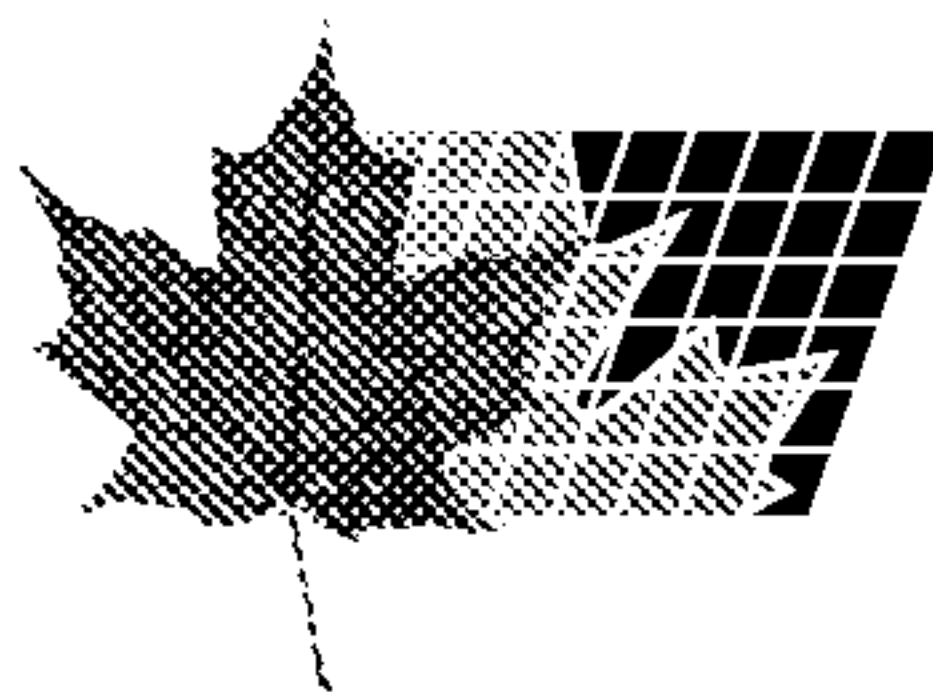




(72) ASSA, STEVEN BRENT, GB

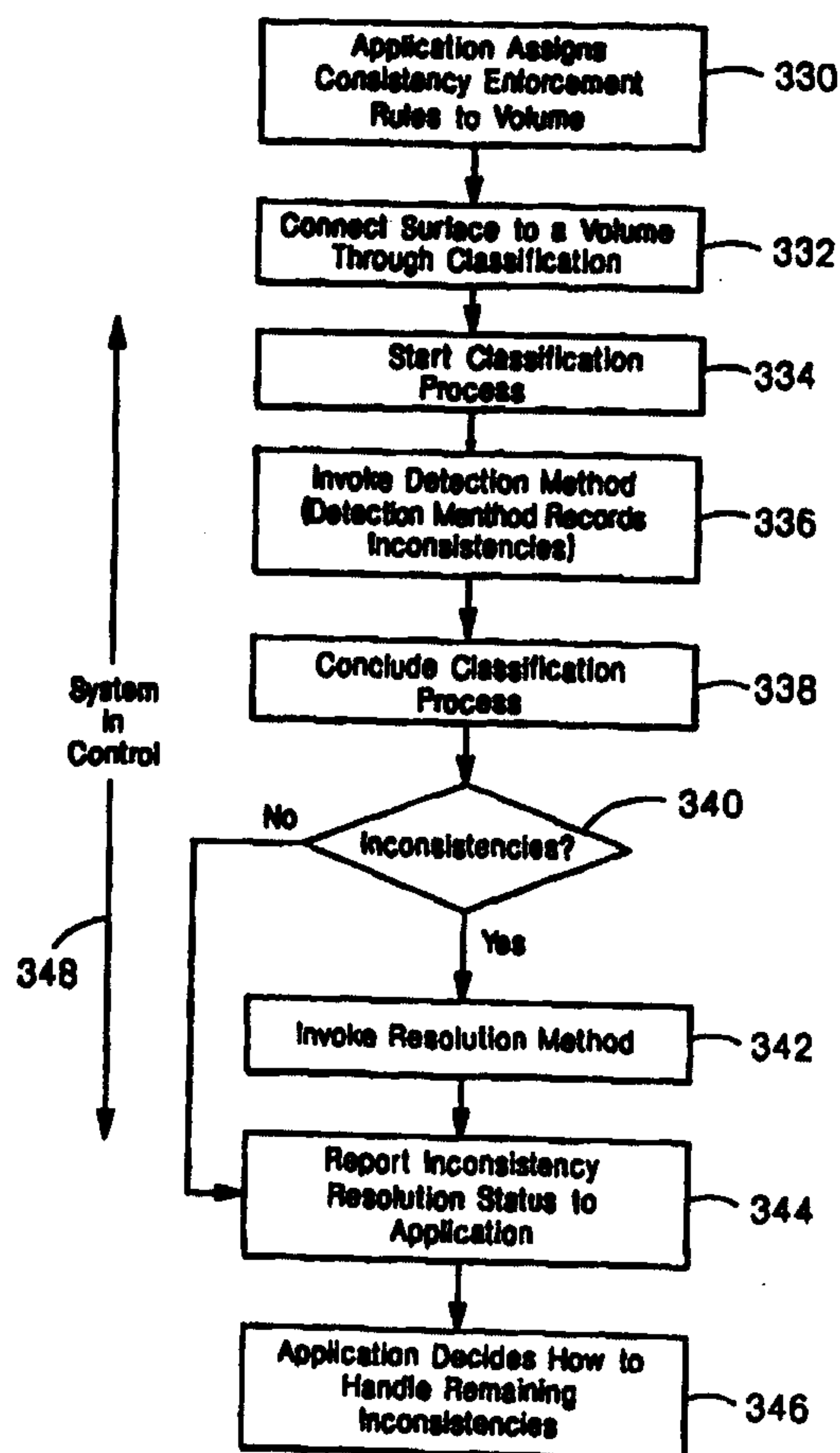
(72) RAMSHORN, CHRISTOPH CLEMENS, GB

(71) SCHLUMBERGER CANADA LIMITED, CA

(51) Int.Cl.⁶ G01V 1/28, G01V 11/00

(54) **RECHERCHE DE COHERENCE DANS DES MODELES
GEOSCIENTIFIQUES**

(54) **ENFORCING CONSISTENCY IN GEOSCIENCE MODELS**



(57) L'invention concerne un procédé, un système informatique et un programme informatique permettant d'analyser des données géologiques échantillonnées sur une région souterraine, stockées dans un modèle géoscientifique, sur un support exploitable par ordinateur, consistant à rechercher une cohérence dans le modèle géoscientifique, à définir une cohérence, à détecter une incohérence et à résoudre cette incohérence.

(57) A method, computer system and computer program for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media including enforcing consistency within the geoscience model. A method, computer system and computer program for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media including defining consistency; detecting an inconsistency; and resolving the inconsistency.



PCTWORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : G01V 1/28, 11/00	A1	(11) International Publication Number: WO 99/53342 (43) International Publication Date: 21 October 1999 (21.10.99)
--	-----------	--

(21) International Application Number: PCT/US98/07119

(22) International Filing Date: 9 April 1998 (09.04.98)

(71) Applicant: SCHLUMBERGER TECHNOLOGY CORPORATION [US/US]; 8311 North R.R. 620, Austin, TX 78720-0015 (US).

(72) Inventors: ASSA, Steven, B.; 10106 Annie Oakley Trail, Austin, TX 78753 (US). RAMSHORN, Christoph, C.; 2108 Elton Lane, Austin, TX 78703 (US).

(74) Agent: OSHA, Jonathan, P.; Rosenthal & Osha L.L.P., Suite 4550, 700 Louisiana, Houston, TX 77002 (US).

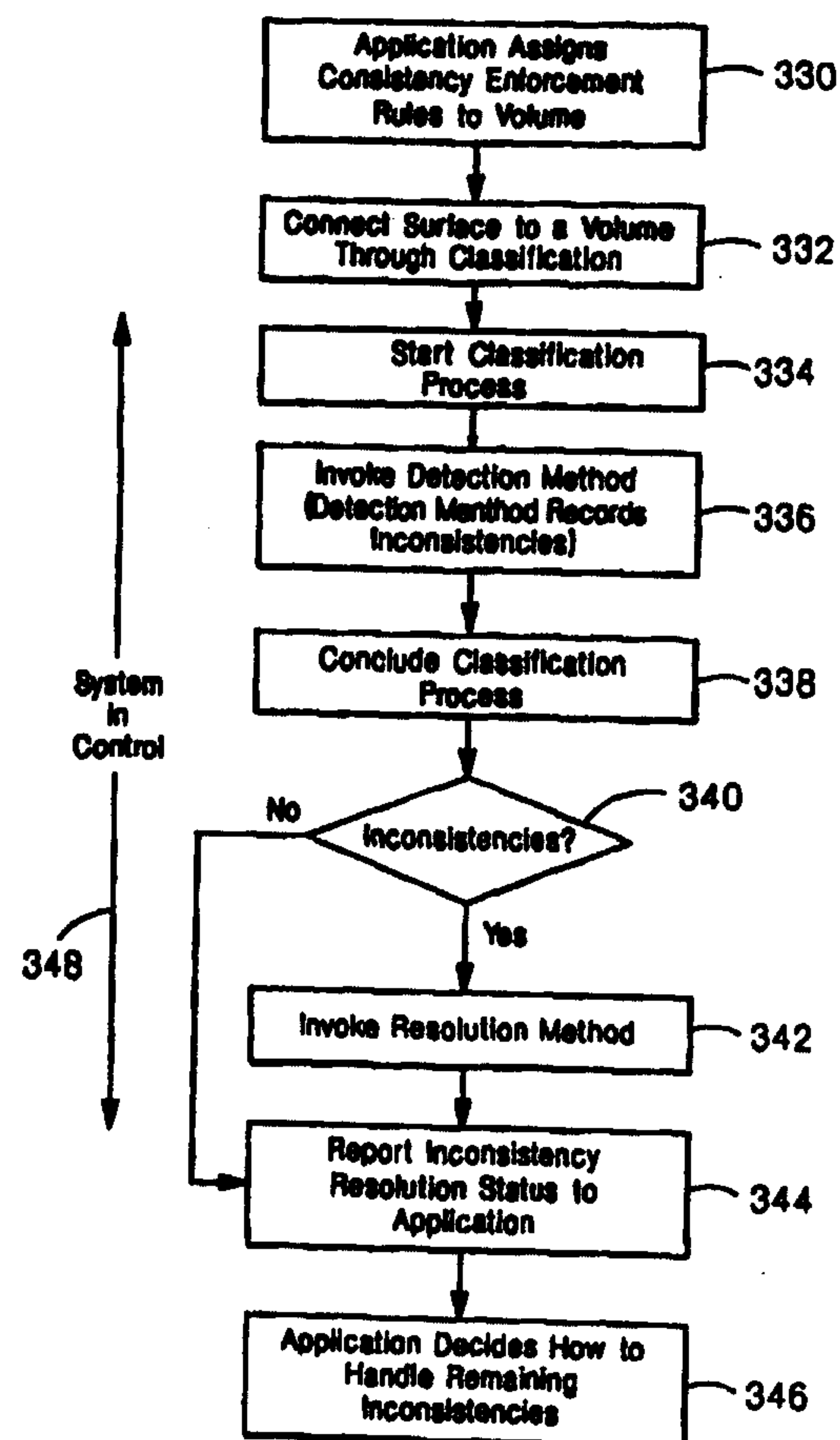
(81) Designated States: CA, NO, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published*With international search report.*

(54) Title: ENFORCING CONSISTENCY IN GEOSCIENCE MODELS

(57) Abstract

A method, computer system and computer program for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media including enforcing consistency within the geoscience model. A method, computer system and computer program for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media including defining consistency; detecting an inconsistency; and resolving the inconsistency.



- 1 -

ENFORCING CONSISTENCY IN GEOSCIENCE MODELSBackground

This invention relates to enforcing consistency in geoscience models.

5 Geologists, geophysicists and petroleum engineers use models, including computerized models, of the earth's shell to plan exploration and production of hydrocarbons and, to a lesser extent, other minerals. As hydrocarbons become more and more scarce, the accuracy of the
10 computerized models becomes increasingly important to limiting the cost of locating and producing hydrocarbons and the associated cost of hydrocarbon products, such as gasoline and heating oil.

Interpretation of log data and seismic surveys
15 produces a description of the structure and properties of rocks in the subsurface, as well as of their fluid and gas contents. Different measurements provide complimentary and overlapping information which has to be integrated to form a consistent model of the subsurface.

20 Summary

In general, in one aspect, the invention features a method for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising enforcing consistency
25 within the geoscience model.

Implementations of the invention may include one or more of the following. Enforcing consistency may comprise applying rules. Enforcing consistency may comprise defining consistency. Enforcing consistency may
30 comprise detecting inconsistency.

Detecting inconsistency may comprise detecting a seed inconsistency. Detecting inconsistency may comprise detecting a background inconsistency. Detecting inconsistency may comprise detecting a seed inconsistency

- 2 -

when a seed is classified into a background. Detecting inconsistency may comprise detecting a background inconsistency when a seed is classified into a background. Detecting inconsistency may comprise
5 detecting a potential inconsistency.

Enforcing consistency may comprise resolving inconsistency. Resolving inconsistency may comprise removing a portion of a seed from a topologically active part of the geoscience model. Resolving inconsistency
10 may comprise removing a portion of a background from a topologically active part of the geoscience model. Resolving inconsistency may comprise changing a geometry of a seed. Resolving inconsistency may comprise changing a geometry of a portion of a background. Resolving
15 inconsistency may comprise flagging an inconsistency for later resolution.

Detecting inconsistency may comprise detecting an embedded surface cell's age being outside the age range of a host layer for the embedded surface cell. Detecting
20 inconsistency may comprise detecting a point in a subvolume having an age not within the range defined by the subvolume's bounding erosional or depositional surfaces. Detecting inconsistency may comprise detecting a bed boundary not connected to two or more distinct
25 subvolumes. Detecting inconsistency may comprise detecting a horizon which terminates in a layer. Detecting inconsistency may comprise detecting a child subvolume having a volume less than a predefined percentage of the child subvolume's parent's volume.
30 Detecting inconsistency may comprise detecting a ratio of a subvolume's volume to its surface being less than a predetermined amount. Detecting inconsistency may comprise detecting a first age assignment from a first horizon that is inconsistent with a second age assignment
35 from a second horizon. Detecting inconsistency may

- 3 -

comprise detecting a pinched-out layer. Detecting inconsistency may comprise detecting a layer forming lenses. Detecting inconsistency may comprise detecting a cell of a feature embedded into a sub-region where the
5 feature is not allowed.

Resolving inconsistency may comprise asking how to extend the surface. Resolving inconsistency may comprise extending a horizon to a nearest surface bounding the layer. Resolving inconsistency may comprise extending a
10 horizon to a geoscience model boundary. Resolving inconsistency may comprise removing a segment of a feature which extends into an unallowed sub-region. Resolving inconsistency may comprise identifying a horizon that contributes to the inconsistency; and
15 removing a segment of the horizon that contributes to the inconsistency. Resolving inconsistency may comprise identifying a first horizon and a second horizon that contribute to the inconsistency; and reshaping the first horizon and the second horizon to maintain a distance
20 between them that is greater than or equal to a preselected amount. Resolving inconsistency may comprise collapsing a small sub-region bounded by a horizon. Collapsing may comprise removing a portion of the horizon that bounds the small sub-region. Resolving
25 inconsistency may comprise identifying a sub-region created by invalid horizons; and reshaping the invalid horizons to produce a correctly-shaped sub-region.

The method may further comprise blocking property
30 updates.

In general, in another aspect, the invention features a method for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising defining

- 4 -

consistency; detecting an inconsistency; and resolving the inconsistency.

In general, in another aspect, the invention features a computer system for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising means for enforcing consistency within the geoscience model.

Implementations of the invention may include one or more of the following. The enforcing consistency means may comprise means for applying rules. The enforcing consistency means may comprise means for defining consistency. The enforcing consistency means may comprise means for detecting inconsistency. The detecting inconsistency means may comprise means for detecting a seed inconsistency. The detecting inconsistency means may comprise means for detecting a background inconsistency. The detecting inconsistency means may comprise means for detecting a seed inconsistency when a seed is classified into a background. The detecting inconsistency means may comprise means for detecting a background inconsistency when a seed is classified into a background. The detecting inconsistency means may comprise means for detecting a potential inconsistency.

The enforcing consistency means may comprise means for resolving inconsistency.

The resolving inconsistency means may comprise means for removing a portion of a seed from a topologically active part of the geoscience model. The resolving inconsistency means may comprise means for removing a portion of a background from a topologically active part of the geoscience model. The resolving inconsistency means may comprise means for changing a geometry of a seed. The resolving inconsistency means may comprise means for changing a geometry of a portion

- 5 -

of a background. The resolving inconsistency means may comprise means for flagging an inconsistency for later resolution.

The detecting inconsistency means may comprise
5 means for detecting an embedded surface cell's age being outside the age range of a host layer for the embedded surface cell. The detecting inconsistency means may comprise means for detecting a point in a subvolume having an age not within the range defined by the
10 subvolume's bounding erosional or depositional surfaces. The detecting inconsistency means may comprise means for detecting a bed boundary not connected to a two or more distinct subvolumes. The detecting inconsistency means may comprise means for detecting a horizon which
15 terminates in a layer. The detecting inconsistency means may comprise means for detecting a child subvolume having a volume less than a predefined percentage of the child subvolume's parent's volume. The detecting inconsistency means may comprise means for detecting a ratio of a
20 subvolume's volume to its surface being less than a predetermined amount. The detecting inconsistency means may comprise means for detecting a first age assignment from a first horizon that is inconsistent with a second age assignment from a second horizon. The detecting
25 inconsistency means may comprise means for detecting a pinched-out layer. The detecting inconsistency means may comprise means for detecting a layer forming lenses. The detecting inconsistency means may comprise means for detecting a cell of a feature embedded into a sub-region
30 where the feature is not allowed.

The resolving inconsistency means may comprise means for asking how to extend the surface. The resolving inconsistency means may comprise means for extending a horizon to a nearest surface bounding the
35 layer. The resolving inconsistency means may comprise

- 6 -

means for extending a horizon to a geoscience model boundary. The resolving inconsistency means may comprise means for removing a segment of a feature which extends into an unallowed sub-region. The resolving

5 inconsistency means may comprise means for identifying a horizon that contributes to the inconsistency; and means for removing a segment of the horizon that contributes to the inconsistency. The resolving inconsistency means may comprise means for identifying a first horizon and a

10 second horizon that contribute to the inconsistency; and means for reshaping the first horizon and the second horizon to maintain a distance between them that is greater than or equal to a preselected amount. The resolving inconsistency means may comprise means for

15 collapsing a small sub-region bounded by a horizon. The collapsing means may comprise means for removing a portion of the horizon that bounds the small sub-region. The resolving inconsistency means may comprise means for identifying a sub-region created by invalid horizons; and

20 means for reshaping the invalid horizons to produce a correctly-shaped sub-region.

The computer system may further comprise means for blocking property updates.

In general, in another aspect, the invention

25 features a computer system for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising means for defining consistency; means for detecting an inconsistency; and means for resolving the inconsistency.

30 In general, in another aspect, the invention features a computer program, residing on a computer-readable medium, for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising instructions for

- 7 -

causing the computer to enforce consistency within the geoscience model.

Implementations of the invention may include one or more of the following. The instructions for causing the computer to enforce consistency may comprise instructions for causing the computer to apply rules. The instructions for causing the computer to enforce consistency may comprise instructions for causing the computer to define consistency. The instructions for causing the computer to enforce consistency may comprise instructions for causing the computer to detect inconsistency.

The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a seed inconsistency. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a background inconsistency. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a seed inconsistency when a seed is classified into a background. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a background inconsistency when a seed is classified into a background. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a potential inconsistency.

The instructions for causing the computer to enforce consistency may comprise instructions for causing the computer to resolve inconsistency.

The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to remove a portion of a seed from a topologically active part of the geoscience model. The

- 8 -

instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to remove a portion of a background from a topologically active part of the geoscience model. The

5 instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to change a geometry of a seed. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the

10 computer to change a geometry of a portion of a background. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to flag an inconsistency for later resolution.

15 The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect an embedded surface cell's age being outside the age range of a host layer for the embedded surface cell. The instructions for causing the

20 computer to detect inconsistency may comprise instructions for causing the computer to detect a point in a subvolume having an age not within the range defined by the subvolume's bounding erosional or depositional surfaces. The instructions for causing the computer to

25 detect inconsistency may comprise instructions for causing the computer to detect a bed boundary not connected to a two or more distinct subvolumes. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the

30 computer to detect a horizon which terminates in a layer. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a child subvolume having a volume less than a predefined percentage of the child subvolume's

35 parent's volume. The instructions for causing the

- 9 -

computer to detect inconsistency may comprise instructions for causing the computer to detect a ratio of a subvolume's volume to its surface being less than a predetermined amount. The instructions for causing the
5 computer to detect inconsistency may comprise instructions for causing the computer to detect a first age assignment from a first horizon that is inconsistent with a second age assignment from a second horizon. The instructions for causing the computer to detect
10 inconsistency may comprise instructions for causing the computer to detect a pinched-out layer. The instructions for causing the computer to detect inconsistency may comprise instructions for causing the computer to detect a layer forming lenses. The instructions for causing the
15 computer to detect inconsistency may comprise instructions for causing the computer to detect a cell of a feature embedded into a sub-region where the feature is not allowed.

The instructions for causing the computer to
20 resolve inconsistency may comprise instructions for causing the computer to ask how to extend the surface. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to extend a horizon to a nearest surface
25 bounding the layer. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to extend a horizon to a geoscience model boundary. The instructions for causing the computer to resolve inconsistency may
30 comprise instructions for causing the computer to remove a segment of a feature which extends into an unallowed sub-region. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to identify a horizon that
35 contributes to the inconsistency; and instructions

- 10 -

for causing the computer to remove a segment of the horizon that contributes to the inconsistency. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the
5 computer to identify a first horizon and a second horizon that contribute to the inconsistency; and instructions for causing the computer to reshape the first horizon and the second horizon to maintain a distance between them that is greater than or equal to a preselected amount.
10 The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to collapse a small sub-region bounded by a horizon. The instructions for causing the computer to collapse the small sub-region may comprise instructions
15 for causing the computer to remove a portion of the horizon that bounds the small sub-region. The instructions for causing the computer to resolve inconsistency may comprise instructions for causing the computer to identify a sub-region created by invalid
20 horizons; and instructions for causing the computer to reshape the invalid horizons to produce a correctly-shaped sub-region.

The computer program may comprise instructions for causing the computer to block property updates.

25 In general, in another aspect, the invention features a computer program, residing on a computer-readable medium, for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising instructions for
30 causing the computer to define consistency; detect an inconsistency; and resolve the inconsistency.

- 11 -

Description

Fig. 1 is a block diagram.

Figs. 2a-c, 3a-c, 4a-b, 5a-c, 6a-c, 7a-c, 8a-e, 9a-b, 10a-b, 11a-b, 12a-b, 13a-b, 14a-c, 15a-f, 16a-c, 5 and 17a-d are examples of items to be modeled.

Fig. 4c is a topology graph.

Fig. 18 is a flow chart.

Simulations may be used to verify interpretations, an approach which is called simulation-based
10 interpretation ("SBI"), described in co-pending patent application serial number _____, entitled "Performing Geoscience Interpretation with Simulated Data", respectively, incorporated by reference.

Simulation-based interpretation, illustrated in
15 Fig. 1, begins by gathering acquired data 10 from the geological structures being modeled. The acquired data is interpreted, and the resulting interpretation is captured as a geoscience model 12. The model is then tested by simulating the original acquisition experiment
20 to produce synthetic data 14. Differences between the synthetic data 14 and the acquired data 10 are determined and used to adjust the model. Repeated iterations of these steps may cause a model to converge toward a consistent model in which the differences between the
25 synthetic and the acquired data are reduced.

Building 3D geometric models of geological structures is not easy because it involves modeling irregular shapes with complex spatial relationships. Irregular Space Partition (ISP) modeling represents
30 geological bodies and their spatial relationships and defines a region of interest whose internal structure is built from surfaces such as horizons and faults that subdivide the region of interest and form sub-regions (such as geologic layer 20a, 20b shown in Fig. 2a). The
35 notion of a region is independent of dimension; a 3D

- 12 -

region is a solid object, a 2D region is an area, etc. An ISP model explicitly stores the spatial relationships of all the subregions contained in it.

An extension to ISP modeling, feature-based
5 modeling, makes it possible to define geometric objects ("features") that consist of one or more pieces that do not have to be topologically connected. For example, all the features in Fig. 2a such as horizon 22a, 22b, 22c, horizon 24a, 24b, both faulted by faults 26 and 28, and
10 layer 20a, 20b, disconnected by fault 28, are treated as single objects, even though they are disconnected.

A wide variety of measurements, processing methods and simulations can create a model that accurately describes a subsurface region. This model can describe a
15 geometry of subsurface structures, a topology relationship among their components, and material properties of subsurface regions such as layers and reservoir compartments. The model may be general and sharable by different applications. Since subsurface
20 structures can be large and complex, the model may support automated, high-level model editing.

Building three-dimensional geometric models of geological structures is complicated by the fact that input geometry is used to build a geoscience model are
25 frequently not "clean." For example, as shown in Fig. 2b, fault ties, which are intersections of horizons and faults, may be missing 30, horizons may cross faults 32, or horizons may penetrate each other 34.

The stratigraphic and structural characteristics
30 of specific features are often known. For example, it may be known that fault 28 in Fig. 2a truncates and offsets horizon 22a, 22b, 22c and horizon 24a, 24b, which are older. It may be known that fault 28 is in turn truncated by the younger erosional contact 36. It may
35 also be known that horizon 22a, 22b, 22c and horizon 24a,

- 13 -

24b separate younger layers above from older layers below. Additionally, it may be known that the layer between erosional contact 36 and another erosional contact 38 has a certain minimal thickness.

5 Exploiting such characteristics can help applications to automatically build correct, clean models or to check an interpretation for consistency. When building irregular spaced partition models of geological structures, three geometrical conditions are encountered
10 that create inconsistencies, as shown in Figs. 2a and 2b. Generally, the three types of inconsistencies are shape, material property, and topology inconsistencies. The inconsistencies can occur in the seed and in the background (both terms are defined below).

15 A geometric component can be too large, such as horizon segment 22a, too short, such as horizon segment 22b (with respect to fault 26), or have the wrong shape, such as erosion contacts 36 and 38. Components that are too large can also be considered to contain excessive
20 pieces, such as the piece of horizon segment 22a extending to the right of fault segment 26. Since it cannot be determined by geometrical analysis only whether or not those conditions constitute an error, additional constraints are necessary to express the structure that
25 is intended to be built. The rule system uses these additional constraints to automatically build correct, clean models such as that depicted in Fig. 2c.

 The formulation of a geological rule requires the analysis of whether to constrain the shape, the
30 topological relationships, the attributes of an object, or a combination of all three. The system separates the representations of shape, topology, and properties of geometric objects for flexibility and to make extensions straightforward.

- 14 -

Shapes can comprise any of a combination of geometry primitives including lines, planes, free-form surfaces, and 3D solids. The shape of a geometry primitive is defined to be the set of points it occupies
5 in space. The primary surface shapes used to build geological models are represented by grids and triangle meshes.

Parametric shape representations may be used to provide a wide range of different shapes. A parametric
10 shape representation is provided by a parametric equation which maps variables in parameter space to image space. For example, consider a regularly gridded surface of N by M nodes with a spacing of d in both directions, and node (0,0) located at $[x_0, y_0, f(0,0)]$. The surface's shape is
15 the set of points given by the function:

$$x = x_0 + n d$$

$$y = y_0 + m d$$

$$z = f(0,0)$$

where n and m are parameters defined in the range
20 from 0 to N and 0 to M, respectively, and $f(n,m)$ is the value of z at node (n,m).

These equations define the mapping from a parameter space where n and m are defined, to the image space where $[x, y, z]$ coordinates are defined. For every
25 (n,m)-pair there is a unique value of $[x, y, z]$. If a continuous interpolation function is defined to obtain z-values for fractional values of n and m, the surface number of points on the surface is infinite and the surface is within its boundaries everywhere defined.

30 Topology, at a low level, includes representations within a geometric model of a collection of 0- to 3-dimensional objects (points, curves, surfaces, and volumes), called cells, whose topological relationships are explicitly represented. Figs. 3a-c depict the
35 concept of a cell.

- 15 -

A cell is defined as a closed but not necessarily bounded path-connected region of fixed dimension. A region is closed if its point set encloses its boundaries. A region is bounded if it is of finite extent. Path-connected means that any two points in a region can be connected by a path of finite length which never leaves the region, as shown by path 40 in Fig. 3a. The fixed dimension restriction implies a cell is a surface or a volume, but is not a surface plus a volume. A surface embedded in a volume is a set of two cells. As a point-set, two distinct cells can overlap along a common boundary or not at all.

In Fig. 3b areas 42 and 44 are distinct cells, as are the line segments between intersection points 46a-k, as are the intersection points 48a-j, as shown in Fig. 3c. The information associated with the cells of a geometric model fully describe the model's geometry and topology. Consequently, the geometric model is itself represented as a cellular model.

The topology of a geometric model is the set of all cell-cell connectivity relationships. These can be represented in a graph in which the arcs of the graphs represent connectivity and the nodes represent cells, as shown in Figs. 4a-c. Box 50, shown in Fig. 4a, comprises two dimensional cell 52, one dimensional cells 54, 56, 58, and 60, and zero dimensional cells 62, 64, 66, and 68, as shown in Fig. 4b.

In a cellular model, cells can have two roles: as a boundary of a region of higher dimension; and as a region of higher dimension bounded by cells of lower dimension. Cells of dimension n are connected to cells of dimension less than n and vice versa. Most frequently, n -dimensional cells are bounded by cells having dimension $n-1$. It is possible for n -dimensional cells to be connected to cells of dimension less than $n-1$

- 16 -

(e.g., a borehole represented as a one-dimensional curve in a three-dimensional volume). In a topology graph, illustrated in Fig. 4c, cells are represented by circles and connections between cells are represented by arcs.

5 Three dimensional cell 52, represented by circle 70, is connected to one dimensional cells 54, 56, 58, and 60, represented by circles 72, 74, 76, and 78, respectively. The connection is represented by arcs 71, 73, 75, and 77. One dimensional cell 54 is connected to zero dimensional

10 cells 62 and 64, represented by circles 80 and 82, respectively. The connections are represented by arcs 88 and 90. One dimensional cell 56 is connected to zero dimensional cells 64 and 66, represented by circles 82 and 84, respectively. The connections are represented by

15 arcs 92 and 94. One dimensional cell 58 is connected to zero dimensional cells 66 and 68, represented by circles 84 and 86, respectively. The connections are represented by arcs 96 and 98. One dimensional cell 60 is connected to zero dimensional cells 68 and 62, represented by

20 circles 86 and 80, respectively. The connections are represented by arcs 100 and 102.

The most frequent representation in ISP models for boundary relationships for geological cells is that: endpoints bound curves; curves bound surfaces; and

25 surfaces bound subvolumes. For example, a surface can bound a subvolume, but can itself be bounded by a set of curves. The one dimensional cells 54, 56, 58 and 60 in Fig. 4b are both subregions, bounded by points 62, 64, 66, and 68, and boundaries of subregion 52. Cells that

30 are contained in higher-dimensional cells but do not split them, such as fault 104 in Fig. 3b, are said to be "embedded."

The operation of combining two geometric objects is called classification. Each geometric object is

35 represented by a set of connected cells, called a cell

- 17 -

complex. As the cell complexes are classified against each other, new cells may be generated by splitting existing cells.

Classification is executed in four steps: (1)
5 compute intersections of shapes and generate cells
representing the intersection geometries (such as the
intersection points in Fig. 3c); (2) embed
lower-dimensional cells of complex 1 in the
higher-dimensional cells of complex 2 that they overlap;
10 (3) split cells that are subdivided by lower-dimensional
cells; (4) update the material properties of all affected
cells (material properties are discussed below).

For example, assume an earth model contains a
layer 110 and a faulted horizon 112a, 112b, as shown in
15 Figs. 5a-c. Initially, there is only one layer in the
model because the bed boundary geometry with its gap does
not subdivide layer 110. During classification, a fault
114 first gets split by horizon cells 112a and 112b, as
shown in Fig. 5b, after which fault cells 116a, 116b,
20 116c get embedded in layer 110, as shown in Fig. 5c.
Then layer 110 is split into layers 118 and 120, as shown
in Fig. 5c. Fault cells 116a and 116c are embedded into
layers 118 and 120, respectively. Embedding of fault
116a, 116b, 116c into layer 110 occurs prior to
25 splitting.

Classification is discussed in detail in co-
pending application serial number _____, entitled
"Modeling Geological Structures and Properties",
incorporated by reference.

30 Attributes, or properties, are assigned to
topological entities in a cellular model, where
topological entities correspond to nodes or arcs in the
model's topology graph. One topological entity can be
assigned multiple properties.

- 18 -

While the primary use of the property system is to handle material properties, other properties such as geological age can be represented equally well. The property system has four important concepts.

5 First, properties can be tied to features and cells and other geometric entities.

Second, property representations can be arbitrarily chosen to be constant values, pointers to sets of values, or pointers to functions. Those
10 functions typically take a point either in parameter space or in image space and return a material property value. Alternatively, they can return a property gradient or some other derived value.

Third, properties can be oriented. A surface in
15 3-space, for example, can have different properties on either side. In Figure 6a, for example, a surface 130 in 3-space is shown with two different material properties 132 and 134, one for each side of the surface.

Fourth, changes to a model's topology graph
20 (called "events") trigger functions (called "event callbacks") that can alter property assignments. Typical events include splitting and merging cells, and embedding of lower dimensional cells. In Figs. 6a-c, as the surface 130 is inserted into a volume 136, the volume 136
25 is subdivided, its 3D cell is split, and two new subregions 138 and 140 are created. In this example, a split event callback propagates the curve's oriented material properties to the appropriate subregions. Consequently, the sub-region 138 above the curve takes
30 the material property 132 associated with the top of the surface and the sub-region below the surface 140 takes the material property 134 associated with the bottom of the surface.

An application is able to keep a model's material
35 properties assigned correctly during classification

- 19 -

because the callback mechanism notifies the application each time a cell with a material property changes its topological state. Material property representations are discussed in detail in co-pending application serial
5 number _____, entitled "Modeling Geological Structures and Properties", incorporated by reference.

While cellular models can represent geometries of arbitrary complexity, it is cumbersome to build and edit them, and to keep track of material property assignments
10 as cells are added, split, merged, and deleted as side effects of geometric operations. Cellular modeling is extended with the concept of features.

Features can be thought of as groups of one or more cells with a number of important conceptual
15 properties (among others; see co-pending application serial number _____, entitled "Modeling Geological Structures and Properties", incorporated by reference, for a full discussion of features).

First, features preserve their point-set. The bed
20 features 150, 152, and 154, shown in Fig. 7a, each continue to represent their entire respective beds although the gas, oil, and water layers 156, 158, and 160 have split feature 150 into five cells 162, 164, 166, 168, and 170, feature 152 into four cells 172, 174, 176
25 and 178, and feature 154 into two cells 180, 182, as shown in Figs. 7b and 7c.

Second, a feature may be discontinuous, that is, a feature may contain unconnected cells. For example, layer feature 20a and 20b in Fig. 2a comprises two
30 unconnected cells 20a and 20b.

Third, features may overlap as point sets. Individual cells may belong to more than one feature. For example, bed feature 150's cells, shown in Fig. 7c, also belong to the water, oil, and gas features 156, 158,
35 and 160.

- 20 -

Fourth, features can carry properties that are inherited by their cells. What properties a cell has depends on the context in which its properties are evaluated.

5 Fifth, features can have 'active' and 'inactive' cells. Active cells can change their connectivity whereas inactive cells cannot. When features are combined to form a larger geometric model, only their active cells change topological state. Inactive cells
10 are created as parts of features and are trimmed off, i.e., removed from the active part of the model, during model building. An application of this mechanism is the modeling of eroded horizons (often referred to as "ghost surfaces") and layers.

15 Sixth, a model may contain active and inactive features. Features are active by default and may contain active and inactive cells. Certain operations, such as dissociating a feature from a model, make a feature inactive. Inactive features only contain inactive cells,
20 some of which may be copies of active cells. Inactive features cannot change their connectivity.

Rule based modeling is an extension of feature-based modeling.

Feature-based modeling can represent the
25 inconsistencies that are generated when constructing a geometric model but cannot automatically report or resolve them. Structurally correct geological models are automatically built by exploiting rules based on feature properties.

30 The properties and structural characteristics of a feature can be translated into specifications that state which values of feature properties and which spatial relationships between features are considered correct. Such specifications are called rules. Rules apply to

- 21 -

properties such as formation age or whether a surface represents a bed boundary or a fault.

Organizationally, a rule consists of a detection method and a resolution method. The detection method's task is to recognize when an inconsistency has arisen. The resolution method's task is to remove the inconsistency, to instruct the system how to remove the inconsistency, or to report the inconsistency to the application. Detection is automatic, in the sense that application-provided detection methods are invoked at the appropriate time to verify consistency. Resolution is performed at the conclusion of the operation giving rise to the violations, in the sense that the detection method calls a specific resolution method to resolve a particular violation. In certain cases, resolution may require complex human input, e.g, instructions on how to tie a bed boundary to a fault.

For example, when a fault 190a, 190b is classified, it should be truncated by an erosional surface 192, as shown in Fig. 8a. Instead, the fault extends across the erosional surface. An application may assign the following rule to each subvolume:

Formation age rule (detection): The age of any point in a subvolume is bounded above and below by the age of the erosional or depositional surfaces that bound the subvolume.

Formation age rule (resolution): Remove any surface cell violating the age bound.

Assume that all points along erosional surface 192 are of fixed age e . Suppose fault 190a, 190b is of constant age f and that $f > e$, i.e., that the fault is older than the erosional surface. During classification, the fault will be split along erosional surface 192 into fault cells 190a and 190b. The splitting does not affect the age

- 22 -

assignment, so both parts are of age f. An inconsistency occurs with respect to the embedding of fault cell 190b in the upper subvolume 194, because the age of the fault is greater than the age of the erosional layer, which is in turn greater than the age of layer 194. Thus, the age of fault cell 190b is greater than the age of layer 194, which violates the formation age rule. In contrast, fault cell 190a's embedding in layer 196 is consistent because the age of the fault is greater than the age of the erosional surface and the age of layer 196 is greater than or equal to the age of the erosional surface. Following the rule's resolution strategy the inconsistency is resolved by removing fault segment 190b from the model as shown in Fig. 8b.

Inconsistent structural properties can also be detected and resolved, as shown in Fig. 8c (here, the age relationships are assumed to be everywhere consistent). Horizon 200a, 200b is offset by fault 202 but the horizon and fault geometries do not match exactly. Specifically, there is a gap between horizon cell 200a and fault 202 and horizon cell 200b crosses fault 202. The corresponding connectivity rule is defined as follows.

Connectivity rule (detection): A bed boundary must be connected to distinct subvolumes.

Connectivity rule (resolution): Ask how to extend the surface to subdivide the subvolume containing it.

Since horizon 200a, 200b is a bed boundary, each piece of it should separate two subregions. Horizon cell 200a should separate subregions 204 and 206 and horizon cell 200b should separate subregions 204 and 208. Horizon cells 200a and 200b each create an inconsistency, because each fails to subdivide the model. Note that although fault 202 does not subdivide subvolume 204 either, this

- 23 -

is not an inconsistency: a fault can terminate (or "die out") anywhere in a layer.

Several forms of resolution are possible. Since horizon cell 200a is in close proximity to fault 202, one
5 form of resolution is to extend horizon cell 200a to match up with fault 202 and remove from the model the portion of horizon cell 200b extending to the left of fault 202, as shown in Fig. 8d. Another possibility is to rotate or shift fault 202 to match horizon cells 200a
10 and 200b, as shown in Fig. 8e.

Inconsistencies in shape can also be detected and resolved. For example, suppose that a region-of-interest 209 is divided by surfaces 210 and 211 into three layers 212, 213 and 214, as shown in Fig. 9a. Suppose that the
15 user wishes to move surface 211 to a new position close to surface 210, but accidentally moves surface 211 so that it overlaps surface 210, as shown in Fig. 9b. This might happen because of the granularity of the device, such as a mouse, being used to move the surface, because of the
20 inability of the user to control the moving device or for some other reason.

The movement of surface 211 occurs in two steps. In the first, surface 211 is dissociated from the region of interest and sub-regions 213 and 214 are merged. In
25 the second step, surface 211 is classified into the region of interest. The classification causes sub-region 212 to be split into two sub-regions 215 and 216, and causes the sub-region which was the merger of sub-regions 213 and 214 to be split into three sub-regions: 217, 218
30 and 219.

Sub-region 212 is split by invoking a split callback. The callback processes the following detection method and resolution method:

35 Small volume rule (detection): A child subvolume of a split of a parent subvolume must have at

- 24 -

least a predefined percentage of the volume of the parent subvolume.

Small volume rule (resolution): Dissociate one of the surfaces bounding the child subvolume.

5 In the example, subvolumes 217 and 218 are found to have a volume above the "predefined percentage" of the volume of the subvolume that was the merger of volumes 213 and 214. When the volume of subvolume 215 is compared to volume of subvolume 212, on the other hand, it is found
10 to violate the small volume rule.

When surface 211 was classified into the position shown in Fig. 9b, surfaces 210 and 211 were split into three cells each, 210a, 210b, 210c and 211a, 211b, 211c, respectively. The resolution method is to delete one of
15 the two surface cells, 210b and 211b, bounding subvolume 216. Assume that the application chooses to dissociate surface 211b. In that case, subvolume 216 is merged into subvolume 215.

A variation on the "small volume rule" may be
20 necessary in some instances when two nearly identical surfaces are moved very close to each other but are still separated. This might happen, for example, when a user is trying to move one surface on top of another knowing that they are the same surface. In such a case, the
25 volume of the subvolume between the two surfaces is very large, because the extent of the surfaces is very large, and the "small volume rule" would not have the desired effect. The variation on the detection method of the rule would determine the ratio of the surface area of a
30 subvolume to its volume and require it to be below a predetermined value. In the example, the ratio would not be below the predetermined value and the resolution method would be invoked.

- 25 -

In the context of a classification involving rules, we call a feature that is being classified against a model a "seed" and the model before classification the "background." Using this terminology, a framework for
5 rule-based modeling is defined that may include a number of components.

First, an expected value or range of values for a property may be specified. In Fig. 8a, the expected value of fault 190a's, 190b's age property is an age
10 greater than the age of surface 192. The expected structural characteristic of surface 192 is that it is connected to two subvolumes 194, 196 and in particular is not a crack within a single subvolume.

Second, an inconsistency detection method is
15 provided. This algorithm detects where properties are different from their expected values, or where topological relationships do not match structural characteristics. Geometric inconsistencies (e.g, subdivision of a volume into two parts with one part
20 being very small; such as the "lens" 210 between layers 212 and 214 in Fig. 2a) or connectivity inconsistencies (such as horizon cell 200a and fault 202 in Fig. 8b) are the expected failure modes. The method is invoked whenever a background feature is split or is the target
25 in an embed operation. The inconsistency detection method returns a status describing the type of inconsistency found, if any, and a list of seed and background features that are affected by the inconsistency.

30 Some inconsistencies cannot be fully detected during classification (see discussion of "potential inconsistencies" below). In these cases, the inconsistency detection algorithm may be automatically invoked after the classification has finished.

- 26 -

Third, an inconsistency resolution method is provided. This algorithm implements the strategy to resolve an inconsistency. The method returns a status and, optionally, a list of affected cells.

5 Inconsistency resolution has to be delayed until the classification has finished. This is the safest and most frequently used mode. It is believed that in the future the geometry engine will allow classification to be aborted before normal completion, which will allow
10 some inconsistencies to be resolved during classification.

For example, suppose it is desired to classify erosional surface 225, shown in Fig. 10a, into volume of interest 226 which is already split by surface 227 into
15 subvolumes 228 and 229, producing the volume shown in Fig. 10b. During the classification, it becomes clear that a portion of surface 227 violates the formation age rule, because the erosional surface is younger than surface 227. When the classification is complete,
20 surface 227 has been split into three surface cells 230, 231 and 232, and the system can ascertain that cell 231 violates the formation age rule and dissociate it.

Fourth, an interface to the geometric modeling system is provided that specifies when detection and
25 resolution algorithms are invoked, which parameters they are passed, and what they return.

The system assumes responsibility for invoking the detection and resolution method at appropriate moments. Each application provides its detection and resolution
30 methods. Alternatively, the system may have a library of predefined detection and resolution methods, in which case the application must choose both a detection and a resolution method as a pair. It is possible to devise very complicated rules, but for practical use simple
35 bounds checks are used for error detection.

- 27 -

Rules are associated with features. Preferably, the inconsistency detection and resolution algorithms are only invoked when a feature forms part of the background. Alternately, a rule can be attached at any time to a
5 topologically stable volume (that is, a volume not undergoing a classification), causing the system to reevaluate the material, shape, and connectivity relationships of every cell connected to the volume. Of course, the corresponding seed features are accessible by
10 those algorithms as they run. If a feature does not have an associated rule, any outcome of a classification with respect to the feature is considered correct.

The example of Fig. 8b shows that even seemingly simple rules can involve rather complex processing.
15 However, regardless of a rule's complexity, it can be broken down into specification of what a consistent value is, analysis of consistency, and resolution of inconsistencies. Rules interact with unstable or stable volumes (that is, volumes undergoing classification and
20 not undergoing classification, respectively) only in a limited number of ways, and methods of dealing with those interactions establish a framework which is suitable to handle arbitrary rule definitions.

As one example, if a surface (the "seed") is
25 classified against a consistent and geologically correct model (the "background"), there are five types of inconsistencies that can occur. They are distinguished by the type of resolution method they require and by the point in the classification process they can be detected.

30 One kind of inconsistency is in the seed's material property assignment or assignments. To resolve such an inconsistency, one or more pieces of the seed must be removed. Such an inconsistency is detectable during embedding the seed. An example is the fault cell

- 28 -

190b in Fig. 8a. The age of seed fault cell 190b is inconsistent with the age of the layer 194 in which it is being embedded.

Another kind of inconsistency occurs in the material property assignment of the background. To resolve such an inconsistency, one or more pieces of the background must disappear, possibly as a result of a dissociate or a merge. The cells that disappear may not be defined until later in the seed feature classification, so their removal might not happen until the seed classification has ended. An example is discussed below in the discussion of Figs. 15a-f.

Another kind of inconsistency occurs in the seed's topology. To resolve a topological seed inconsistency, the geometry of the seed must be changed to obtain a specified topological structure. This is typically done by asking the user for help which the user provides by sketching a trajectory for extending the seed to its attachment site. Inconsistencies in a seed's topology can be detected during splitting of a subvolume by a seed or by the absence of splitting (see discussion of "potential inconsistency" below).

For example, suppose that region of interest 233 is split by a fault 234 and has classified against it surface 235, as shown in Fig. 11a. Suppose that the application has assigned to surface 235 the rule that it connects to all faults in the region of interest. The callback associated with the classification of surface 235 will observe that surface 235 does not satisfy that rule. The resolution method will then ask the user to define how surface 235 is to connect to fault 234. Three responses, 236, 237 and 238, of the infinite possible responses are illustrated in Fig. 11b.

Another kind of inconsistency occurs in the background's topology. To resolve such an inconsistency,

- 29 -

the geometry of a background feature must change. This inconsistency can be detected during splitting of a subvolume by a seed or by the absence of splitting (again, see discussion of "potential inconsistencies" below).

For example, suppose that it is desired to classify erosional surface 239 into volume of interest 240 which is split by depositional surfaces 241, 242, 243 and 244, all younger than erosional surface 239, as shown in Fig. 12a. The result, illustrated in Fig. 12b, shows violations of the formation age rule because segments of each depositional surface 241, 242, 243 and 244 are found within the younger volume enclosed by erosional surface 239. These segments must be dissociated as discussed above with regard to Figs. 10a and 10b.

This example illustrates another advantage of feature-based modeling. If this example were attempted with a modeling system that was not feature-based, cell-to-cell inconsistencies between the erosional surface and the depositional surfaces would have to be resolved each time the user stopped moving the erosional surface. In contrast, in feature-based modeling, the system need only remember that a feature (the erosional surface) is being moved across some other features (the depositional surfaces). It can defer the resolution of inconsistencies until later. When the classification is complete, the system knows that the erosional surface feature was classified against the depositional surface features and takes care of the cell-to-cell inconsistencies at that time.

Another kind of inconsistency is a "potential" inconsistency. Some inconsistencies cannot be detected during classification. Consider the insertion of horizon 246a, 246b in Fig. 13a. Horizon 246a, 246b creates a topological seed inconsistency manifested by the fact

- 30 -

that no splitting occurs. The inconsistency cannot be readily detected at embedding time because at that time it is not known whether or not a split will occur.

Potential inconsistencies therefore are "flagged" during embedding. A subsequent split, such as is caused by the classification of fault 248 will split the region into subregions 250, 252, and 254, as shown in Fig. 13b, and clear the flag. Uncleared flags trigger a property consistency checking method after classification.

10 An application may elect to tolerate certain inconsistencies over the span of several classifications. The faulted horizon 246a, 246b in Fig. 13a, for example, creates a topological seed inconsistency because it fails to subdivide (split) the background. However, as fault 15 248 is inserted, the inconsistency is removed, as shown in Fig. 13b.

While in principle any inconsistency could be resolved after classification has finished, early detection is important because it can be used to reduce the amount of required processing. This is particularly relevant when costly material property computations are involved. The order in which surfaces are inserted to build a model also affects performance. This is now shown by reviewing two model building sequences involving 25 an erosional surface.

The first sequence, illustrated in Figs. 14a-c, is an example of a seed inconsistency. The background model consists of two subregions, a younger layer 260 and a older layer 262, separated by an erosional surface 264. 30 Assume that the rule system has been set up like in the example described in the discussion concerning Fig. 8a. A horizon 266a, 266b, 266c is inserted such that it intersects erosional surface 264, as shown in Fig. 14b. Assume that horizon 266a, 266b, 266c is older than 35 erosional surface 264. Horizon cell 266b embedded in

- 31 -

layer 260 creates a seed inconsistency that can be resolved by removing horizon cell 266b, creating a "ghost surface" 266b, as shown in Fig. 14c.

The system does not abort a classification once it is launched, so the system is compelled to undo the split event. A major efficiency issue is minimizing changes to the model's material property state by the (temporary) split event. Alternatively, the system may abort the classification in order to block the completion of the split.

During classification, the embedding of horizon cell 266b in layer 260 splits layer 260 into a subregions above and below horizon cell 266b, triggering a split event callback to adjust material property assignments. The split event callback creates two material property files, one for each of the subregions on either side of horizon cell 266b. The two subregions above and below horizon cell 266b formed from layer 260 exist only temporarily, however, since the embedding of horizon cell 266b generated an inconsistency. After removal of horizon cell 266b, and merger of the two earlier-created material property files, layer 260's material property assignments will be identical to those before the classification. Therefore, temporarily changing layer 260's property assignment is a waste of time and system resources, such as disk space. The system prevents this by simply blocking property updates when a seed inconsistency has been detected.

The second sequence involves a background inconsistency. Consider a set of layers 270, 272, 274, shown in Fig. 15a, to which an erosional surface 236a-c is added, as shown in Fig. 15b. Assume that erosional surface 276a-c is younger than layers 270 and 272 but older than layer 274. Again, the Formation Age Rule defined previously requires that geological age

- 32 -

relationships be honored. Looking just at the result, shown in Fig. 15f, this example is equivalent to that in Figs. 14a-c, but the order in which the surfaces were introduced causes significant differences in workload.

5 The processing of the classification of erosional surface 276a-c is depicted piece by piece from Figs. 15c-15f; the actual order in which processing of erosional surface 276a-c occurs is not defined and does not matter for this discussion.

10 Initially, the embedding of erosional surface 276a-c in layer 272 causes layer 272 to split into two subregions on either side of erosional surface 276a, as shown in Fig. 15c. At this time, the inconsistent region of layer 272 that eventually is going to be eroded, i.e.
15 merged with layer 274, is not completely defined. At the point shown in Fig. 15c, the portion of layer 272 to the right of erosional surface 276a is inconsistent. When erosional surface 276b is classified into layer 270, the portion of layer 270 above erosional surface 276b becomes
20 inconsistent. Only after all of erosional surface 276a-c has been embedded into layers 270 and 272, as shown in Fig. 15e, is the extent of the inconsistent region known. Consequently, the best that can be done during the embedding of erosional surface 276a-c is to record
25 whenever the seed creates an inconsistency in the background. In a second processing step, after all regions are completely defined, an algorithm can be invoked to resolve inconsistencies.

After the classification of erosional surface
30 276a-c, illustrated in Figs. 15c-e, inconsistent subregions exist within layers 270 and 272 (subregions 278 and 280, respectively). According to the rule, both inconsistent subregions eventually must be merged with layer 274. This has two consequences for the material
35 property assignment in each layer. First, layer 272 is

- 33 -

split into layers 282 and 284, illustrated in Fig. 15f, so their properties must be updated accordingly. Second, merging is a pairwise operation. Intermediate property assignments must be generated as intermediate subregions
5 are formed. For example, subregions 278 and 280 could be merged first, and then the result would be merged with layer 274. Alternatively, subregion 280 could be merged with layer 274 first, and then subregion 278 could be merged with the result. Depending on the work to be
10 performed to update material properties after each merge, this can be an expensive computation.

Resolving a background inconsistency is more expensive than resolving a seed inconsistency, because of the larger number of splits and subsequent merges.
15 Editing an existing model leads unavoidably to background inconsistencies, but careful ordering of the sequence in which a model is built can trade costly background inconsistency detection and resolution for cheaper seed inconsistency detection and resolution. In general, it
20 is best to first classify faults, then erosional surfaces, then intrusional surfaces (such as salt domes), and last, horizons.

Classification of a surface or volume against a model can be treated as a rule-based restriction of an
25 ordinary set-theoretic union. As a consequence, an application can define its own family of custom classification operators by defining an appropriate set of rules. For example, rules referring to structural characteristics can be used in conjunction with the union
30 operator to create customized classification operators. For example, one possible implementation of the insert operator can be used to embed a salt body 290 into a model, as shown in Figs. 16a-c.

Assume that a rule specifies that salt body 290
35 does not have internal boundaries. Unioning of the salt

- 34 -

body 290 and the model consisting of layers 292, 294, and 296 and horizons 298a-c and 300a-c yield the result shown in Fig. 16b. During classification of the salt body, background inconsistencies will be detected because
5 pieces of horizons 298a-c and 300a-c get embedded in the salt body and subdivide it, as shown in Fig. 16b. The resolution algorithm associated with the rule will be called which removes horizon cells 298b and 300b from the salt body. The end result is shown in Fig. 16c. The
10 salt body is embedded in the layer sequence, and has no internal structure as required by the rule. Its shape and internal structure are the same as at the beginning of the classification.

Applications can create their own customized
15 topology operators by defining a classification operator as rule-based set-theoretic union. When the resolution of inconsistencies involves pruning of model parts, these operators can be as efficient as built-in classification methods.

20 Another example is the rule of focused classification which accounts for the tendency of users to create features that are larger than they need to be. For example, a region 310 may be separated into three fault blocks 312, 314, 316, separated by fault 318 and
25 horizon 320, as shown in Fig. 17a. It is desired to classify into region 310 a new horizon 322, shown in Fig. 17b, which is large enough to extend entirely across region 310. It is the user's intent that the horizon extend only across fault block 316, but, in an effort to
30 make the horizon large enough, the user has made horizon 322 too large. If horizon 322 were classified into the region, it would extend across the entire region, as shown in Fig. 17c.

- 35 -

A focused classification satisfies the following rule, which is implemented by the inconsistency resolution method:

5 Focused classification rule (detection): Subvolume subdivision is permitted if and only if the application says so.

Focused classification rule (resolution):

Deactivate the region of overlap in the horizon and make no change in the unallowed block.

10 Applying this rule to the example just discussed, if a horizon 322 overlaps an unallowed fault block 314, as shown in Fig. 17d, the inconsistency resolution method deactivates the portion of the horizon that overlaps the unallowed fault block, illustrated by the dashed portion
15 of horizon 322 in Fig. 17d.

The rules-based system operates as follows: the application assigns consistency enforcement rules to volumes 330, as shown in Fig. 18. When a surface is connected to a volume 332, the system begins the
20 classification process 334. As part of the classification process, a detection method is invoked for each of the callback events spawned by the classification 336. The detection method records the detected inconsistencies. At the conclusion of the classification
25 process 338, the system determines if any inconsistencies were detected during classification 340.

If inconsistencies were detected, a resolution method is invoked for each of the inconsistencies 342. The system then reports to the application the status of
30 the inconsistency resolution methods 344. That is, the system reports what inconsistencies, if any, remain.

The application decides how to handle the remaining inconsistencies 346. The application's processing may include prompting the user for help in

- 36 -

resolving the problem or it may simply report the inconsistency to the user.

Inconsistency resolution is under the control of the system from the time that classification begins 334, until all inconsistencies are reported 344, as shown by line 350.

The invention may be implemented in hardware or software, or a combination of both. However, preferably, the invention is implemented in computer programs
10 executing on programmable computers each comprising a processor, a data storage system (including volatile and non-volatile memory and/or storage elements), at least one input device, and at least one output device. Program code is applied to input data to perform the
15 functions described above and generate output information. The output information is applied to one or more output devices, in known fashion.

Each program is preferably implemented in a high level procedural or object oriented programming language
20 (such as C++ or C) to communicate with a computer system. However, the programs can be implemented in assembly or machine language, if desired. In any case, the language may be a compiled or interpreted language.

Each such computer program is preferably stored on
25 a storage media or device (e.g., ROM or magnetic/optical disk or diskette) readable by a general or special purpose programmable computer, for configuring and operating the computer when the storage media or device is read by the computer to perform the procedures
30 described herein. The inventive system may also be considered to be implemented as a computer-readable storage medium, configured with a computer program, where the storage medium so configured causes a computer to operate in a specific and predefined manner to perform
35 the functions described herein.

WO 99/53342

PCT/US98/07119

- 37 -

Other embodiments are within the scope of the following claims.

What is claimed is:

- 38 -

1. A method for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising enforcing consistency within the geoscience model.
- 5 2. The method of claim 1, wherein enforcing consistency comprises applying rules.
3. The method of claim 1, wherein enforcing consistency comprises
- 10 defining consistency.
4. The method of claim 1, wherein enforcing consistency comprises detecting inconsistency.
5. The method of claim 4, wherein detecting
- 15 inconsistency comprises detecting a seed inconsistency.
6. The method of claim 4, wherein detecting inconsistency comprises detecting a background inconsistency.
- 20 7. The method of claim 4, wherein detecting inconsistency comprises detecting a seed inconsistency when a seed is classified into a background.
8. The method of claim 4, wherein detecting
- 25 inconsistency comprises detecting a background inconsistency when a seed is classified into a background.
9. The method of claim 4, wherein detecting inconsistency comprises
- 30 detecting a potential inconsistency.
10. The method of claim 1, wherein enforcing consistency comprises resolving inconsistency.
11. The method of claim 10, wherein resolving
- 35 inconsistency comprises

- 39 -

removing a portion of a seed from a topologically active part of the geoscience model.

12. The method of claim 10, wherein resolving inconsistency comprises

5 removing a portion of a background from a topologically active part of the geoscience model.

13. The method of claim 10, wherein resolving inconsistency comprises

changing a geometry of a seed.

10 14. The method of claim 10, wherein resolving inconsistency comprises

changing a geometry of a portion of a background.

15. The method of claim 10, wherein resolving inconsistency comprises

15 flagging an inconsistency for later resolution.

16. The method of claim 4, wherein detecting inconsistency comprises

20 detecting an embedded surface cell's age being outside the age range of a host layer for the embedded surface cell.

17. The method of claim 4, wherein detecting inconsistency comprises

25 detecting a point in a subvolume having an age not within the range defined by the subvolume's bounding erosional or depositional surfaces.

18. The method of claim 4, wherein detecting inconsistency comprises

detecting a bed boundary not connected to two or more distinct subvolumes.

30 19. The method of claim 4, wherein detecting inconsistency comprises

detecting a horizon which terminates in a layer.

20. The method of claim 4, wherein detecting inconsistency comprises

- 40 -

detecting a child subvolume having a volume less than a predefined percentage of the child subvolume's parent's volume.

21. The method of claim 4, wherein detecting
5 inconsistency comprises

detecting a ratio of a subvolume's volume to its surface being less than a predetermined amount.

22. The method of claim 4, wherein detecting
inconsistency comprises
10 detecting a first age assignment from a first horizon that is inconsistent with a second age assignment from a second horizon.

23. The method of claim 4, wherein detecting
inconsistency comprises
15 detecting a pinched-out layer.

24. The method of claim 4, wherein detecting
inconsistency comprises

detecting a layer forming lenses.

25. The method of claim 4, wherein detecting
20 inconsistency comprises

detecting a cell of a feature embedded into a sub-region where the feature is not allowed.

26. The method of claim 10, wherein resolving
inconsistency comprises
25 asking how to extend the surface.

27. The method of claim 10, wherein resolving
inconsistency comprises

extending a horizon to a nearest surface bounding the layer.

28. The method of claim 10, wherein resolving
30 inconsistency comprises

extending a horizon to a geoscience model boundary.

29. The method of claim 10, wherein resolving
35 inconsistency comprises

- 41 -

removing a segment of a feature which extends into an unallowed sub-region.

30. The method of claim 10, wherein resolving inconsistency comprises

5 identifying a horizon that contributes to the inconsistency; and

removing a segment of the horizon that contributes to the inconsistency.

31. The method of claim 10, wherein resolving
10 inconsistency comprises

identifying a first horizon and a second horizon that contribute to the inconsistency; and

reshaping the first horizon and the second horizon to maintain a distance between them that is greater than
15 or equal to a preselected amount.

32. The method of claim 10, wherein resolving inconsistency comprises

collapsing a small sub-region bounded by a horizon.

20 33. The method of claim 32, wherein collapsing comprises

removing a portion of the horizon that bounds the small sub-region.

34. The method of claim 10, wherein resolving
25 inconsistency comprises

identifying a sub-region created by invalid horizons; and

reshaping the invalid horizons to produce a correctly-shaped sub-region.

30 35. The method of claim 5, further comprising blocking property updates.

36. A method for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising
35 defining consistency;

- 42 -

detecting an inconsistency; and
resolving the inconsistency.

37. A computer system for analyzing geological
data sampled from a subsurface region stored in a
5 geoscience model on a computer-readable media comprising
means for enforcing consistency within the
geoscience model.

38. The computer system of claim 37, wherein the
enforcing consistency means comprises
10 means for applying rules.

39. The computer system of claim 37, wherein the
enforcing consistency means comprises
means for defining consistency.

40. The computer system of claim 37, wherein the
15 enforcing consistency means comprises
means for detecting inconsistency.

41. The computer system of claim 40, wherein the
detecting inconsistency means comprises
means for detecting a seed inconsistency.

20 42. The computer system of claim 40, wherein the
detecting inconsistency means comprises
means for detecting a background inconsistency.

43. The computer system of claim 40, wherein the
detecting inconsistency means comprises
25 means for detecting a seed inconsistency when a
seed is classified into a background.

44. The computer system of claim 40, wherein the
detecting inconsistency means comprises
means for detecting a background inconsistency
30 when a seed is classified into a background.

45. The computer system of claim 40, wherein the
detecting inconsistency means comprises
means for detecting a potential inconsistency.

46. The computer system of claim 37, wherein the
35 enforcing consistency means comprises

- 43 -

means for resolving inconsistency.

47. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for removing a portion of a seed from a
5 topologically active part of the geoscience model.

48. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for removing a portion of a background from
a topologically active part of the geoscience model.

10 49. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for changing a geometry of a seed.

50. The computer system of claim 46, wherein the resolving inconsistency means comprises

15 means for changing a geometry of a portion of a background.

51. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for flagging an inconsistency for later
20 resolution.

52. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting an embedded surface cell's age
being outside the age range of a host layer for the
25 embedded surface cell.

53. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a point in a subvolume having
an age not within the range defined by the subvolume's
30 bounding erosional or depositional surfaces.

54. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a bed boundary not connected
to a two or more distinct subvolumes.

- 44 -

55. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a horizon which terminates in a layer.

5 56. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a child subvolume having a volume less than a predefined percentage of the child subvolume's parent's volume.

10 57. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a ratio of a subvolume's volume to its surface being less than a predetermined amount.

15 58. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a first age assignment from a first horizon that is inconsistent with a second age assignment from a second horizon.

20 59. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a pinched-out layer.

60. The computer system of claim 40, wherein the detecting inconsistency means comprises

25 means for detecting a layer forming lenses.

61. The computer system of claim 40, wherein the detecting inconsistency means comprises

means for detecting a cell of a feature embedded into a sub-region where the feature is not allowed.

30 62. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for asking how to extend the surface.

63. The computer system of claim 46, wherein the resolving inconsistency means comprises

- 45 -

means for extending a horizon to a nearest surface bounding the layer.

64. The computer system of claim 46, wherein the resolving inconsistency means comprises

5 means for extending a horizon to a geoscience model boundary.

65. The computer system of claim 46, wherein the resolving inconsistency means comprises

10 means for removing a segment of a feature which extends into an unallowed sub-region.

66. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for identifying a horizon that contributes to the inconsistency; and

15 means for removing a segment of the horizon that contributes to the inconsistency.

67. The computer system of claim 46, wherein the resolving inconsistency means comprises

20 means for identifying a first horizon and a second horizon that contribute to the inconsistency; and

means for reshaping the first horizon and the second horizon to maintain a distance between them that is greater than or equal to a preselected amount.

25 68. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for collapsing a small sub-region bounded by a horizon.

69. The computer system of claim 68, wherein the collapsing means comprises

30 means for removing a portion of the horizon that bounds the small sub-region.

- 46 -

70. The computer system of claim 46, wherein the resolving inconsistency means comprises

means for identifying a sub-region created by invalid horizons; and

5 means for reshaping the invalid horizons to produce a correctly-shaped sub-region.

71. The computer system of claim 41, further comprising

means for blocking property updates.

10 72. A computer system for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising
means for defining consistency;
means for detecting an inconsistency; and
15 means for resolving the inconsistency.

73. A computer program, residing on a computer-readable medium, for analyzing geological data sampled from a subsurface region stored in a geoscience model on a computer-readable media comprising instructions for
20 causing the computer to

enforce consistency within the geoscience model.

74. The computer program of claim 73, wherein the instructions for causing the computer to enforce consistency comprise

25 instructions for causing the computer to apply rules.

75. The computer program of claim 73, wherein the instructions for causing the computer to enforce consistency comprise

30 instructions for causing the computer to define consistency.

76. The computer program of claim 73, wherein the instructions for causing the computer to enforce consistency comprise

- 47 -

instructions for causing the computer to detect inconsistency.

77. The computer program of claim 76, wherein the instructions for causing the computer to detect
5 inconsistency comprise

instructions for causing the computer to detect a seed inconsistency.

78. The computer program of claim 76, wherein the instructions for causing the computer to detect
10 inconsistency comprise

instructions for causing the computer to detect a background inconsistency.

79. The computer program of claim 76, wherein the instructions for causing the computer to detect
15 inconsistency comprise

instructions for causing the computer to detect a seed inconsistency when a seed is classified into a background.

80. The computer program of claim 76, wherein the
20 instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a background inconsistency when a seed is classified into a background.

25 81. The computer program of claim 76, wherein the instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a potential inconsistency.

30 82. The computer program of claim 73, wherein the instructions for causing the computer to enforce consistency comprise

instructions for causing the computer to resolve inconsistency.

- 48 -

83. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

instructions for causing the computer to remove a
5 portion of a seed from a topologically active part of the geoscience model.

84. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

10 instructions for causing the computer to remove a portion of a background from a topologically active part of the geoscience model.

85. The computer program of claim 82, wherein the instructions for causing the computer to resolve
15 inconsistency comprise

instructions for causing the computer to change a geometry of a seed.

86. The computer program of claim 82, wherein the instructions for causing the computer to resolve
20 inconsistency comprise

instructions for causing the computer to change a geometry of a portion of a background.

87. The computer program of claim 82, wherein the instructions for causing the computer to resolve
25 inconsistency comprise

instructions for causing the computer to flag an inconsistency for later resolution.

88. The computer program of claim 76, wherein the instructions for causing the computer to detect
30 inconsistency comprise

instructions for causing the computer to detect an embedded surface cell's age being outside the age range of a host layer for the embedded surface cell.

- 49 -

89. The computer program of claim 76, wherein the instructions for causing the computer to detect inconsistency comprise

5 instructions for causing the computer to detect a point in a subvolume having an age not within the range defined by the subvolume's bounding erosional or depositional surfaces.

90. The computer program of claim 76, wherein the instructions for causing the computer to detect
10 inconsistency comprise

instructions for causing the computer to detect a bed boundary not connected to a two or more distinct subvolumes.

91. The computer program of claim 76, wherein the
15 instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a horizon which terminates in a layer.

92. The computer program of claim 76, wherein the
20 instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a child subvolume having a volume less than a predefined percentage of the child subvolume's parent's volume.

25 93. The computer program of claim 76, wherein the instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a ratio of a subvolume's volume to its surface being less
30 than a predetermined amount.

94. The computer program of claim 76, wherein the instructions for causing the computer to detect inconsistency comprise

instructions for causing the computer to detect a
35 first age assignment from a first horizon that is

- 50 -

inconsistent with a second age assignment from a second horizon.

95. The computer program of claim 76, wherein the instructions for causing the computer to detect
5 inconsistency comprise

instructions for causing the computer to detect a pinched-out layer.

96. The computer program of claim 76, wherein the instructions for causing the computer to detect
10 inconsistency comprise

instructions for causing the computer to detect a layer forming lenses.

97. The computer program of claim 76, wherein the instructions for causing the computer to detect
15 inconsistency comprise

instructions for causing the computer to detect a cell of a feature embedded into a sub-region where the feature is not allowed.

98. The computer program of claim 82, wherein the instructions for causing the computer to resolve
20 inconsistency comprise

instructions for causing the computer to ask how to extend the surface.

99. The computer program of claim 82, wherein the instructions for causing the computer to resolve
25 inconsistency comprise

instructions for causing the computer to extend a horizon to a nearest surface bounding the layer.

100. The computer program of claim 82, wherein the instructions for causing the computer to resolve
30 inconsistency comprise

instructions for causing the computer to extend a horizon to a geoscience model boundary.

- 51 -

101. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

instructions for causing the computer to remove a
5 segment of a feature which extends into an unallowed sub-region.

102. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

10 instructions for causing the computer to identify a horizon that contributes to the inconsistency; and
instructions for causing the computer to remove a segment of the horizon that contributes to the inconsistency.

15 103. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

instructions for causing the computer to identify a first horizon and a second horizon that contribute to
20 the inconsistency; and

instructions for causing the computer to reshape the first horizon and the second horizon to maintain a distance between them that is greater than or equal to a preselected amount.

25 104. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

instructions for causing the computer to collapse a small sub-region bounded by a horizon.

30 105. The computer program of claim 104, wherein the instructions for causing the computer to collapse the small sub-region comprise

instructions for causing the computer to remove a portion of the horizon that bounds the small sub-region.

- 52 -

106. The computer program of claim 82, wherein the instructions for causing the computer to resolve inconsistency comprise

instructions for causing the computer to identify
5 a sub-region created by invalid horizons; and

instructions for causing the computer to reshape the invalid horizons to produce a correctly-shaped sub-region.

107. The computer program of claim 77, to further
10 comprising

instructions for causing the computer to block property updates.

108. A computer program, residing on a computer-readable medium, for analyzing geological data sampled
15 from a subsurface region stored in a geoscience model on a computer-readable media comprising instructions for causing the computer to

define consistency;

detect an inconsistency; and

20 resolve the inconsistency.

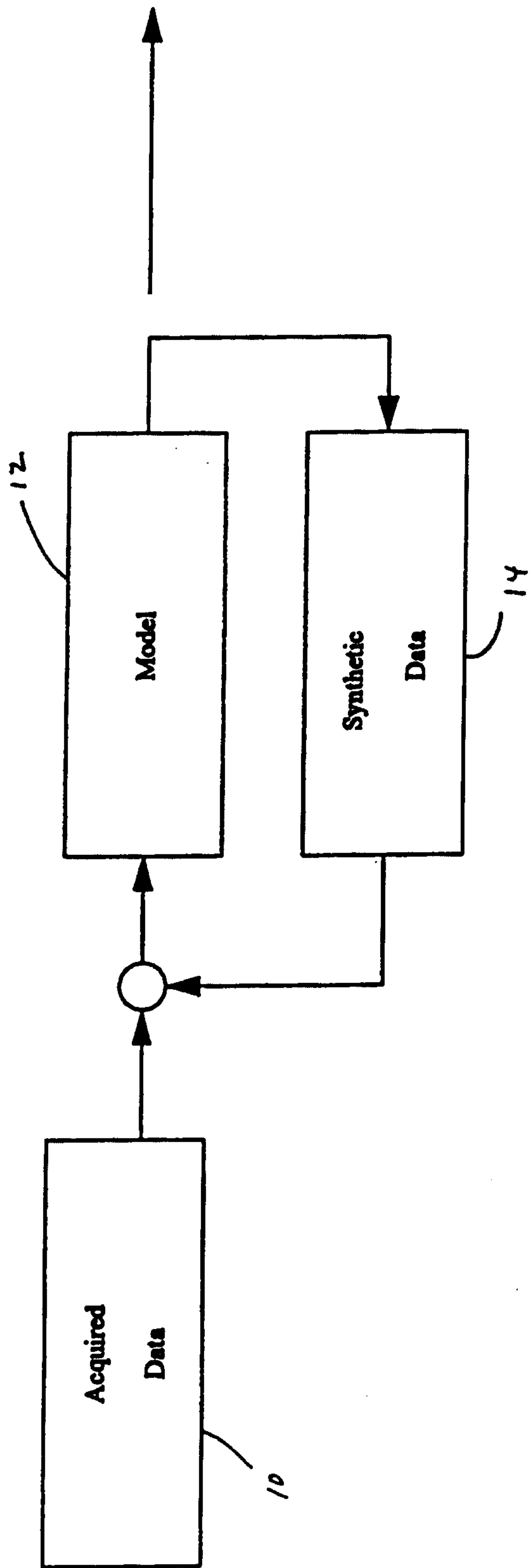
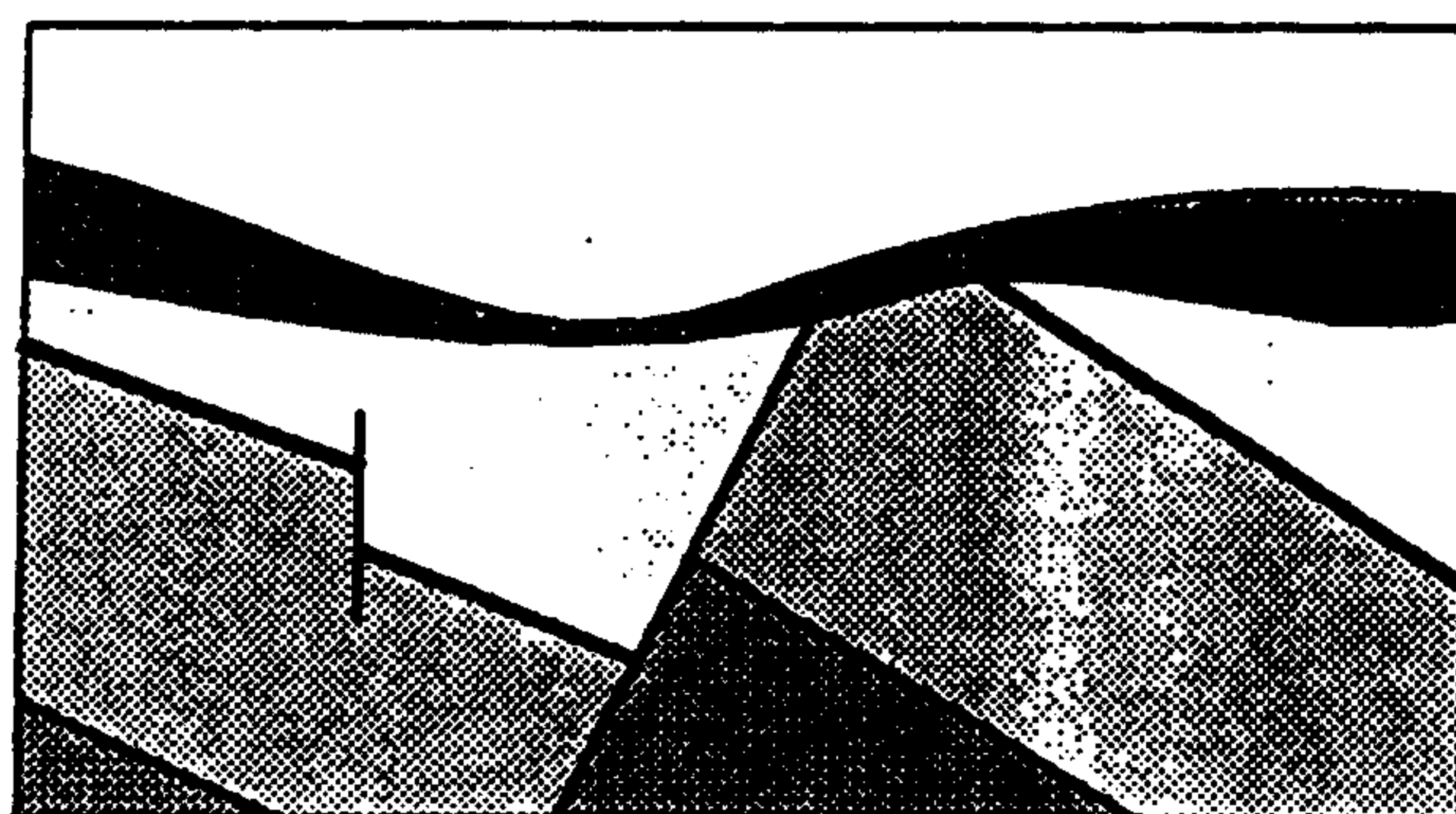
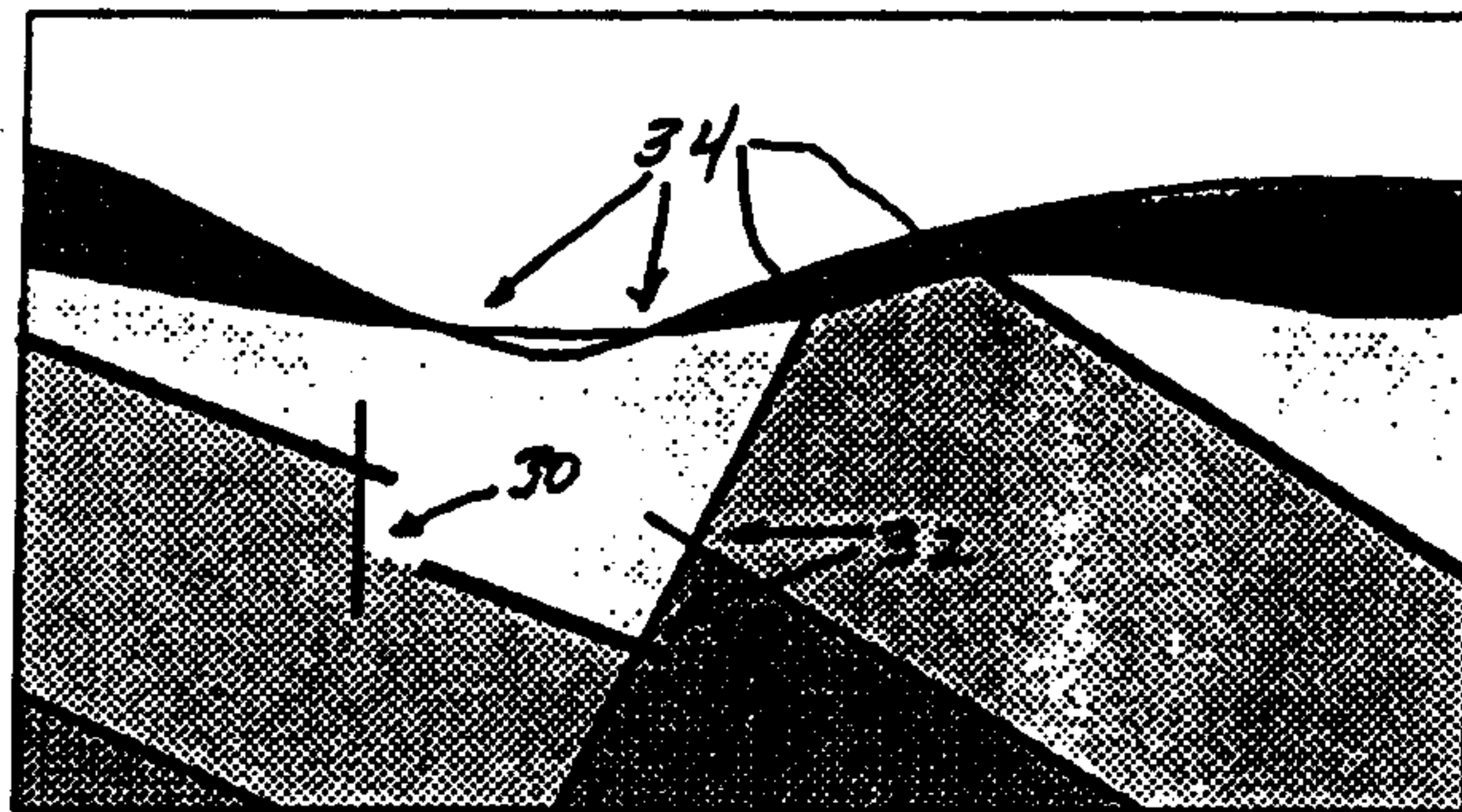
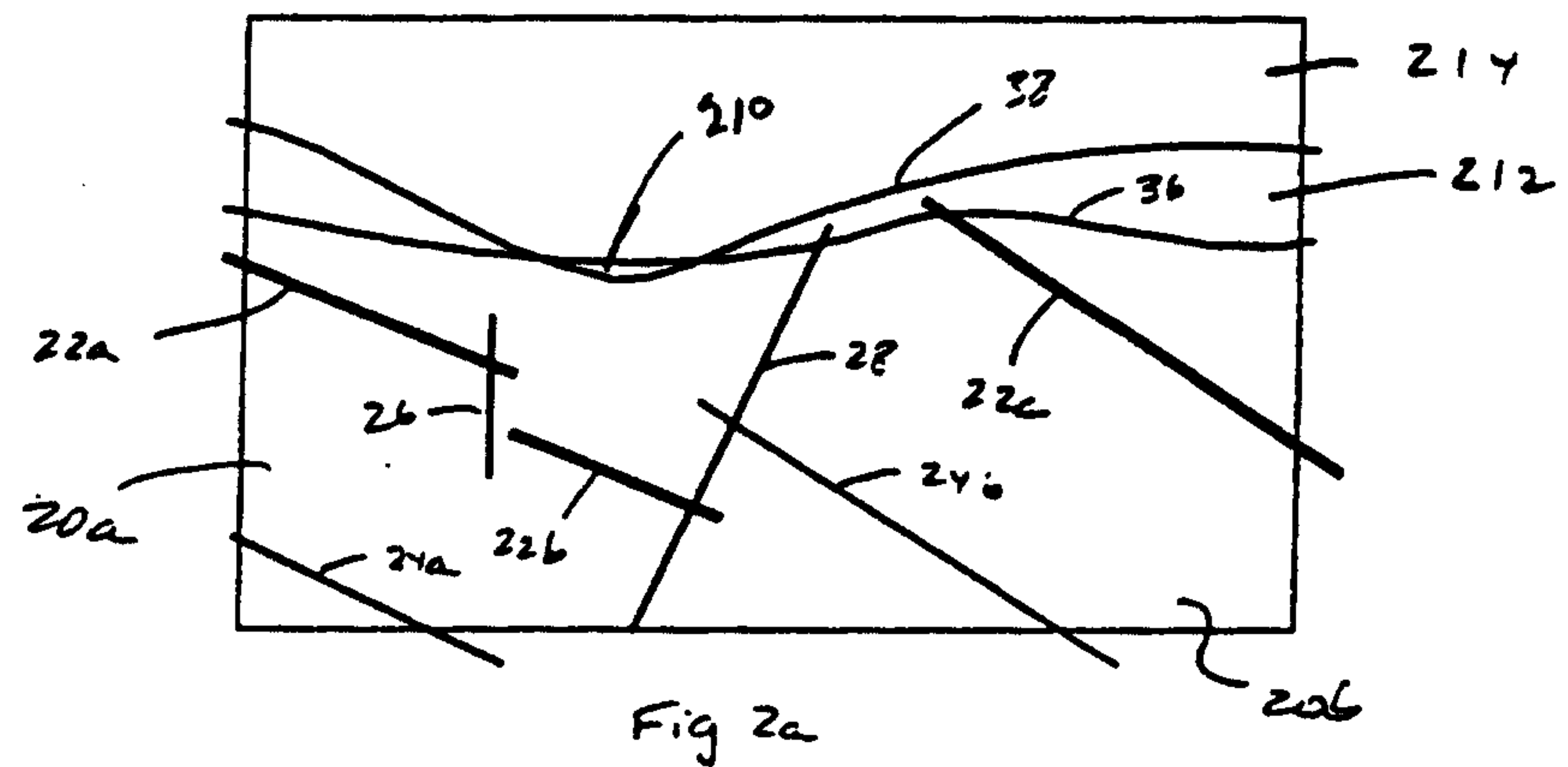


Fig. 1



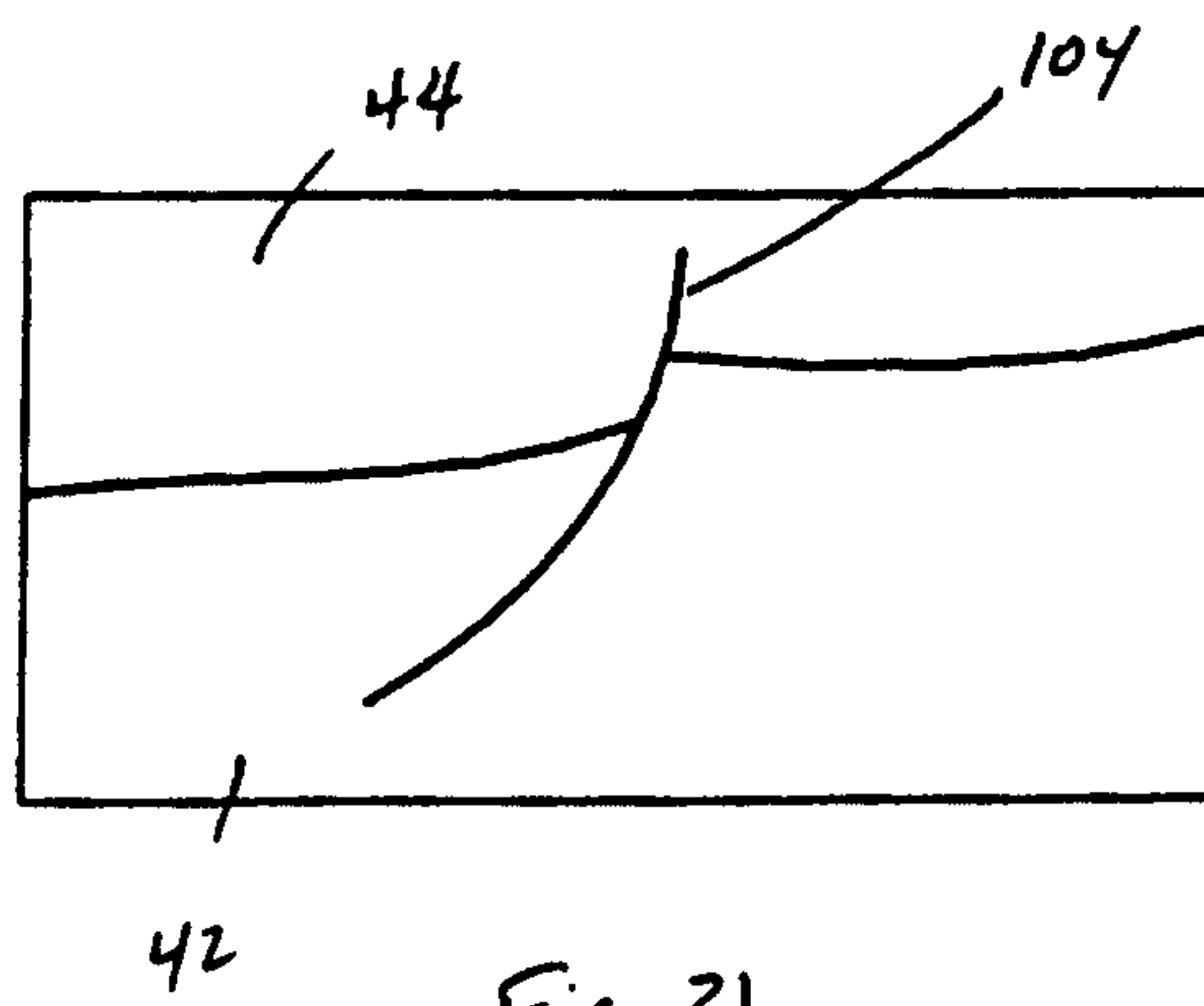


Fig. 3b

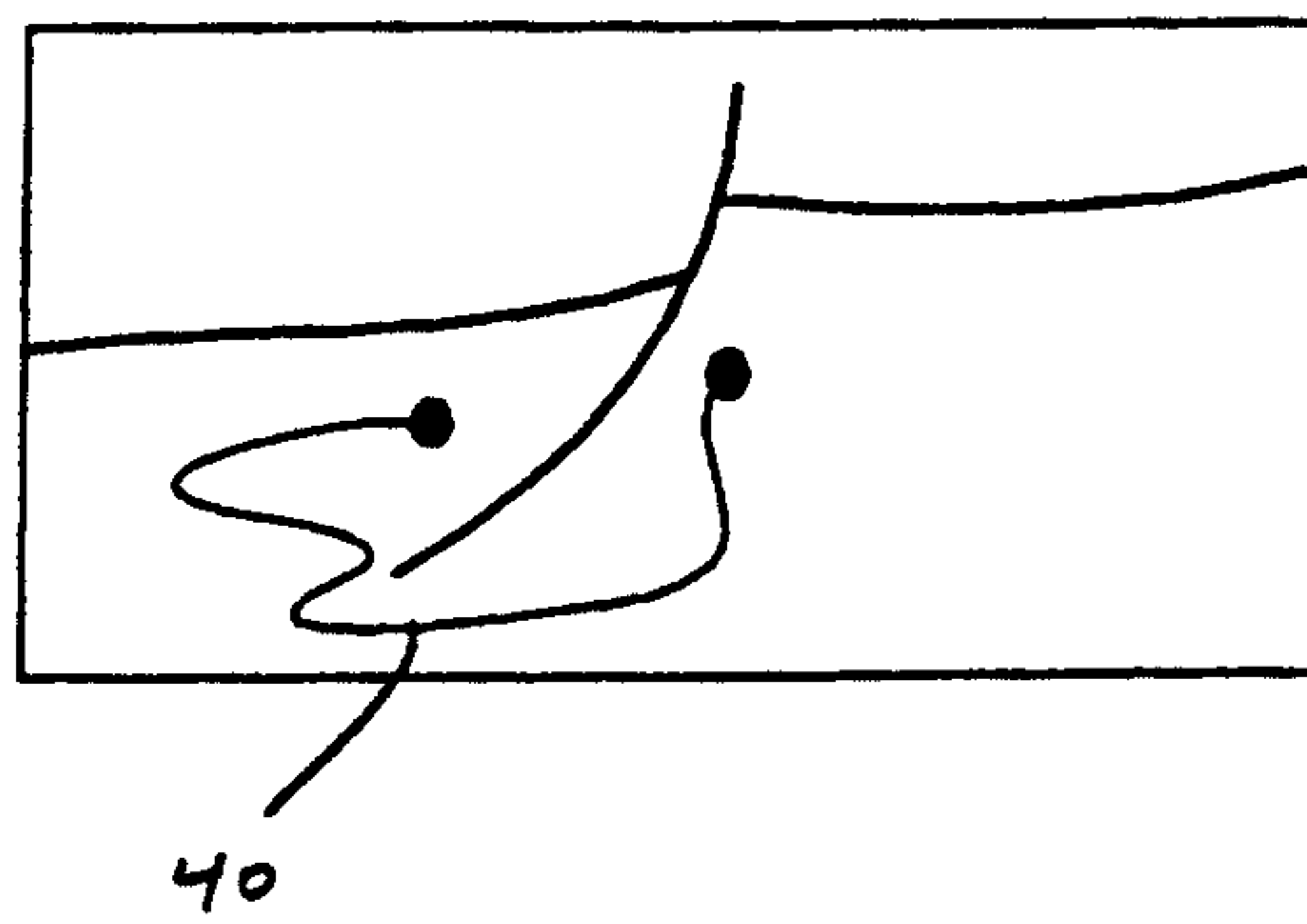


Fig. 3a

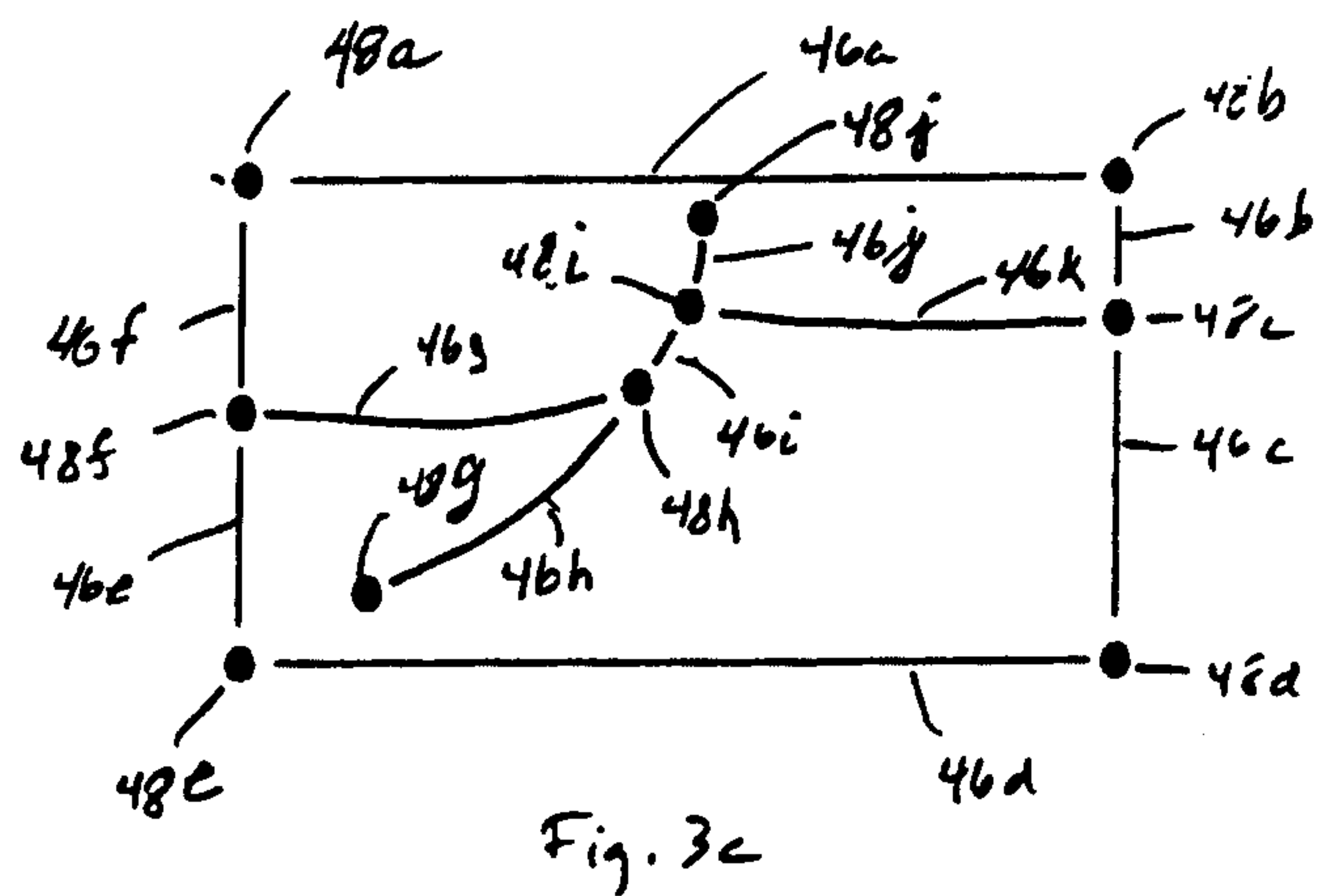


Fig. 3c

4 / 18

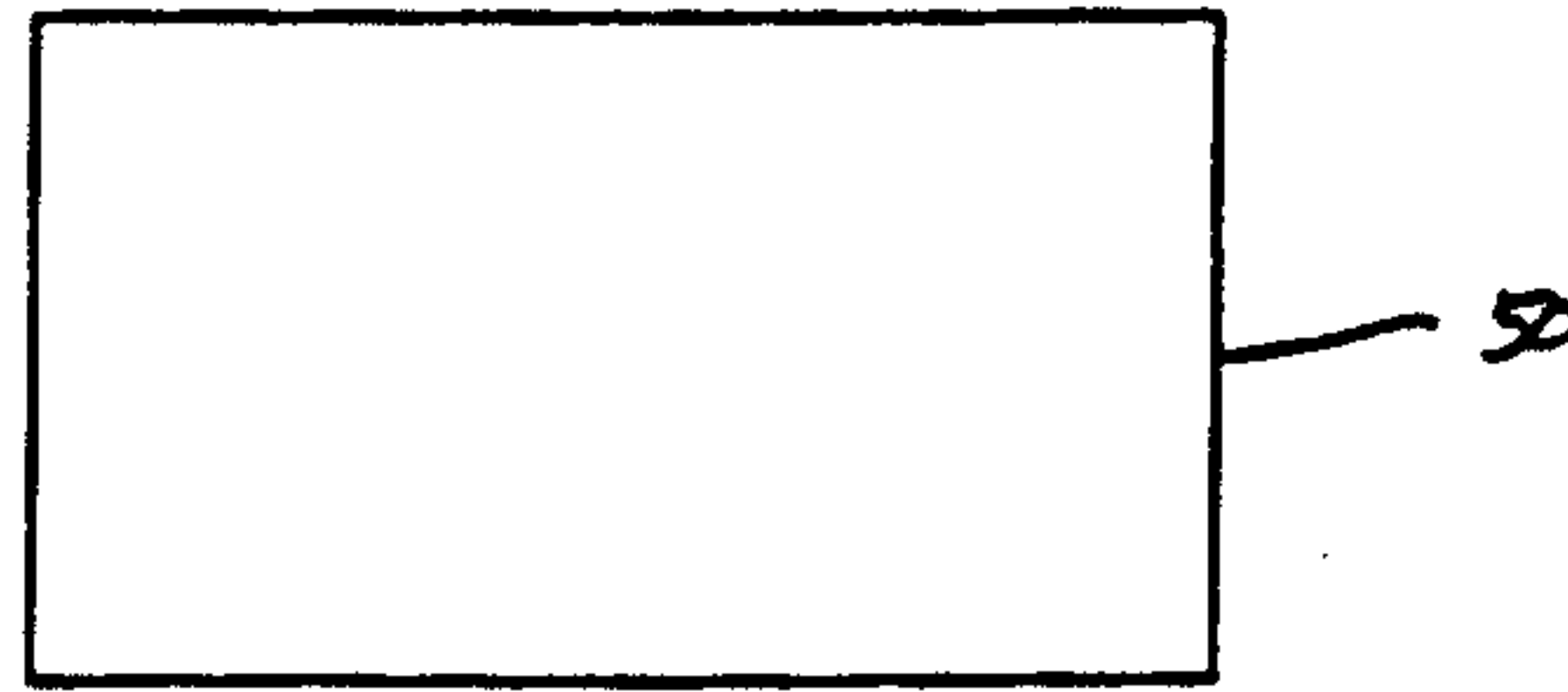


Fig. 4a

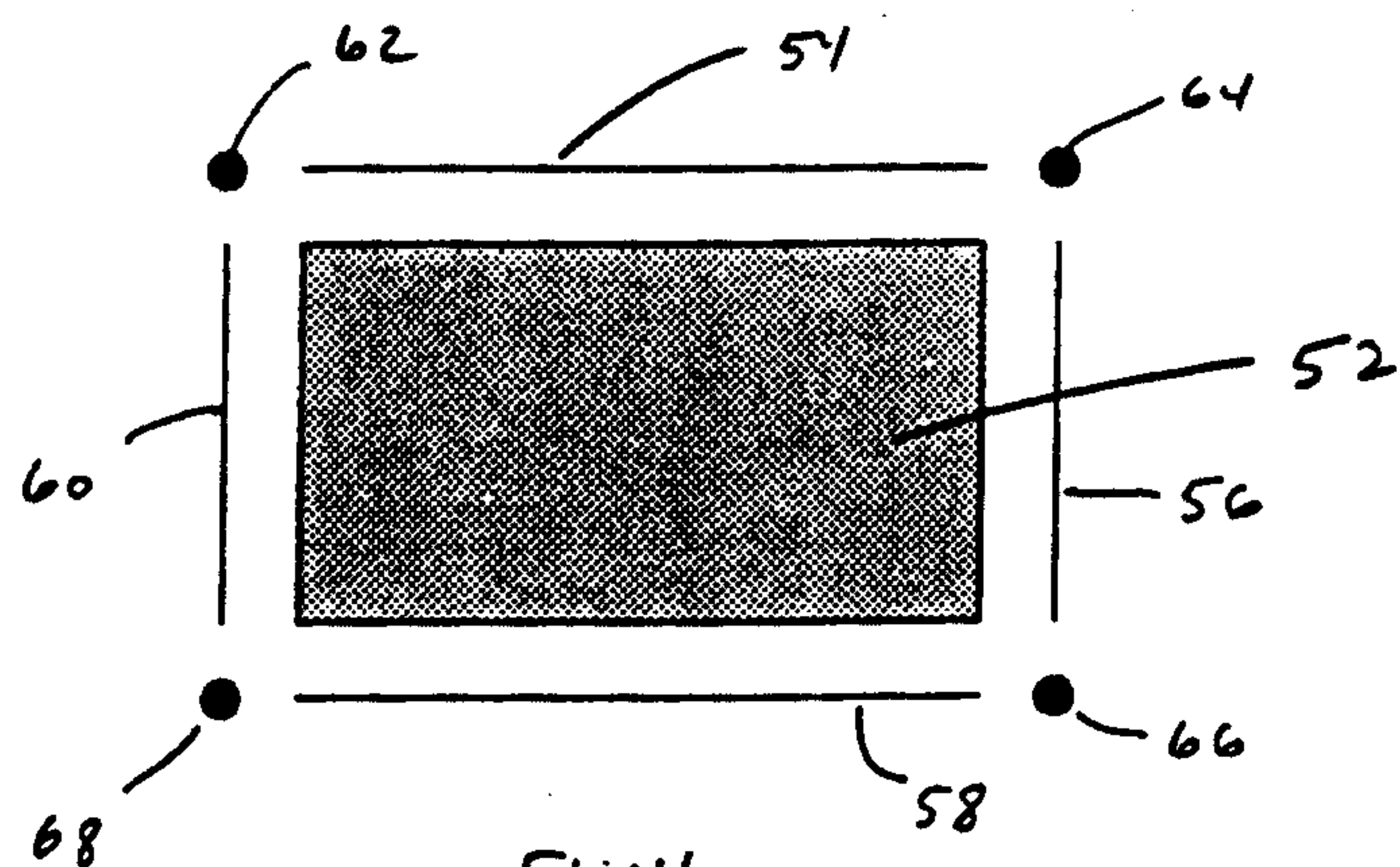


Fig. 4b

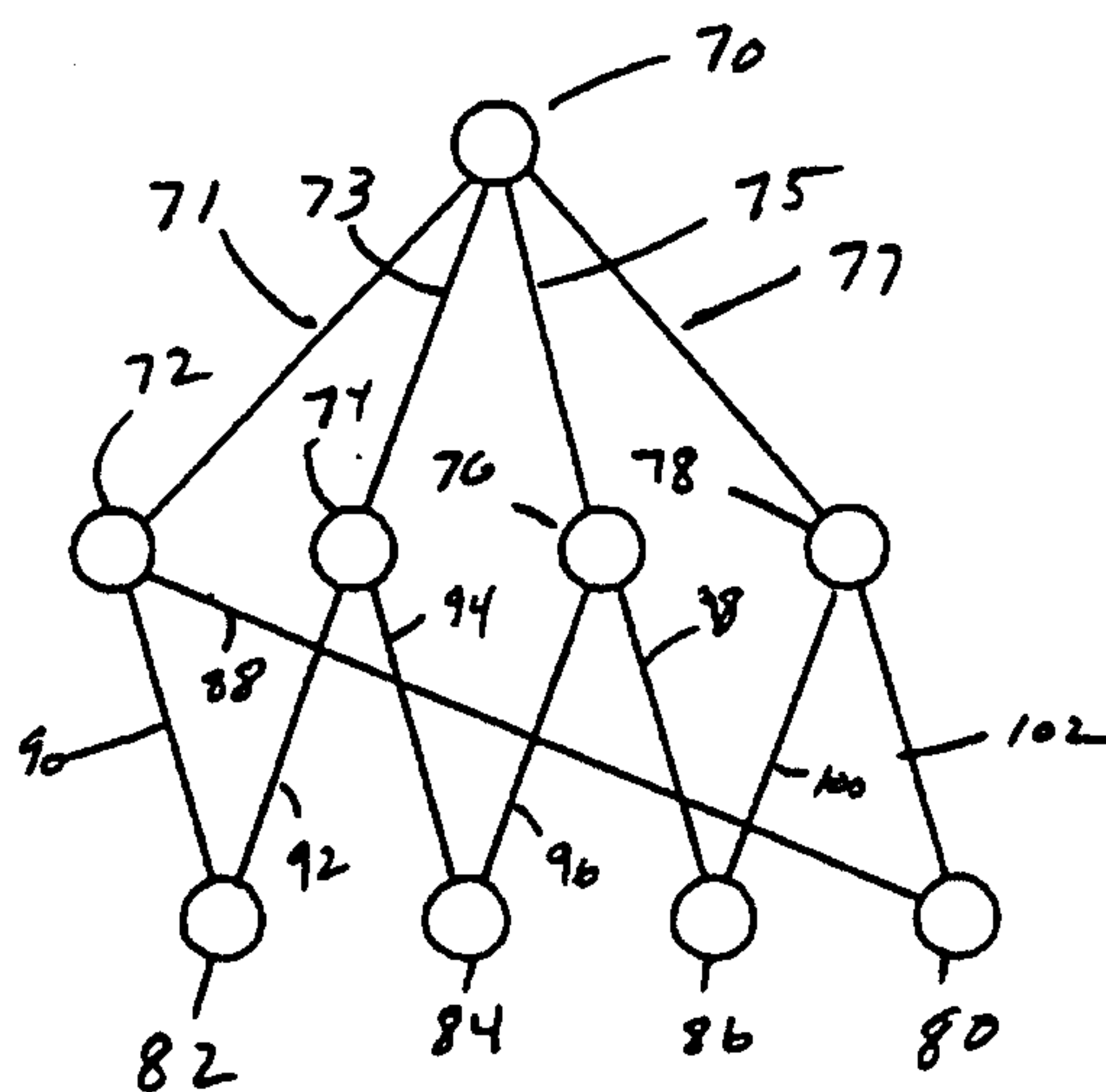


Fig. 4c

5 / 18

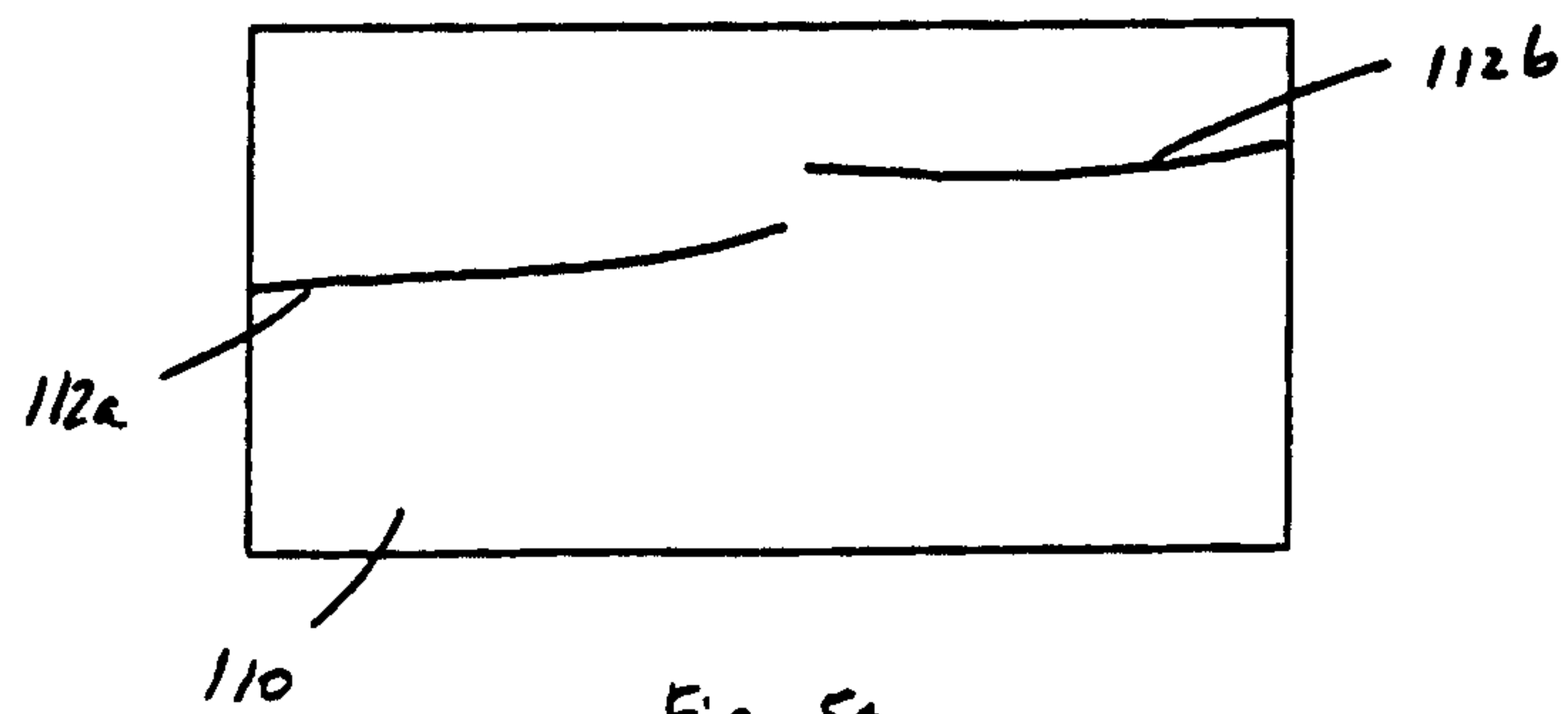


Fig. 5a

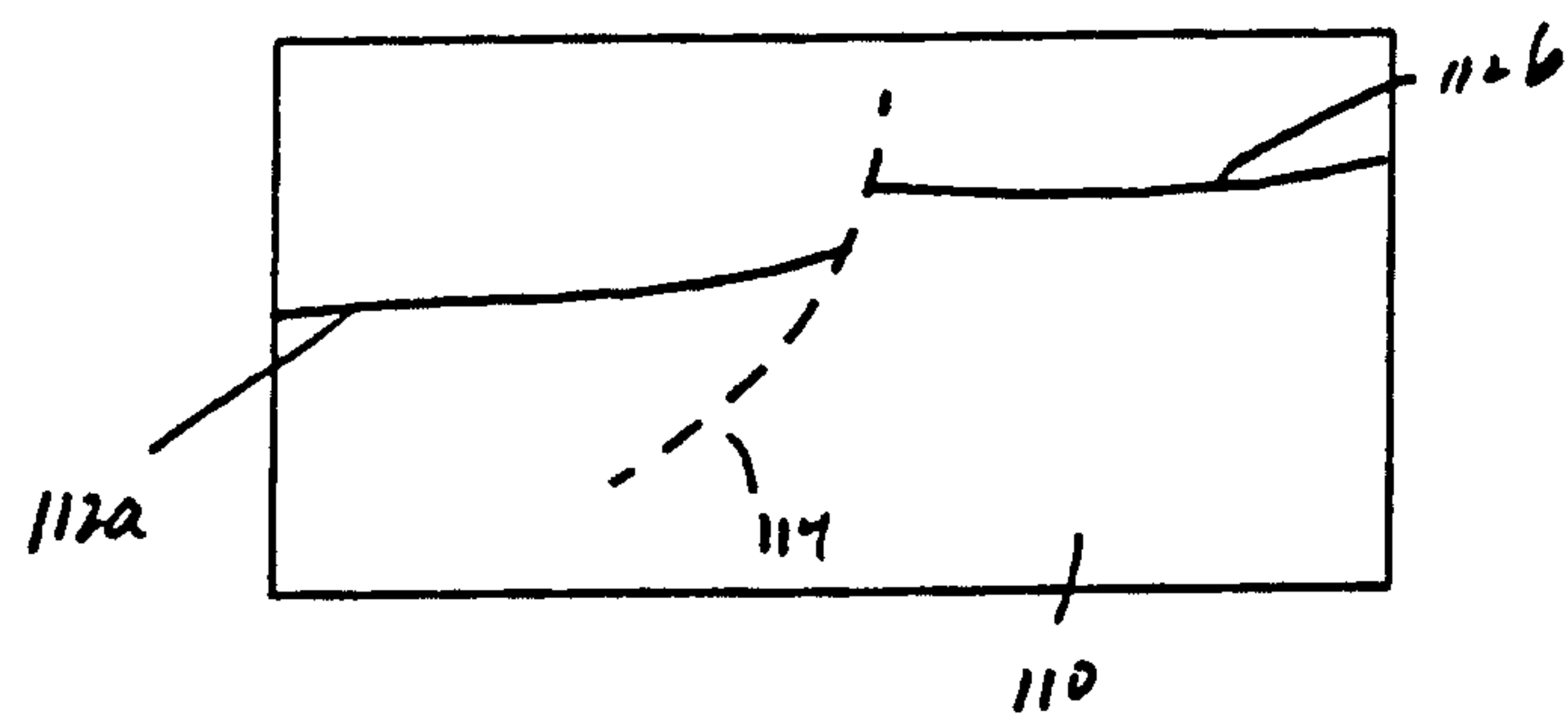


Fig. 5b

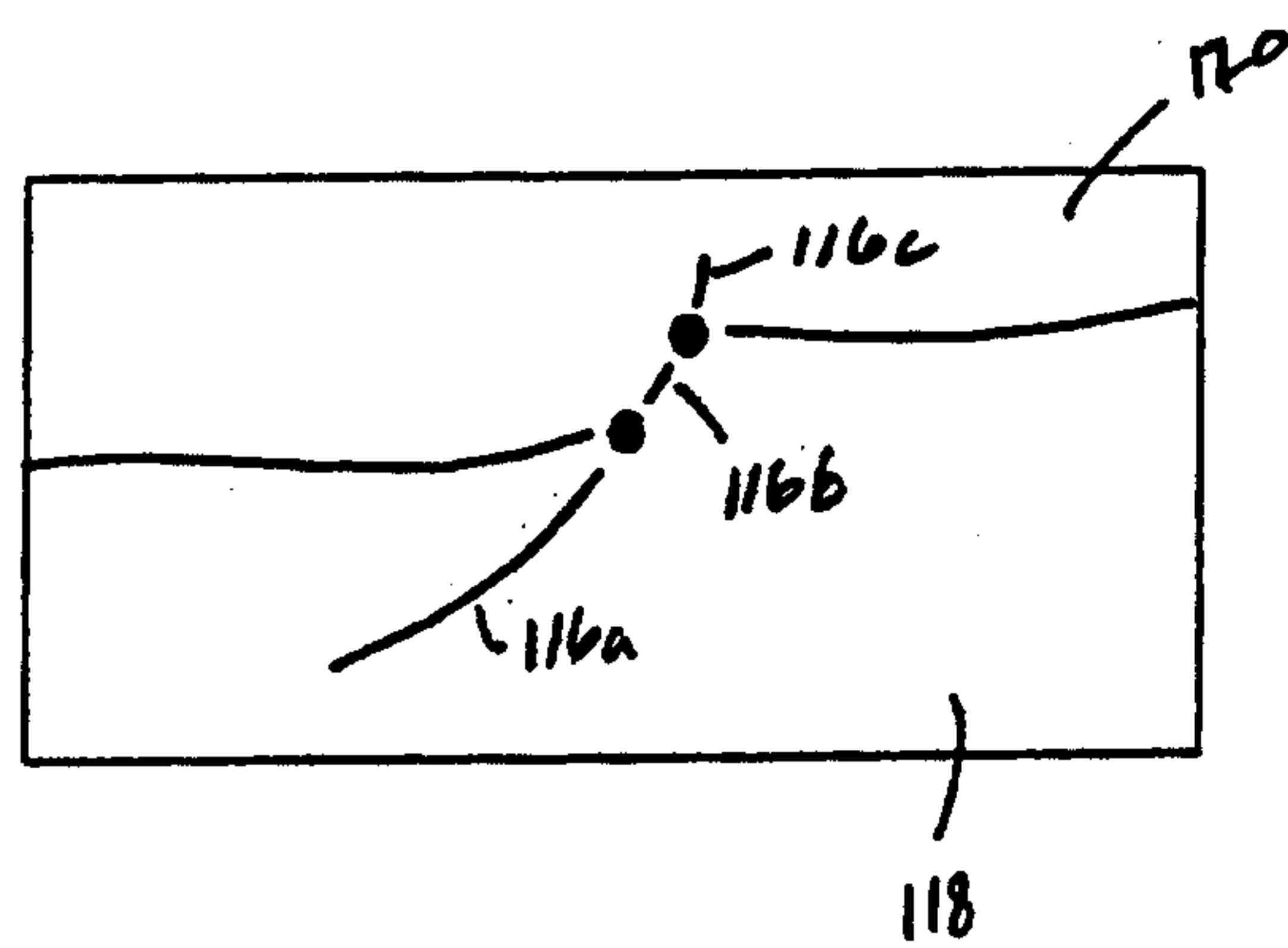
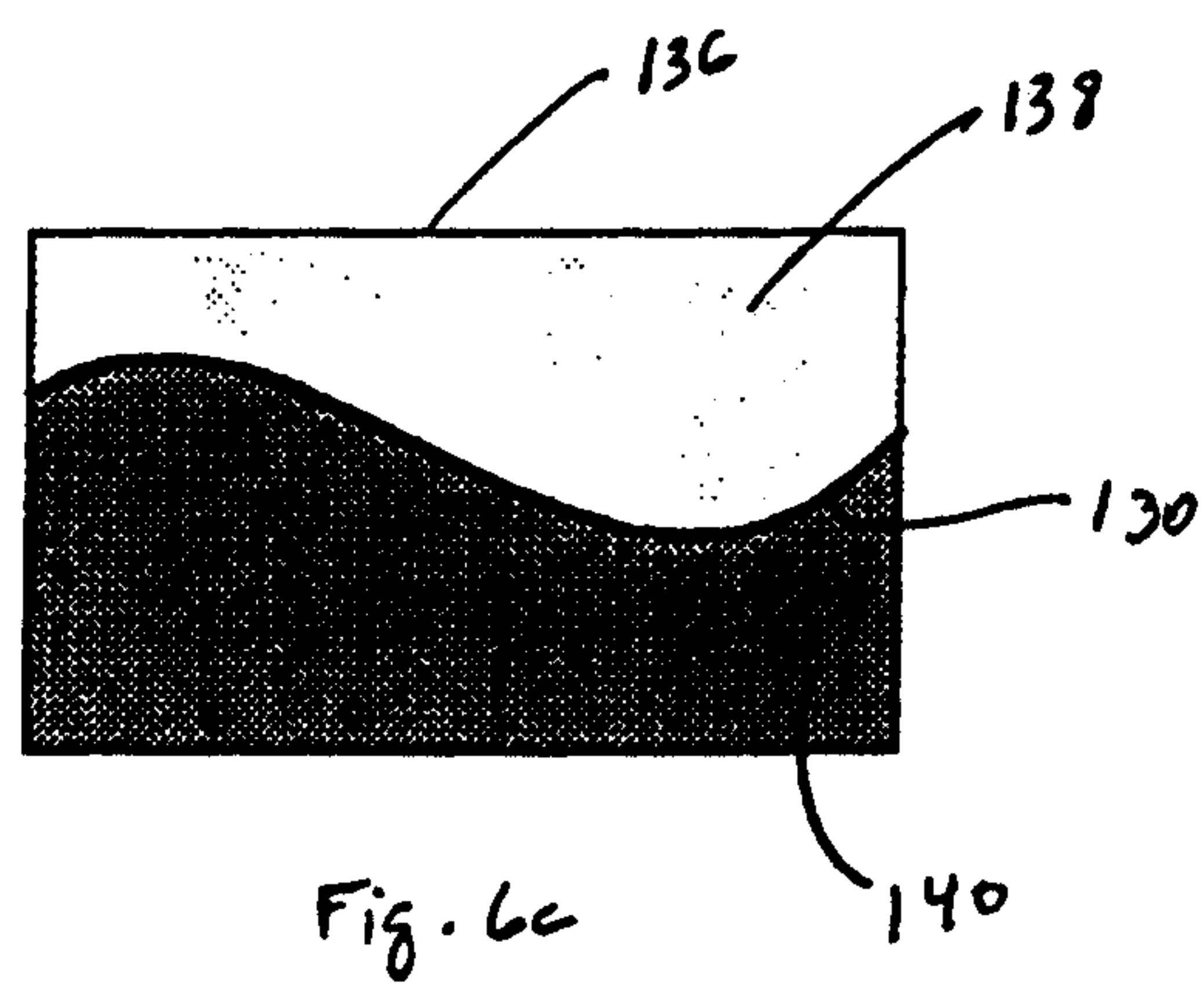
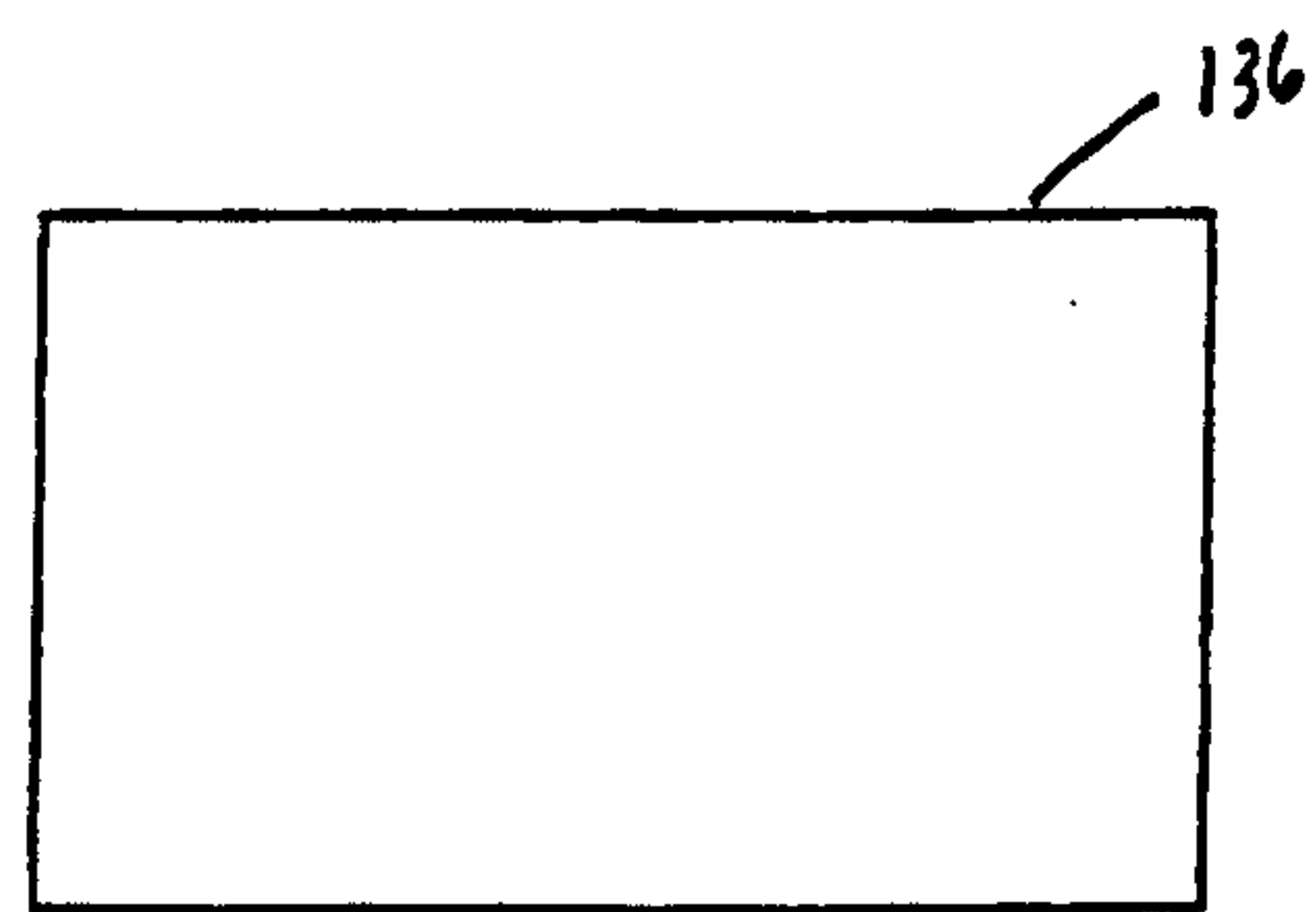
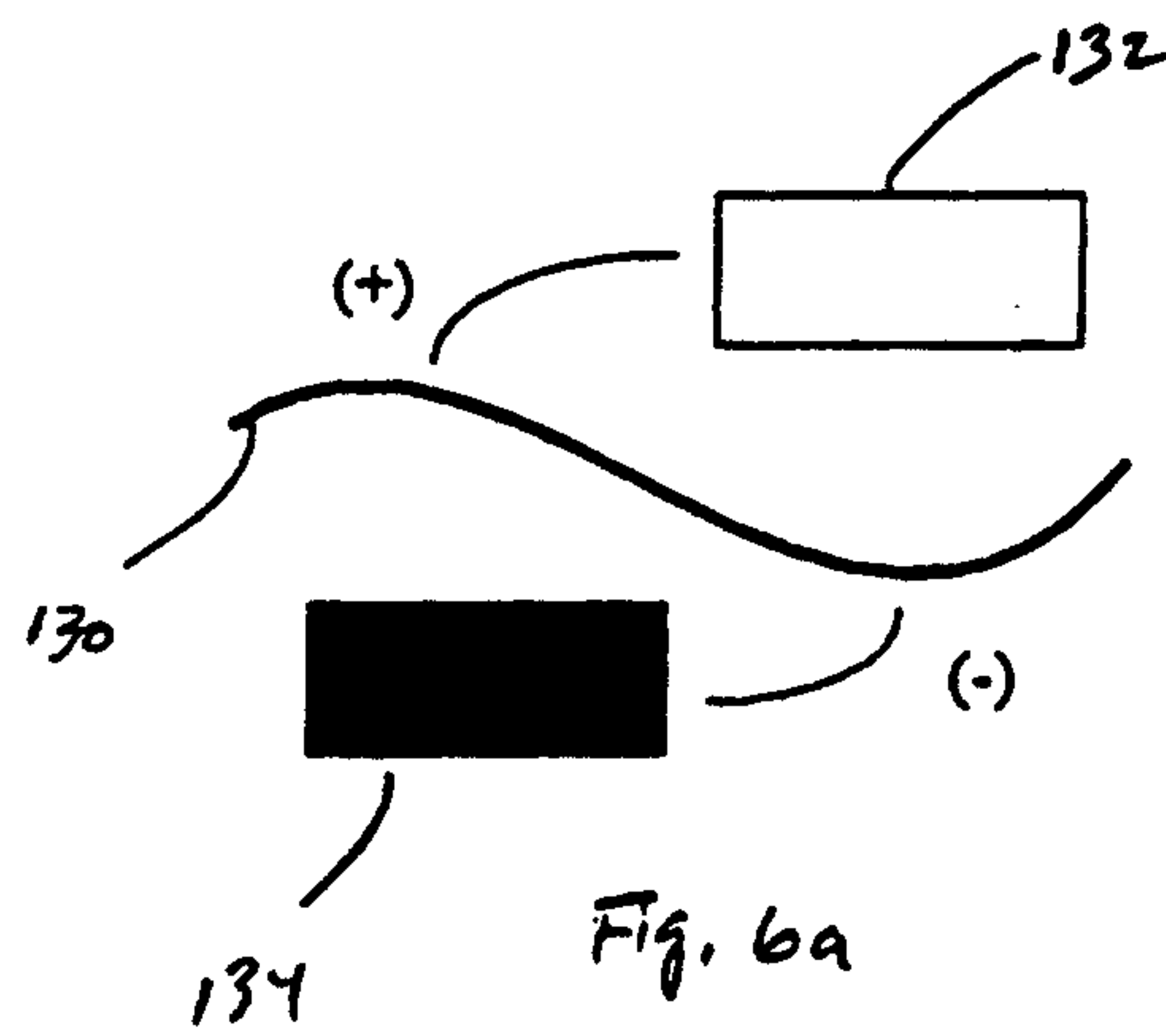
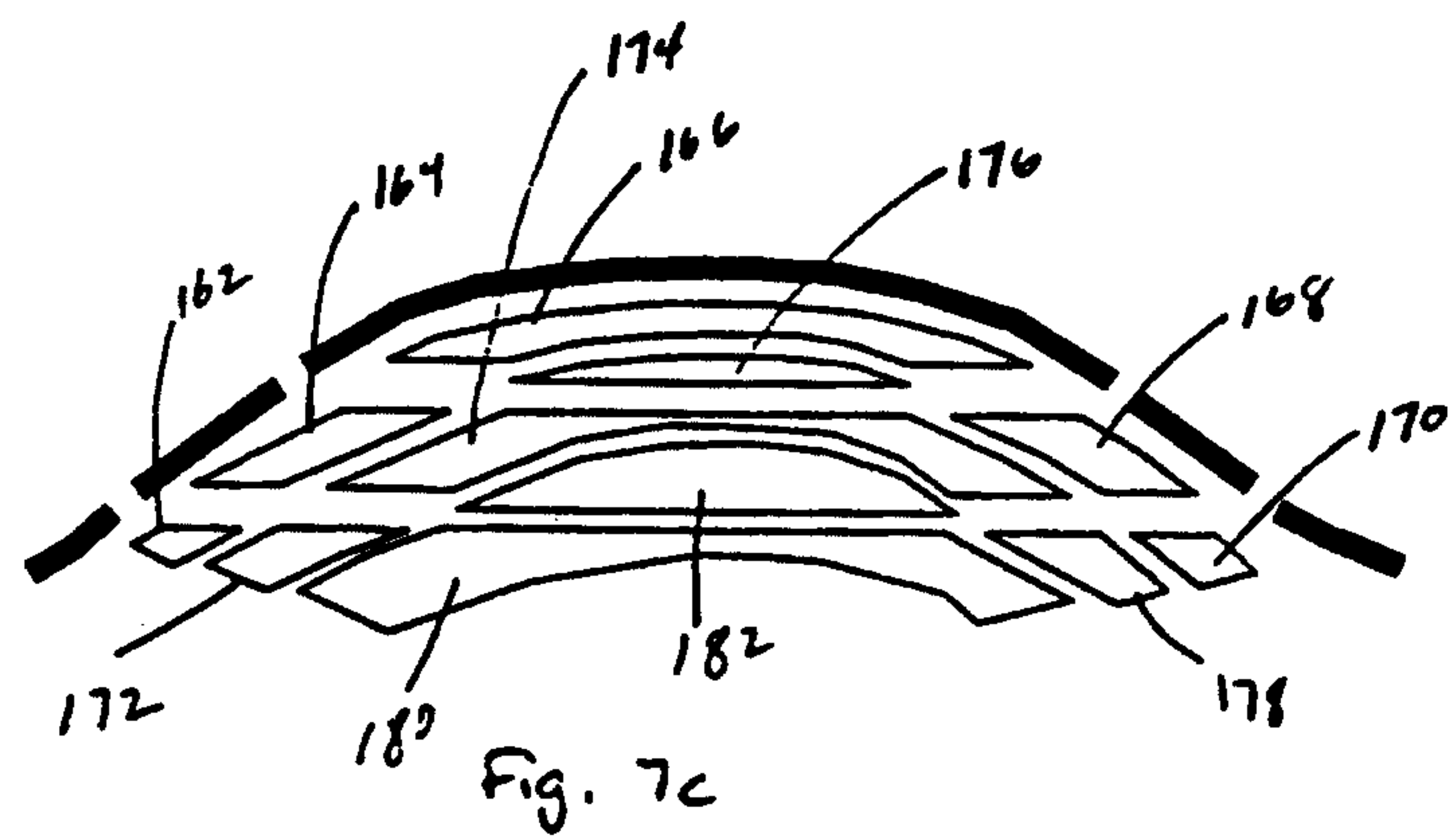
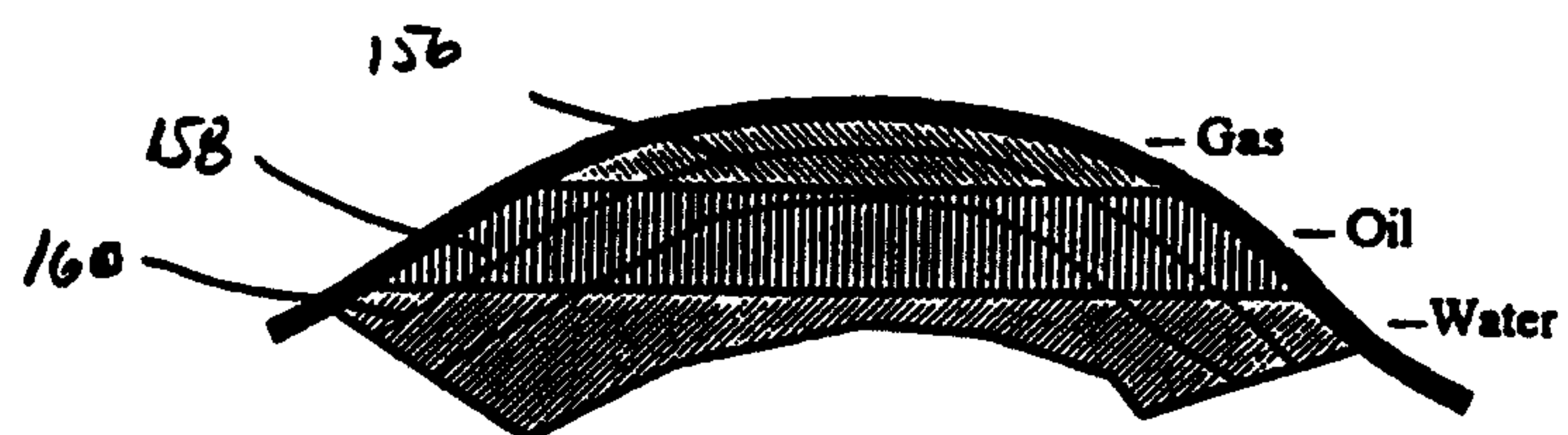
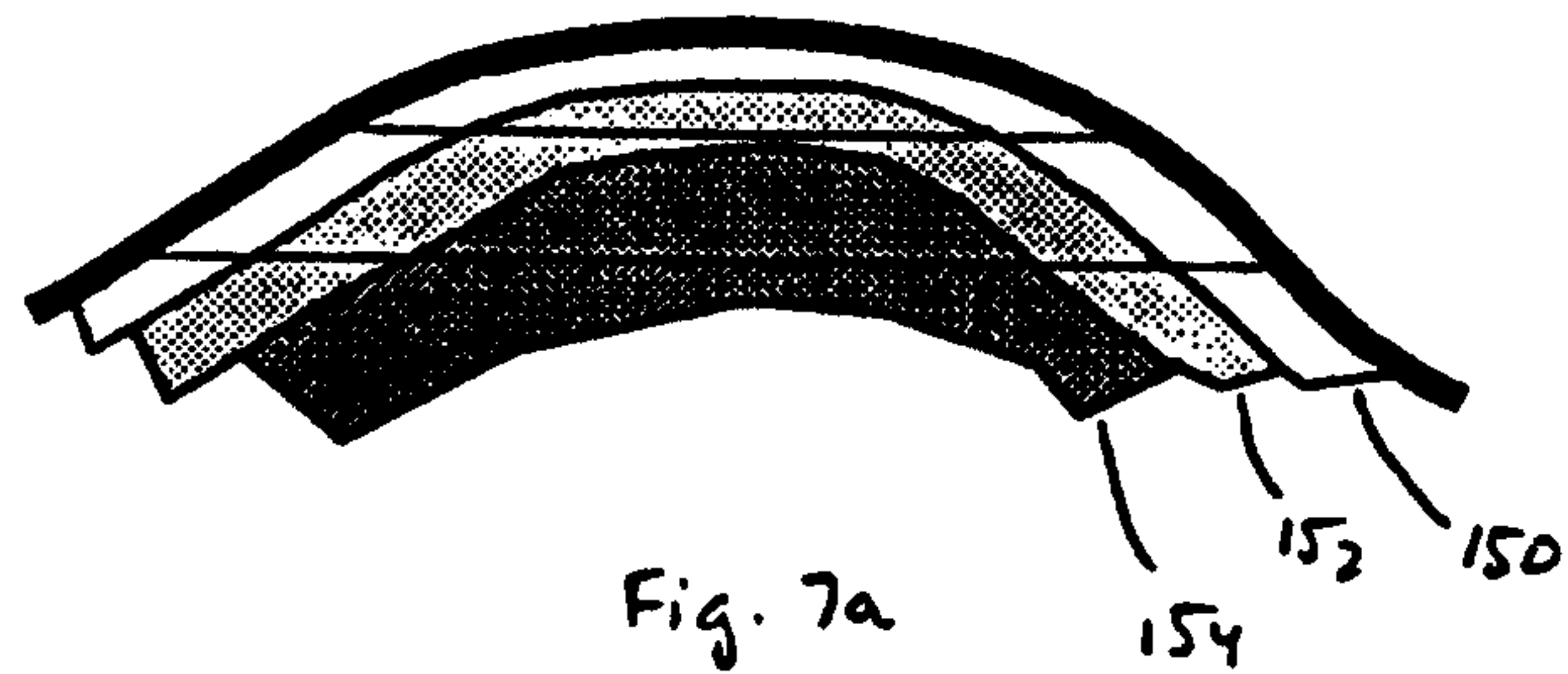


Fig. 5c





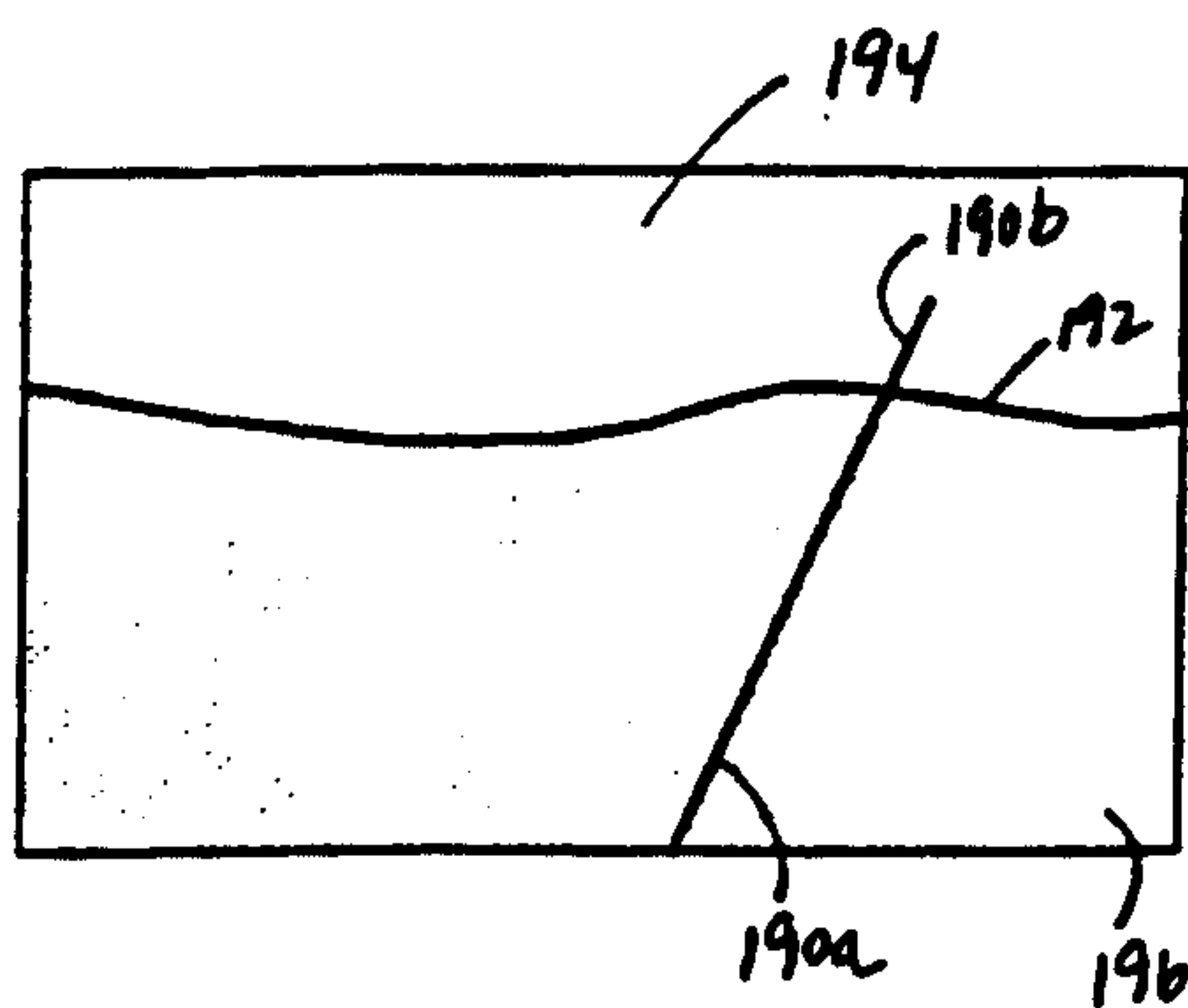


Fig. 8a

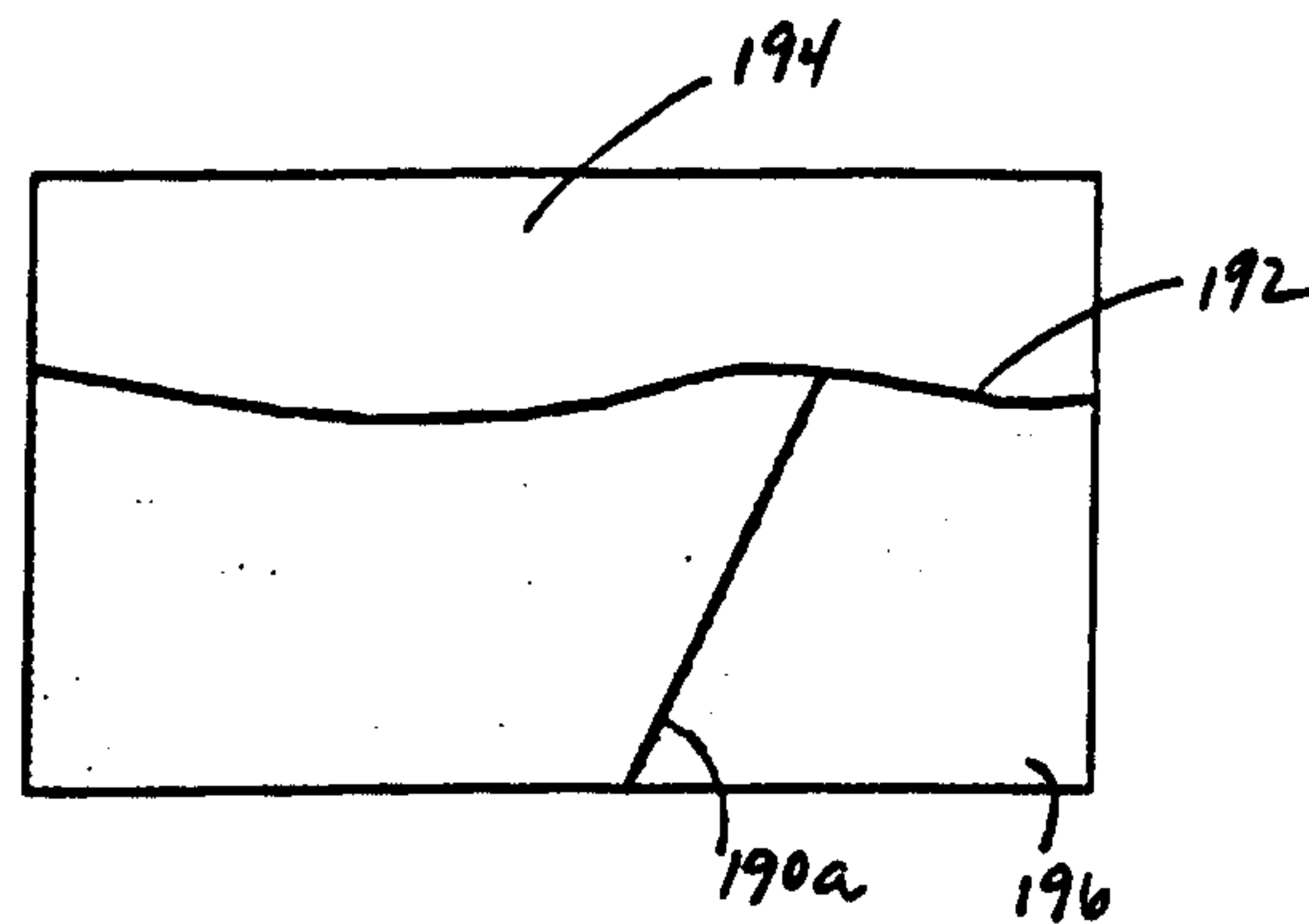


Fig. 8b

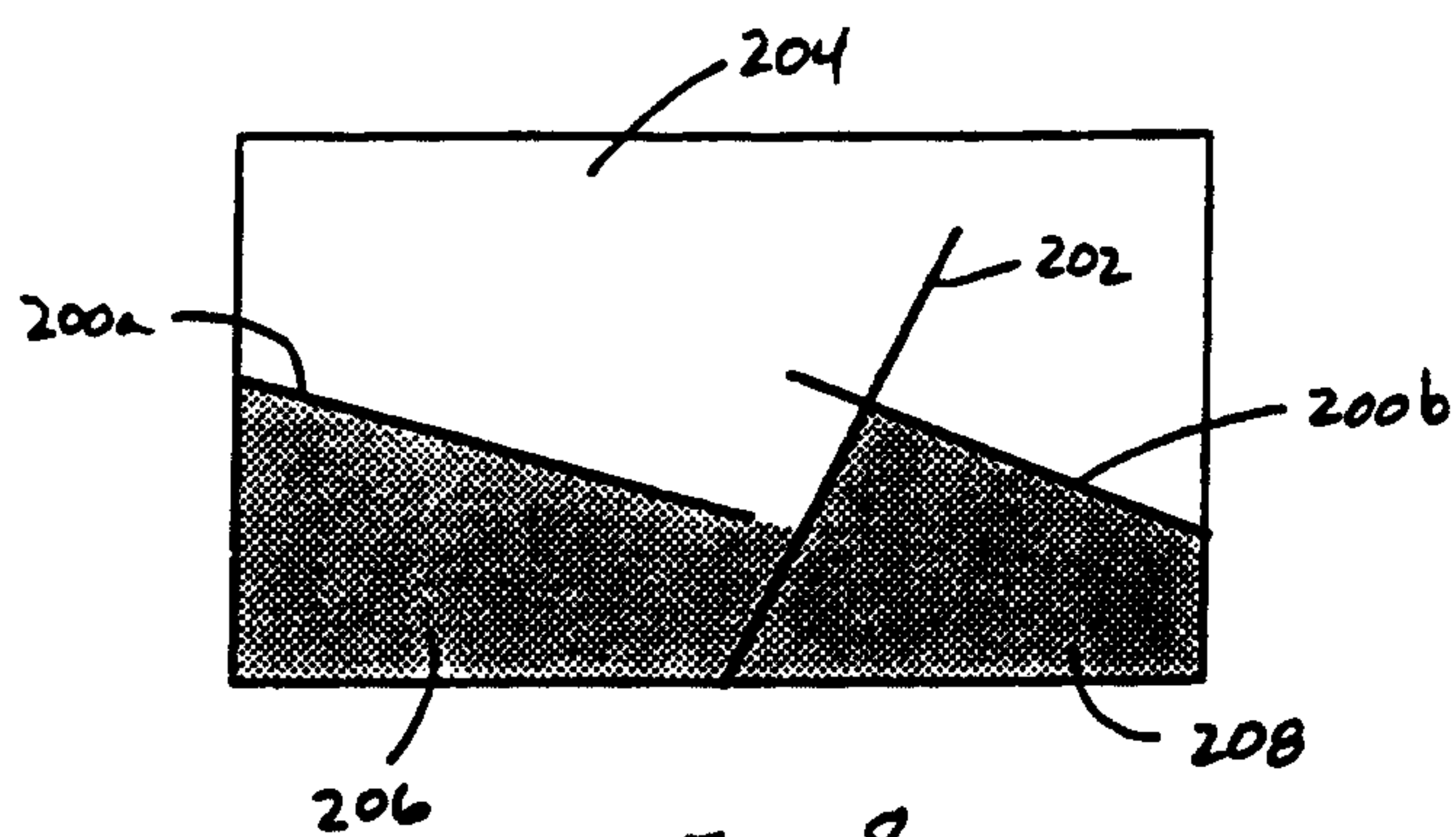


Fig. 8c

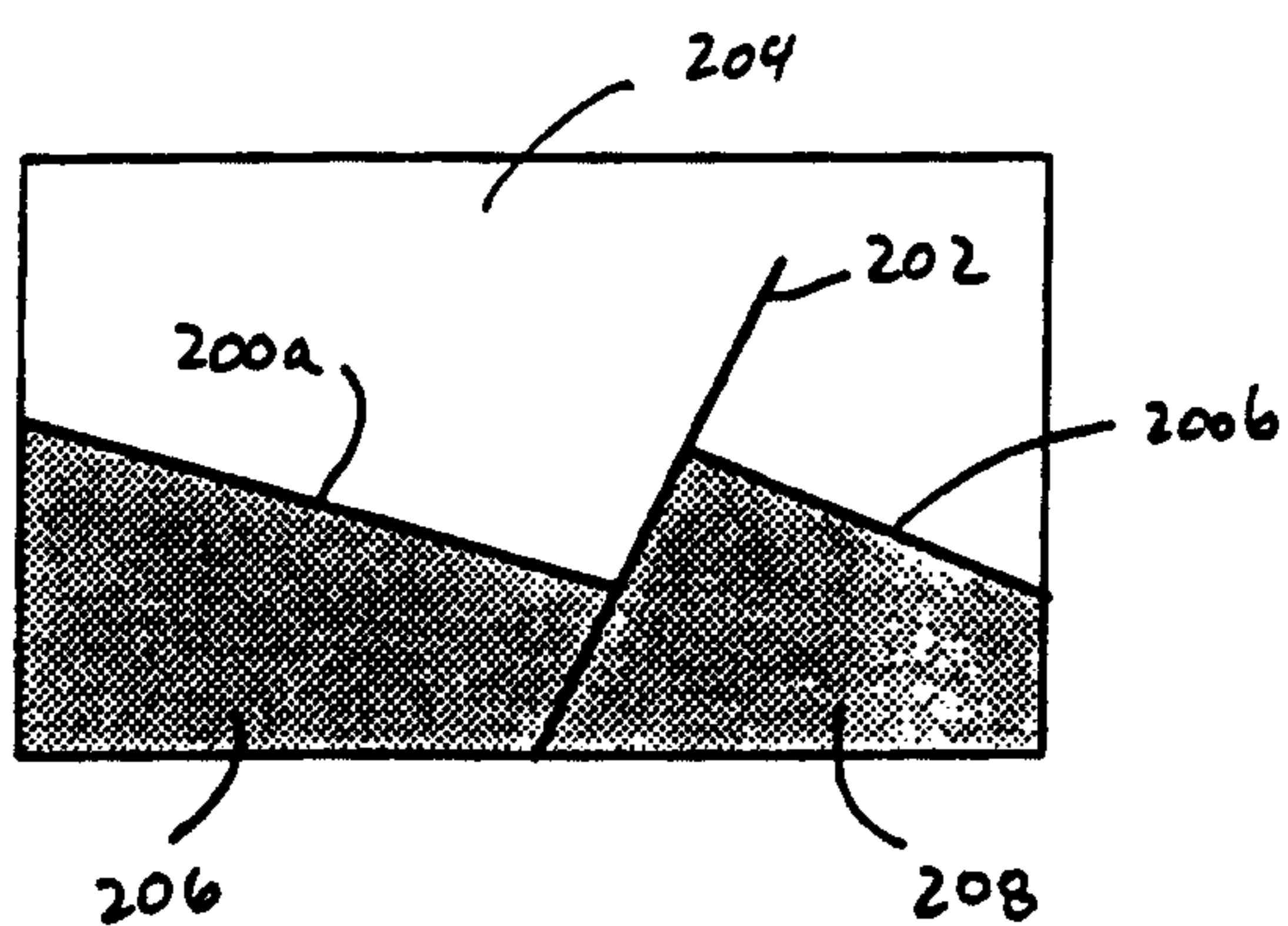


Fig. 8d

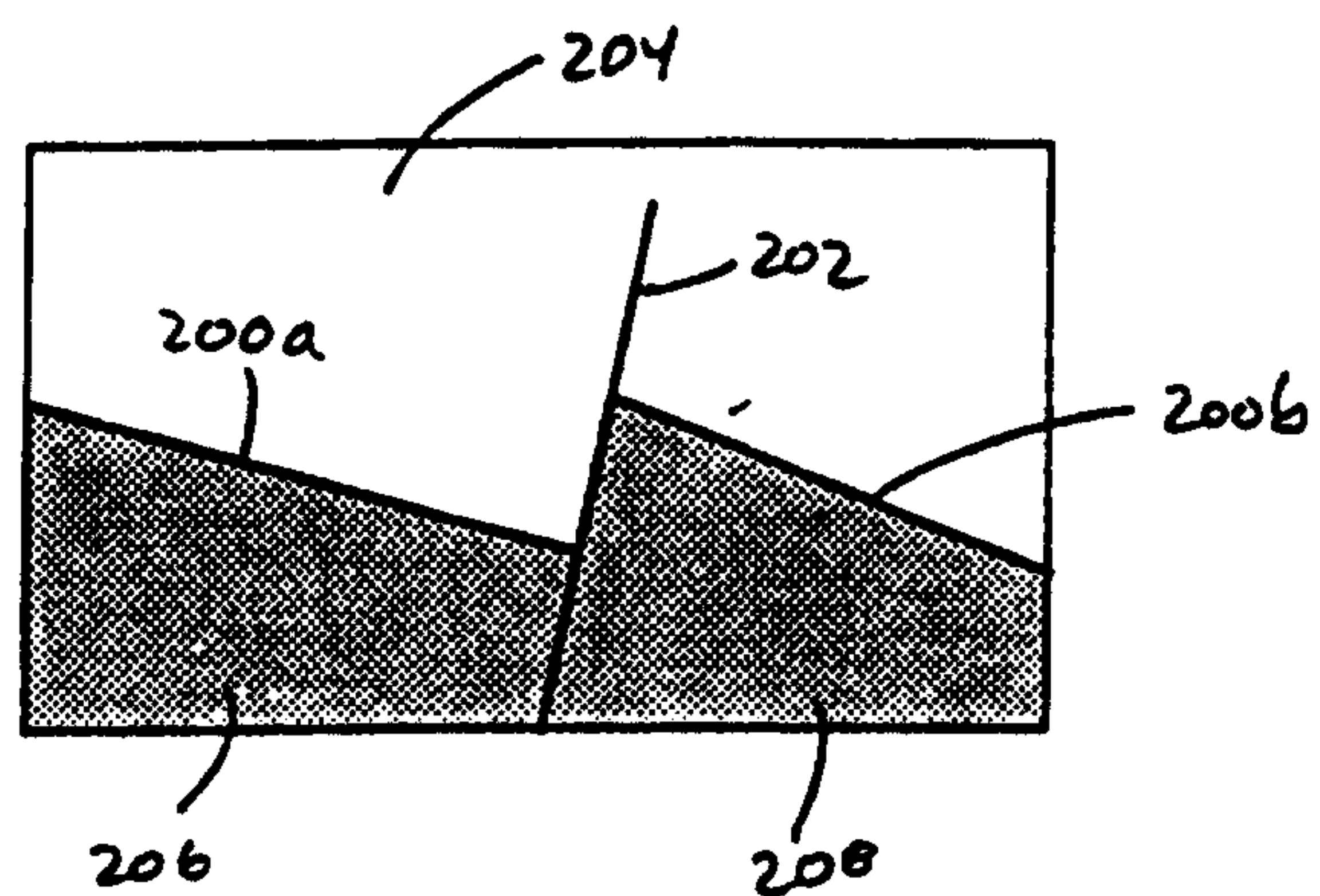


Fig. 8e

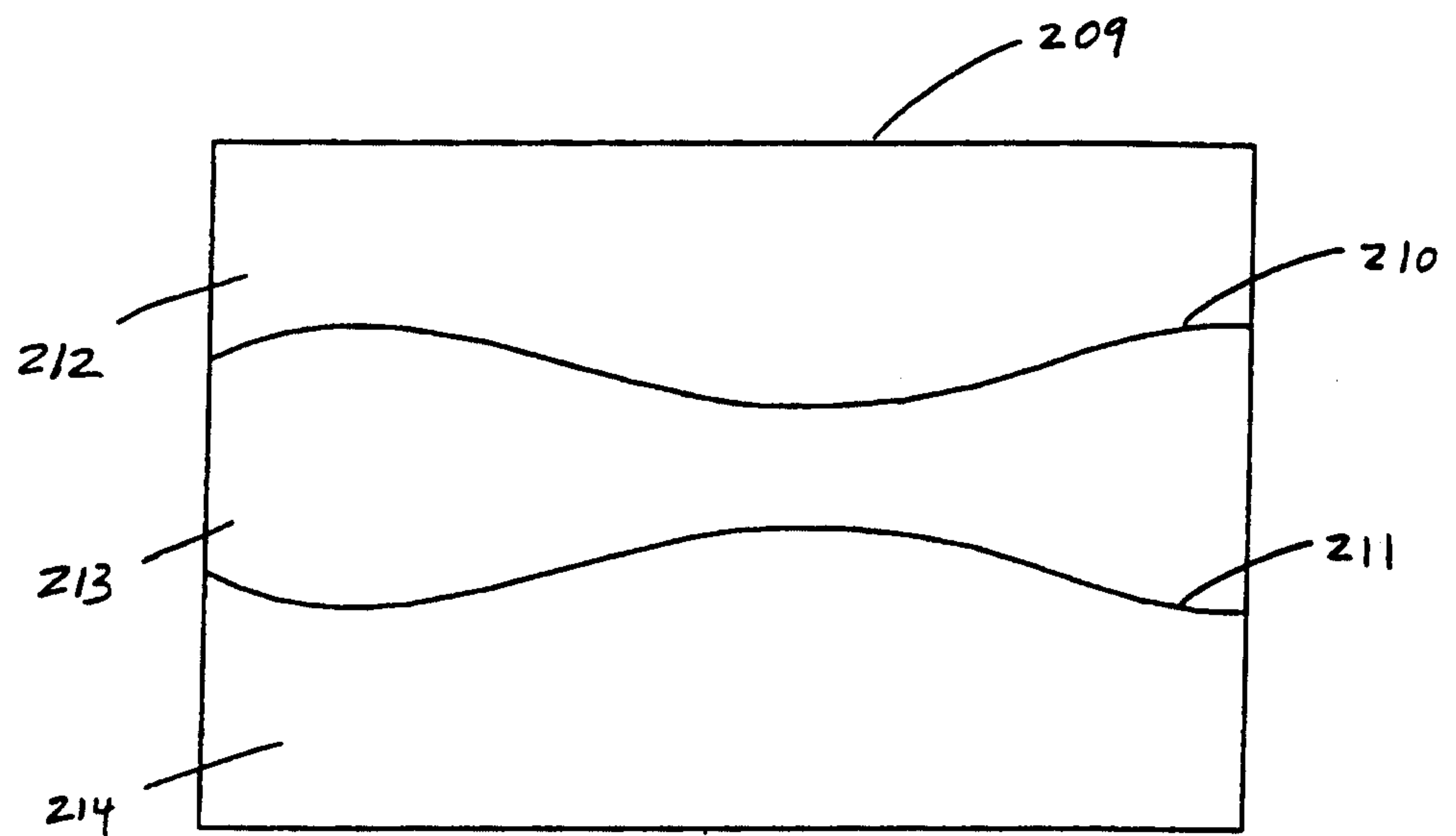


Fig. 9a

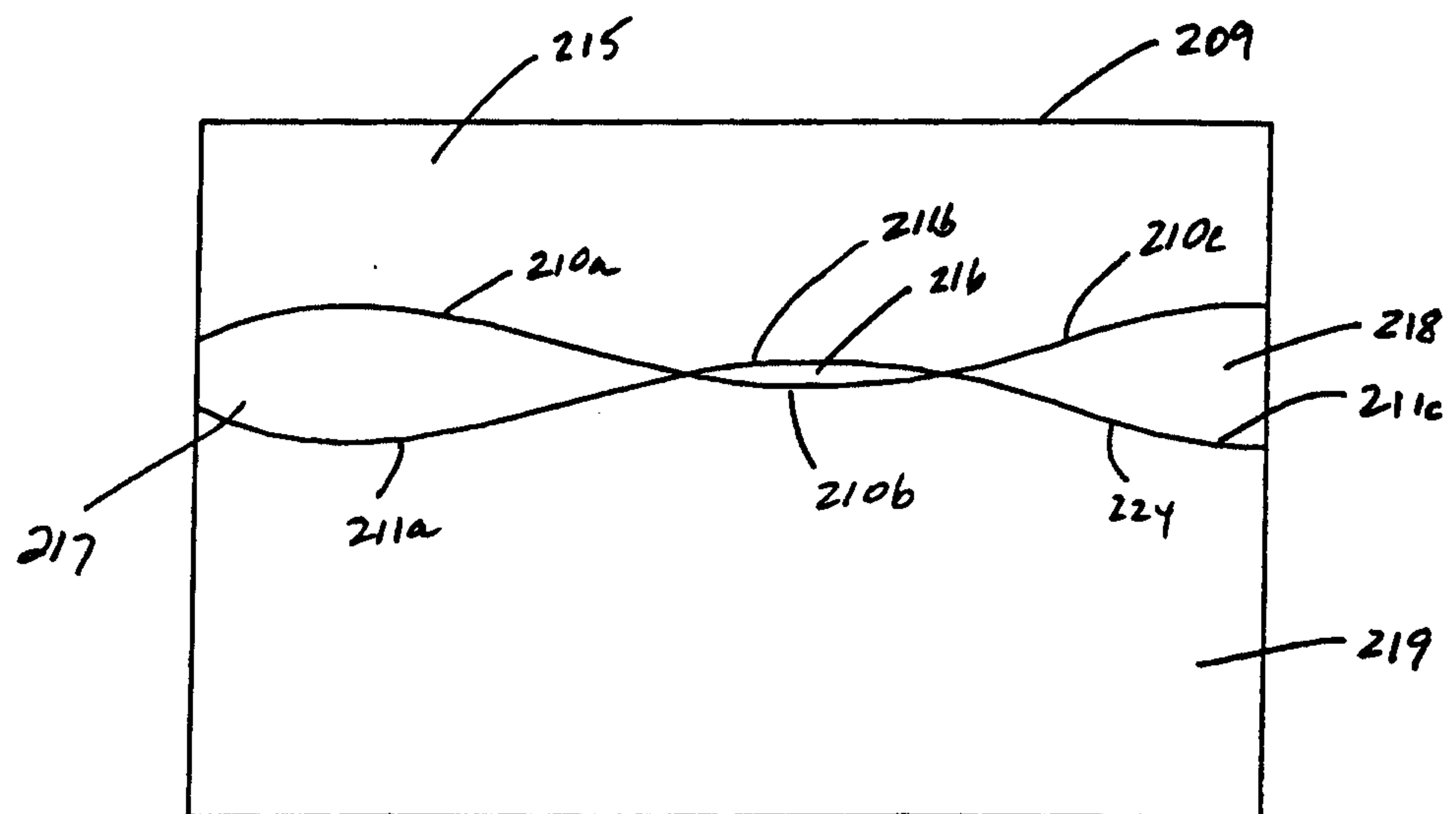


Fig. 9b

10 / 18

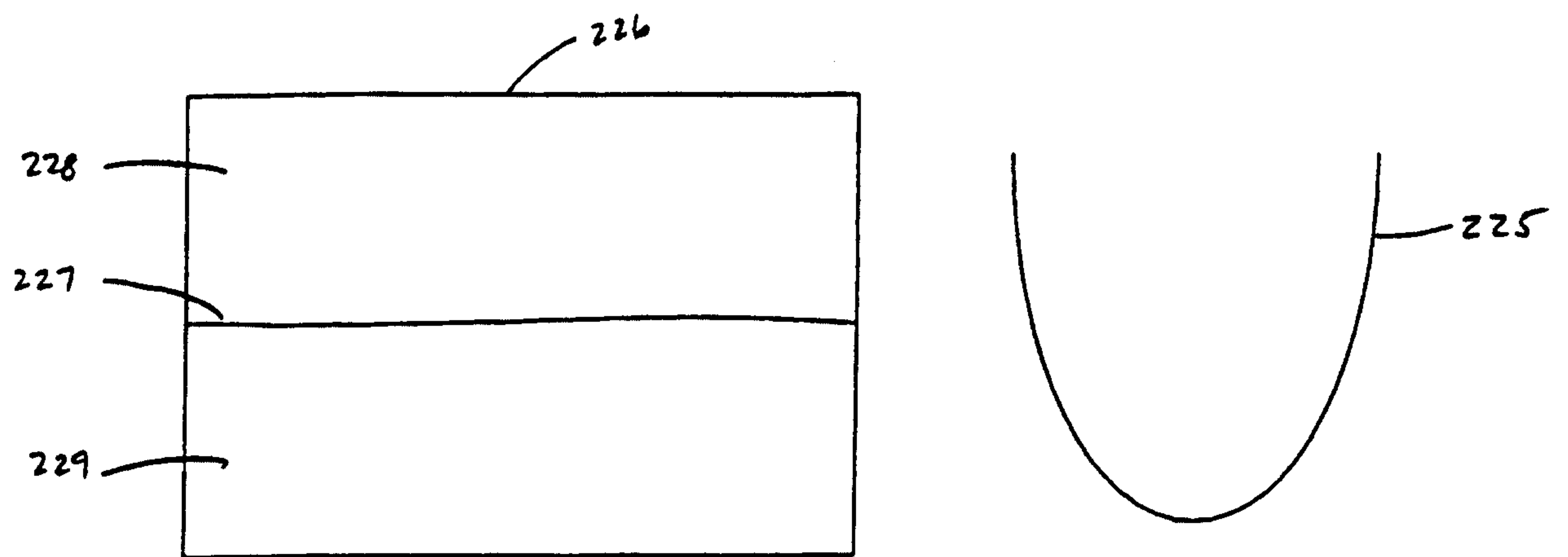


Fig. 10a

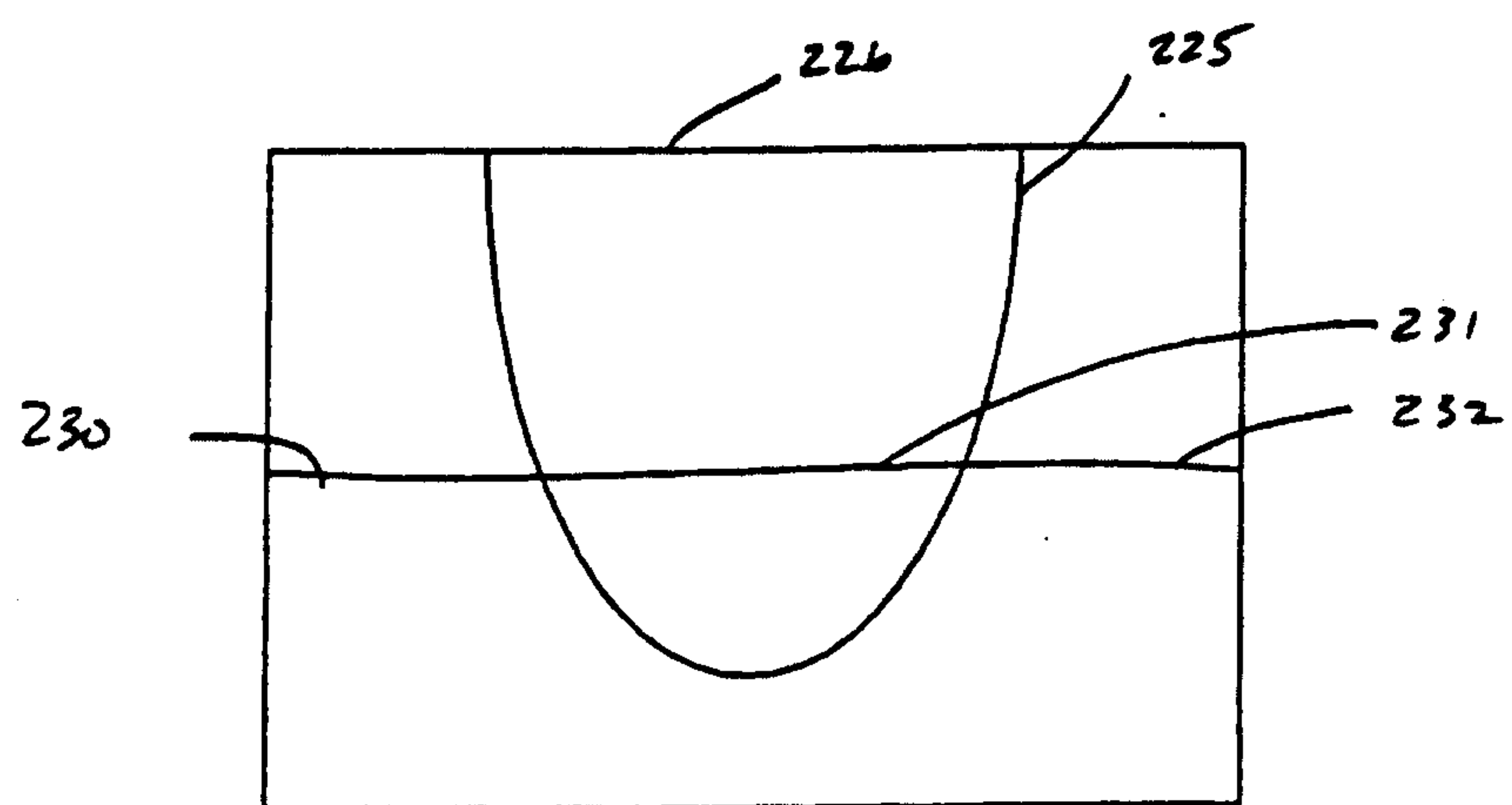


Fig. 10b

11 / 18

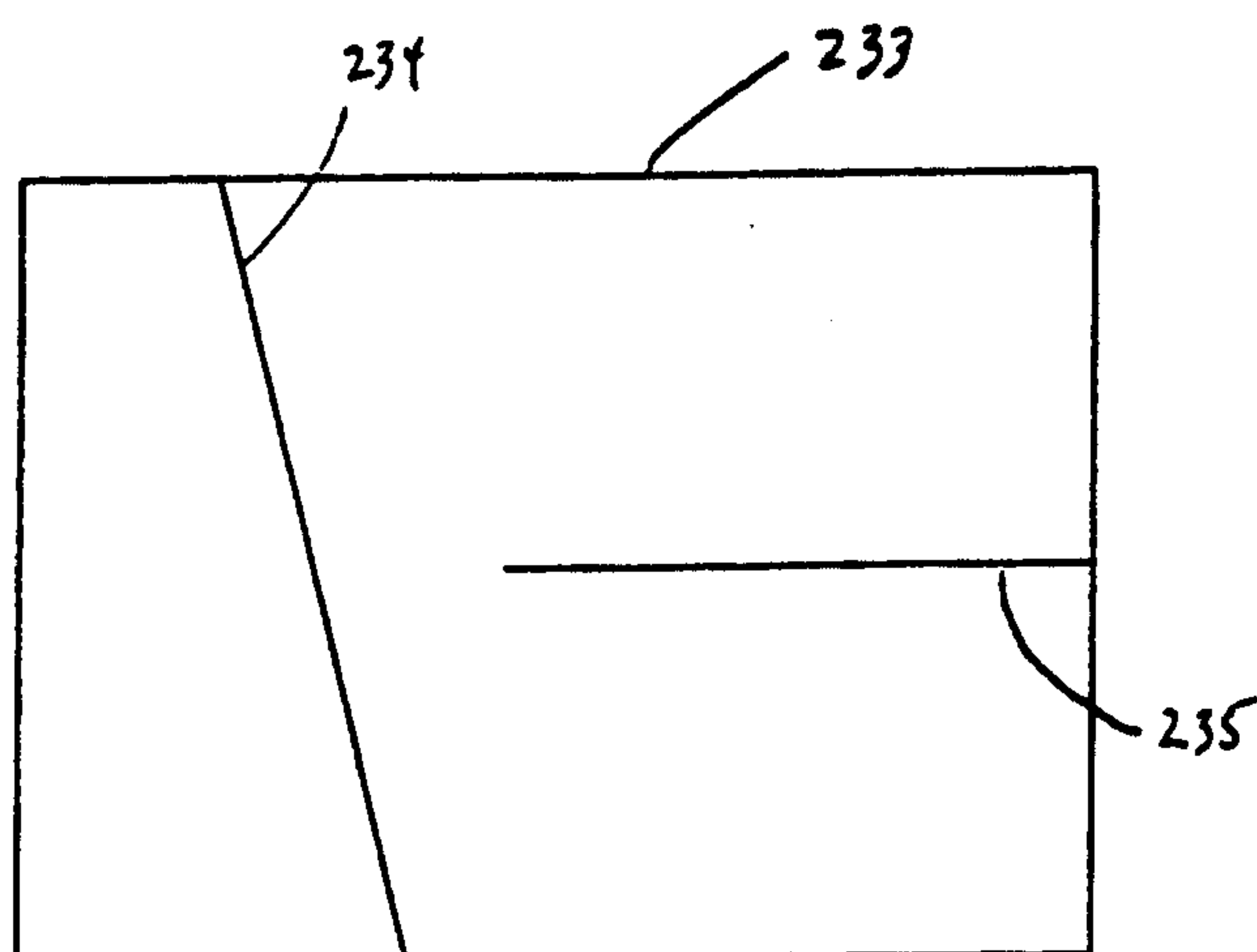


Fig. 11a

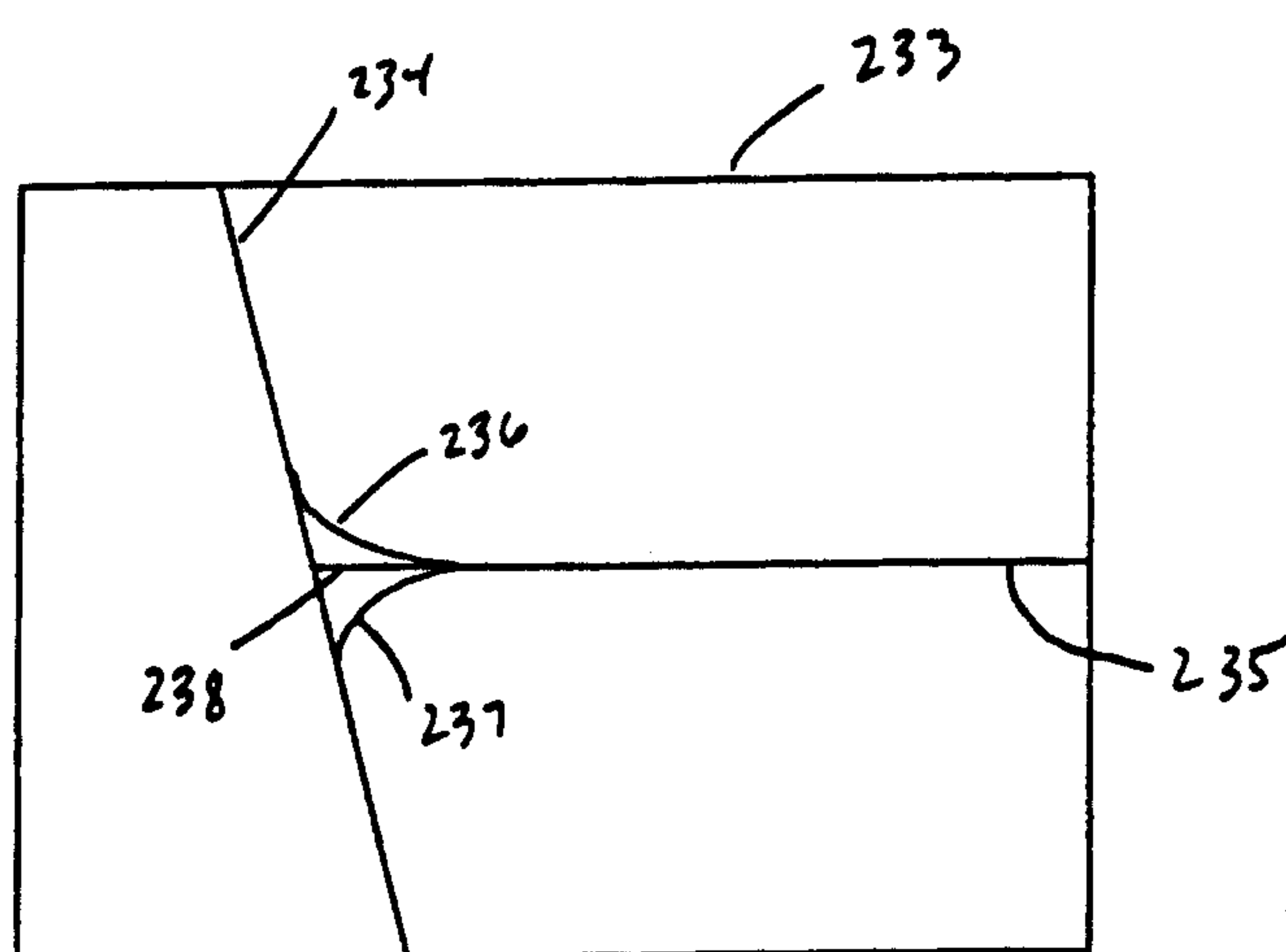


Fig. 11b

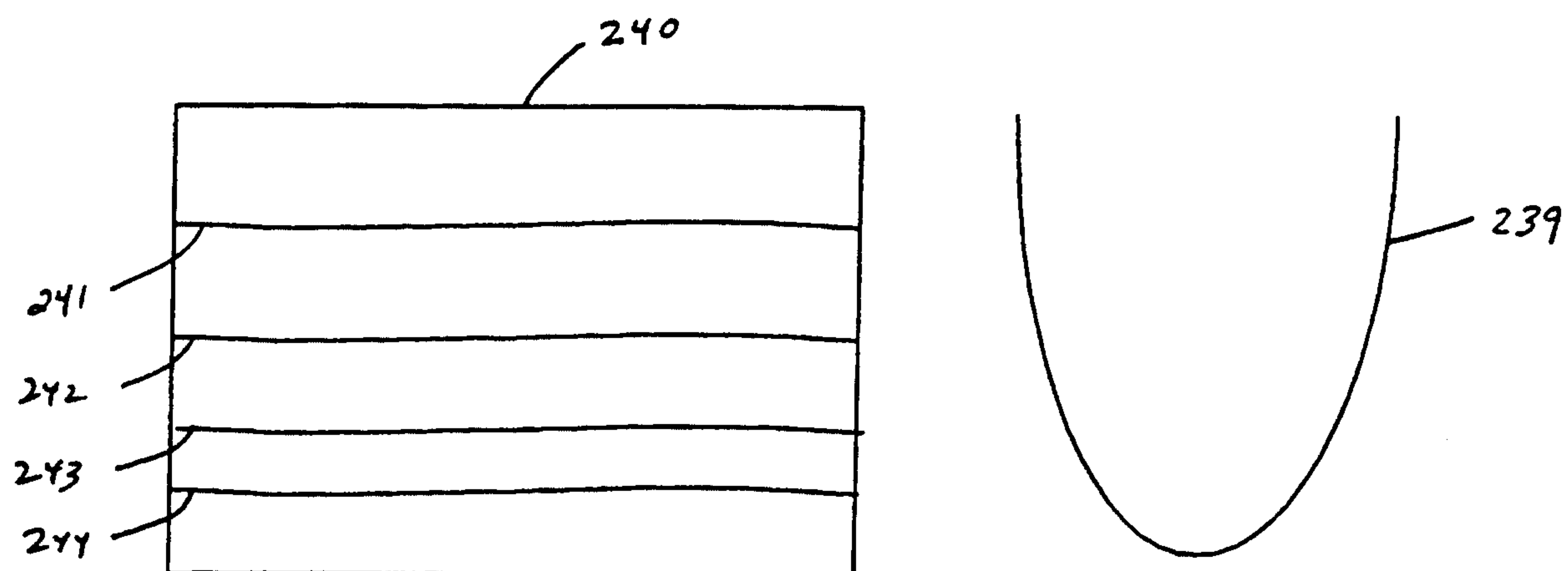


Fig. 12a

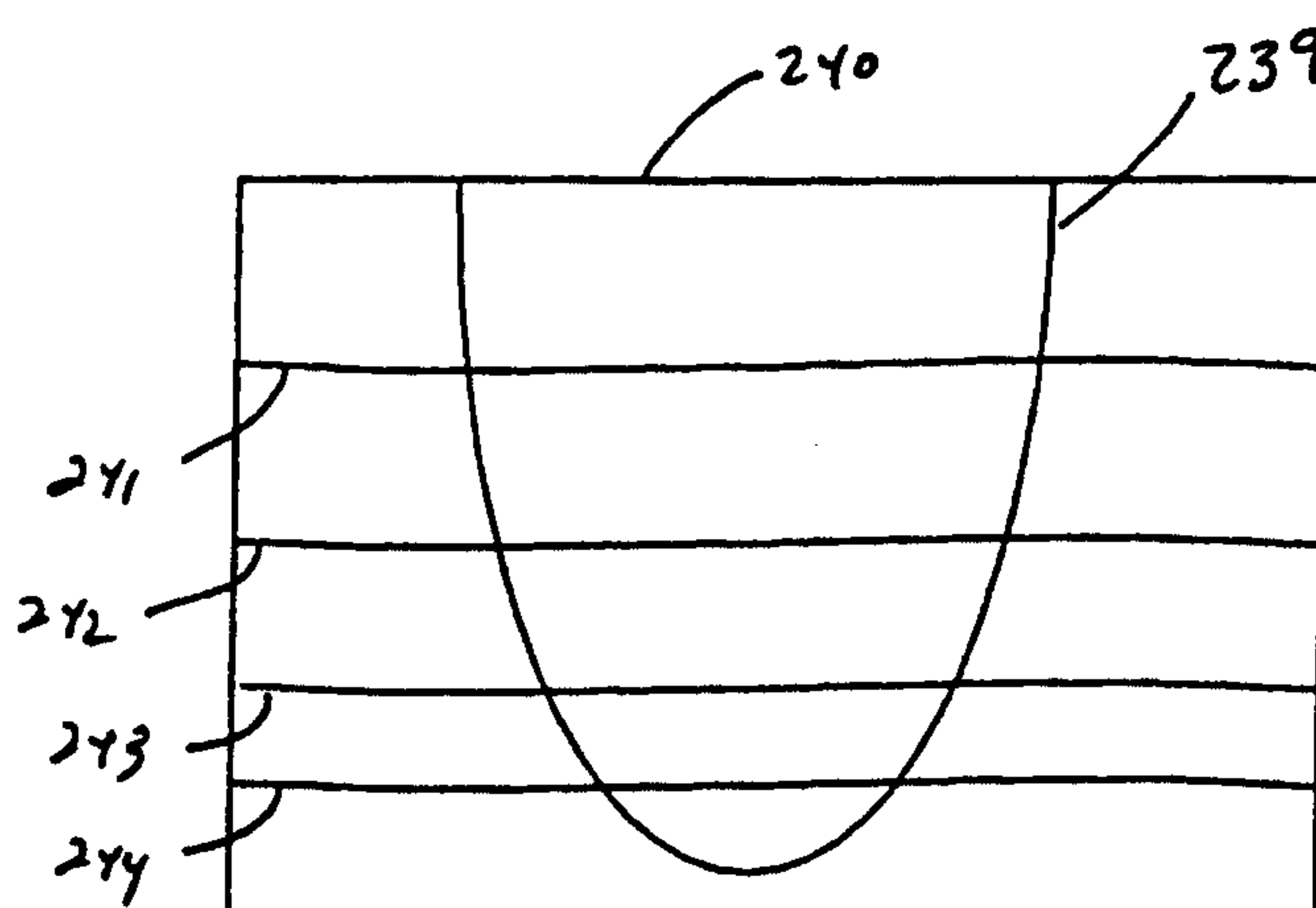


Fig. 12b

13 / 18

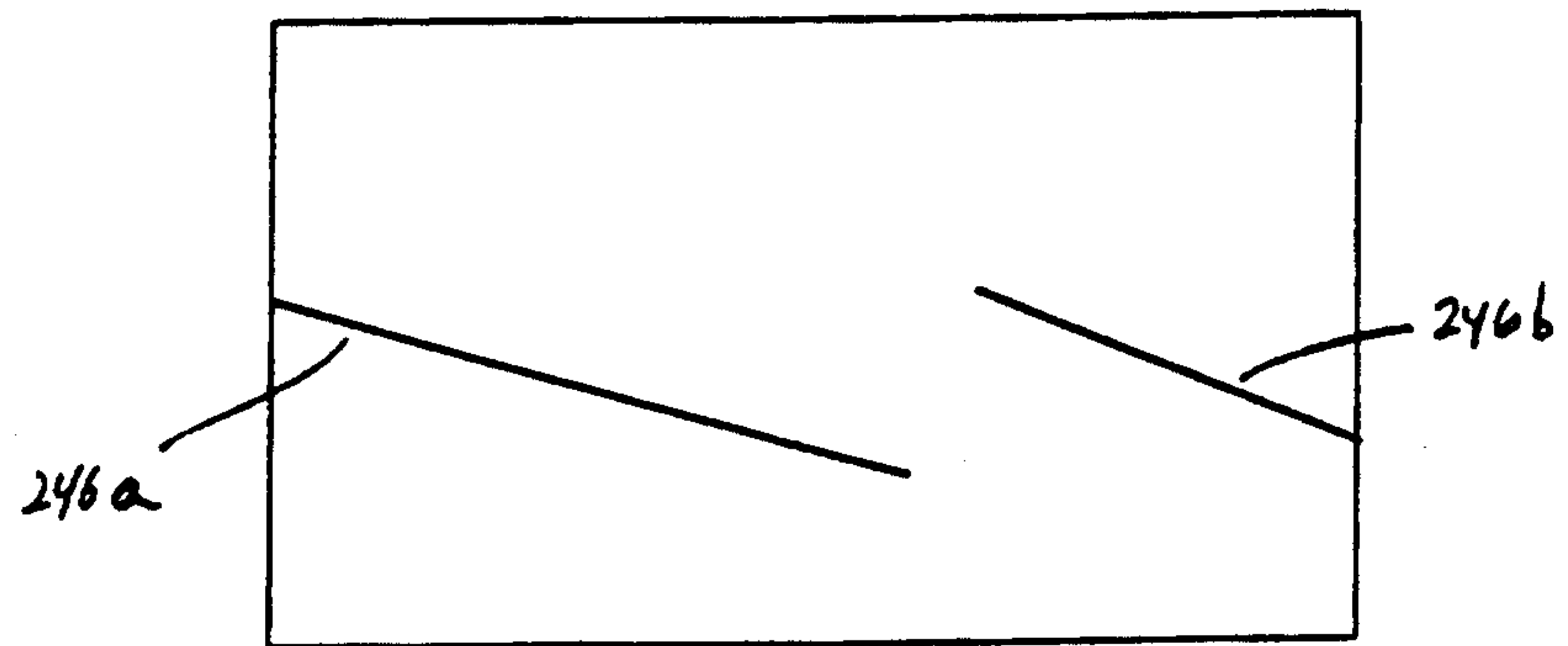


Fig. 13a

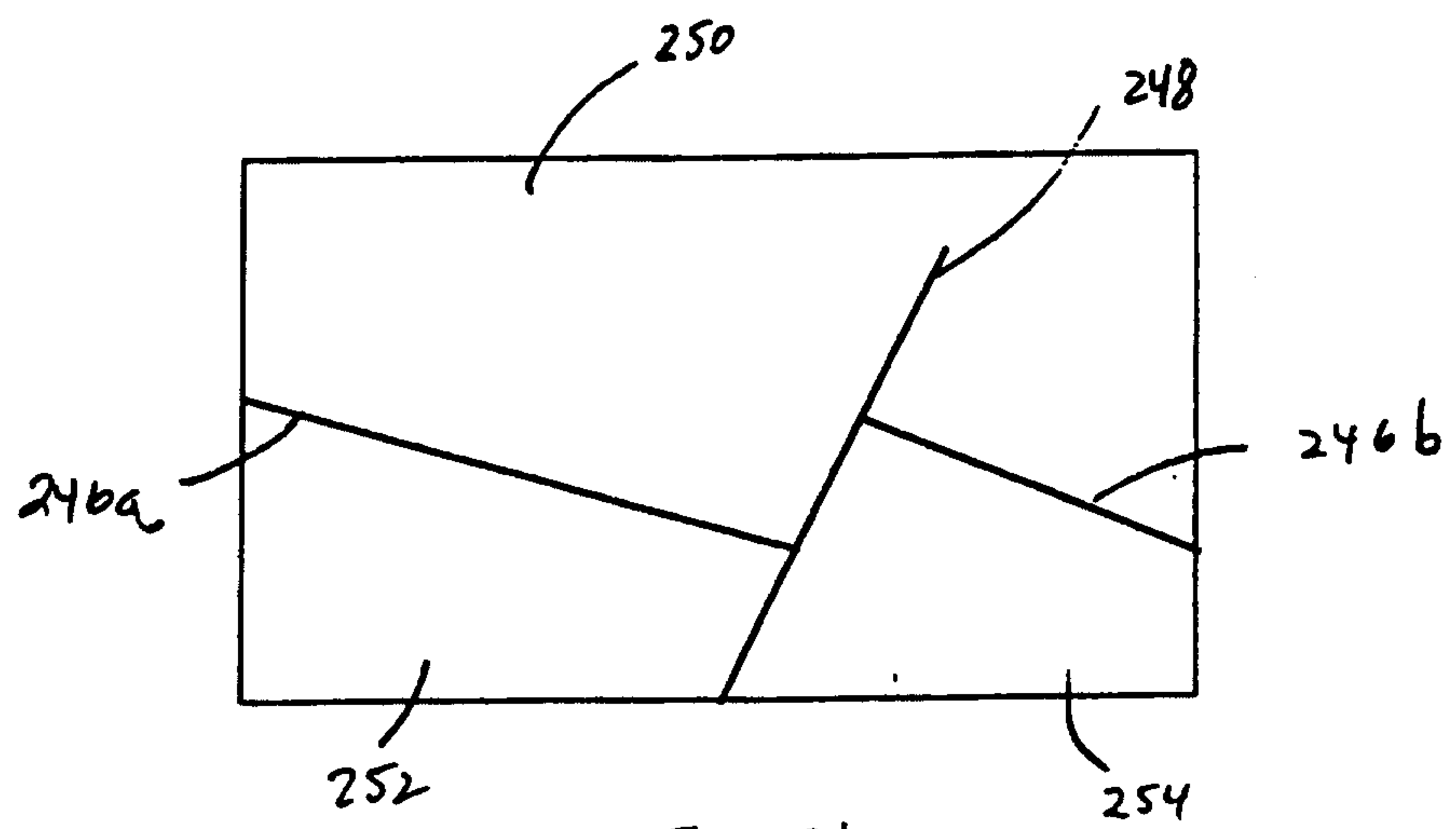


Fig. 13b

14 / 18

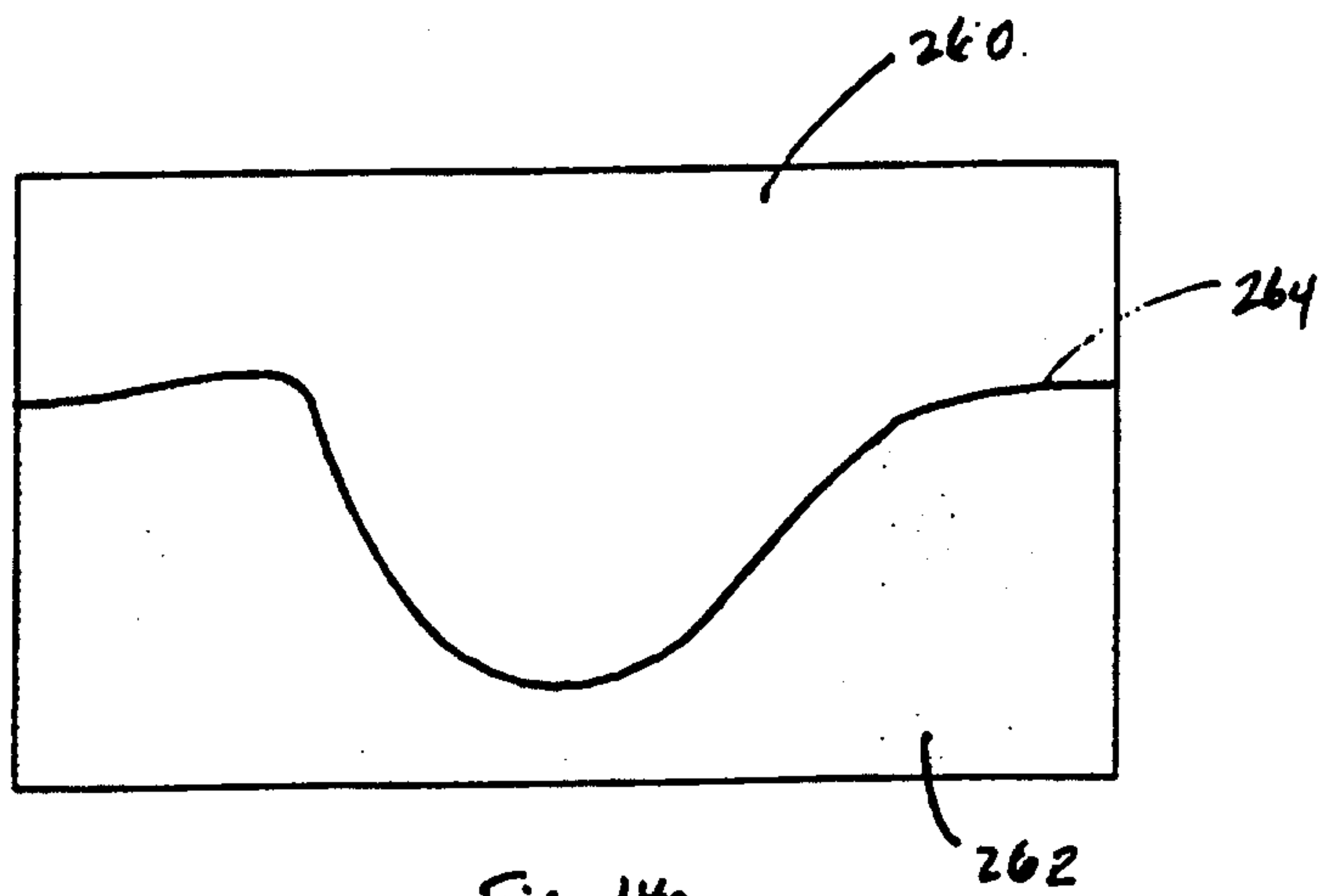


Fig. 14a

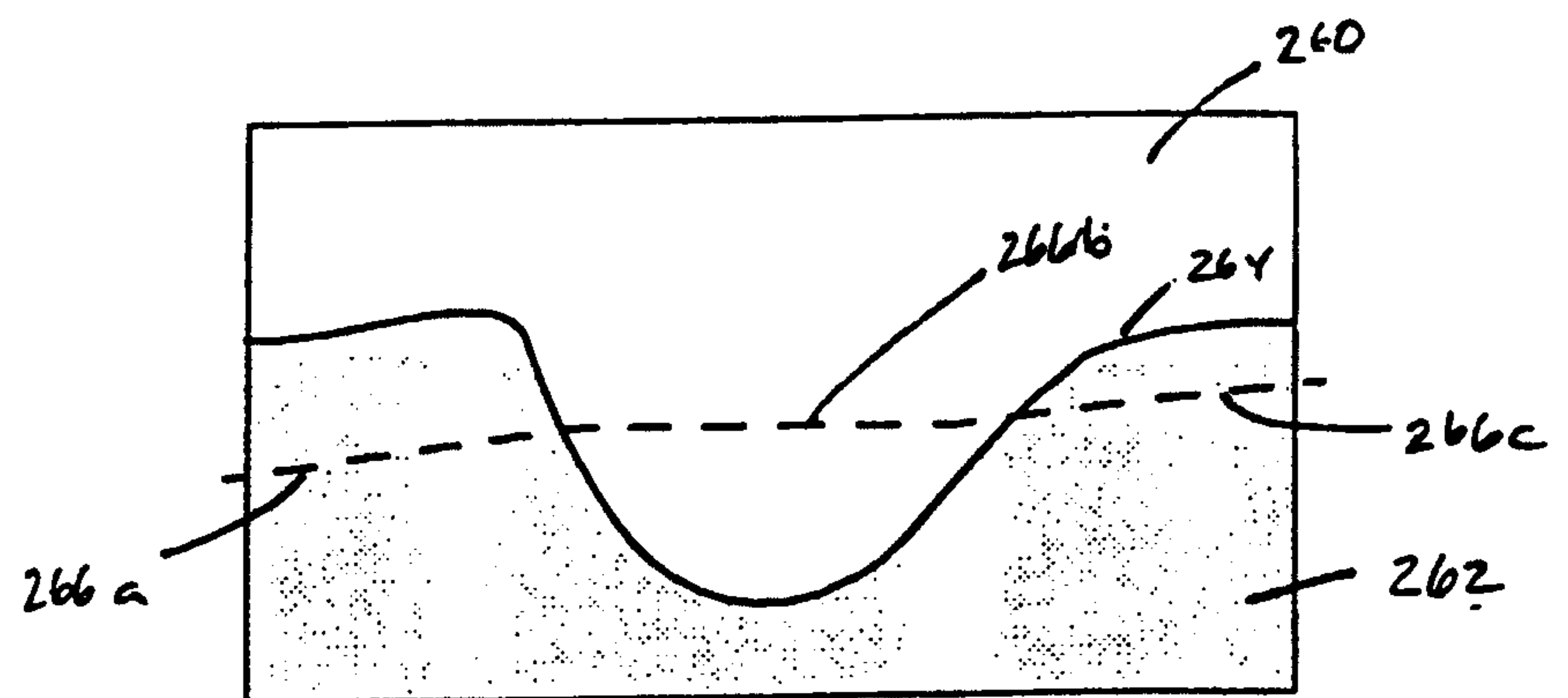


Fig. 14b

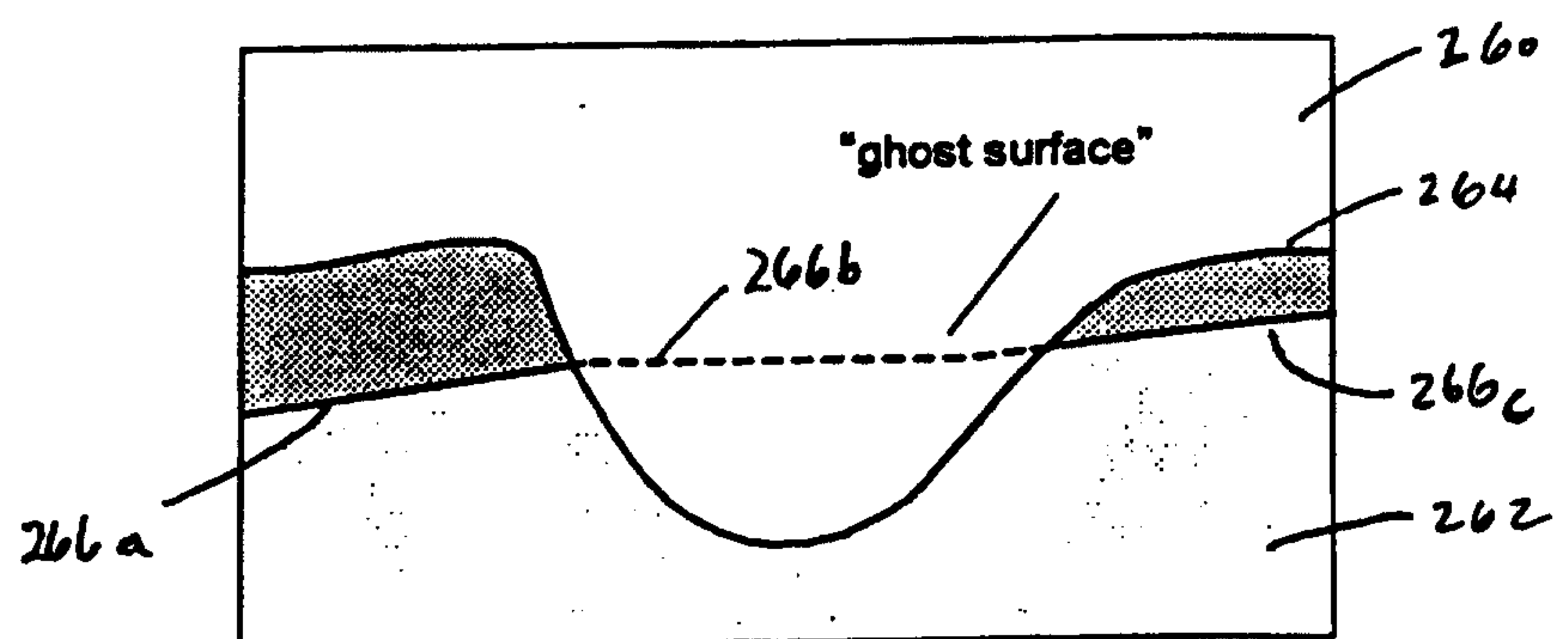


Fig. 14c

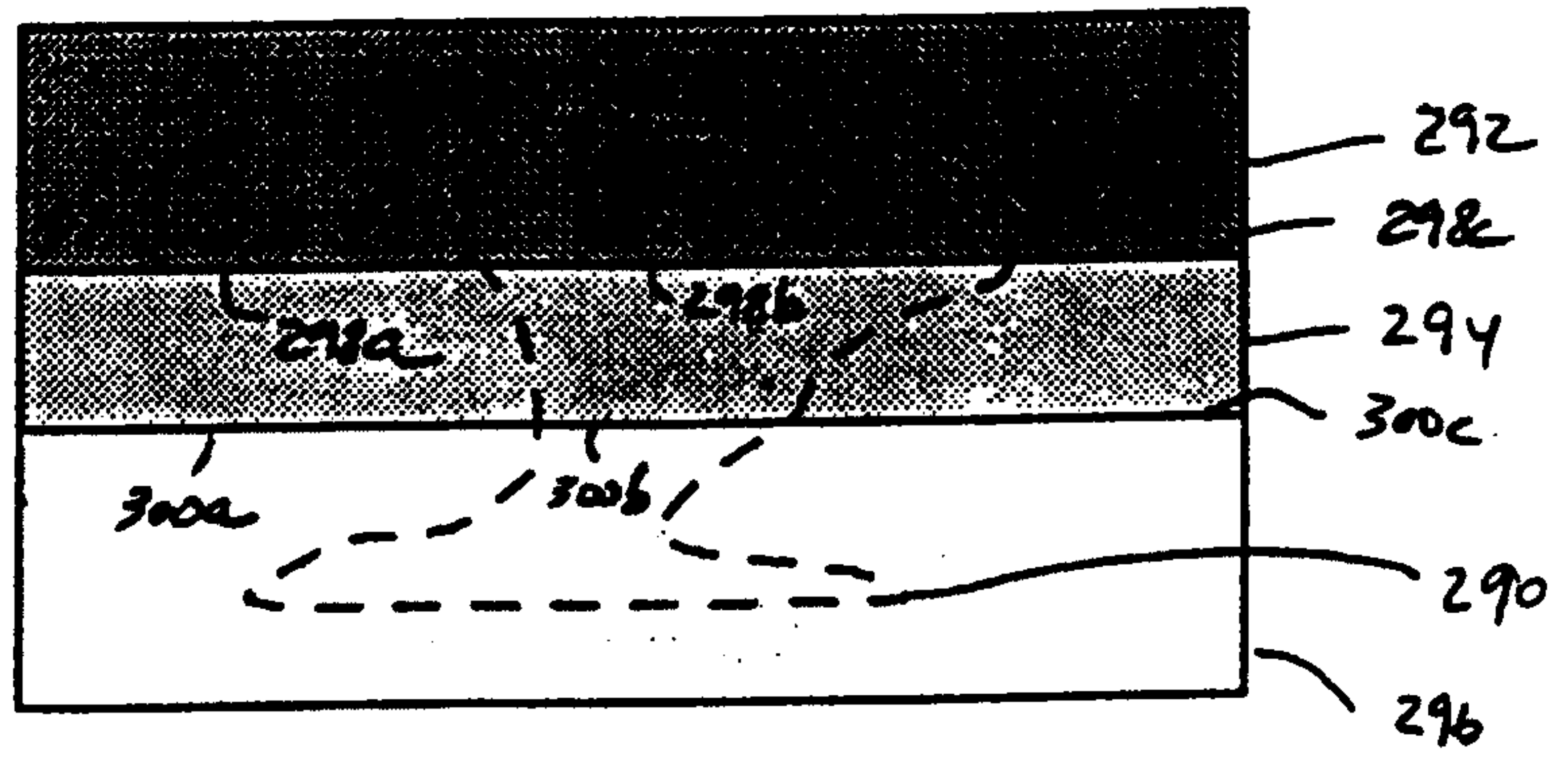


Fig 16a

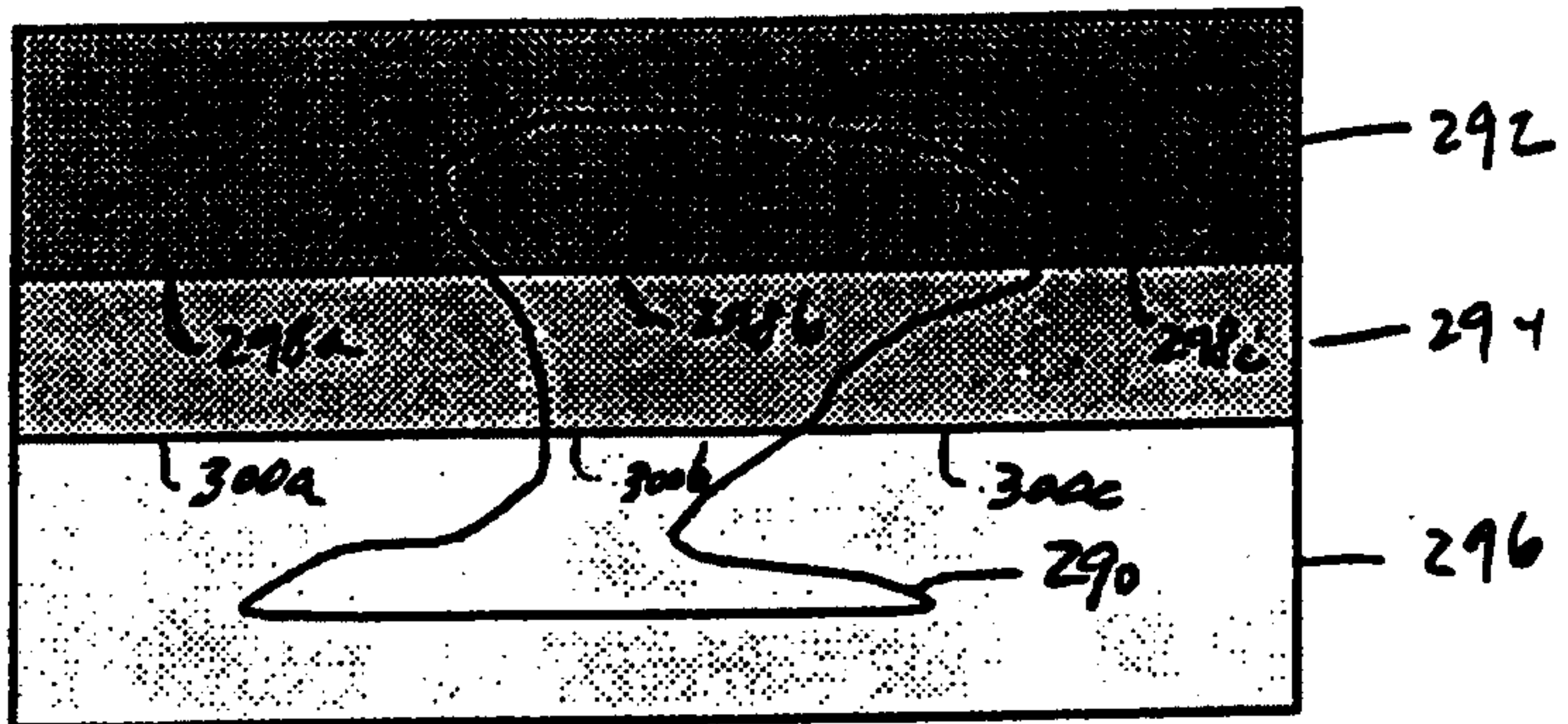


Fig 16b

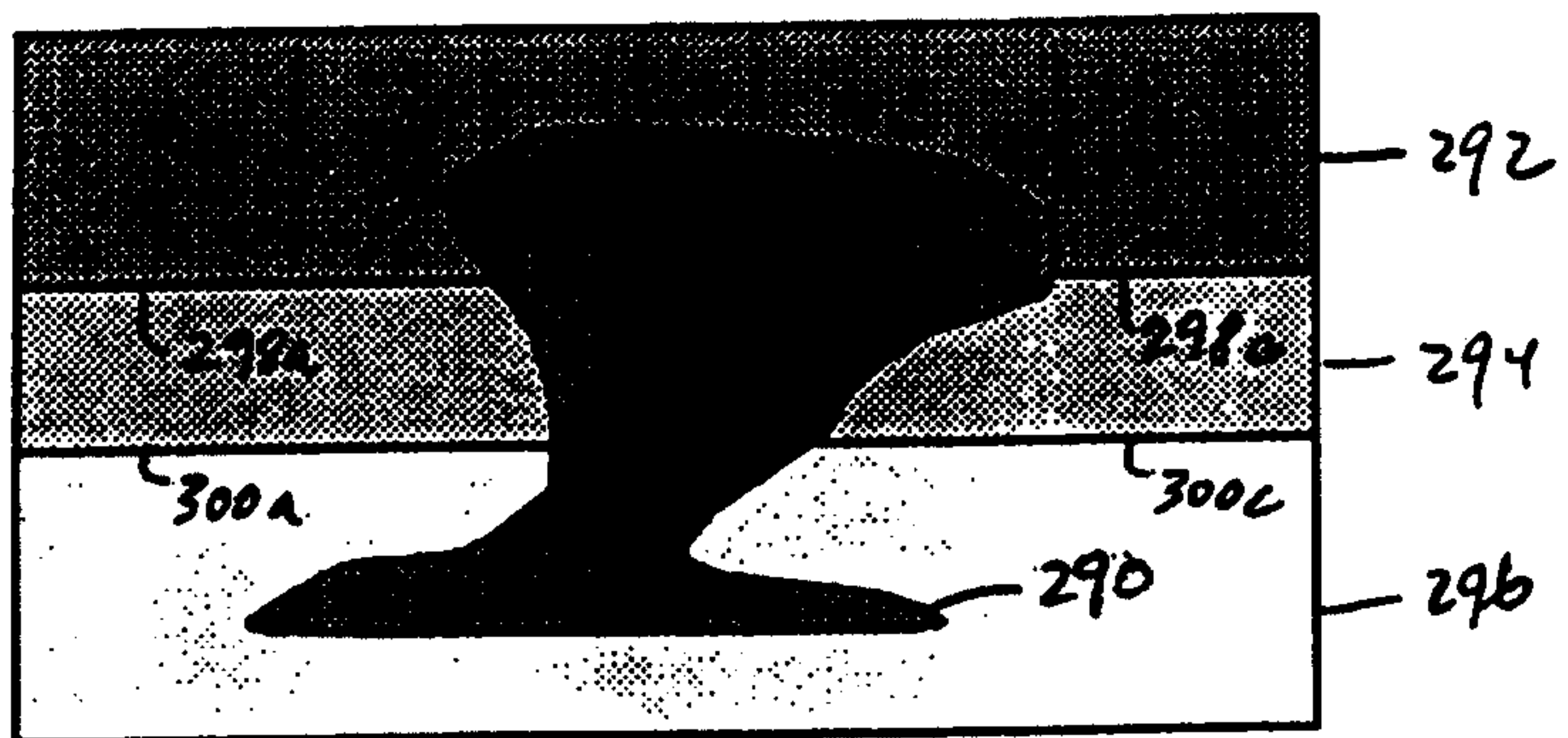
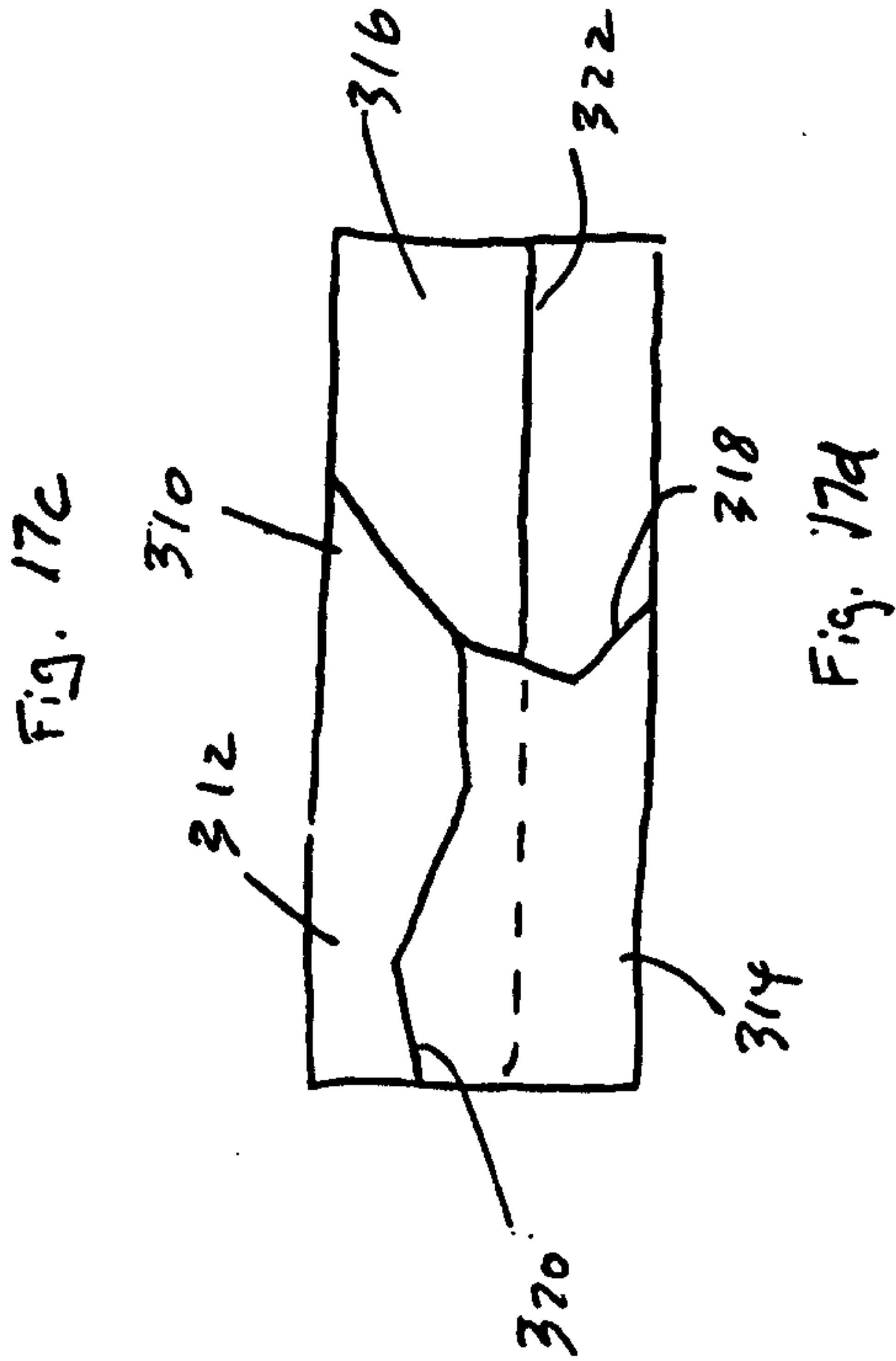
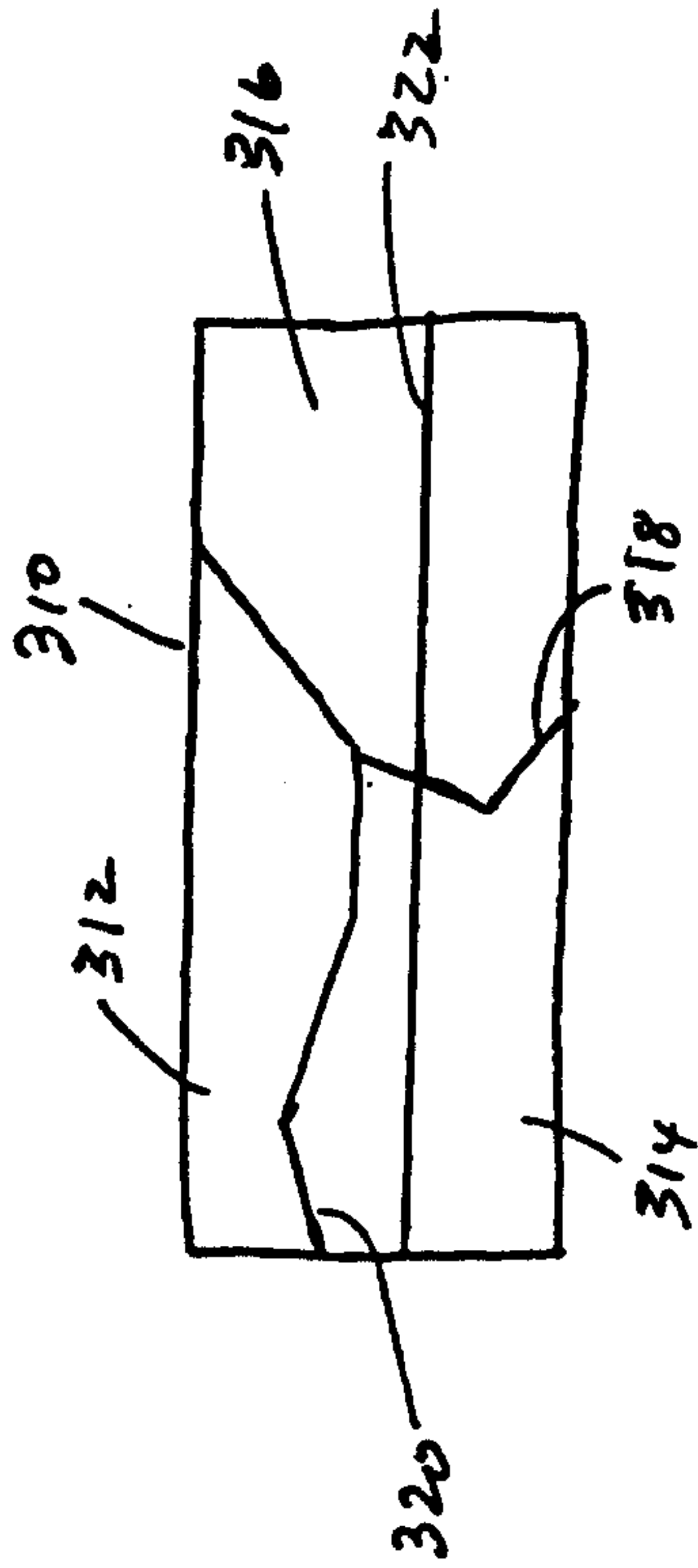
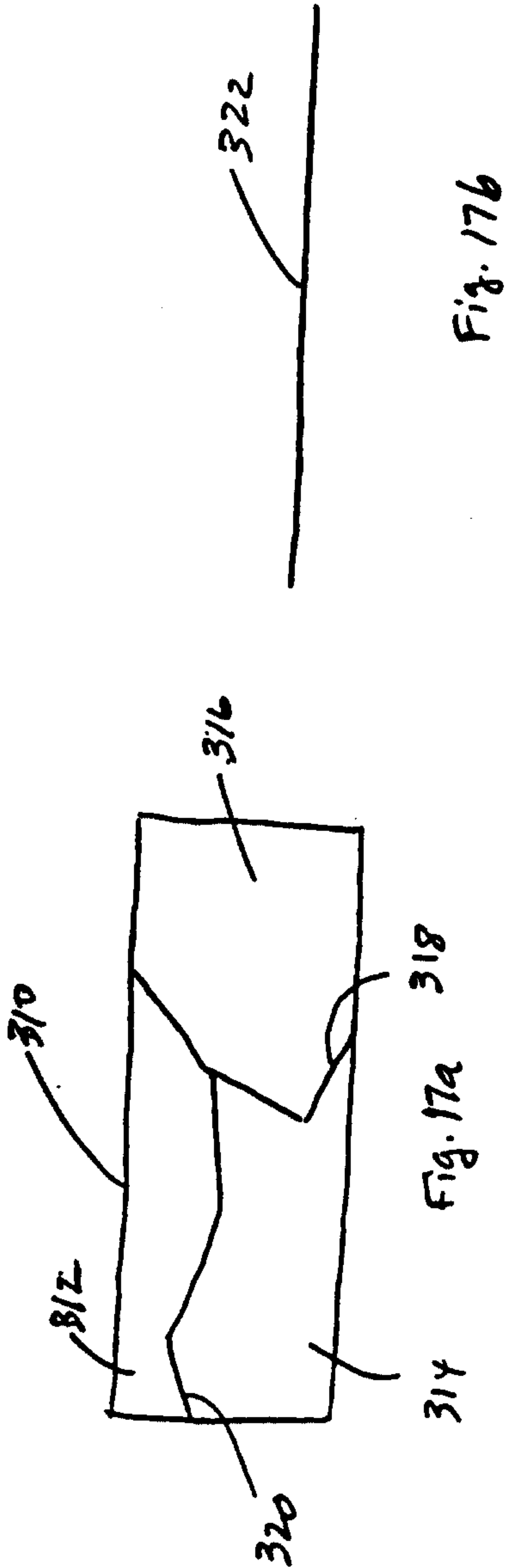


Fig 16c



17 / 18

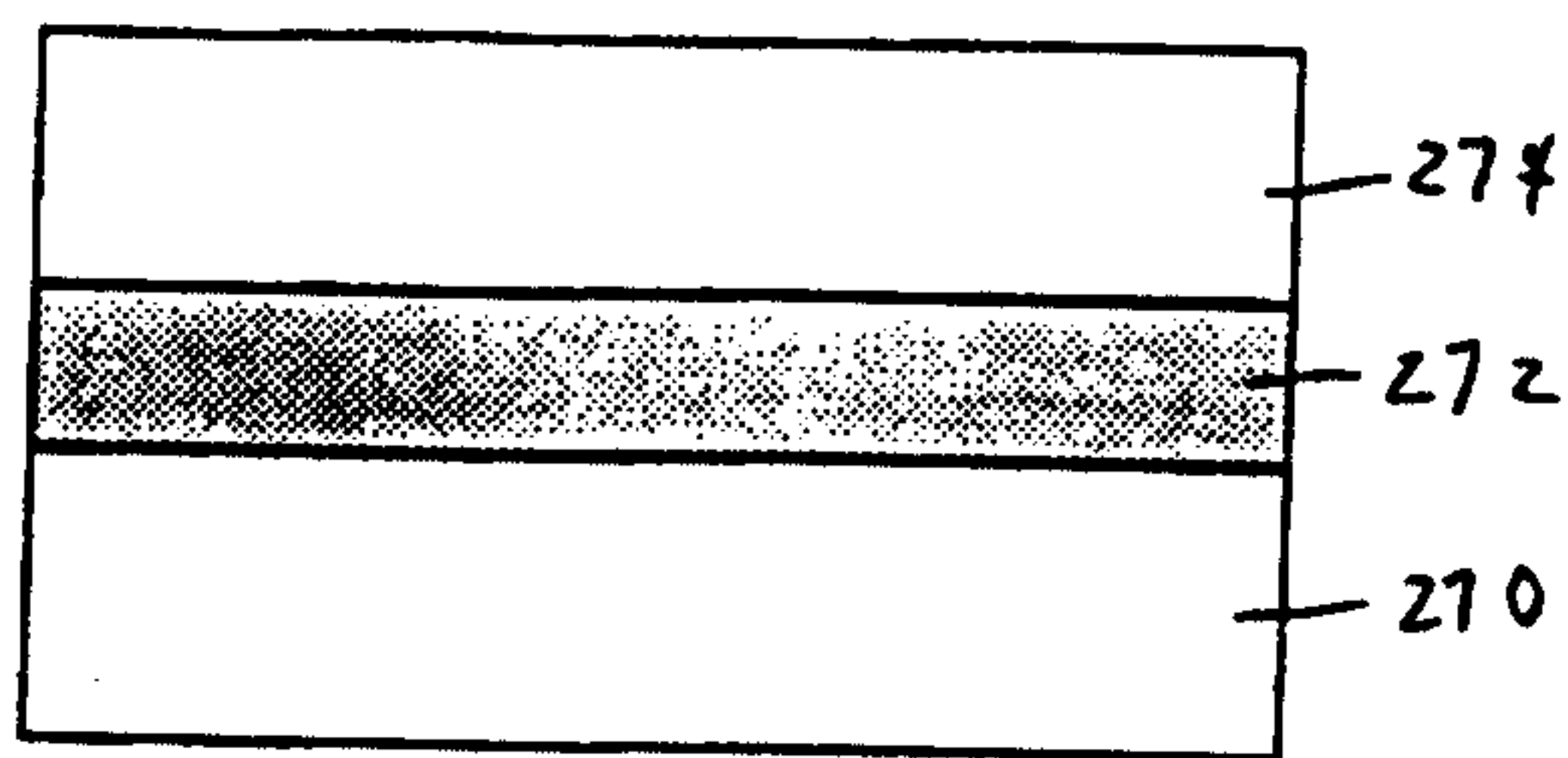


Fig. 15a

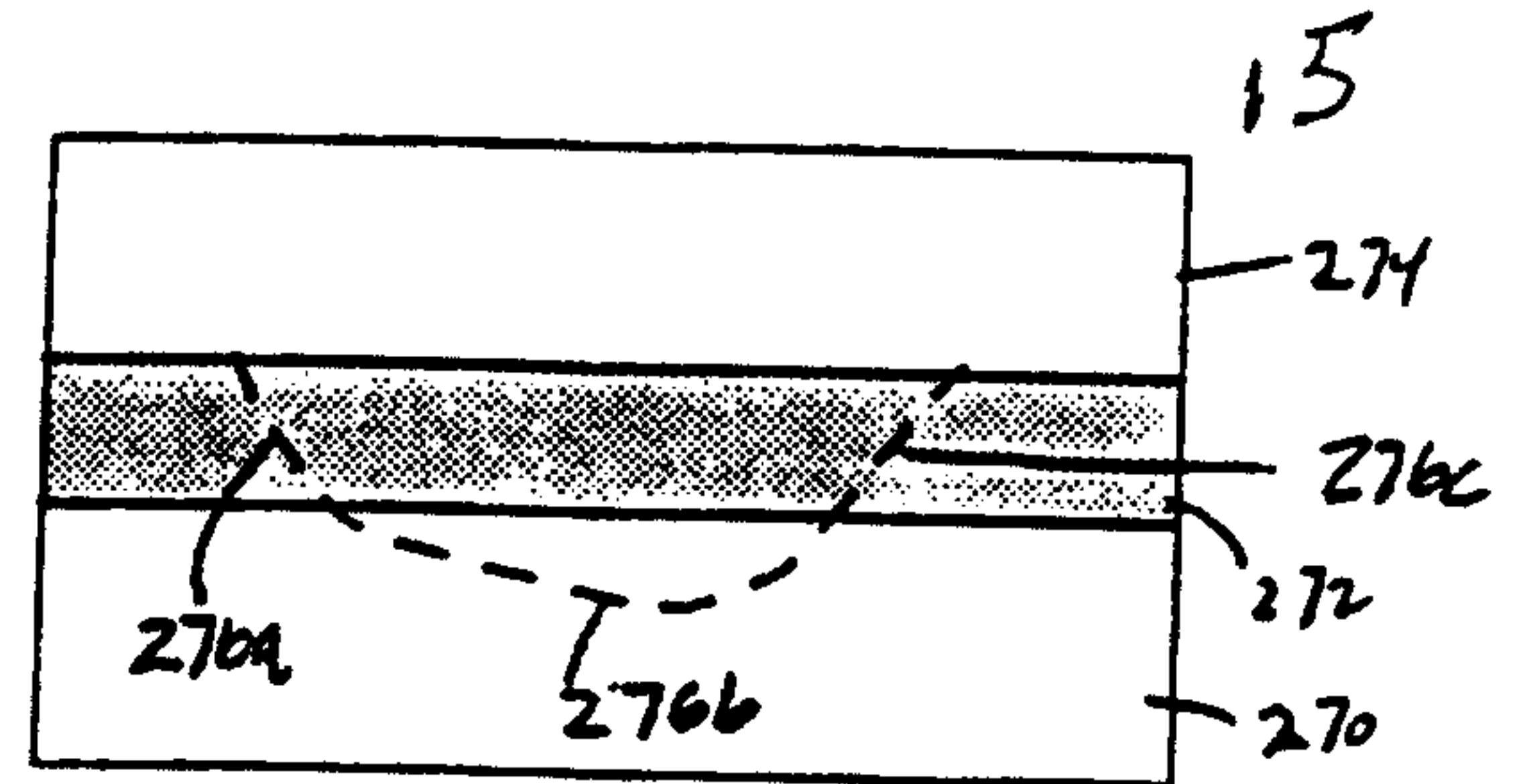


Fig. 15b

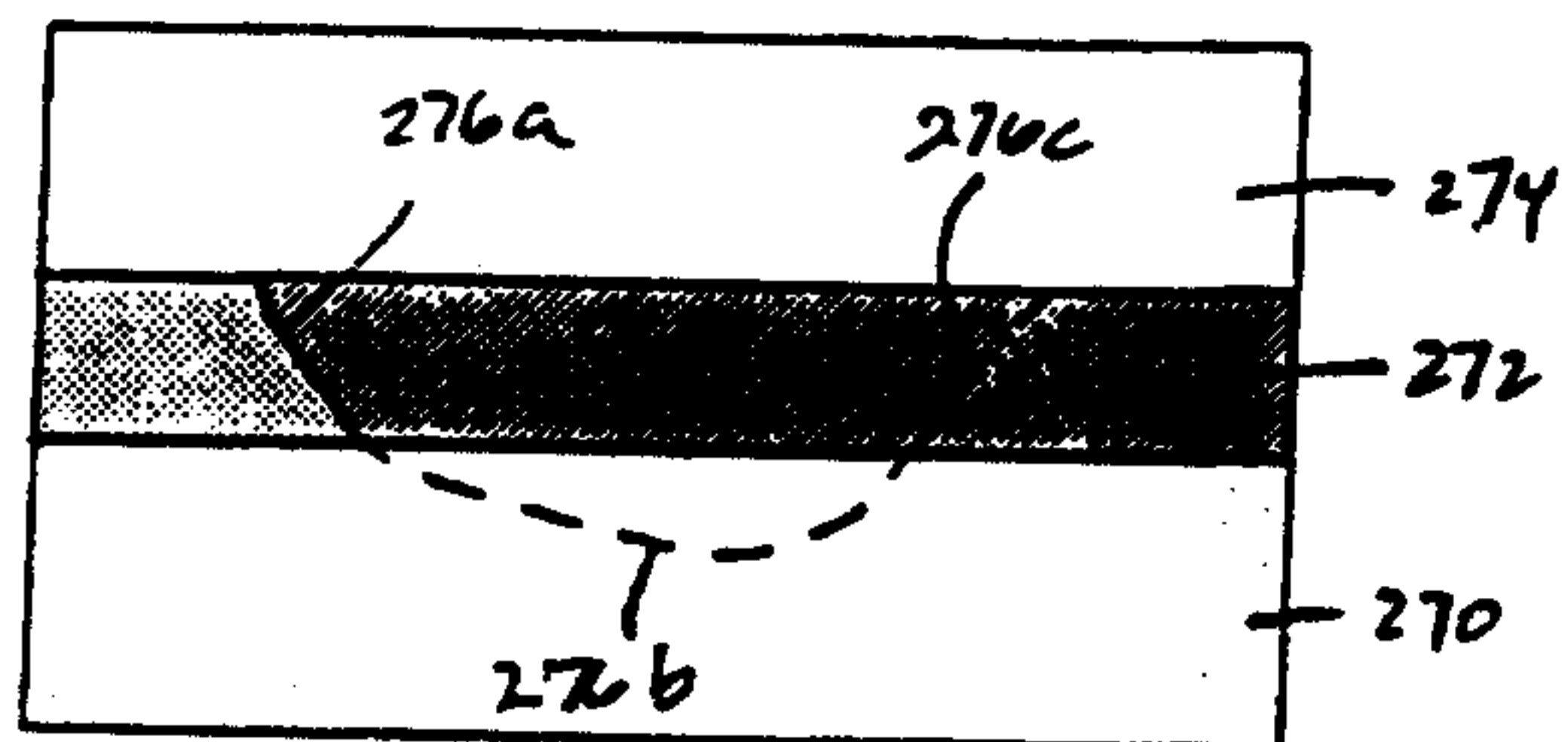


Fig. 15c

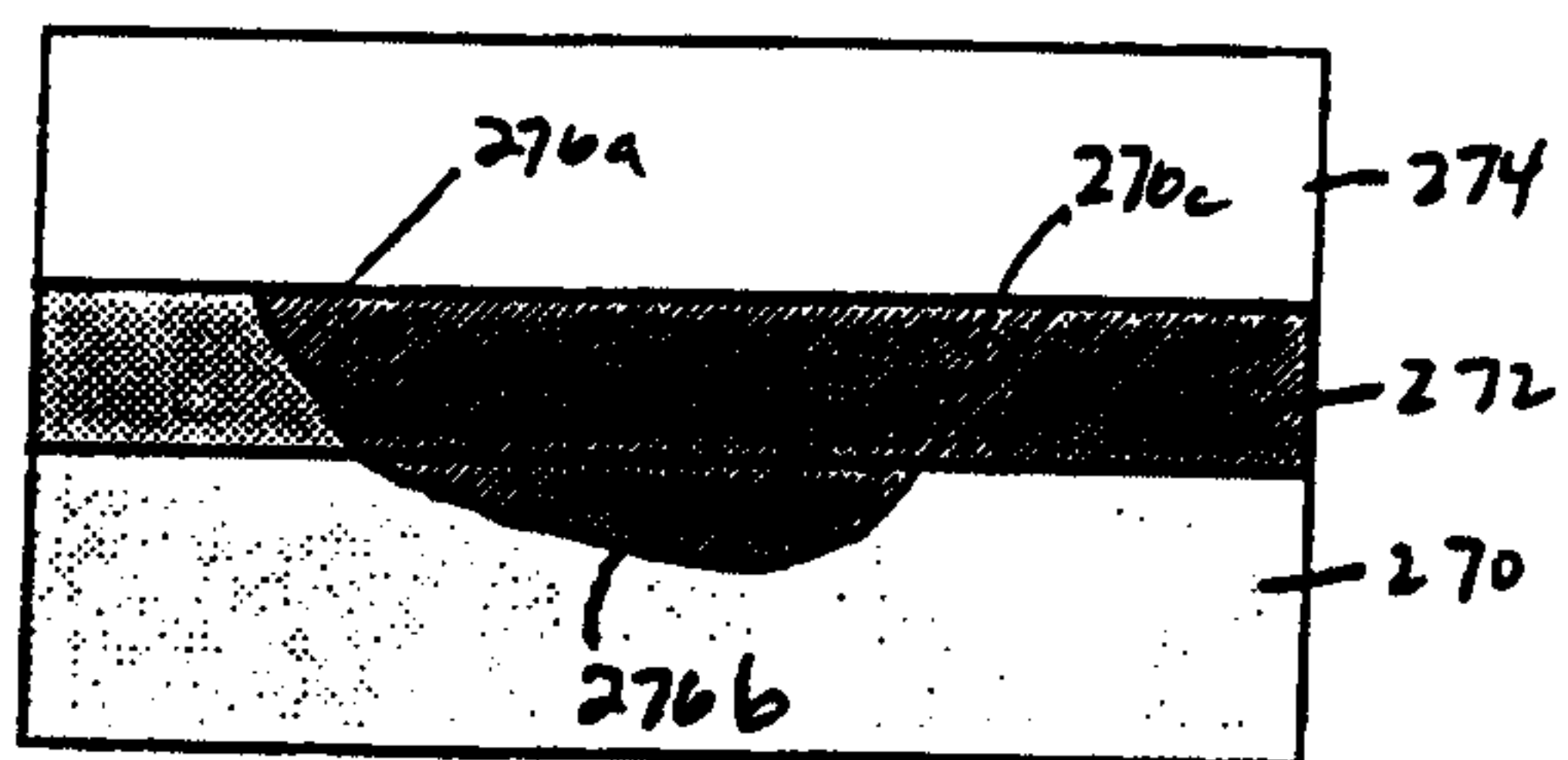


Fig. 15d

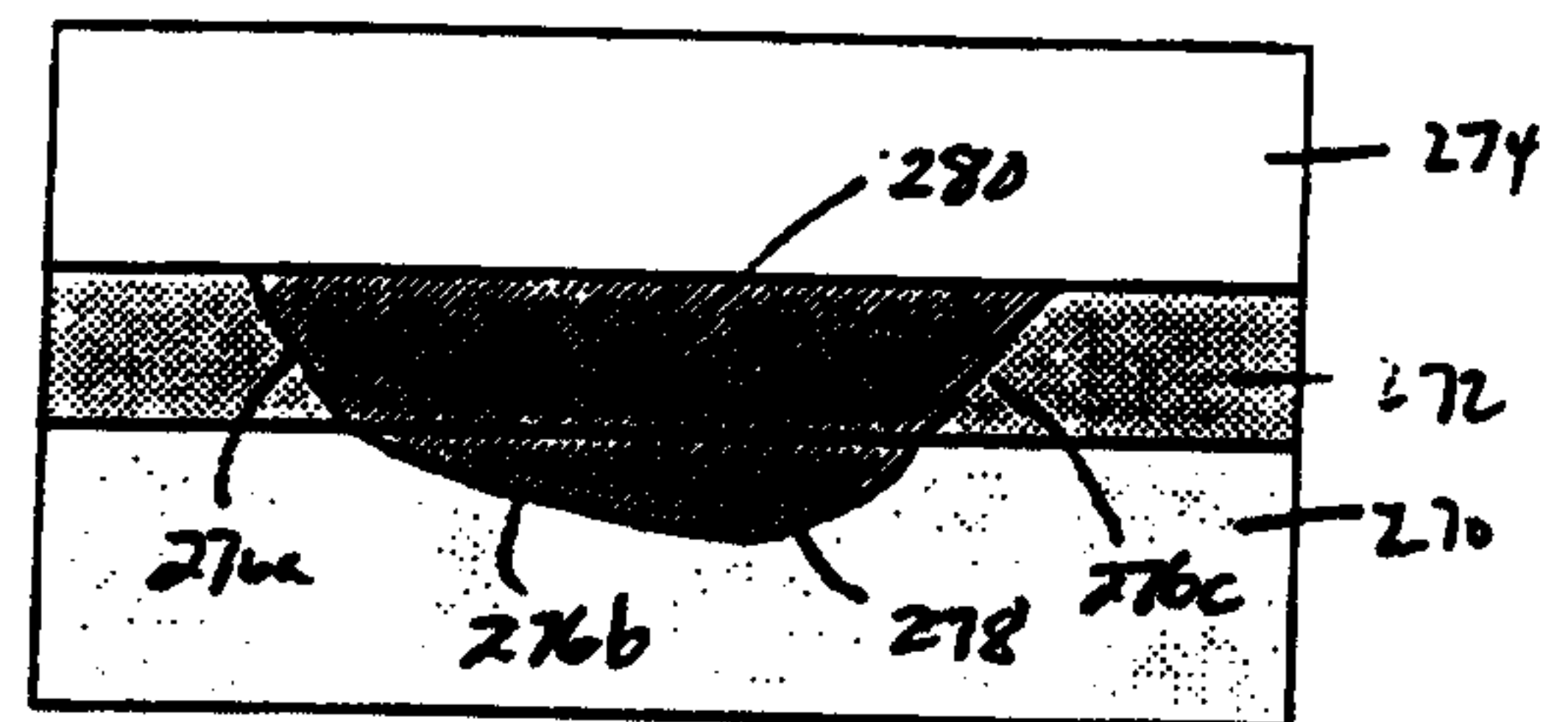


Fig. 15e

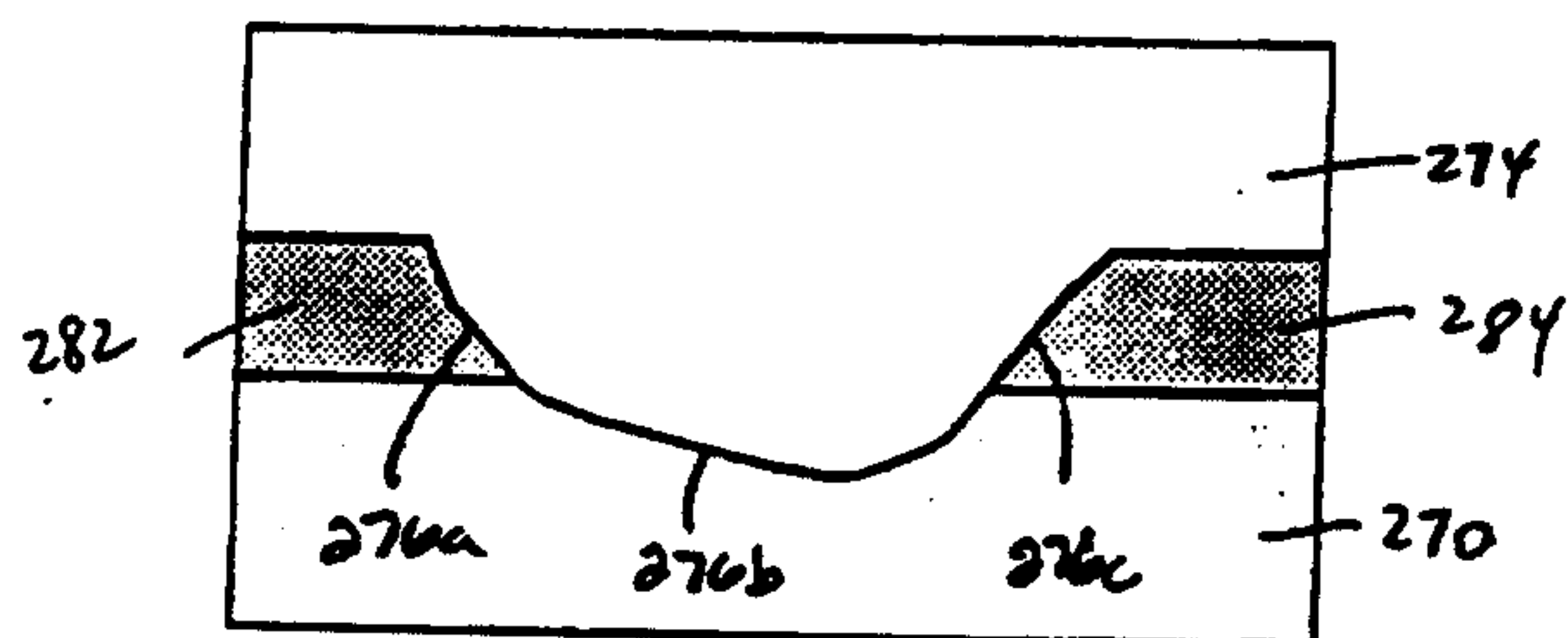


Fig. 15f

 Inconsistent region

18 / 18

