may depend on the design of the component that is used. The speaker(s) **62** may be operatively coupled to a sound circuit **112**, that may comprise a voice- and sound-synthesis circuit or that may comprise a driver circuit. The sound-generating circuit **112** may be coupled to the I/O circuit **108**.

[**0063**] As shown in **FIG. 3**, the components **52**, **54**, **56**, **58**, **66**, **112** may be connected to the I/O circuit **108** via a respective direct line or conductor. Different connection schemes could be used. For example, one or more of the components shown in **FIG. 3** may be connected to the I/O circuit **108** via a common bus or other data link that is shared by a number of components. Furthermore, some of the components may be directly connected to the microprocessor **104** without passing through the I/O circuit **108**.

[**0064**] The graphics processor **107** may be a processor, such as a conventional graphics processor, configured to convert graphical primitives into display information that can be shown on the display unit **120**. Graphical primitives may include, for example, points, lines, polygons, etc. The graphics processor **107** may be a KYRO II® or KYRO III® graphics and video accelerator commercially available from STMICROELECTRONICS®, Inc, a GEFORCE2® graphics processing unit commercially available from NVIDIA® Corporation, a RADEON® 7000 graphics processor commercially available from ATI Technologies Inc., etc.

[**0065**] Although the microprocessor **104** and the graphics processor **107** are shown as separate devices in **FIG. 3**, it should be noted that such a representation is merely exemplary and that the functionality of both devices could be incorporated into a single device. For example, the microprocessor could perform some or all of the operations carried out by the graphics processor **107**.

3D Graphics Overview

[**0066**] In a system employing 3D graphics, a scene to be displayed may be composed in a 3D model space (also referred to herein as "3D graphics space"). The scene may include a plurality of 3D objects. These 3D objects may be composed of a plurality of geometric primitives that help define the object's surface. Such primitives may include, for instance, points, lines, polygons, etc. As an example, the OpenGL graphics application programming interface, promulgated by OpenGL Architecture Review Board, provides various types of primitives that may be employed such as points, lines, line strips, line loops, polygons, quadrilaterals, polygon strips, triangles, triangle strips, and triangle fans.

[**0067**] A 3D cube may comprise, for example, of six quadrilateral primitives, corresponding to the six surfaces of the cube. A polygon that is extensively used in typical 3D graphics systems is the triangle. Thus, as another example, the 3D cube may comprise 12 triangle primitives, where each of the six surfaces of the cube comprises two triangles. By using an appropriate number of primitives, such as triangles, objects may be made to appear round, spherical, tubular, etc. Complex objects, such as a human body, may be composed of numerous primitives.

[**0068**] Primitives in the 3D model space may be indicated by the coordinates of their vertices. For example, a point may be indicated by its 3D coordinates. Also, a line may be indicated by two sets of 3D coordinates corresponding to its two endpoints. Similarly, a triangle may be indicated by three sets of 3D coordinates corresponding to its three comers. Primitives in the 3D model space may also be indicated by a primitive type (e.g., point, line, triangle, quadrilateral, etc.). The 3D model space may be described using a rectangular coordinate system or another coordinate system such as a spherical coordinate system or a cylindrical coordinate system.

[**0069**] Attributes may be assigned to objects or primitives in the 3D model space. Attributes may include color, pattern, reflectance, transparency, translucency, animation, texture, etc. Textures may include smoothness, surface irregularities such as bumps, craters, etc., and may also include the mapping of text, an image, bitmap, animation, video, etc., onto the object or primitive.

[**0070**] **FIG. 4** is an illustration of an example of objects in a model space. The model space **100** includes a cube **102** on a plane **104**. The cube **102** comprises surface polygons **106**, **108**, and **110**. Model space **100** also includes a light source **114**, which causes cube **102** to cast a shadow **118** onto plane **104**. The shadow **118** may be represented as one or more 3D primitives having a darker shade. Additionally, polygon **110** is not illuminated by light source **114**, and therefore may appear shaded.

[**0071**] Model space **100** can be viewed from various viewpoints such as viewpoints **122**, **124**, and **126**. To display a depiction of model space **100** on a display unit, a view point may be chosen, and a depiction of the model space **100** as viewed from that viewpoint may be generated.

[**0072**] A typical technique for generating a depiction of the model space **100** from a particular viewpoint is to "project" the model space **100** onto a virtual display. For example, if the viewpoint is to be viewed on a 2D display such as a monitor, the model space **100** (or a portion thereof) may be "projected" onto a virtual 2D display. **FIGS. 5A and 5B** illustrate an example of objects **140** and **150** in 3D model space being "projected" onto a virtual 2D display **144**. Typically, the projection of an object in 3D model space onto a 2D virtual screen is implemented by "projecting" 3D primitives of which it is comprised. Typically, a mathematical transform is applied to the 3D coordinates of each primitive's vertices to generate the 2D coordinates of the projection's vertices on the virtual 2D display **144**. Thus, application of the mathematical transform may generate primitives in a 2D space (hereinafter "2D primitives"). Additional computations may be performed to generate the effects of light sources, if any, in 3D space on the color, luminance, etc., associated with the generated 2D primitives. These mathematical transformations may be included in a 3D graphics processing step typically known to those of ordinary skill in the art as "Transform and Lighting."**FIG. 8B** illustrates the projection of cube **140** onto the virtual display **144**. In this example, the projection comprises two polygons **146** and **148**. Because the virtual display **144** is a 2D space, polygons **146** and **148** can be indicated by sets of 2D coordinates.

[**0073**] The Transform and Lighting processing step generally may produce data including information relating to 2D primitives. This data relating to a 2D primitive may include the 2D coordinates of its vertices, the primitive's color, luminance, etc.

[**0074**] The data may also include depth information of the primitive in terms of the 3D model space and the viewpoint.