(19) **United States**

(12) **Patent Application Publication** (10) Pub. No.: **US 2009/0082686 A1**

Brown et al. (43) **Pub. Date:** **Mar. 26, 2009**

(54) **SYSTEM AND/OR METHOD FOR INITIATING A MEDICAL TASK INVOLVING MOTION WITH A DEVICE**

(76) Inventors: **Stephen J. Brown**, Woodside, CA (US); **David W. Brown**, Bingen, WA (US)

Correspondence Address:
Schacht Law Office, Inc.
ROY-G-BIV Corp.
2801 Meridian Street, Suite 202
Bellingham, WA 98225 (US)

(21) Appl. No.: **12/326,565**

(22) Filed: **Dec. 2, 2008**

Related U.S. Application Data

(63) Continuation of application No. 10/966,848, filed on Oct. 14, 2004, which is a continuation of application No. 09/780,316, filed on Feb. 9, 2001, now abandoned, Continuation-in-part of application No. 08/944,529, filed on Oct. 7, 1997, now abandoned.

(60) Provisional application No. 60/181,577, filed on Feb. 10, 2000.

Publication Classification

(51) **Int. Cl.**
    A61B 5/08 (2006.01)

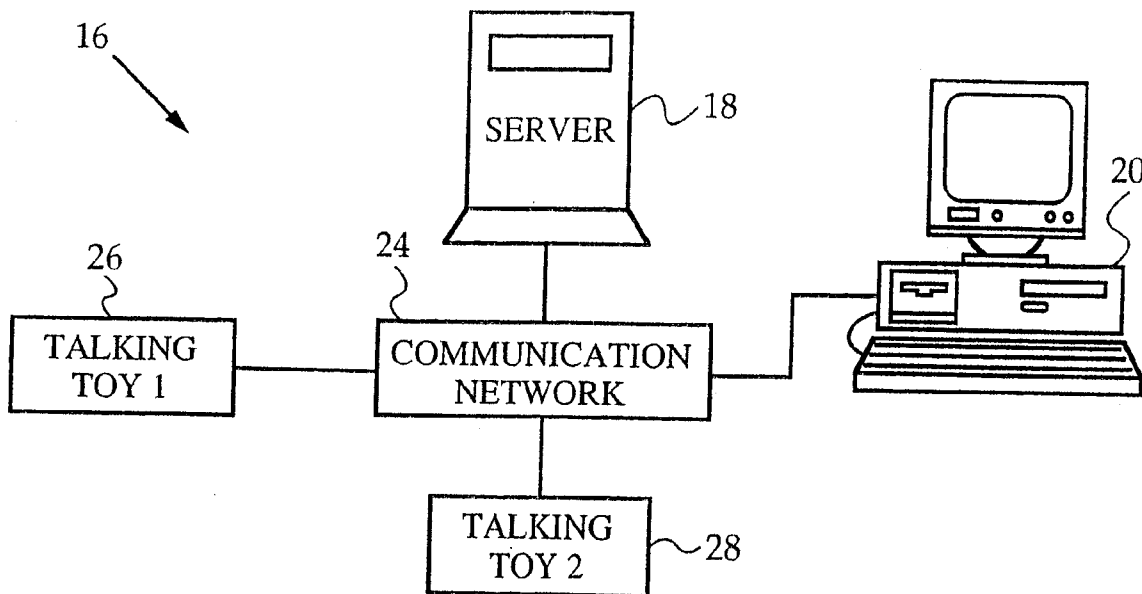(52) **U.S. Cl.** **600/529**

(57) **ABSTRACT**

A system for controlling a motion device. A motion control system comprises a motion control device, and a first user is associated with the motion control system. An output system is arranged to generate motion signals that may be perceived by the first user. A motion server system comprises a plurality of motion scripts associated with messages to be transmitted to the first user and a motion program web page. A processing system comprises a browser program compatible with the motion program web page. A communications system transfers data between the various systems. A second user controls the server system to form a motion message corresponding to a desired motion signal based on the at least one motion script stored by the server system. The motion control system causes the output system to generate the desired motion signal based on the motion message.

16

26 24 18 20

| TALKING TOY 1 | COMMUNICATION NETWORK | SERVER |

TALKING TOY 2 ~28

16

18

SERVER

20

26

TALKING
TOY 1

24

COMMUNICATION
NETWORK

TALKING
TOY 2

28

FIG. 1

18

SERVER

30   DATABASE

32   SCRIPT PROGRAMS

34   LOOK-UP TABLE

36   SOFTWARE
APPLICATION

38   SCRIPT GENERATOR

40   SCRIPT ASSIGNOR

20

REMOTE
INTERFACE

SCRIPT ENTRY   42

SCRIPT ASSIGN   44

26

TALKING TOY

ASSIGNED
SCRIPT PROGRAM

FIG. 2

FIG. 3

FIG. 4

SCRIPT ENTRY SCREEN

42

SCRIPT NAME:

DIABETES SCRIPT 1   86

ENTER PHRASES

88

PLEASE REMEMBER TO TAKE YOUR INSULIN ON TIME

REMEMBER TO MEASURE YOUR BLOOD SUGAR BEFORE BREAKFAST EACH MORNING

IT IS IMPORTANT TO EXERCISE EVERY DAY

MANAGING DIABETES IS TOUGH BUT YOU CAN DO IT

CREATE SCRIPT   90

CANCEL   92
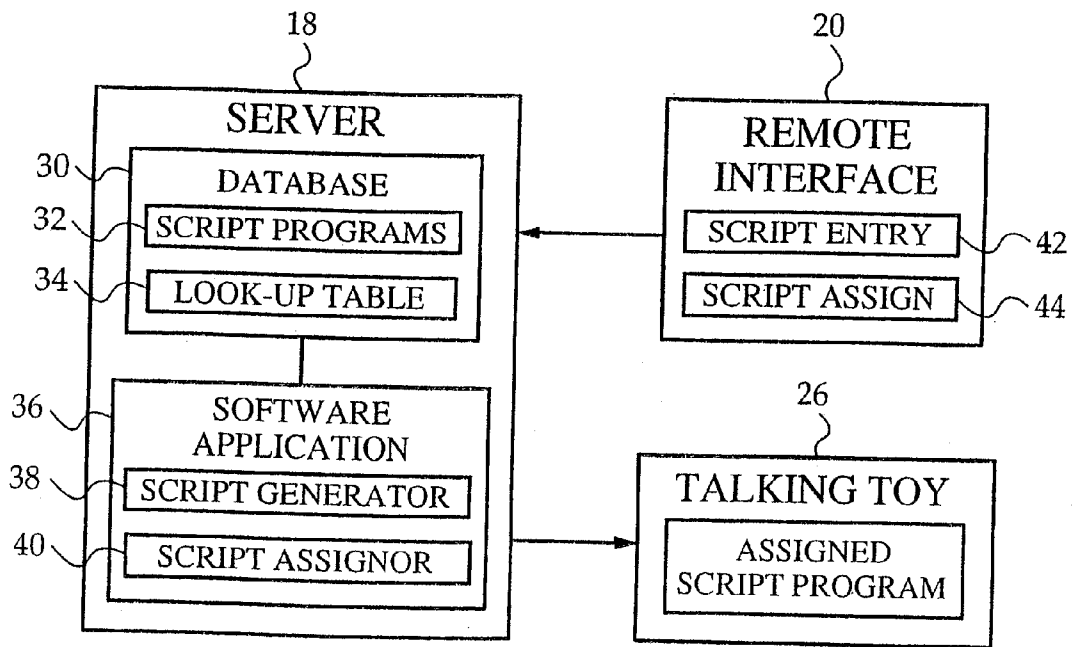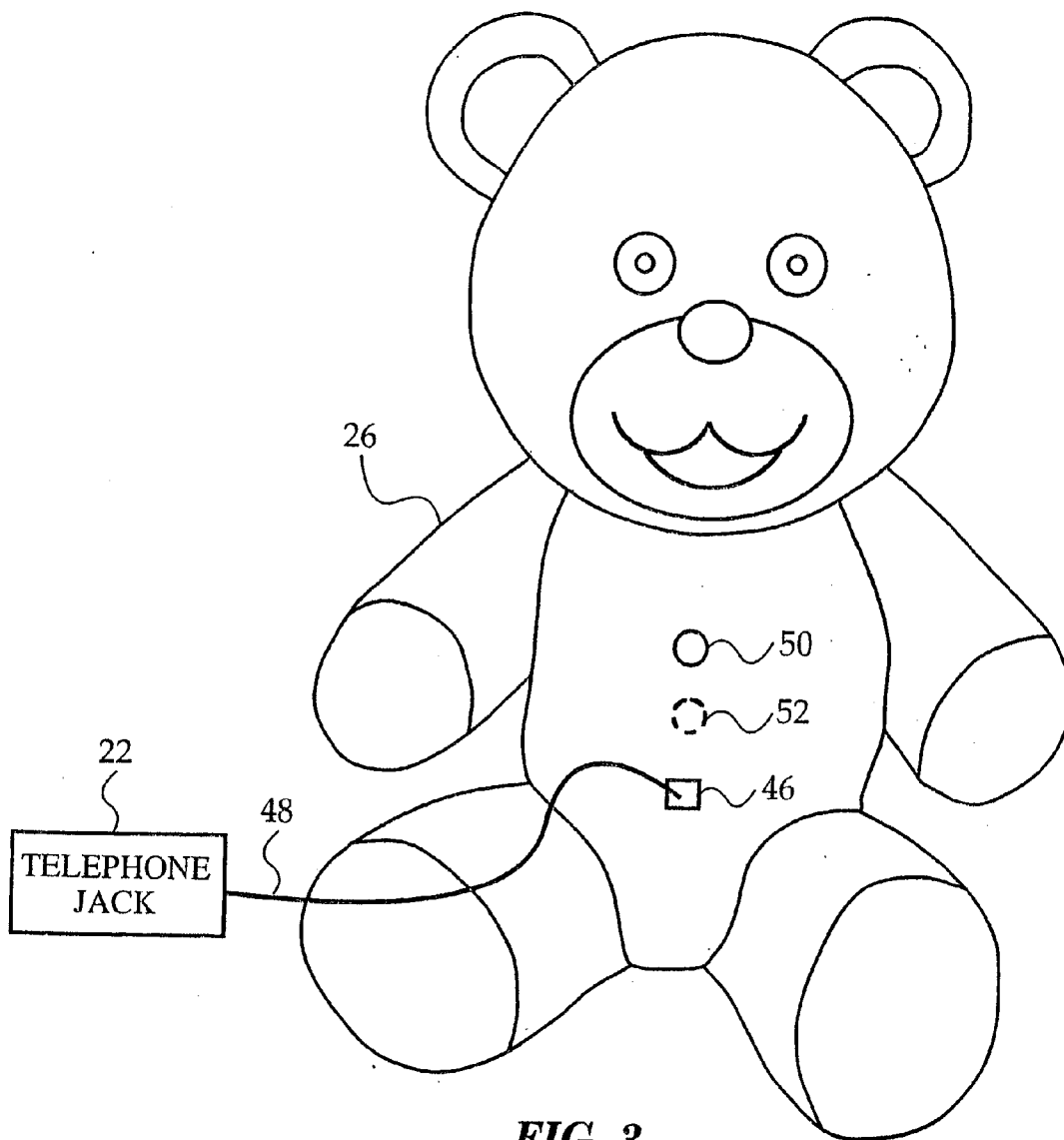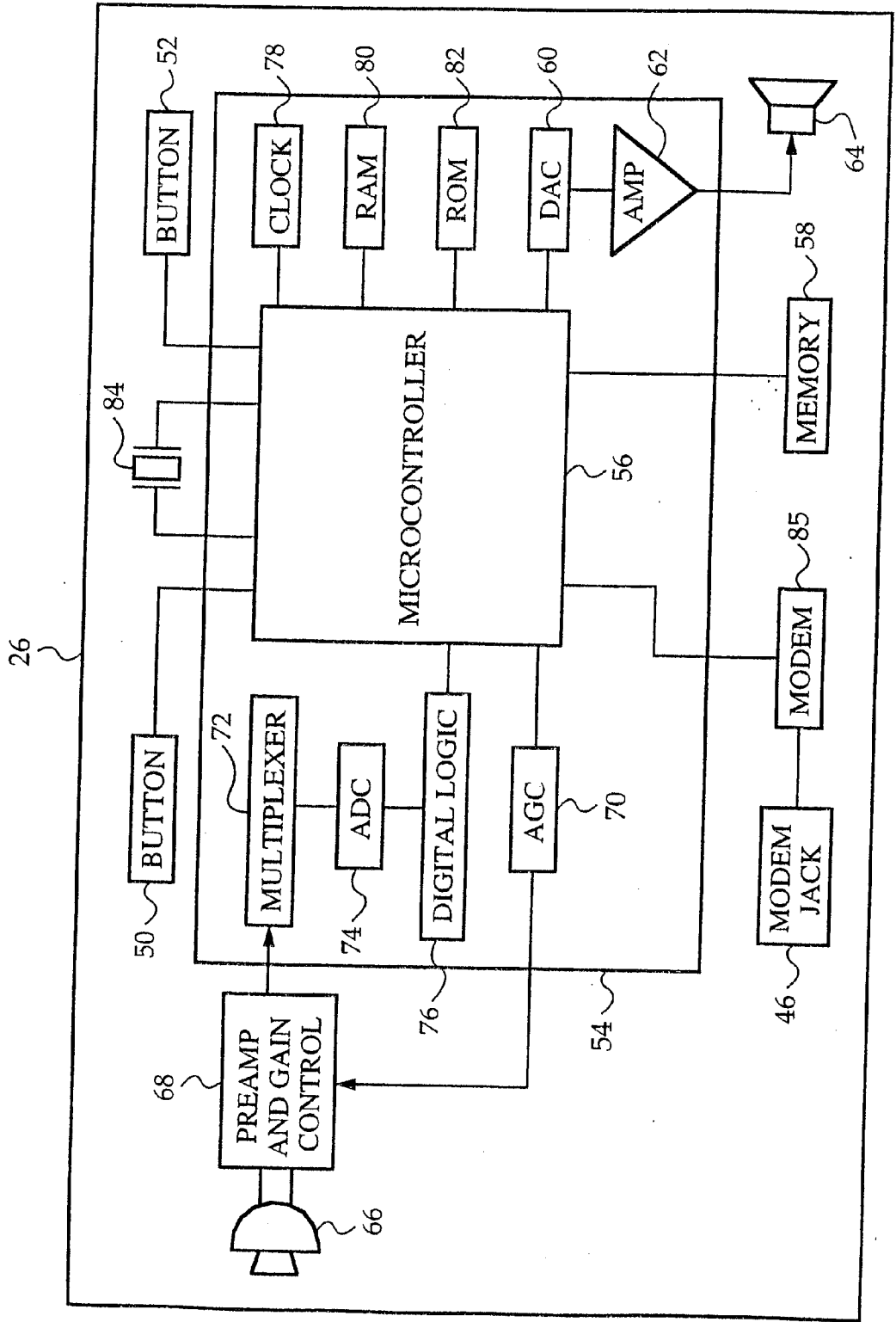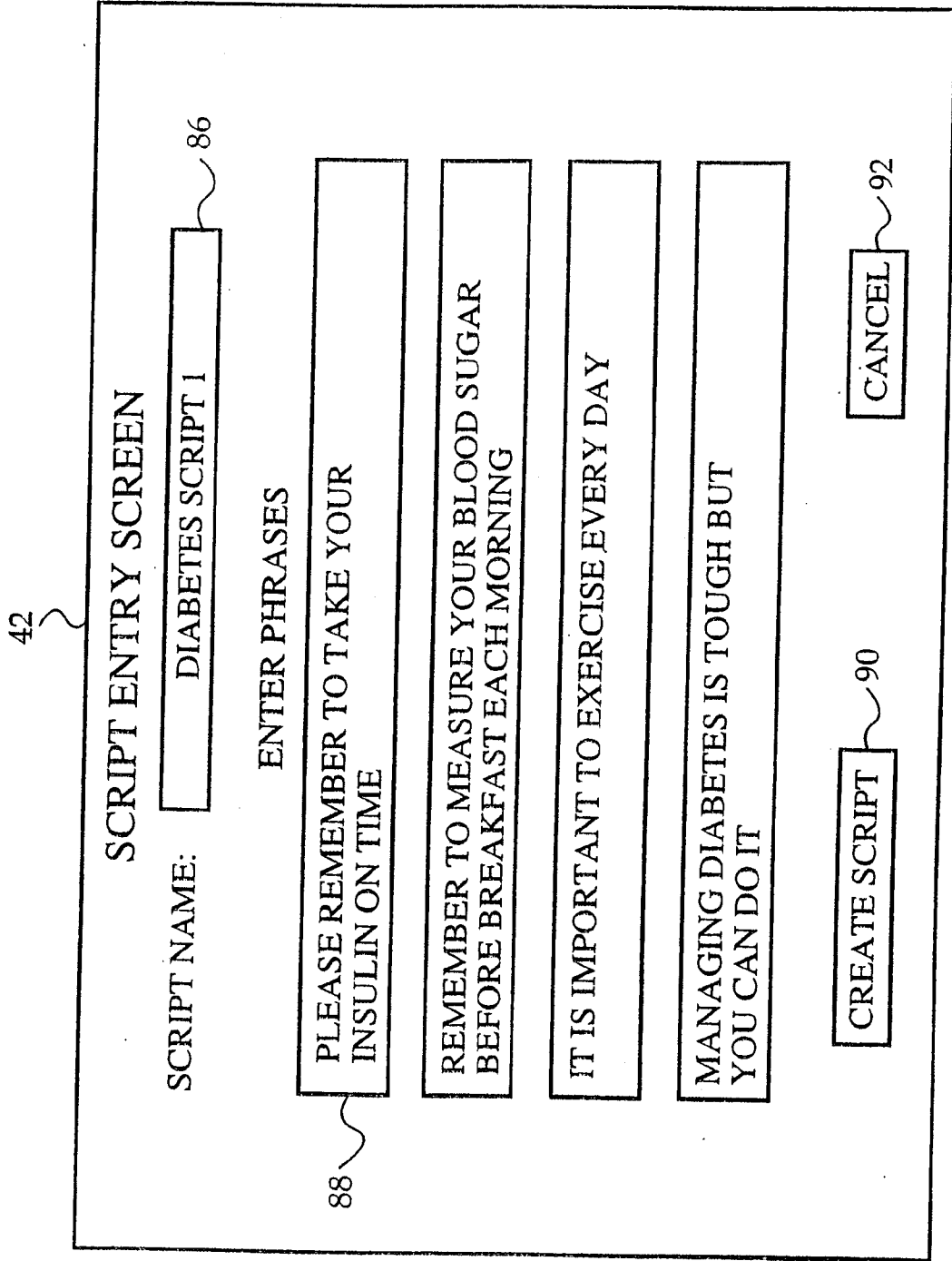
FIG. 5

SPEAK: SAY OK WHEN YOU ARE READY {LF}

RECOGNIZE: OK {LF}

SPEAK: PLEASE REMEMBER TO TAKE YOUR
        INSULIN ON TIME {LF}

DELAY: 3 {LF}

SPEAK: REMEMBER TO MEASURE YOUR BLOOD SUGAR
        BEFORE BREAKFAST EACH MORNING {LF}

DELAY: 3 {LF}

SPEAK: IT IS IMPORTANT TO EXERCISE EVERY DAY {LF}

DELAY: 3 {LF}

SPEAK: MANAGING DIABETES IS TOUGH BUT
        YOU CAN DO IT {LF}

DELAY: 3 {LF}

SPEAK: THAT'S ALL THE NEWS FOR NOW {LF}

DELAY: 1{LF}

SPEAK: CONNECT ME TO THE TELEPHONE LINE
        TO GET NEW MESSAGES {LF}
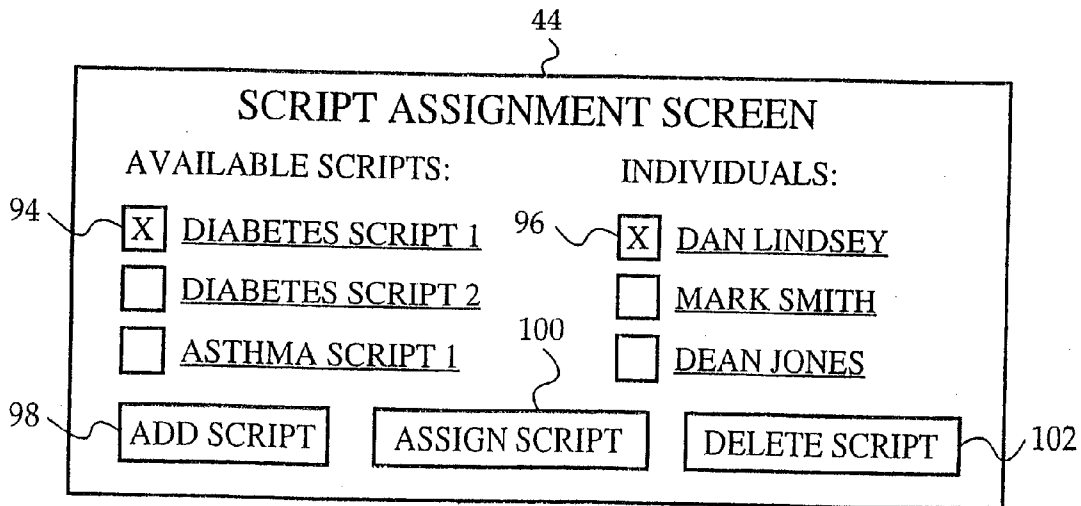{EOF}

FIG. 6

44

SCRIPT ASSIGNMENT SCREEN

AVAILABLE SCRIPTS: INDIVIDUALS:

94   [X] DIABETES SCRIPT 1   96   [X] DAN LINDSEY

        [ ] DIABETES SCRIPT 2   [ ] MARK SMITH

        [ ] ASTHMA SCRIPT 1   100   [ ] DEAN JONES

98   [ADD SCRIPT]   [ASSIGN SCRIPT]   [DELETE SCRIPT]   102

FIG. 7

FIG. 8

START
CONTROL
PROGRAM

302

USER
INPUT? — NO → SLEEP MODE 304

YES

306

SPEECH
REQUEST? — YES → EXECUTE
SCRIPT PROGRAM

308

NO

CONNECT
TO SERVER 310

312

FAILED
CONNECTION? — YES → PROMPT FOR
PHONE LINE
CONNECTION → RECEIVE
REPLY

314

316

NO

318 — TRANSMIT
UNIQUE ID

320 — RECEIVE AND
STORE NEW
SCRIPT PROGRAM

FIG. 9

START SCRIPT
PROGRAM

402 PROMPT INDIVIDUAL

404 RECEIVE REPLY

406 SYNTHESIZE PHRASES

408 PROMPT FOR PHONE LINE
CONNECTION

END SCRIPT
PROGRAM

FIG. 10

SERVER 18

30 DATABASE
31 GENERIC SCRIPTS
33 CUSTOM SCRIPTS
34 LOOK-UP TABLE

26

TALKING TOY

CUSTOM
SCRIPT PROGRAM

APPLICATION
38 SCRIPT GENERATOR
40 SCRIPT ASSIGNOR
41 MERGE PROGRAM

FIG. 11

SCRIPT ENTRY SCREEN

42

SCRIPT NAME: GENERIC SCRIPT 2
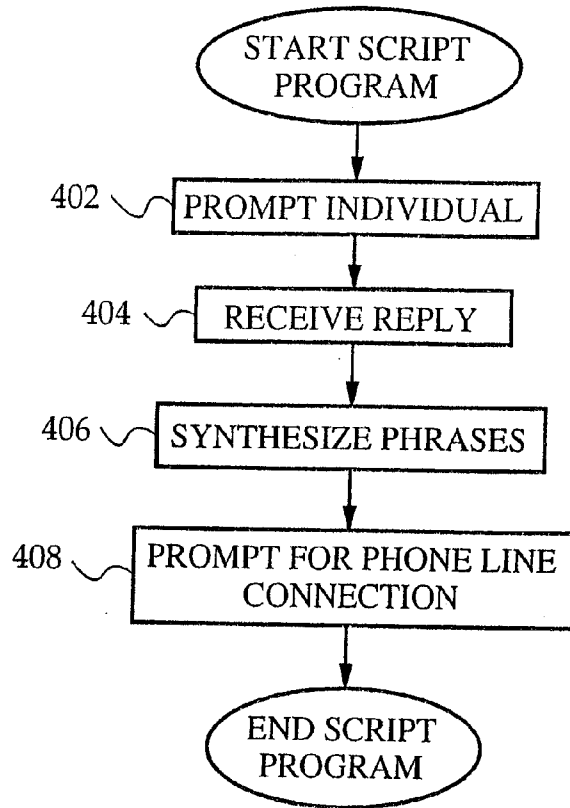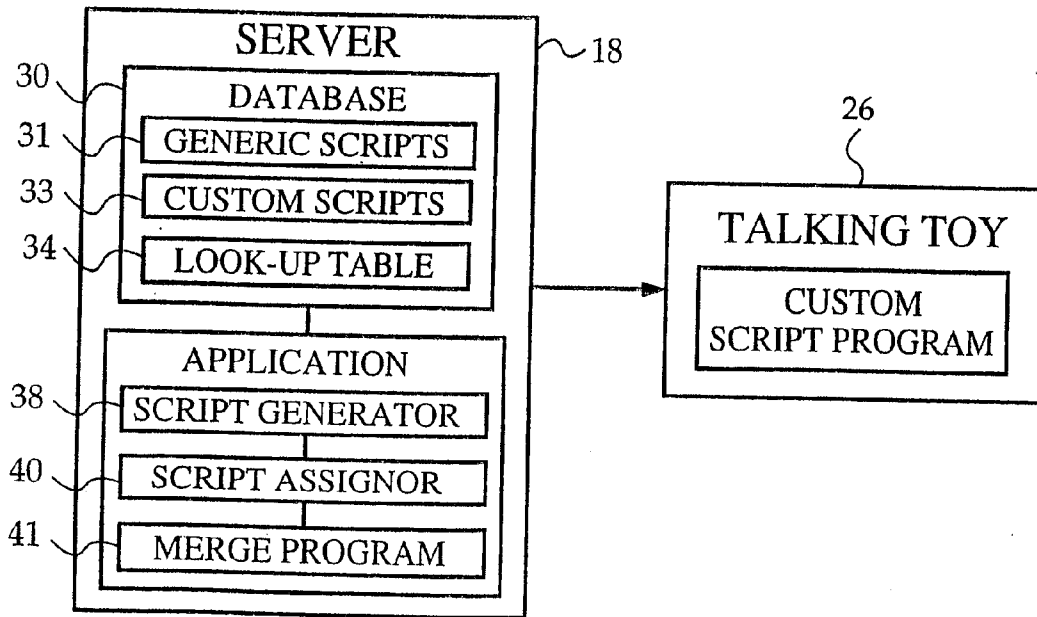
86

ENTER PHRASES

88

<<INSERT PATIENT_NAME>> PLEASE REMEMBER
TO TAKE YOUR <<INSERT MEDICATION_NAME>> ON TIME

<<INSERT PATIENT_NAME>> REMEMBER TO MEASURE YOUR
<<INSERT PHYSIOLOGICAL_PARAMETER>> EACH DAY

MANAGING <<INSERT DISEASE_NAME>> IS TOUGH BUT
YOU CAN DO IT <<INSERT PATIENT_NAME>>

CREATE SCRIPT   90

CANCEL   92

FIG. 12

SPEAK: SAY OK WHEN YOU ARE READY {LF}

RECOGNIZE: OK {LF}

SPEAK: <<INSERT PATIENT_NAME>> PLEASE REMEMBER TO TAKE
      YOUR <<INSERT MEDICATION_NAME>> ON TIME {LF}

DELAY: 3 {LF}

SPEAK: <<INSERT PATIENT_NAME>> REMEMBER TO MEASURE YOUR
      <<INSERT MEASURED_PARAMETER>> EACH DAY {LF}

DELAY: 3 {LF}

SPEAK: MANAGING <<INSERT DISEASE_NAME>> IS TOUGH
      BUT YOU CAN DO IT <<INSERT PATIENT_NAME>> {LF}

DELAY: 3 {LF}

SPEAK: THAT'S ALL THE NEWS FOR NOW {LF}

DELAY: 1{LF}

SPEAK: CONNECT ME TO THE TELEPHONE LINE
      TO GET NEW MESSAGES {LF}
{EOF}

FIG. 13

SPEAK: SAY OK WHEN YOU ARE READY {LF}

RECOGNIZE: OK {LF}

SPEAK: DAN PLEASE REMEMBER TO TAKE
      YOUR INSULIN ON TIME {LF}

DELAY: 3 {LF}

SPEAK: DAN REMEMBER TO MEASURE YOUR
      BLOOD SUGAR EACH DAY {LF}

DELAY: 3 {LF}

SPEAK: MANAGING DIABETES IS TOUGH
      BUT YOU CAN DO IT DAN {LF}

DELAY: 3 {LF}

SPEAK: THAT'S ALL THE NEWS FOR NOW {LF}

DELAY: 1 {LF}

SPEAK: CONNECT ME TO THE TELEPHONE LINE
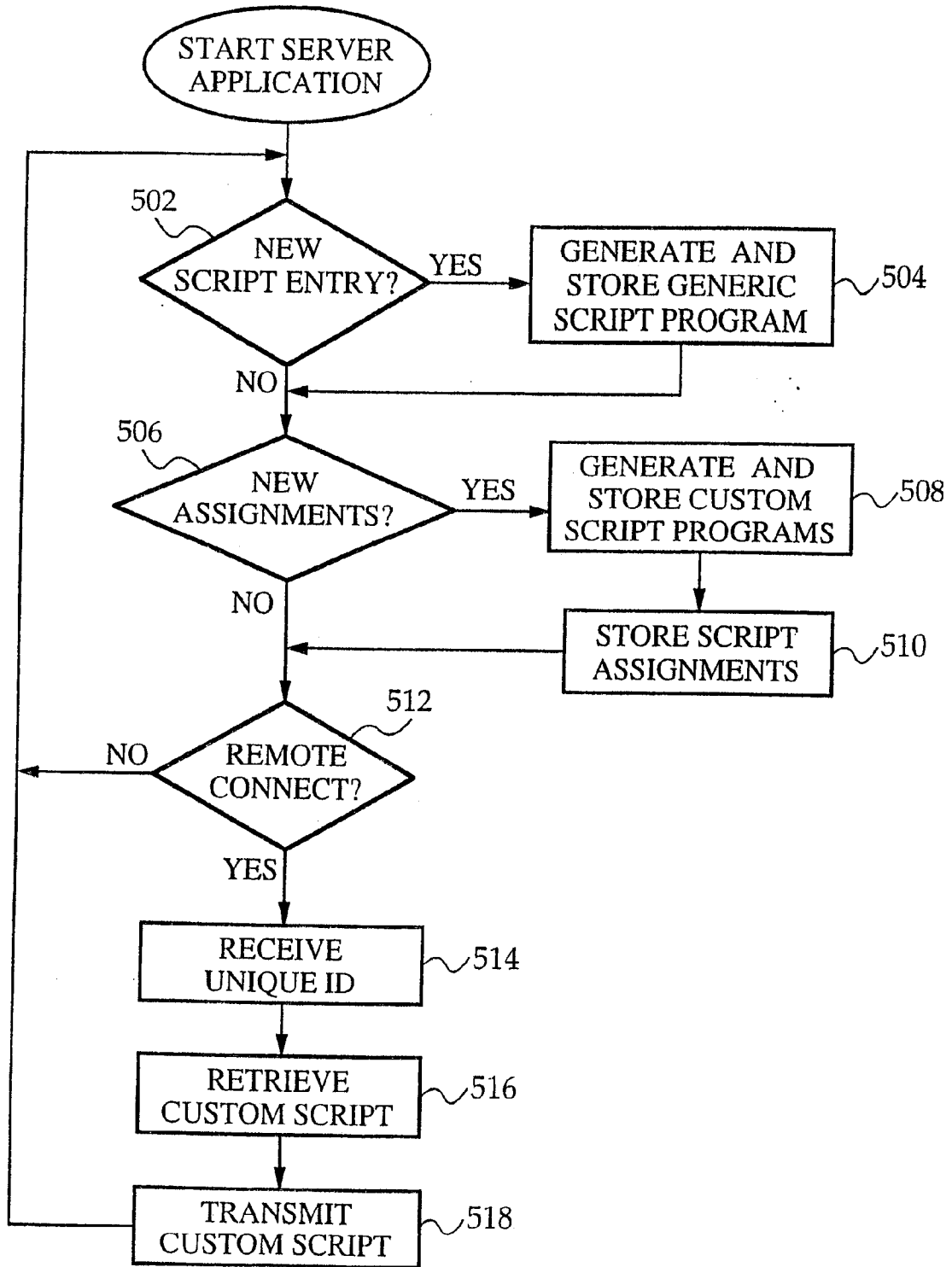   TO GET NEW MESSAGES {LF}
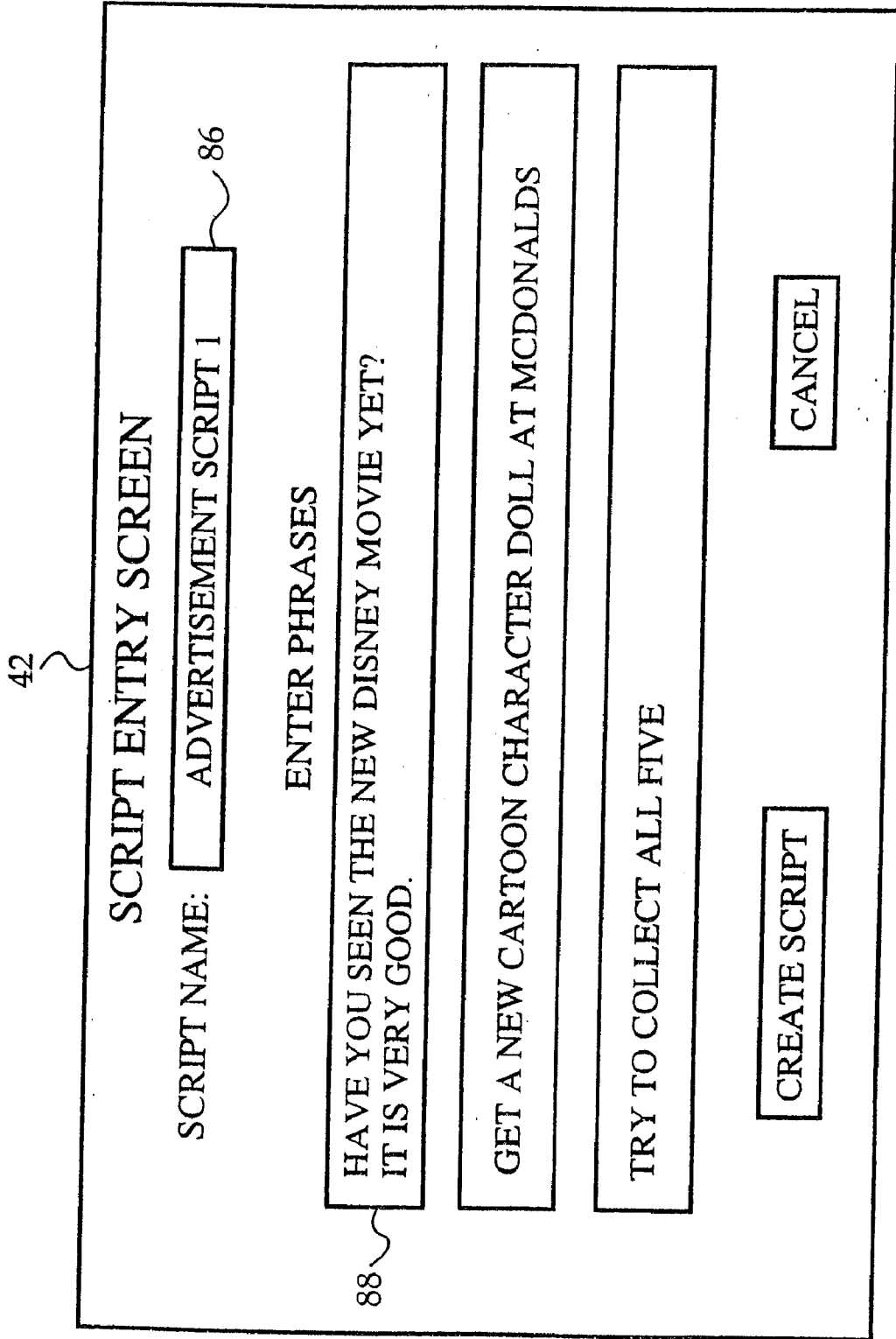{EOF}

FIG. 14

FIG. 15

SCRIPT ENTRY SCREEN

42

SCRIPT NAME: ADVERTISEMENT SCRIPT 1   86

ENTER PHRASES

HAVE YOU SEEN THE NEW DISNEY MOVIE YET?
IT IS VERY GOOD.

88

GET A NEW CARTOON CHARACTER DOLL AT MCDONALDS

TRY TO COLLECT ALL FIVE

CREATE SCRIPT

CANCEL

FIG. 16

SCRIPT ENTRY SCREEN

42

SCRIPT NAME:

DAN LINDSEY'S SCRIPT   86

ENTER PHRASES

HI DAN, REMEMBER TO TAKE YOUR MEDICINE

88

IT IS TIME TO BRUSH YOUR TEETH

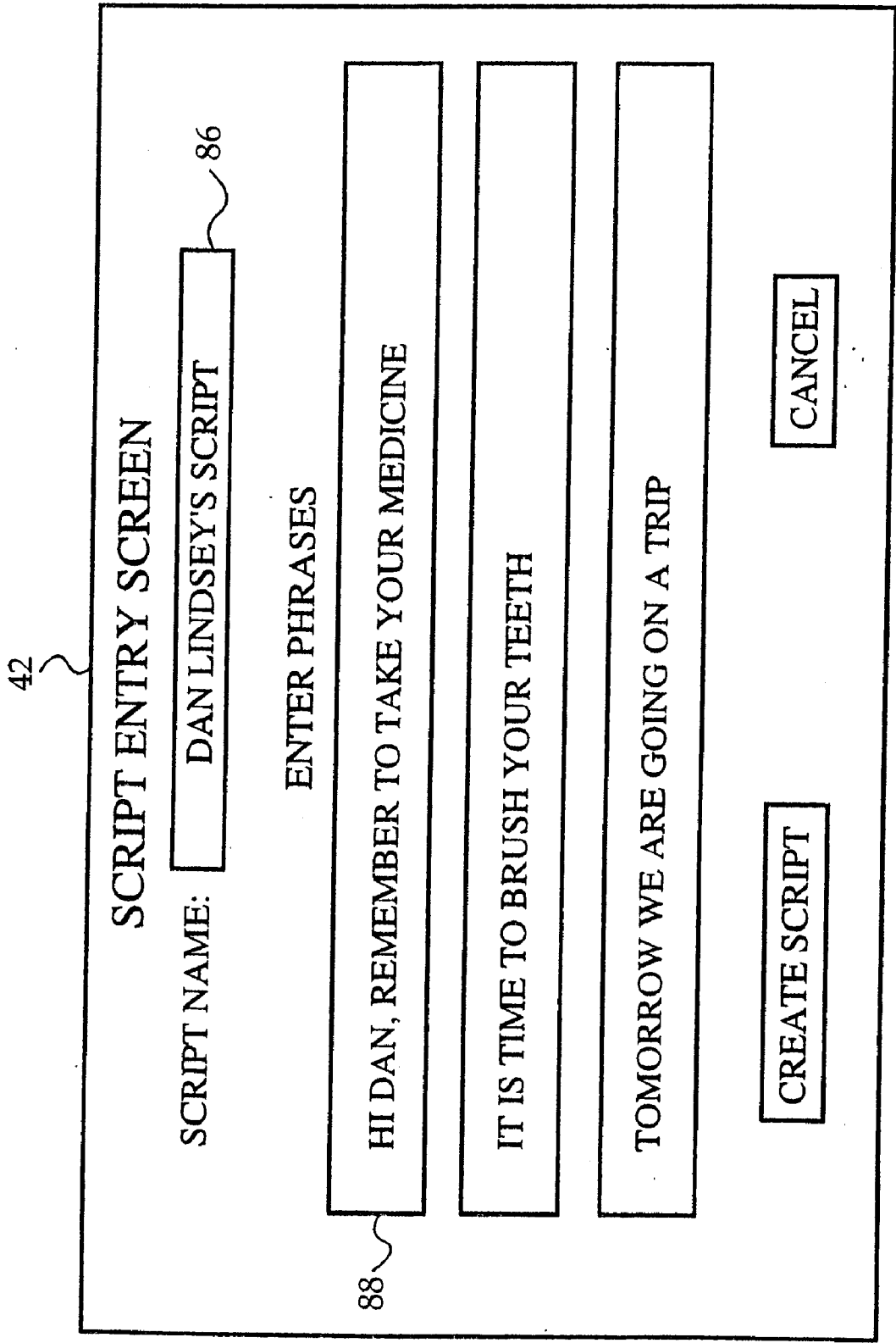TOMORROW WE ARE GOING ON A TRIP

CREATE SCRIPT

CANCEL

FIG. 17

SYSTEM AND/OR METHOD FOR INITIATING A MEDICAL TASK INVOLVING MOTION WITH A DEVICE

RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 10/966,848, filed Jan. 28, 2005, which is a continuation of U.S. patent application Ser. No. 09/780,316 filed Feb. 9, 2001, now abandoned, which claims the benefit of Provisional Application Ser. No. 60/181,577, filed Feb. 10, 2000, each of which is incorporated by reference in its entirety. This application is also a continuation-in-part of U.S. patent application Ser. No. 08/944,529, filed Oct. 7, 1997, now abandoned, which is incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] The present invention relates to control systems for programmable devices and, more particularly, to the generation and distribution of control commands that control the operation of programmable devices.

BACKGROUND OF THE INVENTION

[0003] A wide variety of devices contain a combination of software and hardware that control the operation of the device. These devices will be referred to herein as programmable devices. Programmable devices include a wide variety of items such as toys, industrial motion control systems, exercise equipment, medical devices, household appliances, HVAC systems, and the like.

[0004] A common characteristic of such programmable devices is that they are programmed to perform a limited number of predetermined tasks. For example, a toy may be programmed to speak, move, or react to external stimulation in a predetermined manner. An industrial motion control system is programmed to assemble parts in a precise, repetitive manner. A household appliance may be programmed to perform one or more cooking or cleaning tasks. An HVAC system will be programmed to control a heating element and heat distribution systems to obtain a desired air temperature.

[0005] Some programmable devices contain means for allowing the end user to control the functionality of the system to a limited degree. In the context of a toy, the end user may operate a switch or joystick to select a manner of movement. An HVAC system will normally allow the end user to set the desired temperature. In most cases, however, the input of the end user is limited to changing variables or selecting from among a plurality of stand-alone programs.

[0006] Programmable devices thus take many forms but have certain common characteristics. A programmable device includes some form of memory for storing control commands that define a predetermined command program. The command program may accept input from the user or contain discrete sub-programs from which the end user may select, but the end user may not modify the command program.

[0007] A programmable device further comprises a processor capable of executing the command program and generating control signals. To reduce manufacturing costs, the processor is normally an inexpensive dedicated processor with relatively limited capabilities and resources.

[0008] A programmable device will also comprise control hardware that performs a desired task as defined by the control signals. The control hardware can be as simple as an LED or speaker that generates light or sound or as complicated as a multi-axis industrial motion control device that performs a complex welding procedure.

[0009] The relevance of the present invention is particularly significant given the varying degrees of technical skill possessed by the various patient end users involved in the design, manufacturing, and use of a typical programmable device. The user of a programmable device must be assumed to have little or no capability to create the command programs necessary to operate a programmable device. Certainly a typical child using a toy will not have the skills necessary to create command program for that toy. Even a highly trained technician operating an industrial motion control system typically will likely not have the skill to program the system to perform a desired task.

[0010] Accordingly, in this application the term "end user" will refer to a person who uses a programmable device but cannot be assumed to have the expertise to create a command program for that programmable device.

[0011] In contrast, the term "programmer" will be used herein to refer to a person having the expertise to create a command program for a particular programmable device. The skill level and background of the programmer will vary depending upon the specific programmable device; the term programmer is thus not intended to define a particular level of expertise, but is instead defined in relation to the specific programmable device.

[0012] With some programmable devices, the programmer has no direct contact with the end user. For example, a programmer of a toy or household appliance will typically not have direct contact with the end user. A programmer of an HVAC system or industrial motion control system may, on the other hand, have contact with the end user.

[0013] Without direct contact with the end user, the programmer must anticipate what task the end user will desire of the programmable device. Even with direct contact, the programmer may not fully comprehend the desired task, or the desired task may change after the command program has been created. In either case, obtaining the services of the programmer to modify the command program is likely to be difficult and expensive, if not impossible.

[0014] In general, while the end user may not be able to create a command program, the end user will be able to define the desired task. A technician operating an industrial motion control system will likely be able to observe that a change in the operation of the system will increase product yield or speed up the manufacturing process. Even a child might be able to determine that a doll that walks should also be able to jump.

[0015] The term "end user" may include any other person involved with a programmable device without the technical expertise to qualify as a programmer of that device. For example, a medical device may be used by a patient and controlled by a caregiver, neither of which would have the expertise to be considered a programmer; both the patient and the caregiver would be considered end users in the present application.

[0016] The purpose of the present invention is to facilitate the generation and distribution of command programs for programmable devices. In particular, the present invention is designed to allow an end user of a particular programmable device to define a desired task, interact with a remote computer over a communications network to generate a command

program, and then download the command program into the programmable device over the communications network.

SUMMARY OF THE INVENTION

[0017] The present invention may be embodied as a system for controlling a motion device. A motion control system comprises a motion control device, and a first user is associated with the motion control system. An output system is arranged to generate motion signals that may be perceived by the first user. A motion server system comprises a plurality of motion scripts associated with messages to be transmitted to the first user and a motion program web page. A processing system comprises a browser program compatible with the motion program web page. A communications system transfers data between the various systems. A second user controls the server system to form a motion message corresponding to a desired motion signal based on the at least one motion script stored by the server system. The motion control system causes the output system to generate the desired motion signal based on the motion message.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] FIG. 1 is a block diagram of a networked system according to a preferred embodiment of the invention.

[0019] FIG. 2 is a block diagram illustrating the interaction of the components of the system of FIG. 1.

[0020] FIG. 3 is a perspective view of a remotely programmable talking toy of the system of FIG. 1.

[0021] FIG. 4 is a block diagram illustrating the components of the talking toy of FIG. 3.

[0022] FIG. 5 is a script entry screen according to the preferred embodiment of the invention.

[0023] FIG. 6 is a listing of a sample script program according to the preferred embodiment of the invention.

[0024] FIG. 7 is a script assignment screen according to the preferred embodiment of the invention.

[0025] FIG. 8 is a flow chart illustrating the steps included in a software application executed by the server of FIG. 1 according to the preferred embodiment of the invention.

[0026] FIG. 9 is a flow chart illustrating the steps included in a control program executed by the talking toy of FIG. 3 according to the preferred embodiment of the invention.

[0027] FIG. 10 is a flow chart illustrating the steps included in the script program of FIG. 6.

[0028] FIG. 11 is a block diagram illustrating the interaction of the server of FIG. 1 with the talking toy of FIG. 3 according to a second embodiment of the invention.

[0029] FIG. 12 is a script entry screen according to the second embodiment of the invention.

[0030] FIG. 13 is a listing of a generic script program according to the second embodiment of the invention.

[0031] FIG. 14 is a listing of a custom script program according to the second embodiment of the invention.

[0032] FIG. 15 is a flow chart illustrating the steps included in a software application executed by the server of FIG. 1 according to the second embodiment of the invention.

[0033] FIG. 16 is a script entry screen according to an alternative embodiment of the invention.

[0034] FIG. 17 is a script entry screen according to another embodiment of the invention.

DETAILED DESCRIPTION

[0035] The present invention may be embodied in any programmable device. The present invention will be described below in the context of a toy that may be programmed to speak or move in a desired fashion. The present invention has application to other programmable devices, and the scope of the present invention should be determined by the claims appended hereto and not the following discussion.

[0036] The invention may be embodied as a networked system including one or more programmable toys that can be controlled to perform a desired task such as move and/or communicate messages to end users. In contrast to conventional programmable toys whose desired task is programmed during manufacture or through the insertion of external media, the programmable toys of the present invention are programmed remotely through the use of script programs. The script programs allow flexible and dynamic updating of the movement of or messages delivered by the toys, as well as convenient tailoring of toy movement and/or the communicated messages to the needs of particular end users.

[0037] In an example embodiment of the invention disclosed below, the end users as described above are patients and healthcare providers, and the programmable toys are remotely programmed to encourage healthy behavior in the patients. The terms "patient" and "health care provider" will be used below interchangeably with the term end user.

[0038] In the present exemplary embodiment, the programmable toys may be programmed to encourage children to take their medicine or to tolerate difficult healthcare regimens. The encouragement can take the form of a request audibly delivered to the patient in the form of speech and feedback in the form of movement when the request is followed.

[0039] As generally discussed throughout this application, the system of the present invention is equally well suited for purposes other than healthcare, such as industrial motion control systems, exercise equipment, HVAC systems, advertising, home appliances, education, entertainment, or any other application which involves the control of programmable devices to perform a desired task for an end user.

[0040] The preferred embodiment of the invention is illustrated in FIGS. 1-7. Referring to FIG. 1, a networked system 16 includes a server 18 and a workstation 20 connected to server 18 through a communication network 24. Server 18 is preferably a world wide web server and communication network 24 is preferably the Internet. It will be apparent to one skilled in the art that server 18 may comprise a single stand-alone computer or multiple computers distributed throughout a network. Workstation 20 is preferably a personal computer, remote terminal, or web TV unit connected to server 18 via the Internet. Workstation 20 functions as a remote interface for entering in server 18 the end task to be performed for the benefit of the end user.

[0041] System 16 also includes first and second programmable toys 26 and 28. Each programmable toy interacts with a patient end user in accordance with script programs received from server 18. Each programmable toy is connected to server 18 through communication network 24, preferably the Internet. Alternatively, the programmable toys may be placed in communication with server 18 via wireless communication networks, cellular networks, telephone networks, or any other network which allows each programmable toy to

3

exchange data with server **18**. For clarity of illustration, only two programmable toys are shown in FIG. **1**. It is to be understood that system **16** may include any number of programmable toys for communicating messages to any number of patient end users.

[0042] In general, a healthcare provider end user will operate the workstation **20**, a programmer will design and operate the server **18**, and the patient end user will use the toys **26** and **28**.

[0043] FIG. **2** shows server **18**, workstation **20**, and programmable toy **26** in greater detail. Server **18** includes a database **30** for storing script programs **32**. The script programs are executed by the programmable toys to communicate messages to the patients. Database **30** further includes a look-up table **34**. Table **34** contains a list of the patients who are to receive messages, and for each of the patient end user, a unique identification code and a respective pointer to the script program assigned to the end user. Each programmable toy is designed to execute assigned script programs which it receives from server **18**.

[0044] FIGS. **3-4** show the structure of each programmable toy according to the preferred embodiment. For clarity, only programmable toy **26** is illustrated since each programmable toy of the exemplary preferred embodiment has substantially identical structure to toy **26**. Referring to FIG. **3**, toy **26** is preferably embodied as a doll, such as a teddy bear. Alternatively, toy **26** may be embodied as an action figure, robot, or any other desired toy.

[0045] Toy **26** includes a modem jack **46** for connecting the toy to a telephone jack **22** through a connection cord **48**. Toy **26** also includes first and second user control buttons **50** and **52**. Button **50** is pressed to instruct the toy to execute a script program. Button **52** is pressed to instruct the toy to establish a communication link to the server and download a new script program. In alternative embodiments, the control buttons may be replaced by switches, keys, sensors, or any other type of interface suitable for receiving user input.

[0046] FIG. **4** is a schematic block diagram illustrating the internal components of toy **26**. Toy **26** includes an audio processor chip **54**, which is preferably an RSC-164 chip commercially available from Sensory Circuits Inc. of 1735 N. First Street, San Jose, Calif. 95112. Audio processor chip **54** has a microcontroller **56** for executing script programs received from the server. A memory **58** is connected to microcontroller **56**. Memory **58** stores the end user's unique identification code, script programs received from the server, and a script interpreter used by microcontroller **56** to execute the script programs.

[0047] The script interpreter translates script commands into the native processor code of microcontroller **56**. Specific techniques for translating and executing script commands in this manner are well known in the art. Memory **58** also stores a control program executed by microcontroller **56** to perform various control functions which are described in the operation section below. Memory **58** is preferably a non-volatile memory, such as a serial EEPROM.

[0048] Toy **26** also includes a modem **85** which is connected between microcontroller **56** and modem jack **46**. Modem **85** operates under the control of microcontroller **56** to establish communication links to the server through the communication network and to exchange data with the server. The data includes the end user's unique identification code which modem **85** transmits to the server, as well as assigned script programs which modem **85** receives from the server. Modem

85 is preferably a complete **28**. 8 K modem commercially available from Cermetek, although any suitable modem may be used.

[0049] Toy **26** further includes a speaker **64** and a microphone **66**. Audio processor chip **54** has built in speech synthesis functionality for audibly communicating messages and prompts to an end user through speaker **64**. For speech synthesis, chip **54** includes a digital to analog converter (DAC) **60** and an amplifier **62**. DAC **60** and amplifier **62** drive speaker **64** under the control of microcontroller **56** to communicate the messages and prompts.

[0050] Audio processor chip **54** also has built in speech recognition functionality for recognizing responses spoken into microphone **66**. Audio signals received through microphone **66** are converted to electrical signals and sent to a preamp and gain control circuit **68**. Circuit **68** is controlled by an automatic gain control circuit **70**, which is in turn controlled by microcontroller **56**. After being amplified by preamp **68**, the electrical signals enter chip **54** and pass to through a multiplexer **72** and an analog to digital converter (ADC) **74**. The resulting digital signals pass through a digital logic circuit **76** and enter microcontroller **56** for speech recognition.

[0051] Audio processor chip **54** also includes a RAM **80** for short term memory storage and a ROM **82** which stores audio sounds for speech synthesis and programs executed by microcontroller **56** to perform speech recognition and speech synthesis. Chip **54** operates at a clock speed determined by a crystal **84**. Chip **54** further includes a clock **78** which provides the current date and time to microcontroller **56**. Microcontroller **56** is also connected to control buttons **50** and **52** to receive user input. Toy **26** is preferably powered by one or more batteries (not shown). Alternatively, the toy may be powered by a standard wall outlet. Both methods for supplying power to a toy are well known in the art.

[0052] The toy **26** further comprises a motion system **79** that receives control signals from the microcontroller **56**. The motion system **79** can be similar to the motion system used in the "FURBY" doll; this system **79** allows the toy **26** to shake, move its hands and feet, and open and close its mouth and eyes. The motion system **79** is well-known in the art, but is conventionally preprogrammed at the factory for particular ranges and sequences of movement.

[0053] Referring again to FIG. **2**, server **18** includes a controlling software application **36** which is executed by server **18** to perform the various functions described below. The controlling software application **36** may be a system for generating a sequence of control commands based on an application program for motion control systems such as is disclosed in U.S. Pat. Nos. 5,867,385 and 5,691,897 to Brown et al., which are incorporated herein by reference.

[0054] The controlling software application **36** includes a script generator **38** and a script assignor **40**. Script generator **38** is designed to generate script programs **32** from script information entered through workstation **20**. The script programs **32** are a specific type of command program such as those typically executed by programmable devices. The script programs **32** contain the information necessary for the microcontroller **56** to cause the toy **26** to perform a desired task.

[0055] The script information is entered through a script entry screen **42**. In the preferred embodiment, script entry screen **42** is implemented as a web page on server **18**. Workstation **20** includes a web browser for accessing the web page to enter the script information.

[0056] FIG. 5 illustrates a sample script entry screen 42 as it appears on workstation 20. Screen 42 includes a script name field 86 for specifying the name of a script program to be generated. Screen 42 also includes entry fields 88 for entering information defining the desired task, such as a message containing instructions from the healthcare provider end user to be communicated to the patient end user and a movement to be performed when patient end user complies with the instructions.

[0057] FIG. 5 illustrates an exemplary set of statements which encourage the end user to comply with his or her diabetes care regimen. However, it is to be understood that any type of desired task may be entered in screen 42, including movement, sounds, or other messages such as advertisements, educational messages, and entertainment messages. Screen 42 further includes a CREATE SCRIPT button 90 for instructing the script generator to generate a script program from the information entered in screen 42. Screen 42 also includes a CANCEL button 92 for canceling the information entered.

[0058] In the preferred embodiment, each script program created by the script generator conforms to the standard file format used on UNIX systems. In the standard file format, each command is listed in the upper case and followed by a colon. Every line in the script program is terminated by a linefeed character {LF}, and only one command is placed on each line. The last character in the script program is a UNIX end of file character {EOF}. Table 1 shows an exemplary listing of script commands used in the preferred embodiment of the invention.

TABLE 1

SCRIPT COMMANDS

Command	Description
SPEAK: {words} {LF}	Synthesize the words following the SPEAK command.
RECOGNIZE: {word} {LF}	Recognize the word following the RECOGNIZE command.
DELAY: t {LF}	Wait a period of seconds specified by time parameter t.

[0059] The script commands illustrated in Table 1 are representative of the preferred embodiment and are not intended to limit the scope of the invention. After consideration of the ensuing description, it will be apparent to one skilled in the art many other suitable scripting languages and sets of script commands may be used to implement the invention.

[0060] Script generator 38 preferably stores a script program template which it uses to create each script program. To generate a script program, script generator 38 inserts into the template the information entered in screen 42. For example, FIG. 6 illustrates a sample script program created by the script generator from the script information shown in FIG. S. The script program includes speech commands to synthesize the phrases or statements entered in fields 88. The steps included in the script program are also shown in the flow chart of FIG. 10 and will be discussed in the operation section below.

[0061] Referring again to FIG. 2, script assignor 40 is for assigning script programs 32 to the patient end users. Script programs 32 are assigned in accordance with script assignment information entered through workstation 30. The script

assignment information is entered through a script assignment screen 44, which is preferably implemented as a web page on server 18.

[0062] FIG. 7 illustrates a sample script assignment screen 44 as it appears on workstation 20. Screen 44 includes check boxes 94 for selecting a script program to be assigned and check boxes 96 for selecting the patient end users to whom the script program is to be assigned. Screen 44 also includes an ASSIGN SCRIPT button 100 for entering the assignments. When button 100 is pressed, the script assignor creates and stores for each patient end user selected in check boxes 96 a respective pointer to the script program selected in check boxes 94. Each pointer is stored in the look-up table of the database. Screen 44 further includes an ADD SCRIPT button 98 for adding a new script program and a DELETE SCRIPT button 102 for deleting a script program.

[0063] The operation of the preferred embodiment is illustrated in FIGS. 1-10. FIG. 8 is a flow chart illustrating the steps included in the software application executed by server 18. In step 202, server 18 determines if new script information has been entered through script entry screen 42. If new script information has not been entered, server 18 proceeds to step 206. If new script information has been entered, server 18 proceeds to step 204.

[0064] In the preferred embodiment, the script information is entered in server 18 by one or more healthcare provider end users, such as a physician or case manager assigned to the patient, as generally discussed above. Of course, any person desiring to communicate with the end users may be granted access to server to create and assign script programs.

[0065] Further, it is to be understood that the system may include any number of remote interfaces for entering script generation and script assignment information in server 18. In a toy created for entertainment rather than healthcare purposes, a child may log on to the server 18, custom design a script program, and download the program into the toy.

[0066] As shown in FIG. 5, the script information specifies a desired task, such as a message containing a set of statements or phrases, to be communicated to one or more patient end users. The desired task may further comprise movements selected and/or entered in a similar manner.

[0067] In step 204, the script generator 38 generates a script program from the information entered in screen 42. The script program is stored in database 30. Steps 202 and 204 are preferably repeated to generate multiple script programs, e.g. a script program for diabetes patients, a script program for asthma patients, etc. Each script program corresponds to a respective one of the sets of statements entered through script entry screen 42. In step 206, the server 18 determines if new script assignment has been entered through assignment screen 44. If new script assignment information has not been entered, server 18 proceeds to step 210. If new script assignment information has been entered server 18 proceeds to step 208.

[0068] As shown in FIG. 7, the script assignment information is entered by the healthcare provider end user by selecting a desired script program through check boxes 94, selecting the patient end users to whom the selected script program is to be assigned through check boxes 96, and pressing the ASSIGN SCRIPT button 100. When button 100 is pressed, script assignor 40 creates for each end user selected in check boxes 96 a respective pointer to the script program selected in check boxes 94. In step 208, each pointer is stored in look-up

table **34** of database **30**. In step **210**, server **18** determines if any one of the programmable toys is remotely connected to the server.

[0069] Each patient end user is preferably provided with his or her own programmable toy which has the end user's unique identification code stored therein. Each patient end user is thus uniquely associated with a respective one of the programmable toys. If none of the programmable toys is connected, server **18** returns to step **202**. If a programmable toy is connected, server **18** receives from the programmable toy the patient end user's unique identification code to retrieve from table **34** the pointer to the script program assigned to the patient end user. In step **214**, server **18** retrieves the assigned script program from database **30**. In step **216**, server **18** transmits the assigned script program to the patient end user's programmable toy through communication network **24**. Following step **216**, the server returns to step **202**.

[0070] Each programmable toy is initially programmed with its user's unique identification code, the script interpreter used by the toy to interpret and execute script program commands, and a control program executed by the toy to control its overall operation. The initial programming may be achieved during manufacture or during an initial connection to server **18**. FIG. **9** illustrates the steps included in the control program executed by microcontroller **56** of programmable toy **26**.

[0071] In step **302**, microcontroller **56** determines if any user input has been received. In the preferred embodiment, user input is received through control buttons **50** and **52**. Control button **50** is pressed to instruct the programmable toy to move and/or speak, and control button **52** is pressed to instruct the toy to connect to the server and download a new script program. If no user input is received for a predetermined period of time, such as two minutes, toy **26** enters sleep mode in step **304**. The sleep mode conserves battery power while the toy is not in use. Following step **304**, microcontroller **56** returns to step **302** and awaits user input.

[0072] If user input has been received, microcontroller **56** determines if the input is a task request, step **306**. If the user has pressed control button **50**, if microcontroller **56** executes the script program last received from the server, step **308**. The steps included in a sample script program are shown in the flow chart of FIG. **10** and will be discussed below. Following step **308**, microcontroller **56** returns to step **302** and awaits new user input.

[0073] If the user presses control button **52** requesting a connection to the server, microcontroller **56** attempts to establish a communication link to the server through modem **85** and communication network **24**, step **310**. In step **312**, microcontroller determines if the connection was successful. If the connection failed, the user is prompted to connect toy **26** to telephone jack **22** in step **314**. Microcontroller **56** preferably prompts the user by synthesizing the phrase "PLEASE CONNECT ME TO THE TELEPHONE JACK USING THE CONNECTION CORD AND SAY 'DONE' WHEN YOU HAVE FINISHED."

[0074] In step **316**, microcontroller **56** waits until the appropriate reply is received through microphone **66**. Upon recognizing the reply 'DONE', microcontroller **56** repeats step **310** to get a successful connection to the server. Once a successful connection is established, microcontroller **56** transmits the unique identification code stored in memory **58** to server **18** in step **318**.

[0075] In step **320**, microcontroller **56** receives a new script program from the server through communication network **24** and modem **85**. The new script program is stored in memory **58** for subsequent execution by microcontroller **56**. Following step **320**, microcontroller **56** returns to step **302** and awaits new user input. FIG. **10** is a flow chart illustrating the steps included in a sample script program executed by microcontroller **56**. In step **402**, microcontroller **56** prompts the user by synthesizing through speaker **64** "SAY 'OK' WHEN YOU ARE READY". In step **404**, microcontroller **56** waits until a reply to the prompt is received through the microphone **66**. When the reply 'OK' is recognized, microcontroller **55** proceeds to step **406**. If no reply is received within a predetermined period of time, such as two minutes, toy **26** preferably enters sleep mode until it is reactivated by pressing one of the control buttons.

[0076] In step **406**, microcontroller **56** executes successive speech commands to synthesize through speaker **64** the phrases or statements specified in the script program. Referring again to FIG. **6**, the speech commands are preferably separated by delay commands which instruct microcontroller **56** to pause for a number of seconds between statements. The number of seconds is selected to allow the user sufficient time to absorb each statement. Alternatively, the user may be prompted to acknowledge each statement before a subsequent statement is synthesized. For example, the script program may include commands which instruct microcontroller **56** to synthesize the phrase "SAY 'OK' WHEN YOU ARE READY TO HEAR THE NEXT STATEMENT." Upon recognizing the reply 'OK', microcontroller **56** proceeds to the next speech command in the script program. Movement commands are processed for execution by the motion system in a similar manner.

[0077] In step **408**, the user is reminded to connect toy **26** to telephone jack **22** to download a new script program. Microcontroller **56** synthesizes through speaker **64** "PLEASE CONNECT ME TO THE TELEPHONE JACK TO GET NEW MESSAGES." Following step **408**, the script program ends.

[0078] One advantage of the system of the present invention is that it allows each programmable toy to be programmed remotely through the use of script programs. This allows the task performed by each programmable toy to be tailored to the specific needs of a specific end user or group of end users. Moreover, each script program may be easily created, assigned, and downloaded by simply accessing a server through a communication network, such as the Internet. Thus, the invention provides a powerful, convenient, and inexpensive system for communicating messages to a large number of end users.

[0079] FIGS. **11-15** illustrate a second embodiment of the invention in which messages are further customized to each patient end user by merging personal data with the script programs, much like a standard mail merge application. Referring to FIG. **11**, personal data relating to each patient end user is preferably stored in look-up table **34** of database **30**. By way of example, the data may include each patient end user's name, the name of each patient end user's medication or disease, or any other desired data. As in the preferred embodiment, database **30** also stores generic script programs **31** created by script generator **38**.

[0080] In the second embodiment, server **18** includes a data merge program **41** for merging the data stored in table **34** with generic script programs **31**. Data merge program **41** is

designed to retrieve selected data from table **34** and to insert the data into statements in generic script programs **31**, thus creating custom script programs **33**. Each custom script program contains a message which is customized to a patient end user. For example, the message may be customized with the patient end user's name, medication name, disease name, etc.

[0081] The operation of the second embodiment is illustrated in FIGS. **11-15**. The operation of the second embodiment is similar to the operation of the preferred embodiment except that server **18** transmits custom script programs to each programmable toy rather than generic script programs. FIG. **15** is a flow chart illustrating the steps included in a software application executed by server **18** according to the second embodiment.

[0082] In step **502**, server **18** determines if new script information has been entered through script entry screen **42**. If new script information has not been entered, server **18** proceeds to step **506**. If new script information has been entered, server **18** proceeds to step **504**. As shown in FIG. **12**, the script information specifies a message, such as a set of statements or phrases, to be communicated to the patient end users. Each statement preferably includes one or more insert commands specifying data from table **34** to be inserted into the statement. The insert commands instruct data merge program **41** to retrieve the specified data from database **30** and to insert the data into the statement. For example, the first statement shown in FIG. **12** includes insert commands instructing the data merge program to insert a patient name and a medication name into the statement.

[0083] Following entry of the statements and insert commands, CREATE SCRIPT button **90** is pressed. When button **90** is pressed, script generator **38** generates a generic script program from the information entered in screen **42**, step **504**. A sample generic script program is illustrated in FIG. **13**. The generic script program includes speech commands to synthesize the statements entered in fields **88**. Each statement preferably includes one or more insert commands specifying data to be inserted into the script program. The generic script program is stored in database **30**.

[0084] In step **506**, server **18** determines if new script assignment information has been entered through assignment screen **44**. If new script assignment information has not been entered, server **18** proceeds to step **512**. If new script assignment information has been entered, server **18** proceeds to step **508**. As shown in FIG. **7**, the script assignment information is entered by selecting a desired script program through check boxes **94**, selecting the patient end users to whom the selected script program is to be assigned through check boxes **96**, and pressing the ASSIGN SCRIPT button **100**.

[0085] When button **100** is pressed, data merge program **41** creates a custom script program for each patient end user selected in check boxes **96**, step **508**. Each custom script program is preferably created by using the selected generic script program as a template. For each patient end user selected, data merge program **41** retrieves from database **30** the data specified in the insert commands. Next, data merge program **41** inserts the data into the appropriate statements in the generic script program to create a custom script program for the patient end user.

[0086] For example, FIG. **14** illustrates a custom script program created from the generic script program of FIG. **13**. Each custom script program is stored in database **30**.

[0087] As each custom script program is generated for a patient end user, script assignor **40** assigns the custom script program to the patient end user, step **510**. This is preferably accomplished by creating a pointer to the custom script program and storing the pointer with the patient end user's unique identification code in table **34**. In step **512**, server **18** determines if any one of the programmable toys is remotely connected to the server. If a programmable toy is connected, server **18** receives from the programmable toy the patient end user's unique identification code in step **514**.

[0088] Server **18** uses the received identification code to retrieve from table **34** the pointer to the custom script program assigned to the patient end user. In step **516**, server **18** retrieves the custom script program from database **30**. In step **518**, server **18** transmits the custom script program to the patient end user's programmable toy. The programmable toy receives and executes the script program in the same manner described in the preferred embodiment. The remaining operation of the second embodiment is analogous to the operation of the preferred embodiment described above.

[0089] Although it is presently preferred to generate a custom script program for each patient end user as soon as script assignment information is received for the patient end user, it is also possible to wait until the patient end user's programmable toy s connects to the server before generating the custom script program. This is accomplished by creating and storing a pointer to the generic script program assigned to the patient end user, as previously described in the preferred embodiment. When the patient end user's programmable toy connects to the server, the data merge program creates a custom script program for the patient end user from the generic script program assigned to the patient end user. The custom script program is then transmitted to the patient end user's programmable toy for execution.

[0090] Although the first and second embodiments focus on healthcare applications, the system of the present invention may be used for any messaging application. For example, the system is particularly well suited for advertising. In a third embodiment of the invention, an advertising service is provided with a remote interface to the server for creating and assigning script programs which contain advertising messages. As shown in FIG. **16**, each advertising message may be conveniently entered through script entry screen **42**, like the health-related messages of the preferred embodiment. The operation of the third embodiment is analogous to the operation of the preferred embodiment, except that the talking toys communicate advertising messages rather than health-related messages.

[0091] Of course, the system of the present invention has many other applications. Typically, the user of each programmable toy is a child. In a fourth embodiment of the invention, the child's parent or guardian is provided with a remote interface to the server for creating and assigning script programs which contain messages for the child. As shown in FIG. **17**, each message may be conveniently entered through script entry screen **42**. The operation of the fourth embodiment is analogous to the operation of the preferred embodiment, except that script information is entered in the server by a parent or guardian rather than a healthcare provider.

[0092] Alternatively, the child may be provided with a remote interface to the server to create and assign his or her own script programs. It should also be noted that script programs may be generated from information received from multiple sources, such as a healthcare provider, an advertiser, and a parent. In a fifth embodiment of the invention, the script entry screen includes a respective section for each of the

7

sources to enter a message to be communicated. Each of the sources is provided with a remote interface to the server and a password for accessing the script entry screen. After each source has entered one or more messages in the server, a script program is generated which contains a combination of health-related messages, advertisements, educational messages, or entertainment messages. The remaining operation of the fifth embodiment is analogous to the operation of the preferred embodiment described above.

[0093] Although the above description contains many specificities, these should not be construed as limitations on the scope of the invention but merely as illustrations of some of the presently preferred embodiments. Many other embodiments of the invention are possible. For example, the scripting language and script commands shown are representative of the preferred embodiment. It will be apparent to one skilled in the art many other scripting languages and specific script commands may be used to implement the invention.

[0094] Moreover, the programmable device need not be embodied as a doll. The toys may be embodied as action figures, robots, or any other type of toy. Further, each programmable toy need not include a control button for triggering speech output. In alternative embodiments, speech is triggered by other mechanisms, such as voice prompts, the absence of the user's voice, position sensitive sensors, switches, or the like. Specific techniques for triggering speech in a programmable toy are well known in the art. In addition, the system of the present invention is not limited to healthcare applications.

[0095] The system may be used in any application which involves the communication of messages, including advertising, education, or entertainment. Of course, various combinations of these applications are also possible. For example, messages from multiple sources may be combined to generate script programs which contain a combination of health-related messages, advertisements, or educational messages. Further, the system may include any number of remote interfaces for entering and assigning script programs, and any number of programmable toys for delivering messages.

[0096] More generally, the programmable device need not be a toy.

[0097] For example, the programmable device may be a handheld computing device adapted to control and monitor an athlete end user's performance during athletic training. The device may monitor the athlete end user's heart rate and lactose levels. The device may be remotely programmed by a trainer end user or automated system with instructions for training for a specific athletic event based on data collected by the handheld device.

[0098] The programmable device may be a house hold appliance such as a refrigerator that monitors its contents. A remote end user such as a service that delivers groceries could reprogram the refrigerator as necessary to reflect delivered goods.

[0099] The programmable device may also be an industrial motion control system such as a robotic welding machine. An engineer in charge of the product being welded may change its design and remotely generate a new command program for the robotic welding machine in response to the design changes.

[0100] The programmable device may be an HVAC system that communicates with a remote weather monitoring system that generates a new command program for the HVAC system in response to changes in the weather.

[0101] The embodiments of the present invention disclosed above are merely an illustrative, and not exhaustive, list of the environments in which the present invention may be applied. Therefore, the scope of the invention should be determined not by the examples given, but by the appended claims and their legal equivalents.

I claim:

1. A system comprising:
a server configured to generate a script configured to store information for performing a plurality of tasks;
a remote device configured to (i) receive said script, and (ii) initiate at least one of said tasks including a medical task involving motion in response to said information stored in said script;
a communications network configured to transmit said script from said server to said remote device; and
a user interface configured to receive information from a medical caregiver used by said server to generate said script.

2. The system according to claim 1, wherein said user interface is further configured to graphically display data received from said remote device after execution of said script.

3. The system according to claim 1, wherein said medical task comprises using one or more medical sensors to monitor a heart rate of a patient.

4. The system according to claim 1, wherein said medical task comprises using one or more medical sensors to measure a glucose level of a diabetes patient.

5. The system according to claim 1, wherein said medical task comprises using one or more medical sensors to measure a breathing peak flow of an asthma patient.

6. The system according to claim 1, wherein said medical task comprises using one or more medical sensors to monitor a lactose level of a patient.

7. The system according to claim 1, wherein said remote device is further configured to encourage healthy behavior of a patient.

8. The system according to claim 1, wherein said remote device is further configured to encourage a patient to take medication.

9. The system according to claim 1, wherein said remote device is further configured to encourage a patient to tolerate a difficult health regimen.

10. A system comprising:
a server configured to generate a script having one or more commands;
a toy configured to (i) receive said script, (ii) synthesize one or more audible words relating to a medical task involving motion in response to said commands stored in said script, (iii) initiate at least one of a plurality of motions in response to said one or more commands in said script, and (iv) confirm completion of said medical task by performing speech recognition on a response received from a microphone;
a communications network configured to transmit said script from said server to said toy; and
a user interface configured to generate said script in response to input by a user.

11. The system according to claim 10, wherein said toy is remotely programmable.

12. The system according to claim 10, wherein said toy is further configured to audibly deliver said medical task to a patient.

8

13. The system according to claim 12, wherein said medical task includes encouraging healthy behavior of said patient.

14. The system according to claim 12, wherein said medical task includes encouraging said patient to take medicine.

15. The system according to claim 12, wherein said medical task includes encouraging said patient to tolerate a difficult health regimen.

16. The system according to claim 12, wherein said toy provides feedback in the form of movement when said medical task is completed by said patient.

17. The system according to claim 12, wherein said initiated motion includes encouraging healthy behavior of said patient.

18. The system according to claim 12, wherein said initiated motion includes encouraging said patient to take medicine.

19. The system according to claim 12, wherein said initiated motion includes encouraging said patient to tolerate a difficult health regimen.

20. The system according to claim 10, wherein said user includes a medical caregiver.

21. The system according to claim 20, wherein said medical caregiver generates said script by providing a series of instructions.

22. The system according to claim 20, wherein said medical caregiver generates said script by clicking on a set of pre-defined icons each relating to a different medical task.

23. The system according to claim 10, wherein said response received by said microphone is converted into an audio signal.

* * * * *