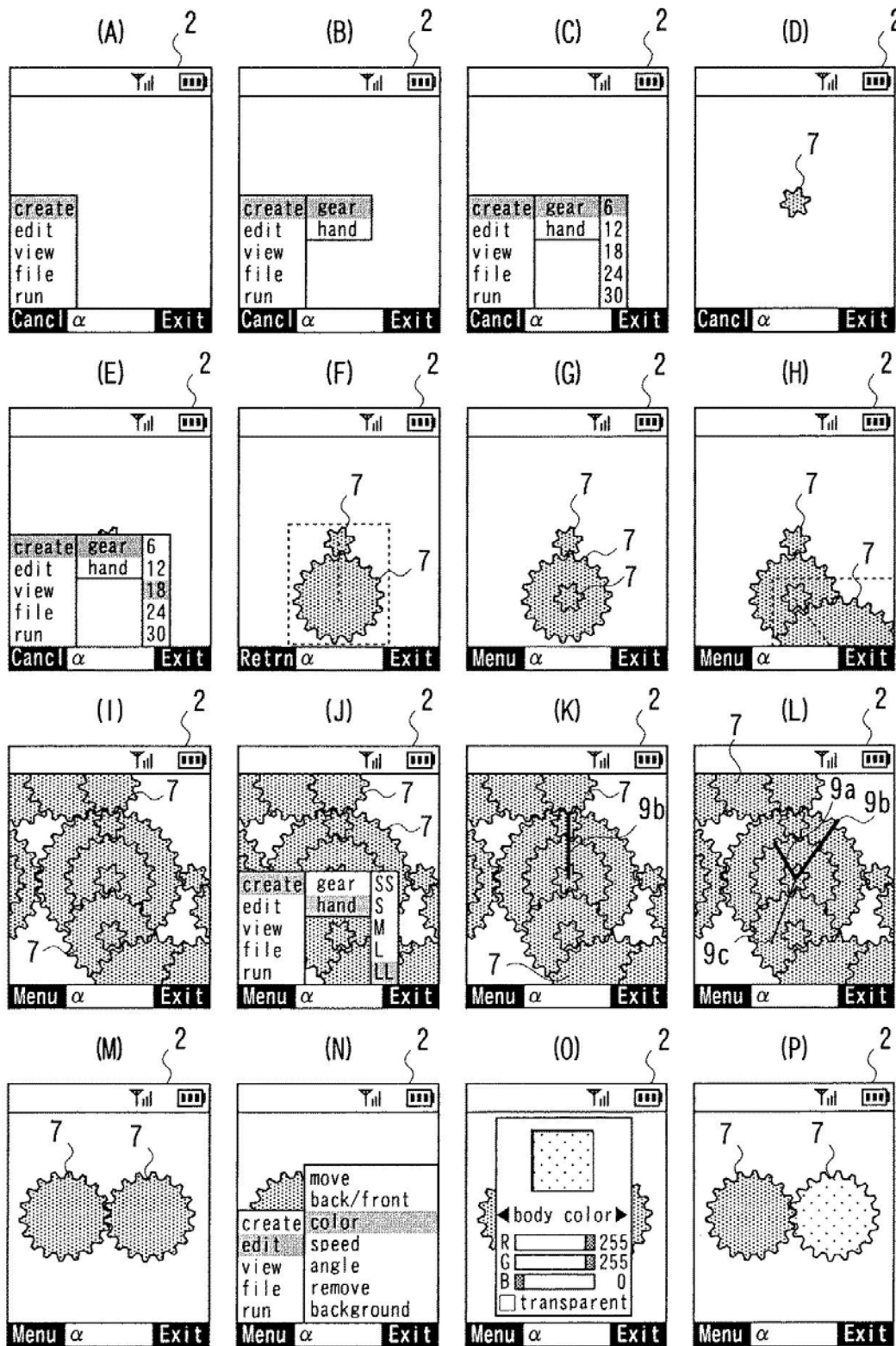
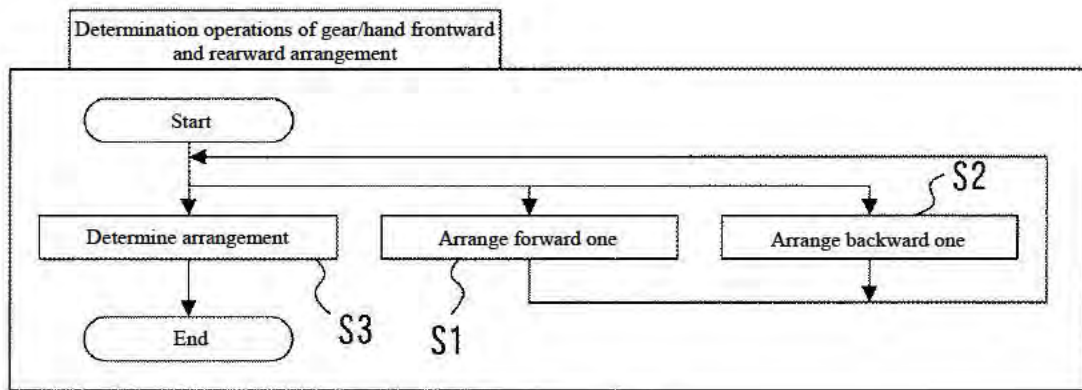


FIG. 6



[FIG. 7]

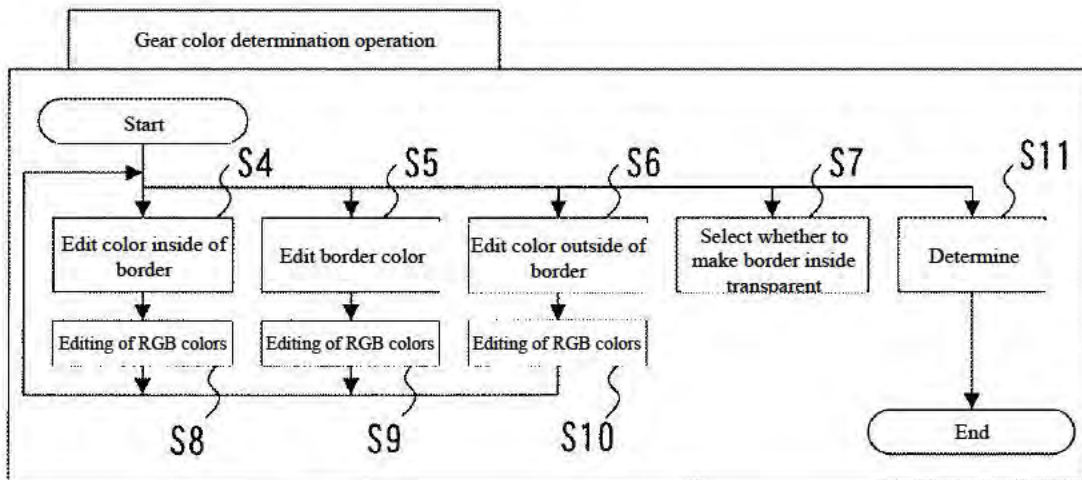


FIG. 8

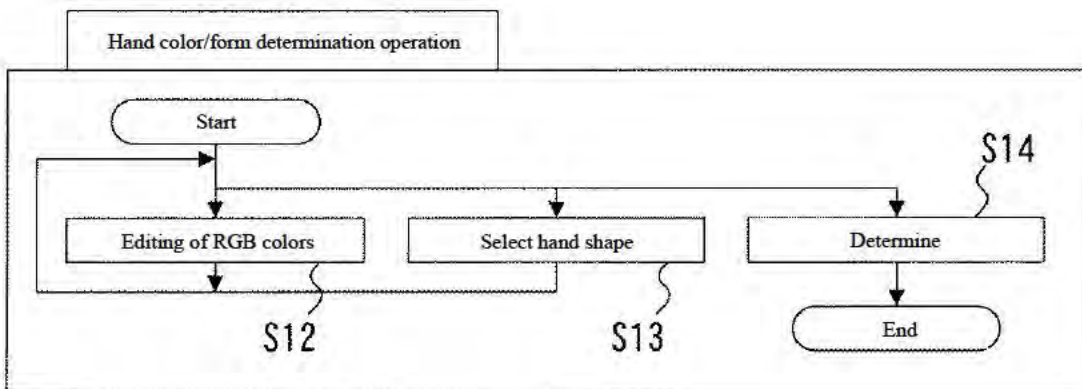


FIG. 9

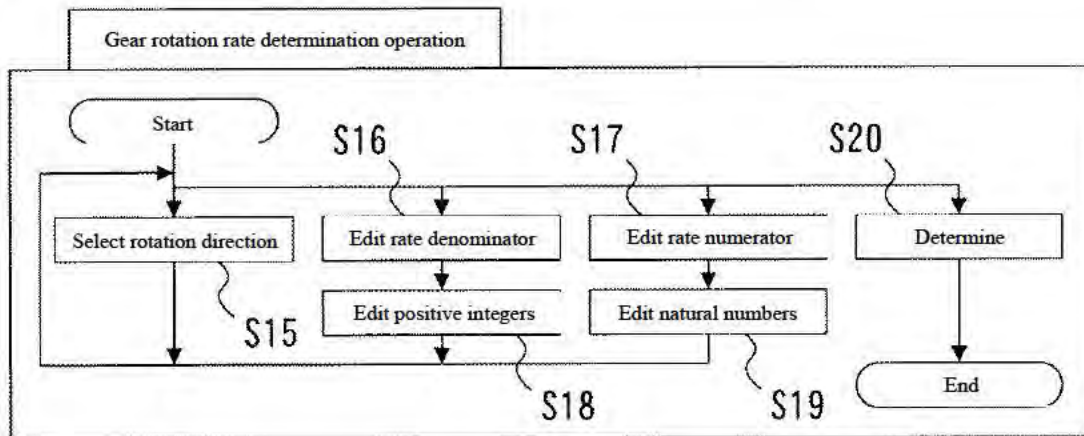


FIG. 10

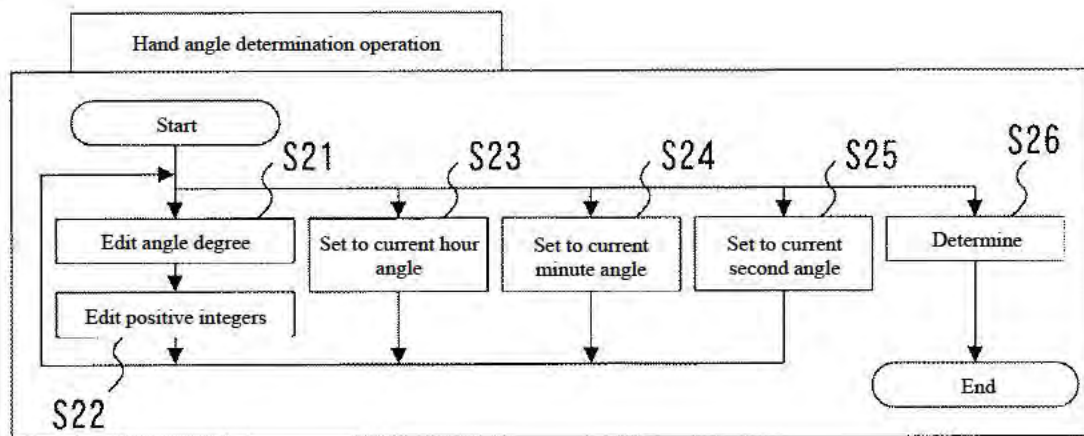


FIG. 11

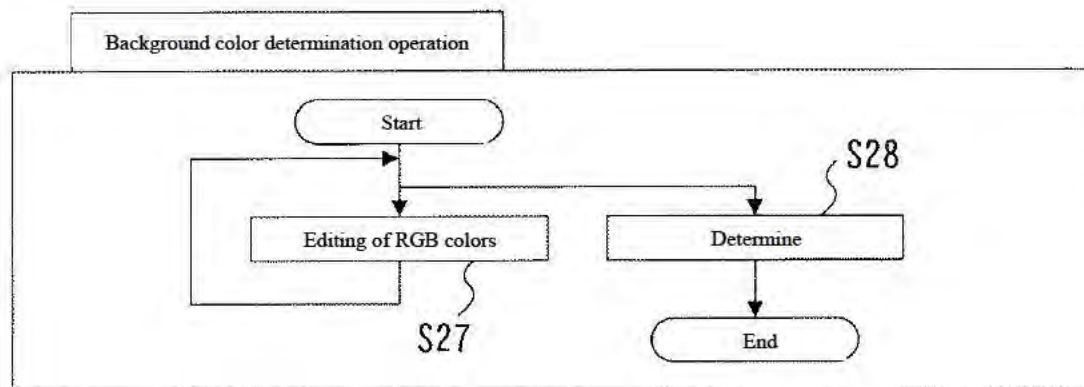


FIG. 12

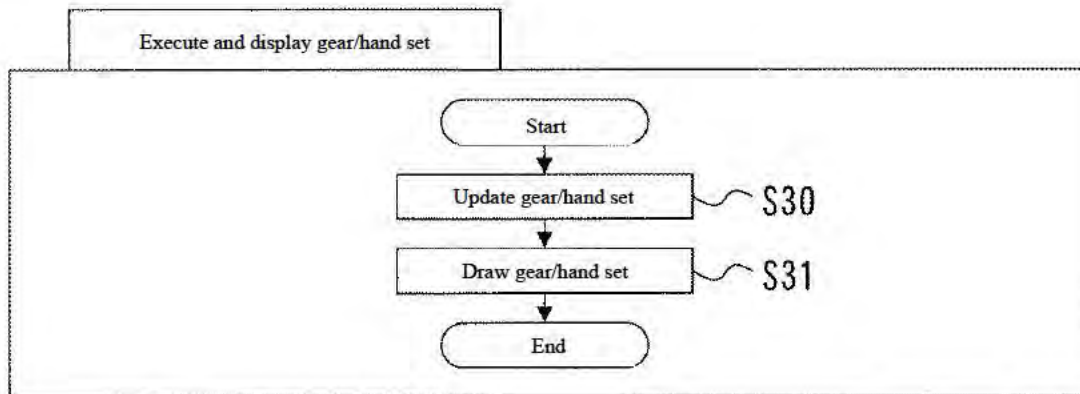


FIG. 13

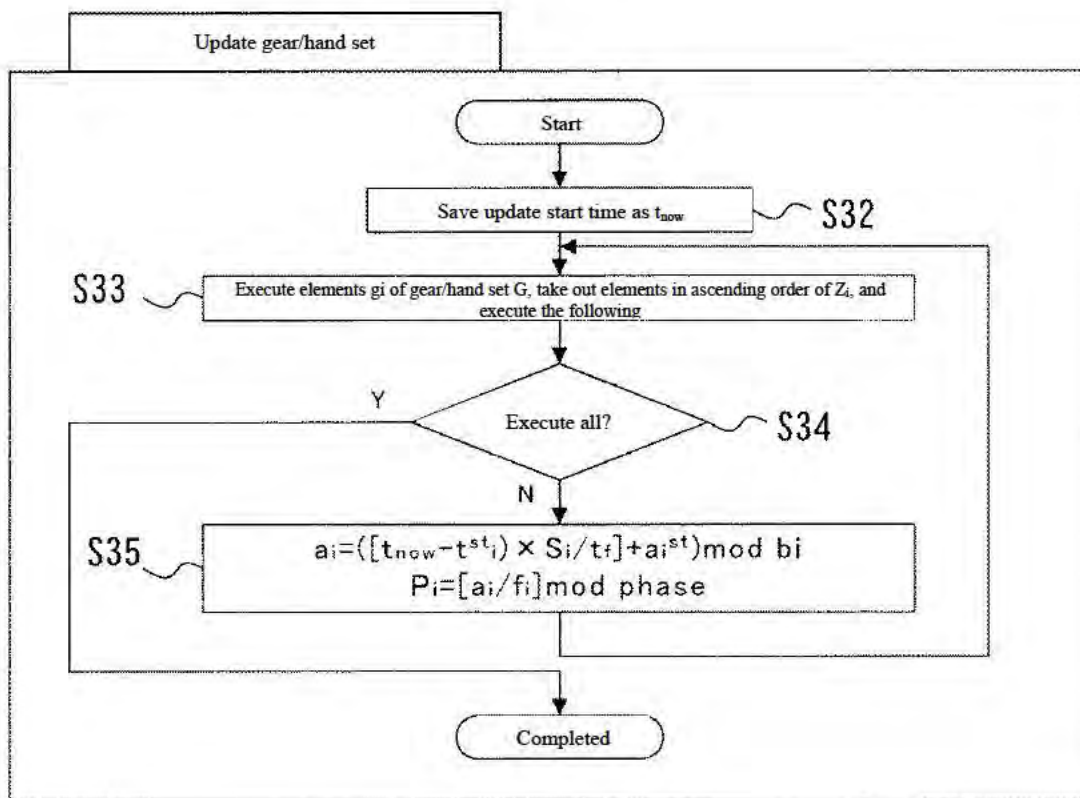


FIG. 14

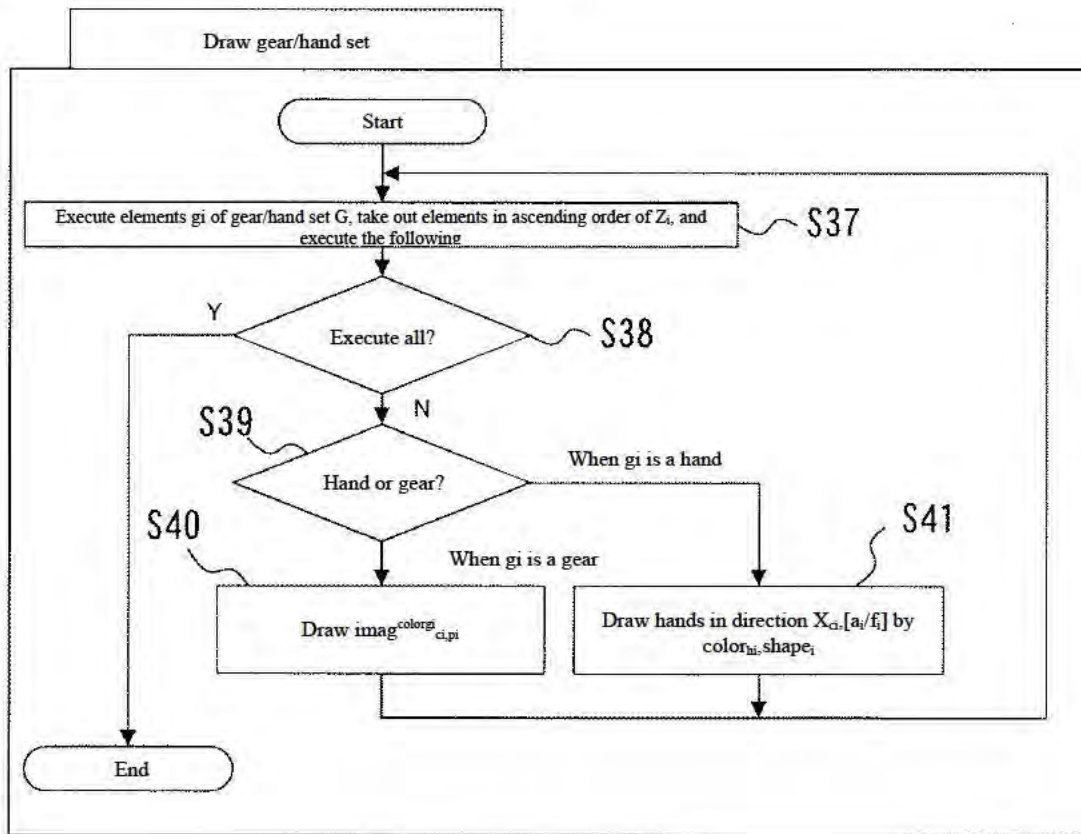


FIG. 15

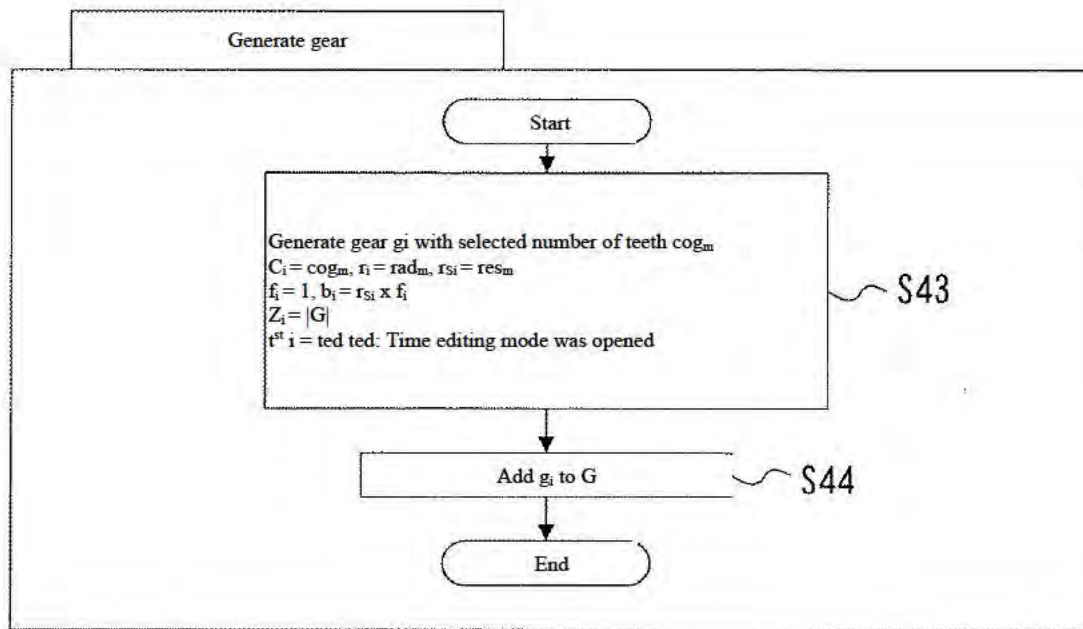


FIG. 16

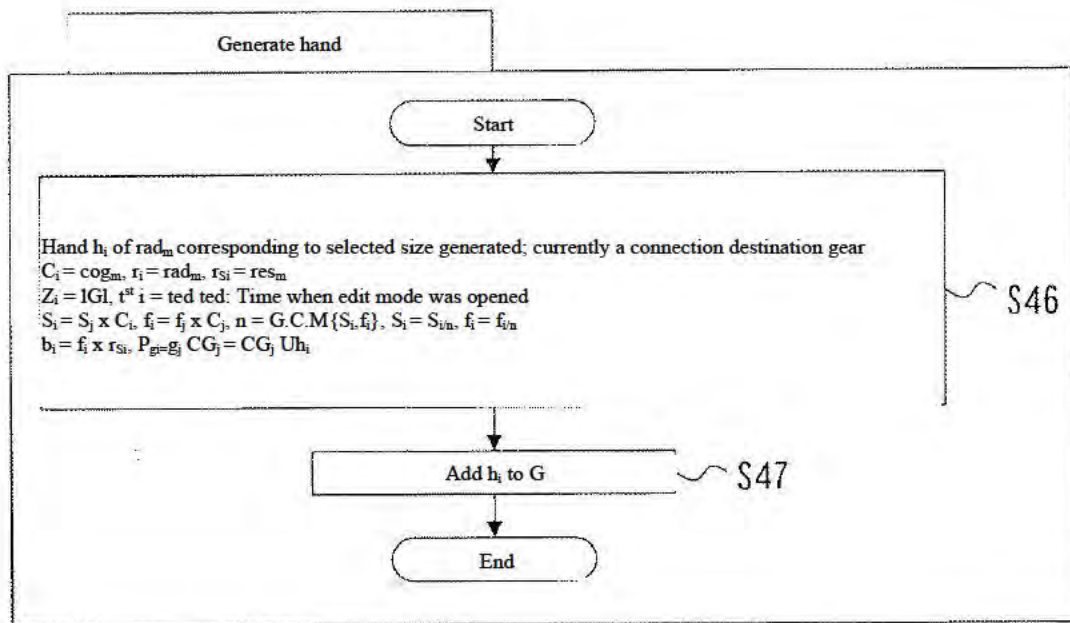
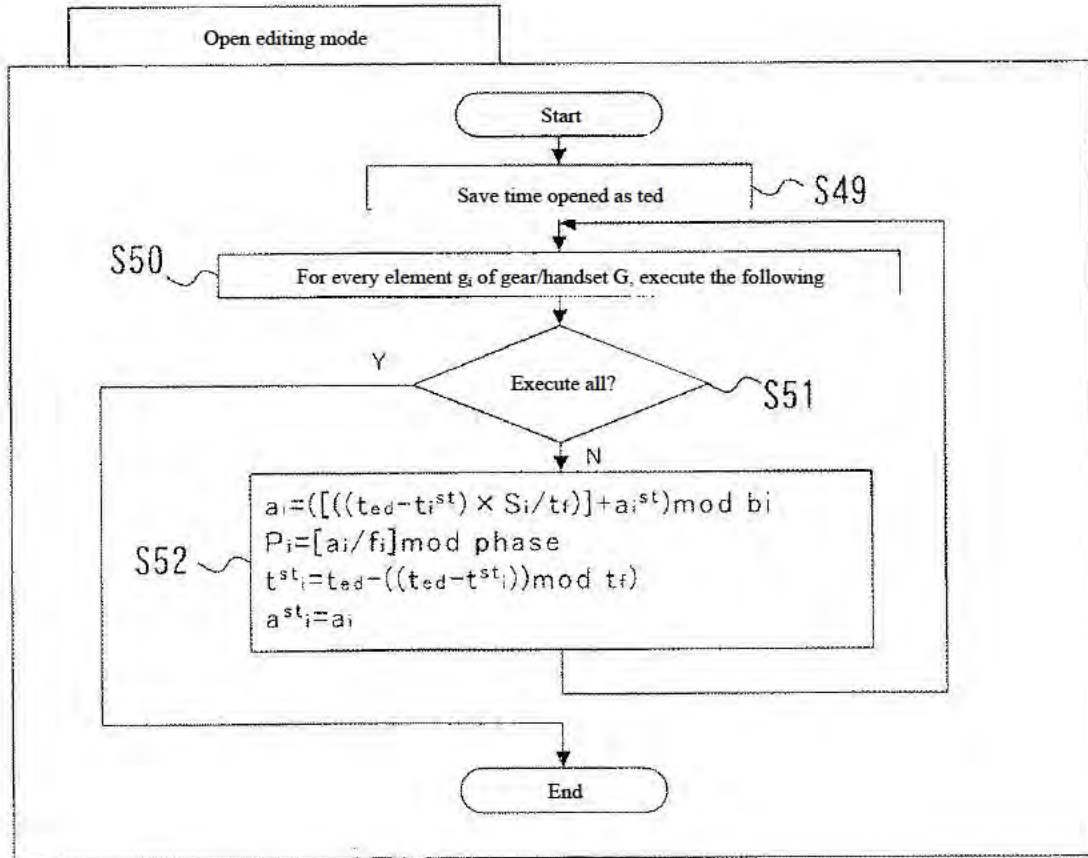
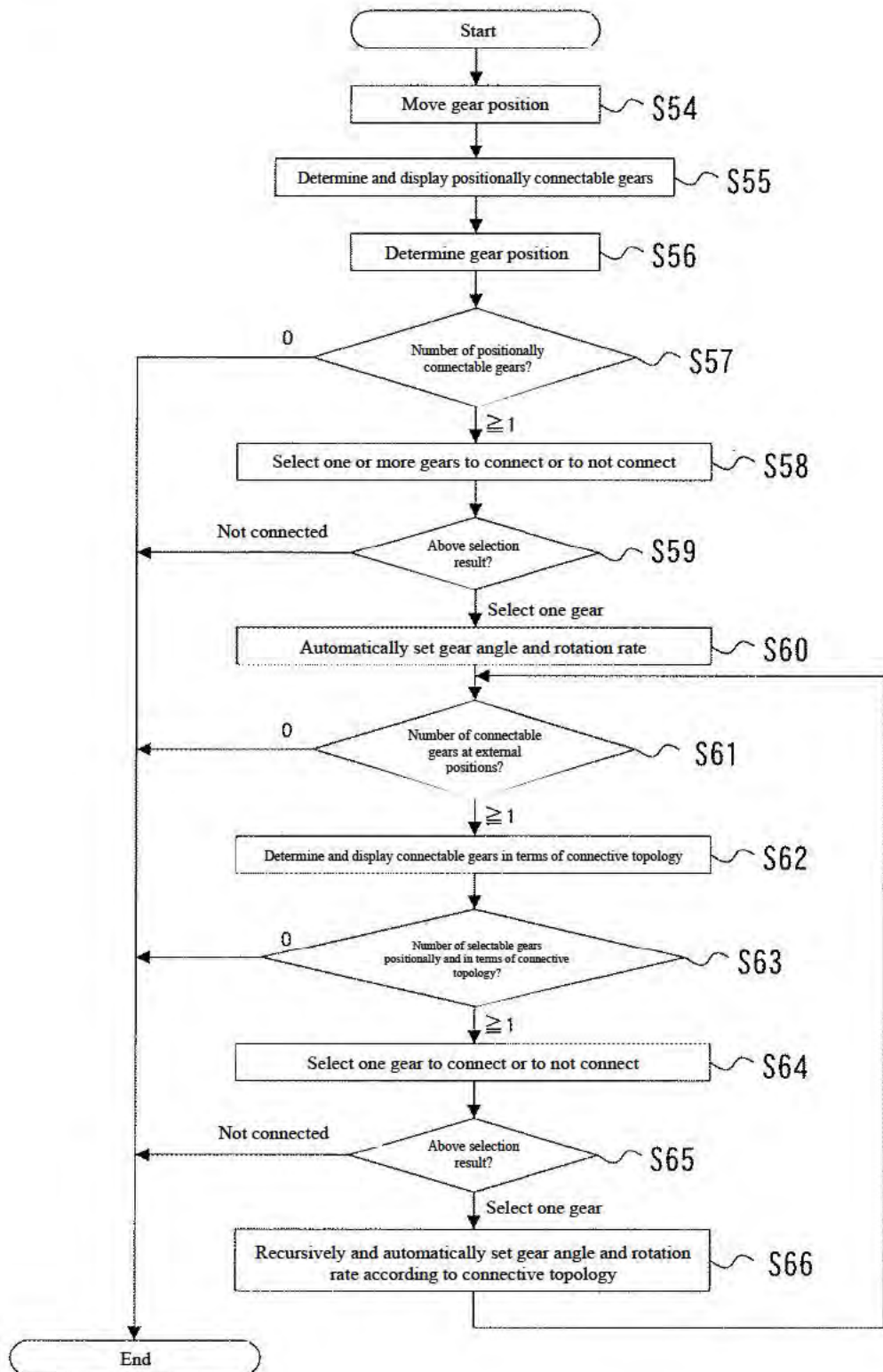


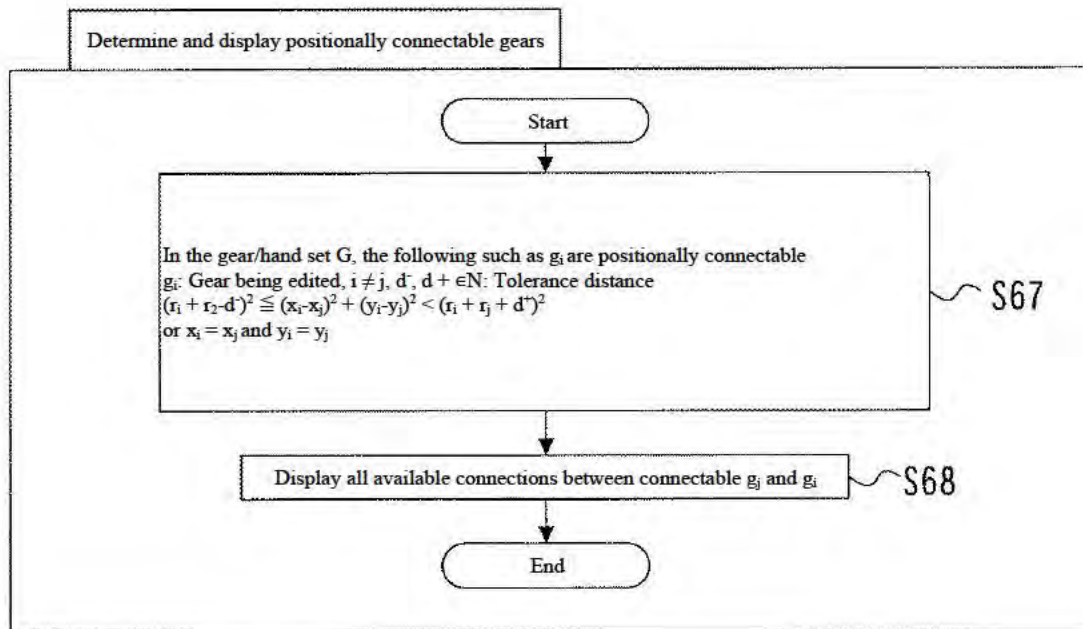
FIG. 17



[FIG. 18]



[FIG. 19]



[FIG. 20]

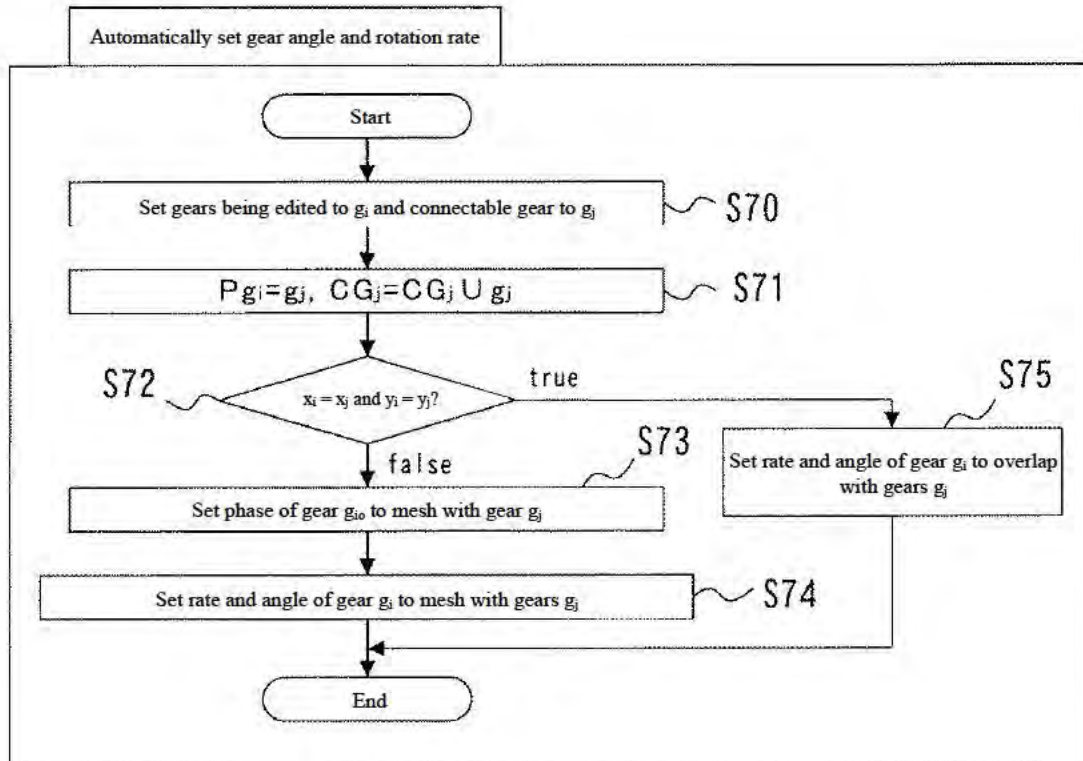


FIG. 21

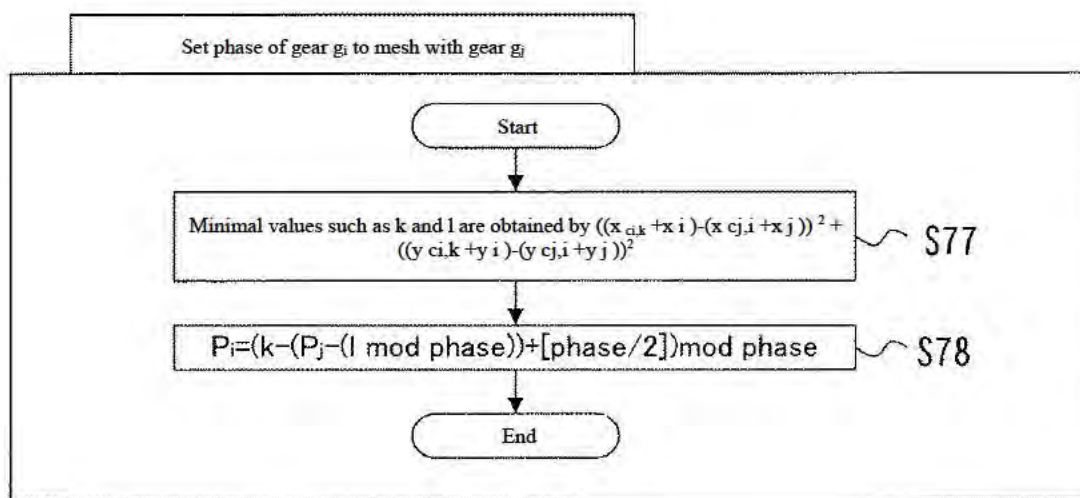


FIG. 22

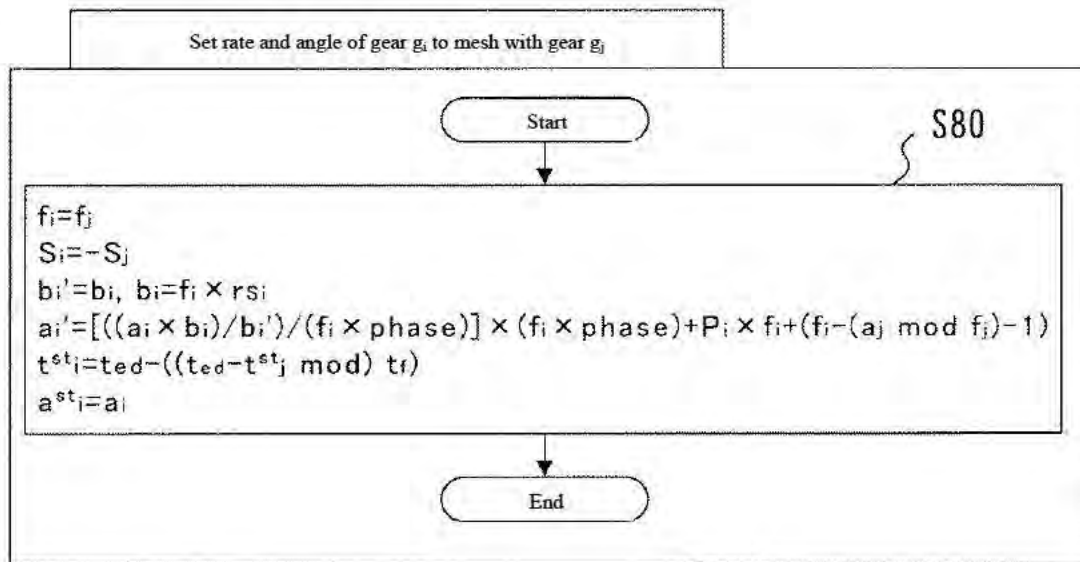


FIG. 23

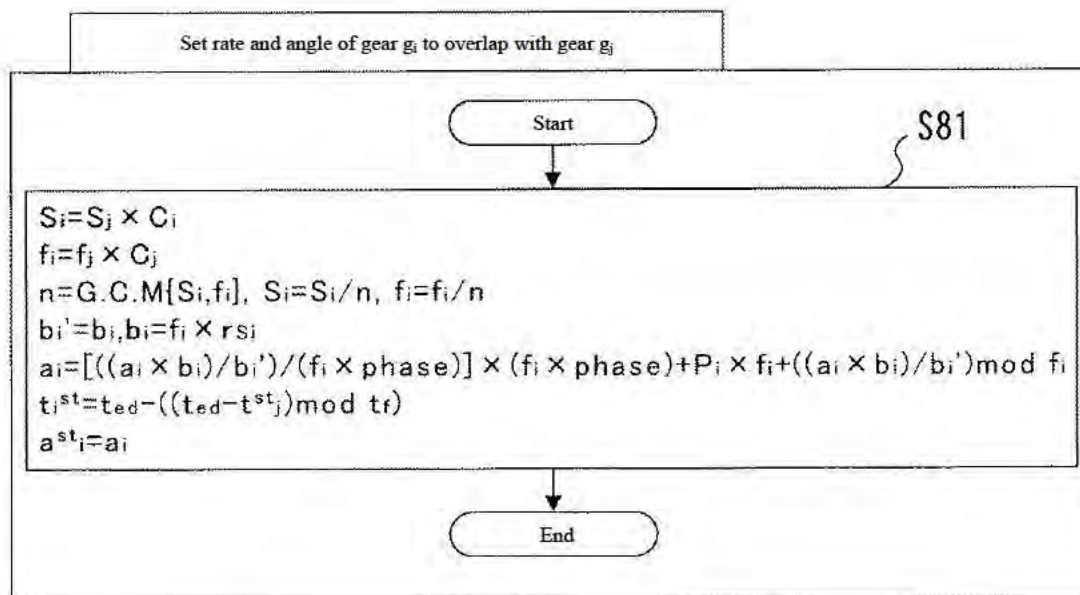


FIG. 24

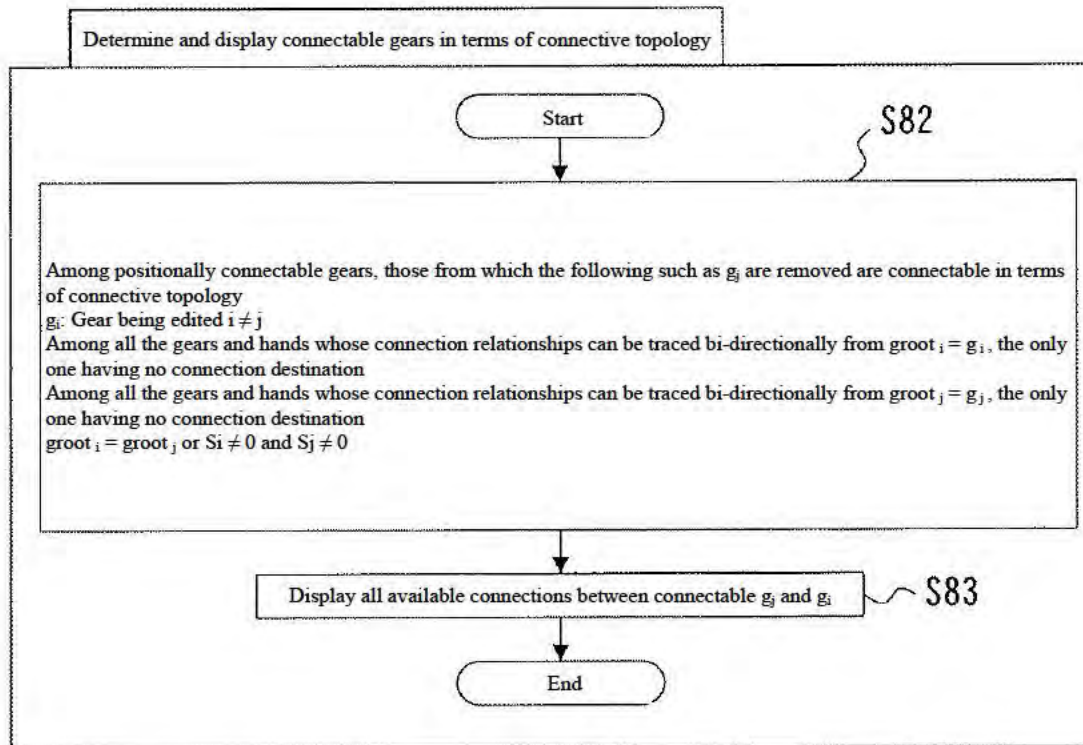


FIG. 25

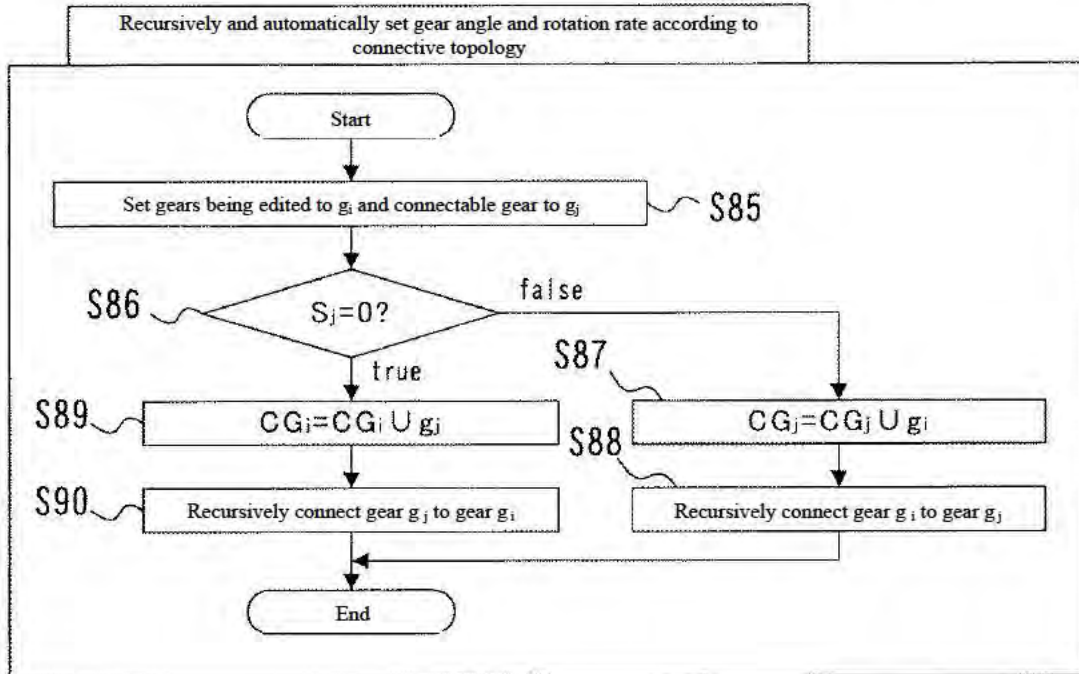


FIG. 26

