



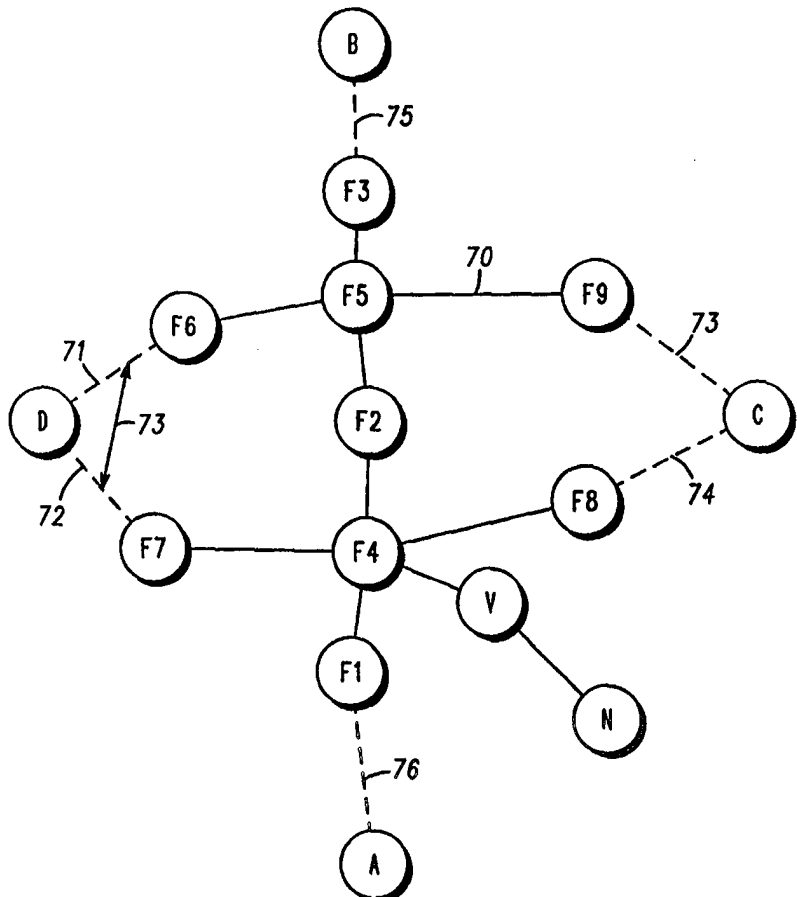
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<p>(51) International Patent Classification <sup>6</sup> : <b>G06F 17/50</b></p>	<p><b>A1</b></p>	<p>(11) International Publication Number: <b>WO 98/16891</b> (43) International Publication Date: 23 April 1998 (23.04.98)</p>
<p>(21) International Application Number: PCT/EP97/05633 (22) International Filing Date: 13 October 1997 (13.10.97) (30) Priority Data: 08/730,046 15 October 1996 (15.10.96) US (71) Applicant: MOTOROLA, INC. [US/CH]; Wilmington Geneva Branch Office, 207, route de Ferney, Grand Saconnex, CH-1218 Geneva (CH). (72) Inventors: BRACHA, Gabriel; Steiner 7, 69860 Tel Aviv (IL). WEISBERGER, Eytan; Dolev 3, 47235 Ramat Hasharon (IL). ALGOR, Ilan; Harama 6, 55900 Ganei Tikva (IL). (74) Agent: RICHARDT, Markus; Motorola GmbH, Hagenauer Strasse 47, D-65203 Wiesbaden (DE).</p>		<p>(81) Designated States: JP, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).  <b>Published</b> <i>With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i></p>

(54) Title: A METHOD FOR ROUTING OF NETS IN AN ELECTRONIC DEVICE

(57) Abstract

In laying out electronic devices on a substrate the routing of nets is important in minimizing conductor area and improving performance. A method for routing of nets in an electronic device which has a plurality of clusters is disclosed which separates the intra cluster routing and the channel routing between the clusters. For the intra cluster routing a topological graph is proposed to be used for mapping a subgraph which is representative of the nodes in a net belonging to a cluster to be routed.



**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

<b>AL</b>	Albania	<b>ES</b>	Spain	<b>LS</b>	Lesotho	<b>SI</b>	Slovenia
<b>AM</b>	Armenia	<b>FI</b>	Finland	<b>LT</b>	Lithuania	<b>SK</b>	Slovakia
<b>AT</b>	Austria	<b>FR</b>	France	<b>LU</b>	Luxembourg	<b>SN</b>	Senegal
<b>AU</b>	Australia	<b>GA</b>	Gabon	<b>LV</b>	Latvia	<b>SZ</b>	Swaziland
<b>AZ</b>	Azerbaijan	<b>GB</b>	United Kingdom	<b>MC</b>	Monaco	<b>TD</b>	Chad
<b>BA</b>	Bosnia and Herzegovina	<b>GE</b>	Georgia	<b>MD</b>	Republic of Moldova	<b>TG</b>	Togo
<b>BB</b>	Barbados	<b>GH</b>	Ghana	<b>MG</b>	Madagascar	<b>TJ</b>	Tajikistan
<b>BE</b>	Belgium	<b>GN</b>	Guinea	<b>MK</b>	The former Yugoslav Republic of Macedonia	<b>TM</b>	Turkmenistan
<b>BF</b>	Burkina Faso	<b>GR</b>	Greece			<b>TR</b>	Turkey
<b>BG</b>	Bulgaria	<b>HU</b>	Hungary	<b>ML</b>	Mali	<b>TT</b>	Trinidad and Tobago
<b>BJ</b>	Benin	<b>IE</b>	Ireland	<b>MN</b>	Mongolia	<b>UA</b>	Ukraine
<b>BR</b>	Brazil	<b>IL</b>	Israel	<b>MR</b>	Mauritania	<b>UG</b>	Uganda
<b>BY</b>	Belarus	<b>IS</b>	Iceland	<b>MW</b>	Malawi	<b>US</b>	United States of America
<b>CA</b>	Canada	<b>IT</b>	Italy	<b>MX</b>	Mexico	<b>UZ</b>	Uzbekistan
<b>CF</b>	Central African Republic	<b>JP</b>	Japan	<b>NE</b>	Niger	<b>VN</b>	Viet Nam
<b>CG</b>	Congo	<b>KE</b>	Kenya	<b>NL</b>	Netherlands	<b>YU</b>	Yugoslavia
<b>CH</b>	Switzerland	<b>KG</b>	Kyrgyzstan	<b>NO</b>	Norway	<b>ZW</b>	Zimbabwe
<b>CI</b>	Côte d'Ivoire	<b>KP</b>	Democratic People's Republic of Korea	<b>NZ</b>	New Zealand		
<b>CM</b>	Cameroon	<b>KR</b>	Republic of Korea	<b>PL</b>	Poland		
<b>CN</b>	China	<b>KZ</b>	Kazakistan	<b>PT</b>	Portugal		
<b>CU</b>	Cuba	<b>LC</b>	Saint Lucia	<b>RO</b>	Romania		
<b>CZ</b>	Czech Republic	<b>LI</b>	Liechtenstein	<b>RU</b>	Russian Federation		
<b>DE</b>	Germany	<b>LK</b>	Sri Lanka	<b>SD</b>	Sudan		
<b>DK</b>	Denmark	<b>LR</b>	Liberia	<b>SE</b>	Sweden		
<b>EE</b>	Estonia			<b>SG</b>	Singapore		

**A method for routing of nets in an electronic device**Technical Field

5

This invention relates to the design of electronic devices, such as integrated circuit chips and, more particularly, to the routing of nets in such chips to minimize the interconnection cost and to maximize the speed of operation.

10

Background of the Invention

From the prior art a number of approaches for designing very large scale integrated (VLSI) circuit chips is known especially as far as the routing of nets between the internal components of a chip are concerned. Channel routing is used extensively in the layout of integrated circuits and printed circuit boards. It is flexible enough to allow its use in various design styles such as gate-arrays, standard cells, and macro cells.

A channel router is designed to route nets that interconnect nodes on two opposite sides of a region called the channel. Criteria for a good channel router are that it routes the interconnections in a minimum area, the added length of the net wiring being also at a minimum. Examples for prior art channel routers are described for example in IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN, VOL. CAD-4, NO. 3, JULY 1985, pages 208-219, by James Reed et al and in "CHAMELEON: A NEW MULTI-LAYER CHANNEL ROUTER ", Douglas Braunt et al. , IEEE, 1986, conference proceedings of the 23rd design automation conference.

In order to increase the overall quality of the routing attempts have been made to include information about the so called "feed-throughs" in the process of channel routing. Such approaches are known from "FEED THROUGH RIVER ROUTING", A. Joseph et al, INTEGRATION, the VLSI Journal 8, pages 41-50, El Revier Science Publishers B.V. , 1989 and "MULTI CHANNEL OPTIMIZATION IN GATE-ARRAY LSI LAYOUT", K. Aoshima and E. S. Kuh, IEEE, 1983.

Since there is a constant need to further increase the wiring density in VLSI technology it is an underlying problem of the invention to devise an improved method for routing of nets for the design of electronic devices.

5

### Summary of the Invention

The underlying problem of the invention is solved basically by applying the features laid down in the independent claims. Preferred embodiments of  
10 the invention are given in the dependent claims.

The invention is advantageous in that it allows to design a VLSI layout requiring a minimum of silicon floor space which has a positive effect on chip yield and price. These extreme layout densities can be obtained by a  
15 fully automated solution which only requires moderate computing resources in terms of computer memory and run time.

The invention allows to generate separate sub graphs which contain the nodes to be connected both for the so called "cluster areas" and the "channel  
20 areas". These sub graphs are derived from input data which comprise the nodes of each net to be routed as well as the assignment of the nodes to clusters. Additional logical nodes which are defined by means of a minimal Steiner tree are inserted into the input data whereby the subgraphs are  
25 found. When the clusters are aligned horizontally it is advantageous to use rectilinear Steiner trees for defining such additional logical nodes.

Each cluster has one subgraph for each net to be wired. Analogously each channel has a dedicated subgraph for each net. As the subgraphs are descriptive of the interconnections to be established any method, including  
30 hand crafted design, can be used to actually route the subgraphs - or with other words - to map the subgraphs onto the physical design of the integrated circuit chip.

For routing the channel area a lot of automated prior art methods are  
35 known. If such a method is to be employed the subgraphs of the channel areas can be used as input data for such prior art methods.

-3-

For routing of the clusters it is advantageous to employ the automated routing method which is described in the dependent method claims. For this method first the actual electrical connectivities which can be established between the nodes in each of the clusters have to be described.

5 This is done by a graph representation which forms the basis of the intra cluster routing method.

According to this method subgraphs which represent the desired electrical interconnections to be established between the nodes in a given cluster are mapped on a graph which represents the potential electrical connectivities

10 within the given cluster. To carry out this procedure no human interaction is required. The steps of the method of the invention can be realized by programming a general purpose digital computer, which is fed with the required design data, i.e. the placement of the clusters, the nets to be wired

15 and the assignment of the nodes of the nets to the clusters.

#### Brief Description of the Drawing

20

FIG. 1 is a schematic circuit diagram illustrating the structure of a bit cell;

FIG. 2 is a simplified diagram to illustrate the process of concatenating individual bit cells and cluster routing;

25

FIG. 3 is a portion of a circuit diagram which illustrates the formation of clusters and channels between the clusters;

FIG. 4 is a simplified circuit diagram which illustrates the definition of

30 logical nodes;

FIG. 5 is a diagram which shows the transformation of the input data into two separate subgraphs;

35 FIG. 6 and 7 show the modeling of the feed throughs according to the invention;

FIG. 8 is a flow chart illustrating the generation of subgraphs for cluster and channel routing;

FIG. 9 is a flow chart illustrating the generation of a graph which  
5 represents potential connectivities in a cluster; and

FIG. 10 is a flow chart which illustrates the intra cluster routing according to the invention.

10 In FIG. 1 there is shown a schematic view of the circuit layout of bit cell 19. A bit cell which sometimes is also called a leave cell is the basic building block of a functional block in an integrated circuit, such as a data-path block. In general terms, a bit cell is planned to perform a given functionality, to provide a certain interface for connections, and to support  
15 traversing routing through and/or over itself.

The cell functionality of a bit cell is established by transistors and predefined via geometries. Certain points within a bit cell are designated as internal pins or potential connection points. Such points will be referred to  
20 as "nodes" in the following. The nodes are typically implemented in the top most metal layer of the circuit structure but can also be implemented in any other layer.

The typical way of establishing a connection to a pin is by vias. Vias are  
25 layout structures that allow traversing from one layer of the circuit topology to another. In order to optimize the layout process optional or potential vias are available in the underlying structure of the electronic device to be designed. When the potential vias are not required for the circuit layout they do not obstruct and allow other routing to traverse.

30 In addition one can designate special paths that are potential connections of the node of a bit cell to the boundary of the bit cell, or that traverse the bit cell from one side to another. Such structures are termed feed-throughs. These are also potential geometries of connectivities that are only  
35 implemented if they are actually required to establish the routing between the nodes of bit cells or to traverse over a bit cell.

By way of example the bit cell 19 has a number of horizontal feed-throughs 10 and horizontal feed-throughs 12. The feed throughs 10 and 12 define potential wiring geometries which can be used for routing the integrated circuit. Vias 14 which are symbolized by the filled dashed boxes in FIG. 1  
5 serve to interconnect nodes 18 to the feed throughs.

Vias 16 which are symbolized by the empty dashed boxes in FIG. 1 define potential vias which are not directly assigned to one of the nodes 18. Hence the feed throughs 10, 12 and the vias 14, 16 define allowed electrical  
10 connectivities for interconnecting the nodes 18 within the bit cell 19 to nodes within the same bit cell 19 or to other bit cells or external pins of the integrated circuit chip not shown in FIG. 1.

FIG. 2 includes a sequence of three figures, i.e. FIG. 2A, 2B and 2C to  
15 illustrate the process of concatenation and routing. FIG. 2A shows a bit cell 29 which is similar to bit cell 19 of Fig. 1. For clarity the layout of the bit cell 29 is simplified: the bit cell 29 has only feed throughs 20 and 22 as well as a node 28 even though in practice a much larger number of components and wiring geometries typically is present.

20 FIG. 2B illustrates how a plurality of bit cells 29 are concatenated. The feed throughs 20, 22 are designed so that when cells are concatenated or abutted, their feed throughs also abbut to create longer and more complex potential paths for connectivities to form a lattice or grid that spans a  
25 plurality of concatenated bit cells. Such a grid simplifies the task for the router software.

FIG. 2C shows by way of example how the concatenated bit cells 29 of FIG. 2B are routed. The hatched or "painted" feed throughs 24 and 26 are  
30 actually used for the physical implementation of the integrated circuit whereas the other potential electrical connectivities as defined by the remaining feed throughs 20, 22 are not used to establish electrical interconnections between nodes 28.

35 FIG. 3 shows a portion of integrated circuit chip 39. The integrated circuit chip 39 comprises a large number of clusters. For simplicity only clusters 30, 31 and 32 are shown in FIG. 3. Each of the clusters 30, 31 and 32

-6-

comprises a plurality of bit cells which are concatenated like shown in FIG. 2. The gaps between the clusters 30, 31 and 32 are referred to as channels. Channel 37 is situated between the clusters 30 and 31; channel 38 is situated between the clusters 31 and 32. The clusters 30, 31 and 32 are interconnected by the wiring 36. The wiring 36 is composed of horizontal and vertical routes as well as vias.

One portion of the wiring 36 is implemented inside the clusters 30, 31 and 32 which is called intra cluster routing. Another portion of the wiring 36 is implemented in the channel areas 37, 38; this portion of the wiring 36 is referred to as channel routing. For simplicity only a small number of routes are shown in FIG. 3.

FIG. 4 comprises a schematic representation of clusters 40, 41 and 42 of the integrated circuit chip 39. Each of the clusters 40, 41 and 42 comprises a number of concatenated bit cells. The bit cells of the cluster 40 have nodes 1.1, 2.1, 1.2, 2.2 and 2.3; the bit cells of cluster 41 have nodes 2.4, 2.5, 1.3, 2.6 and 1.4; the bit cells of cluster 42 have nodes 1.5, 2.7, 1.6, 2.8 and 1.7. These nodes belong to different nets: a set of nodes which are to be interconnected to the same signal will be referred to as a net in the following.

All nodes 1.x belong to net 1 whereas all nodes 2.x belong to net 2. For proper operation of the integrated circuitry to be implemented it is essential that the routes of different nets which interconnect the required nodes do not contact. On the other hand for good utilization of the available silicon floor space and for short signal propagation delays a minimum over all length of the circuit wiring is highly desirable. This applies both to the intra cluster routing of the clusters 40, 41 and 42 as well as to the channel routing.

In the example considered here there is a channel area A in the horizontal direction between clusters 40 and 41 and a channel area B between the clusters 41 and 42 as well as channel areas C and D in the vertical direction.

For routing the nets in a first step a minimal steiner tree 44 which interconnects the nodes belonging to net 1 of all of the clusters of the



-7-

integrated circuit including clusters 30, 31, 32, 40, 41 and 42 is established. The condition for the minimal steiner tree 44 is that

- 5 a) it interconnects at least one node of net 1 of each of the clusters, provided that there is at least one node belonging to net 1 in a given cluster; and that
- b) its overall length is minimal.

This applies analogously for minimal steiner trees for other nets, e.g. net 2. Since only rectilinear routes are allowed in the technical environment considered here also the steiner tree 44 has to be rectilinear. The theory of steiner trees as such is known from the art, e.g. from M. Hanan, "ON STEINER MINIMUM TREES WITH RECTILINEAR DISTANCES", SIAM, J. Appl. Math. Vol 16, no. 1, pages 255-265, 1966. The usage of rectilinear Steiner trees is also devised in US patent 4,577,276 to Dunlop et al. for laying out integrated circuits on a substrate.

The Steiner tree 44 for the net 1 interconnects one node belonging to net 1 in each of the clusters. As shown in FIG. 4 the Steiner tree 44 logically interconnects the node 1.2 in cluster 40, node 1.3 in cluster 41 and node 1.6 in cluster 42. Other clusters which contain at least one node which belongs to net 1 which are not shown in FIG. 4 are also spanned by the steiner tree 44 in the same way.

25 Analogously a minimum steiner tree is also established for the net 2. For simplicity the steiner tree of net 2 is not shown in FIG. 4. In a practical design much more than 2 nets are present so that a large number of steiner trees is established in a practical design.

30 Not all clusters necessarily comprise a node of each net in a practical design. Hence for each net a subset of clusters exists. Each cluster belonging to such a subset comprises at least one node belonging to the net to which the subset of that cluster is assigned. In the more general case considered here a minimal steiner tree of one of the nets to be routed has to span all clusters of this net. Thereby at least one node of each cluster  
35 having a node belonging to the net to be routed is logically interconnected to

at least another node of the same net in the other clusters of the same subset of clusters.

It is to be noted that the minimal steiner trees, especially steiner tree 44 shown in FIG. 4, do not directly define the placement of the circuit wiring. Once the steiner trees are found, i.e. one minimal steiner tree for each net, additional logical nodes are defined for generating an enhanced graph representation of each of the nets to be routed.

These logical nodes are determined by the intersections of the steiner trees with the boundaries of the channel area. In the example considered in FIG. 4 these are the logical nodes V1, V2, V3 and V4. The logical node V1 is situated at the intersection of the steiner tree 44 and the lower boundary of cluster 40 to the channel area A. Likewise the logical node V2 is situated at the upper boundary of cluster 41 to the channel area A; logical node V3 is situated at the lower boundary of cluster 41 to the channel area B and logical node V4 is situated at the upper boundary of cluster 42 to the channel area B. The minimal steiner tree of each of the nets to be routed yields such logical nodes.

20

FIG. 5 shows a graph representation  $G_1$  of the portion of net 1 shown in FIG. 4. The graph representation  $G_1$  comprises a list of the nodes 1.x which are assigned to the net 1. Further, the graph representation comprises topological information as regards the placement of the nodes 1.x in the clusters. This information is symbolized by the sets 50, 51 and 52 corresponding to clusters 40, 41 and 42 of FIG. 4, respectively. The information contained in the graph representation  $G_1$  is required as input data.

The additional logical nodes V1, V2, V3 and V4 which are found by means of the minimal steiner tree 44 are included into the graph representation of the net 1 so that the graph representation  $G_1$  is transformed to the graph representations  $g_{1C}$  and  $g_{1CH}$ . This transformation is performed by including the additional logical nodes V1, V2, V3 and V4 into the list of nodes contained in the graph representation  $G_1$ .

35

The additional logical nodes are inserted into the subset of nodes 50, 51 and 52 so that the subgraphs  $g_{10}$ ,  $g_{11}$  and  $g_{12}$  result. The portions of the steiner tree 44 which are in the channel areas are reflected in the graph representation  $g_{1CH}$  which comprises the subgraphs  $g_{1A}$  and  $g_{1B}$ . The  
5 subgraph  $g_{1A}$  is representative of the interconnections to be established for the net 1 over the channel area A, whereas the subgraph  $g_{1B}$  is representative of the interconnections to be established for the net 1 over the channel area B - the so called channel routing.

10 Hence, the subgraph  $g_{1A}$  includes the logical nodes V1 and V2 which are situated at the boundary of channel A and which are interconnected by the steiner tree 44, whereas the subgraph  $g_{1B}$  includes the interconnected logical nodes V3 and V4 which are situated at the boundary of the channel area B.

15

Thus two different sets of subgraphs result for the net 1: the first set of subgraphs comprises subgraphs  $g_{10}$ ,  $g_{11}$ ,  $g_{12}$ . Each of these subgraphs covers one of the clusters as far as the nodes and the logical nodes for net 1 inside that cluster are concerned.

20

The second set of subgraphs comprises the subgraphs  $g_{1A}$  and  $g_{1B}$ . This set of subgraphs covers the channel areas A and B.

For simplicity and clarity the corresponding graphs for net 2 are not shown  
25 in the FIG. 5.

FIG. 6 shows a simplified example of potential connectivities which can be physically implemented within a cluster area 60. Within the cluster area 60 there is a node N which can be interconnected by a via V to a top plate T4 of  
30 the via V. The top plate T4 can be contacted by one of the feed throughs F1, F2, F7 and F8. By way of the feed throughs F2 and F7 other feed throughs F3, F5, F6 or F9 can be accessed. The feed throughs as well as the top plate T4 and the via V define allowed electrical connectivities which can be used by the router to wire the nets. The information contained in FIG. 6 is  
35 modelled by graph representation shown in FIG. 7:

There are three different types of edges in the graph of FIG. 7: The first type of edge establishes geometrical adjacency between its vertices. Geometrical adjacency means that there is an potential electrical connectivity between the vertices of that edge. For example, the edge 70 which links the vertices  
5 F5 and F9 represents the geometry of the cluster area 60 of FIG. 6 in which the feed throughs F5 and F9 abut so that an electrical connectivity making usage of the concatenated feets F5 and F9 can be established. This type of geometrical adjacency is symbolized by a normal straight line between  
10 vertices in FIG. 7.

The second type of edges establishes a mutual exclusion relation between its vertices with respect to different nets. This is the case for the edges 71 and 72 which link the vertices D, F6 and D, F7, respectively. The mutual exclusion relation between the edges 71 and 72 is symbolized by the double  
15 headed arrow 73 between these edges. The mutual exclusion relation is representative of the fact that the feets F6 and F7 can not be used for wiring different nets. This is due to the fact that in the topology of allowed electrical connectivities shown in FIG. 6 the feed throughs F7 and F6 contact so that  
20 both feed throughs F6 and F7 can not be used for wiring different nets having different signals. Otherwise a shortcircuit between the nets carrying different signals would result. Such a relationship of mutual exclusion does not exist between the feed through F8 and F9 since these feed  
25 throughs do not contact in the circuit topology shown in FIG 6.

The third type of edge establishes a channel border relation for  
30 representation of the additional logical nodes. The vertices A, B, C and D are representations of such logical nodes analogous to the logical nodes V1, V2, V3 and V4 of FIG. 4. This third type of edge symbolizes an allowed electrical connectivity by way of a feed through, via or node to an additional  
35 logical node of one of the channel areas. In the example considered here these are the vertices 71 to 76 which are represented by dashed lines.

For example the feed through F3 directly interconnects the cluster area 60 to the channel area A and thus can serve to establish an interconnection to  
35 the logical point V1 which has been found by means of the minimal steiner tree 44 (cf. FIG. 4).

Relationships of mutual exclusion can only exist between edges of the third type.

5 The graph shown in FIG. 7 is used for mapping the subgraphs which are representative of the interconnections of nodes belonging to the same net within one cluster, for example  $g_{10}$ ,  $g_{11}$ ,  $g_{12}$  of FIG. 5, onto the actual allowed electrical connections of the given circuit topology, including potential connectivities .

10 FIG. 8 is a flow chart illustrating one embodiment of the method of the invention. According to this embodiment the method is carried out by a computer program without human interaction.

15 In step 80 the input data are provided to the computer program. The input data are sets of nodes  $NO_j$ . Each set of nodes  $NO_j$  defines one net  $N_i$  to be routed. All nodes within the same net eventually have to be interconnected. Further information is provided for each of the nodes  $NO_j$  to which one of the clusters  $C_k$  it belongs. This information is equivalent to the graph representation  $G_1$  of FIG. 5.

20

In step 81 the variable  $i$  is initialized to select one of the nets  $N_i$  for routing. In step 82 a minimal rectilinear steiner tree is generated for the selected net  $N_i$ . Like in the example described in FIG. 4 the minimal rectilinear steiner tree connects nodes  $NO_j$  of the same net  $N_i$  in different clusters  
25 across the channels. In step 83 the intersections of the steiner tree found in step 82 with the boundary between the channel areas and the clusters are determined.

30 In step 84 the logical nodes  $V_{ik}$  are defined as the intersections of the steiner tree with the channel cluster boundary as determined in step 82. This is analogous to the definition of the logical nodes  $V_1$ ,  $V_2$ ,  $V_3$  and  $V_4$  of FIG. 4.

35 In step 85 the logical nodes  $V_{ik}$  found in step 84 are included in the net  $N_i$ . Thereby subgraphs  $g_k$  for the clusters and subgraphs  $g_{CH}$  for the channels are generated in the same way as shown in the example of FIG. 5 where the subgraphs  $g_{10}$ ,  $g_{11}$ ,  $g_{12}$  for the clusters and  $g_{1A}$ ,  $g_{1B}$  for the channels result. After step 85 is carried out the variable  $i$  is incremented to select

-12-

another net  $N_i$ . This incrementation is performed in step 86. The above described loop of steps 82 to 85 is carried out for all nets  $N_i$  until  $i$  reaches its maximum value as defined by the number of nets which are inputted.

5 If  $i$  reaches its maximum value all required subgraphs are generated and the control goes to steps 87 and 88. In step 87 the routing of the interconnections of nodes belonging to the same net is carried out for the internal area of each cluster - so called intra cluster routing - whereas in step 88 each of the subgraphs  $g_{CH}$  belonging to different ones of the nets  $N_i$   
10 are channel routed.

This task of routing the subgraphs  $g_k$  and  $g_{CH}$  can be described as mapping the subgraphs onto the actual circuit topology which has a certain number of allowed electrical connectivities. The task of routing - or with other words  
15 mapping - the subgraphs  $g_k$  and  $g_{CH}$  can be carried out by any appropriate method, including hand crafted mapping. However, in order to optimize this process, it is advantageous to employ a preferred embodiment of a method according to the invention which allows to carry out the routing automatically. Therefore the generation of a model which represents  
20 potential electrical connectivities of the physical circuit topology is required.

FIG. 9 is a flow chart which illustrates the generation of such a model. In step 90 topology data of one of the cluster regions  $C_k$  are inputted. These topology data are representative of the placement of feed throughs, vias, top  
25 and bottom plates and the like as shown in the example of FIG. 6. The representation of the topology data of potential connectivities of the clusters can be in any form.

In step 91 the variable  $k$  is initialized to select one of the clusters  $C_k$ . In step  
30 92 the input data of step 90 is analyzed to find all potential connectivities of nodes  $NO_j$  in the selected cluster to other nodes in the same cluster or to channel boundaries of the same cluster irrespective of any assignment of the nodes to specific nets. After the analyzation of the input data in step 92 a graph is generated in step 93 which represents the allowed connectivities.  
35 Each vertice in the graph is representative of an element of the circuitry like nodes, feed throughs, vias and top or ground plates whereas the edges

between these vertices are representative of allowed connectivities between these potential circuit elements.

In step 94 the information about the type of each edge is added to the graph  
5 generated in step 93. A type 1 edge defines geometrical adjacency between its vertex elements, an edge of type 2 defines mutual exclusion of the electrical connectivities as indicated for example by the double headed arrow 73 in FIG. 7. An edge of type 3 defines a channel border relation.

10 Subsequently in step 95 the variable  $k$  is incremented so that another cluster  $C_k$  is selected. Hence the steps 91 to 94 are repeated for all clusters  $C_k$  which are inputted in step 90.

15 FIG. 10 illustrates one embodiment of the method of the invention for performing the intra cluster routing of nets by mapping the cluster subgraphs generated by means of the steiner trees onto the model of allowed electrical connectivities established in accordance with the method of FIG. 9.

20 In step 100 the input data are provided. The required input data are the topological graphs of all the clusters  $C_k$  which have been found as a result of the method of FIG. 9. Further all the subgraphs  $g_k$  of the clusters  $C_k$  for each net  $N_i$  are also inputted. The subgraphs  $g_k$  are generated according to the method as described above with reference to FIG. 8. The method  
25 described here with reference to FIG. 10 is one way of implementing step 88 of FIG. 8.

In step 105 the variable  $k$  is initialized to select one of the clusters  $C_k$ . In  
30 step 110 the variable  $i$  is initialized to select one of the nets  $N_i$  to be routed.

In step 115 a list of logical components for the selected cluster is initialized for purposes of routing the selected net. Initially each node in the selected cluster  $C_k$  which belongs to the selected net  $N_i$  is defined to be a separate logical component so that the initial list of logical components comprises  
35 all nodes of the selected cluster which belong to the selected net  $N_i$ . With other words: The list of logical components comprises all nodes comprised

-14-

in the relevant subgraph  $g_k$  of the selected cluster  $C_k$  for the selected net  $N_i$ ,  
in particular the logical nodes  $V_{ik}$ .

5 In step 120 one of the logical components is selected from the list of logical  
components for processing.

10 In step 125 a route is searched between the selected component and another  
component from the same list of logical components which has to be  
connected to the selected component according to the subgraph  $g_k$  of the  
selected cluster  $C_k$  for routing of the selected net  $N_i$ . For searching a route  
between the selected component and the other component the topological  
graph of the selected cluster  $C_k$  is used.

15 The usage of the topological graph for searching such a route is possible  
because the topological graph is representative of the allowed electrical  
connectivities which can be established in the circuit topology. Any known  
search algorithm can be used for finding a path in the given topological  
graph between the two components which are represented by vertices in the  
topological graph. If such a route is found in step 130 it is decided to go to  
20 step 135.

In step 135 the edge or the edges in the topological graph which represent  
the route which is found in step 125 is/are painted to indicate that these  
edge/edges are no longer available for routing another net. This is to  
25 prevent that electrical connectivities are established between different nets  
which would lead to mal functioning of the integrated circuit.

30 In step 140 the logical components for which an interconnecting route has  
been found in step 125 are logically united to one single component. The  
original components are deleted from the list of logical components and  
replaced by one single united component.

35 In step 150 the list of logical components is checked for the number of  
components which are left in the list. If there is more than one logical  
component left the control goes back to step 120 to select one of the residual  
logical components and to find a route between another residual  
component. After all the logical components are united into one logical



-15-

component the variable  $i$  is incremented in step 155 to select the next net  $N_i$  to be routed. As a consequence in step 115 a new list of logical components for the selected net is established. The selected cluster  $C_k$  remains the same. When all nodes of all nets  $N_i$  within the selected cluster  $C_k$  are  
5 routed the variable  $k$  is incremented to select another cluster  $C_k$  until all clusters are routed.

If no route is found in step 125 it is decided in step 130 to carry out step 160. In step 160 an obstructing route of another net which has already been  
10 routed previously is ripped up. This removal of the obstructing route allows to establish a route between the selected component and the other component as described with reference to step 125.

As a consequence of the ripping up of a route in the other net which had  
15 already been routed previously the corresponding logical components in the list of logical components are separated in the other net which has been ripped up. This is done in step 165. As a consequence at least two separate logical components result in the list of logical components for the ripped up net. The ripped up net is put back on the stack for later processing to find  
20 another route to interconnect the separated logical components.

After step 165 the control returns back to step 135 to paint the edge or the edges in the topological graph which represent the route which has been found due to the removal of the route of the other net.  
25

In order to further speed up and enhance the above described method it is possible to select the logical components in step 120 according to a certain criteria: it is advantageous to select the logical component from the list which is the most constraint component. The most constraint component  
30 has the least options for electrical connectivities according to the topological graph.

For example the node  $N$  which is considered as a logical component here, can be accessed from four different directions according to the diagram  
35 shown in FIG. 6. This can be done through the feed throughs  $F_1$ ,  $F_2$ ,  $F_7$  or  $F_8$ . Other nodes might have lesser or larger number of potential connectivities which determines the constraint factor. Hence each logical

-16-

component in the list should have an associated constraint factor which is indicative of the number of options for electrical connectivities to or from the logical component.

- 5 Also when logical components are united in step 140 a new constraint factor is to be established. Usually when two components are united, the resulting logical component has a lower constraint factor because the bigger united logical component has usually more options for electrical connectivities.

10

15

Claims

1. A method for routing of nets in an electronic device having a plurality of  
5 clusters, said clusters being separated by a channel area between the  
clusters, and said clusters having a plurality of nodes, each of said nodes  
being assigned to one of said nets, said method comprising the steps of:

a) for each set of nodes belonging to the same net:

10

establishing a minimal steiner tree between a sub set of said clusters  
which have at least one of said nodes of said set of nodes, said  
minimal steiner tree logically interconnecting at least one of said  
nodes of said set of nodes of each of said clusters of said sub set of  
15 clusters;

b) for each of said steiner trees found in step a):

20

determining logical nodes where said steiner tree intersects a  
boundary of said channel area; including said logical nodes in a  
graph representation of said net, so that first sub graphs covering  
said clusters and second sub graphs covering said channel area  
result;

25 c) for each cluster:

routing each of said first sub graphs found in step b) independently  
from channel routing of said second sub graphs found in step b).

30 2. The method according to claim 1 further comprising the steps of

d) for each of said clusters: for each node of said cluster forming a second  
graph representation to model potential electrical connectivities to said  
node within said cluster;

35

-18-

e) routing each of said first sub graphs found in step b) of claim 1 in accordance with the electrical connectivities as modeled by said second graphs.

5 3. The method according to claim 2 said second graph representation including vertices to represent potential electrical conductors and edges to represent potential electrical connections between said vertices, whereby an edge can have one of the following types:

10 i) a first type of edge which establishes geometrical adjacency between its vertices;

ii) a second type of edge which establishes a mutual exclusion relation between its vertices with respect to different nets; and

15

iii) a third type of edge which establishes a channel border relation for representation of said logical nodes.

20 4. The method according to claim 2, said method further comprising the steps of:

for each of said clusters:

for each of said first sub graphs covering said cluster:

25 f) establishing a list of logical components comprising said nodes of said first sub graph, each of said nodes of said first sub graph being initially defined to be a separate logical component in said list;

30 g) repetitively selecting one of said logical components from said list until there is only one single logical component left in said list;

35 h) routing from the selected component to another logical component of the same list, if possible;

-19-

5 i) in case of finding a route in said step h): removing said selected component and said other component from said list; uniting said selected component and said other component to create a new logical component; inserting said united component into said component list;

10 j) in case of not finding a route in said step h): ripping up an obstructing route of another one of said first sub graphs which has been processed previously;

k) in case of ripping up an obstructing route in said step j): finding an alternative route to replace said obstructing route.

15 5. The method according to claim 4 whereby said step k) is carried out by logically splitting said united component of said other first sub graph into at least two logical components by ripping up said obstructing route and carrying out said step g) and said step h) to find said alternative route.

20 6. The method according to claim 4 whereby said one of said logical components selected in said step g) of claim 9 is the logical component of said list which is the most constrained component.

25 7. An electronic device having a plurality of clusters, said clusters being separated by a channel area between the clusters, and said clusters having a plurality of nodes, each of said nodes being assigned to one of a plurality of nets, said nets being routed in accordance with a method comprising the steps of

30 a) for each set of nodes belonging to the same net:

establishing a minimal steiner tree between a sub set of said clusters which have at least one of said nodes of said set of nodes, said minimal steiner tree logically interconnecting at least one of said nodes of said set of nodes of each of said clusters of said sub set of clusters;

35

b) for each of said steiner trees found in step a):

5 determining logical nodes where said steiner tree intersects a boundary of said channel area; including said logical nodes in a graph representation of said net, so that first sub graphs covering said clusters and second sub graphs covering said channel area result;

c) for each cluster:

10 routing each of said first sub graphs found in step b) independently from channel routing of said second sub graphs found in step b).

8. The electronic device according to claim 7, said method further comprising the steps of

15

d) for each of said clusters: for each node of said cluster forming a second graph representation to model potential electrical connectivities to said node within said cluster;

20 e) routing each of said first sub graphs found in step b) of claim 7 in accordance with the electrical connectivities as modeled by said second graphs.

9. The electronic device according to claim 8, said method further comprising the steps of

25

for each of said clusters:

for each of said first sub graphs covering said cluster:

30 f) establishing a list of logical components comprising said nodes of said first sub graph, each of said nodes of said first sub graph being initially defined to be a separate logical component in said list;

35 g) repetitively selecting one of said logical components from said list until there is only one single logical component left in said list;

h) routing from the selected component to another logical component of the same list, if possible;

5 i) in case of finding a route in said step h): removing said selected component and said other component from said list; uniting said selected component and said other component to create a new logical component; inserting said united component into said component list;

10

j) in case of not finding a route in said step h): ripping up an obstructing route of another one of said first sub graphs which has been processed previously;

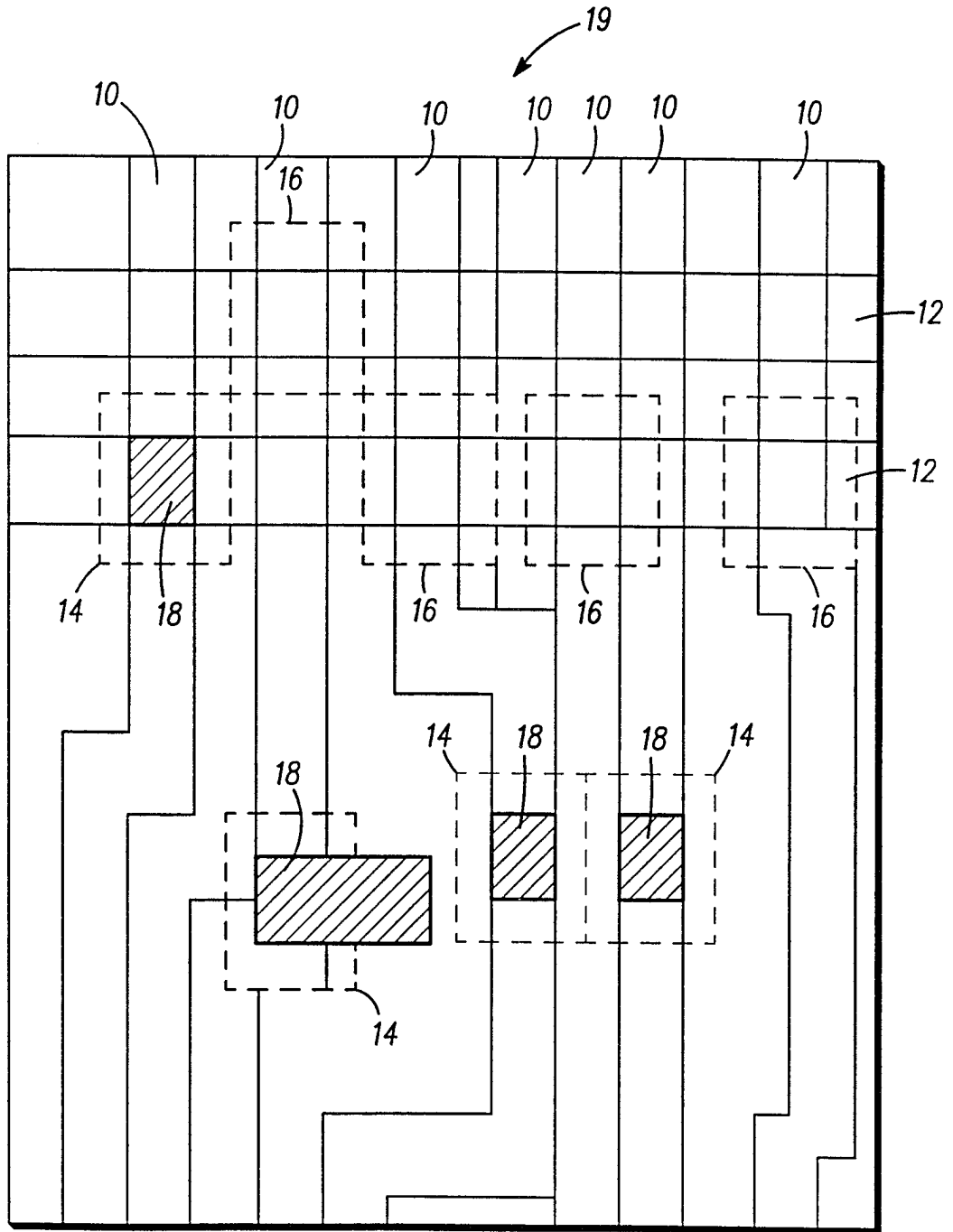
15

k) in case of ripping up an obstructing route in said step j): finding an alternative route to replace said obstructing route.

10. The electronic device according to claim 9, said method being characterized in that said step k) of claim 9 is carried out by logically  
20 splitting said united component of said other first sub graph into at least two logical components by ripping up said obstructing route and in that said step g) and said step h) of claim 9 are carried out to find said alternative route.

25 11. The electronic device according to claim 10, said method being characterized in that said one of said logical components selected in said step g) of claim 9 is the logical component of said list which is the most constrained component.

30



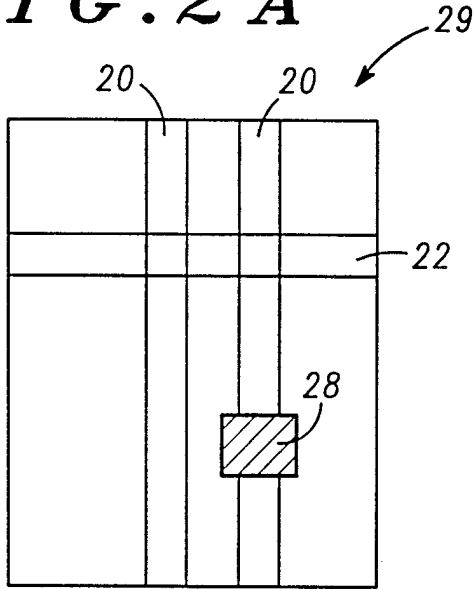
▨ = NODE

□ = VIA

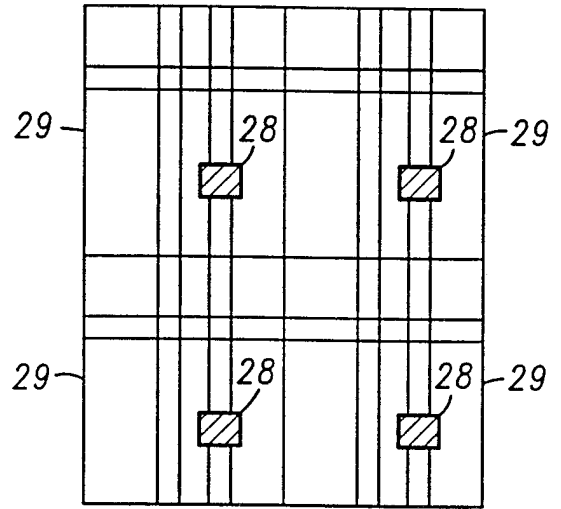
*FIG. 1*



**FIG. 2A**



**FIG. 2B**



**FIG. 2C**

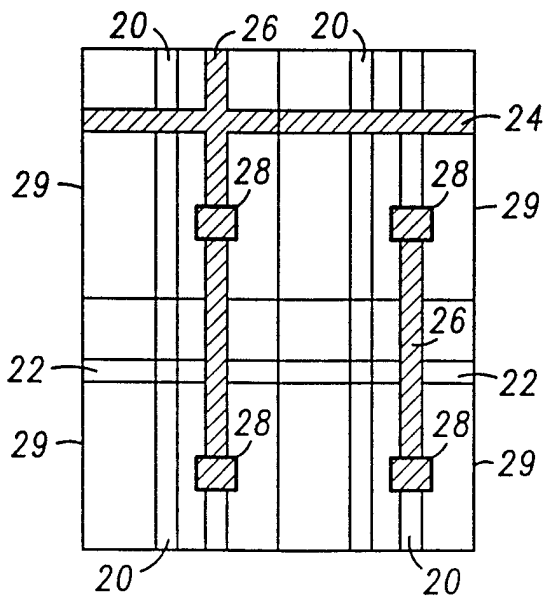
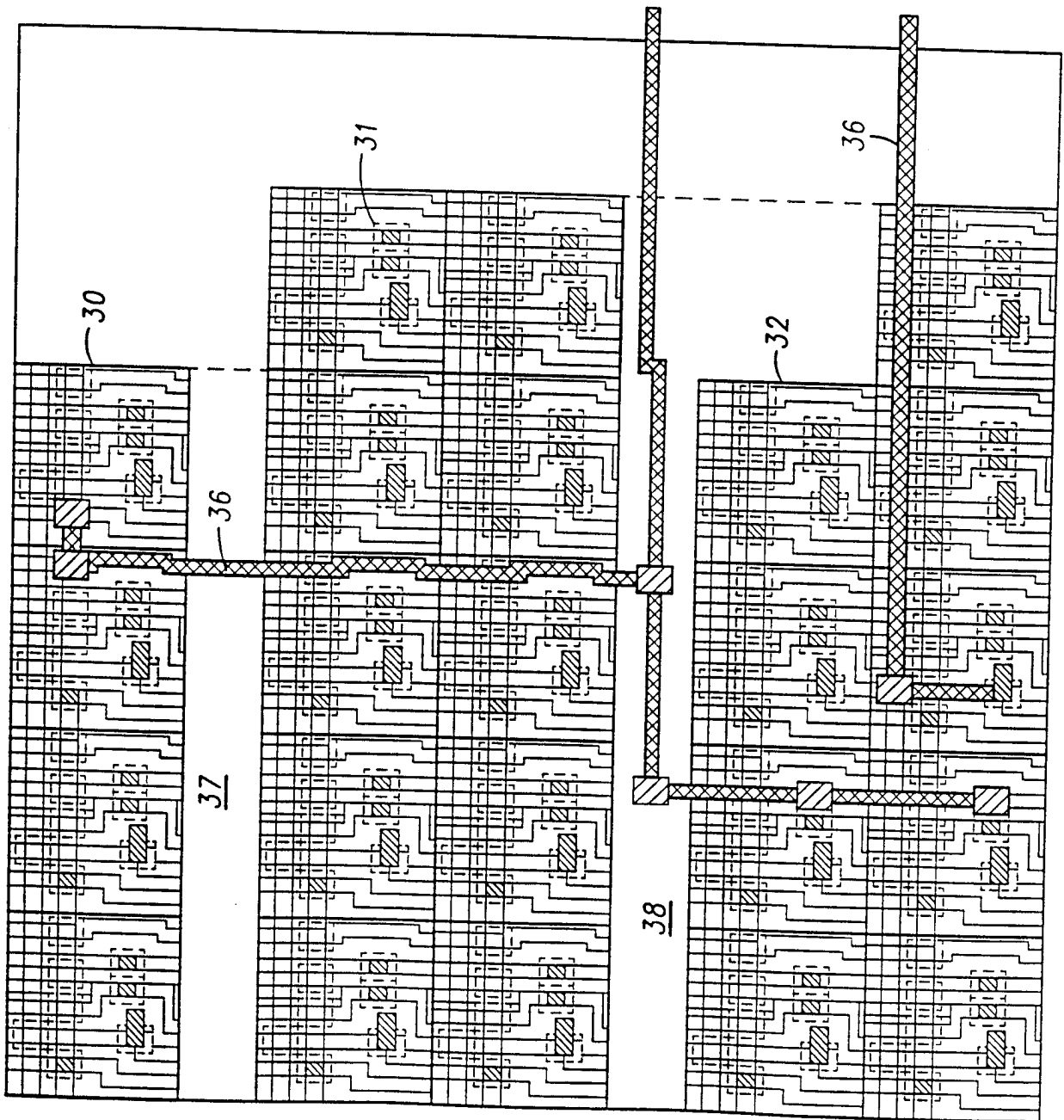


FIG. 3



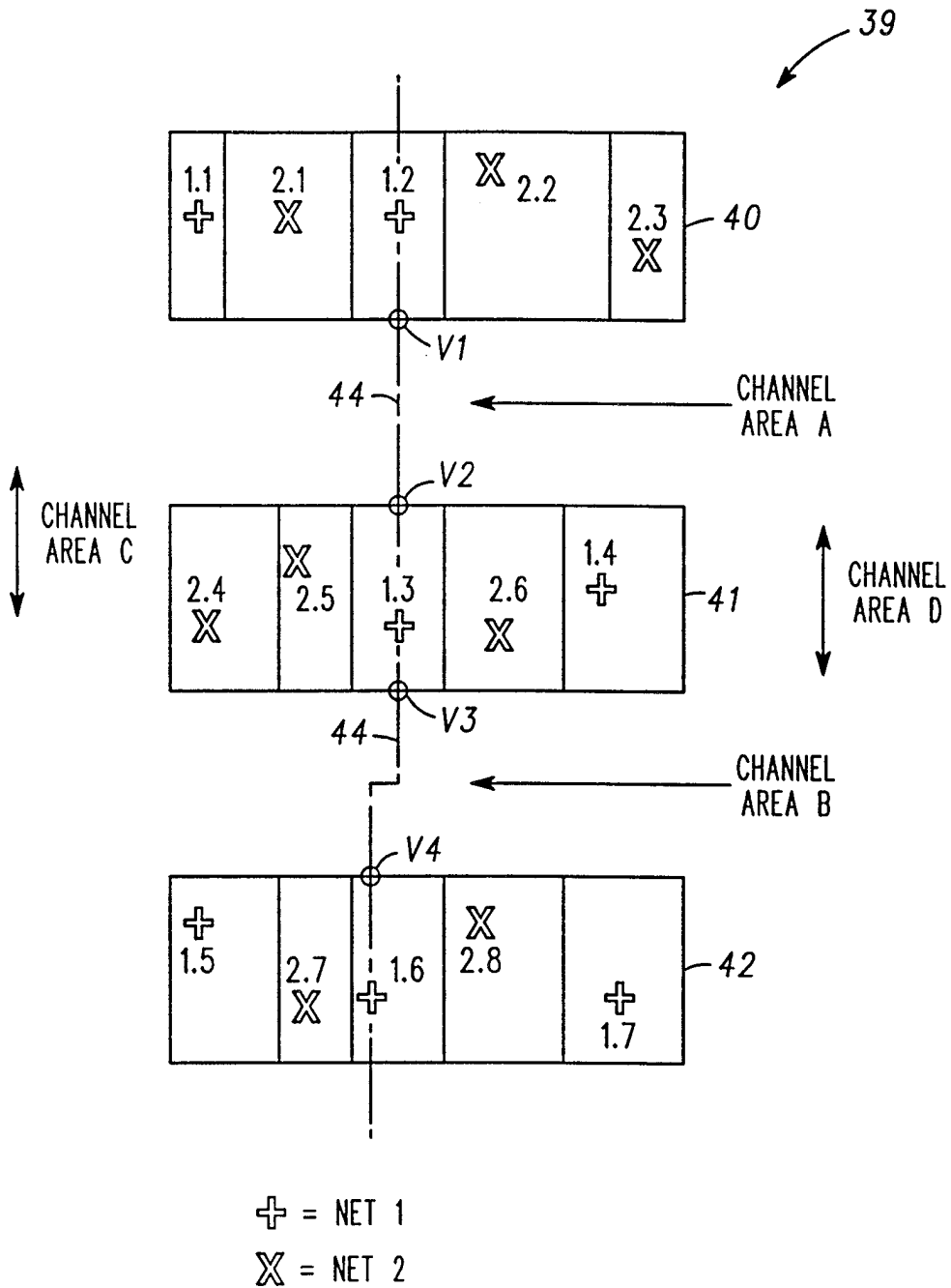


FIG. 4

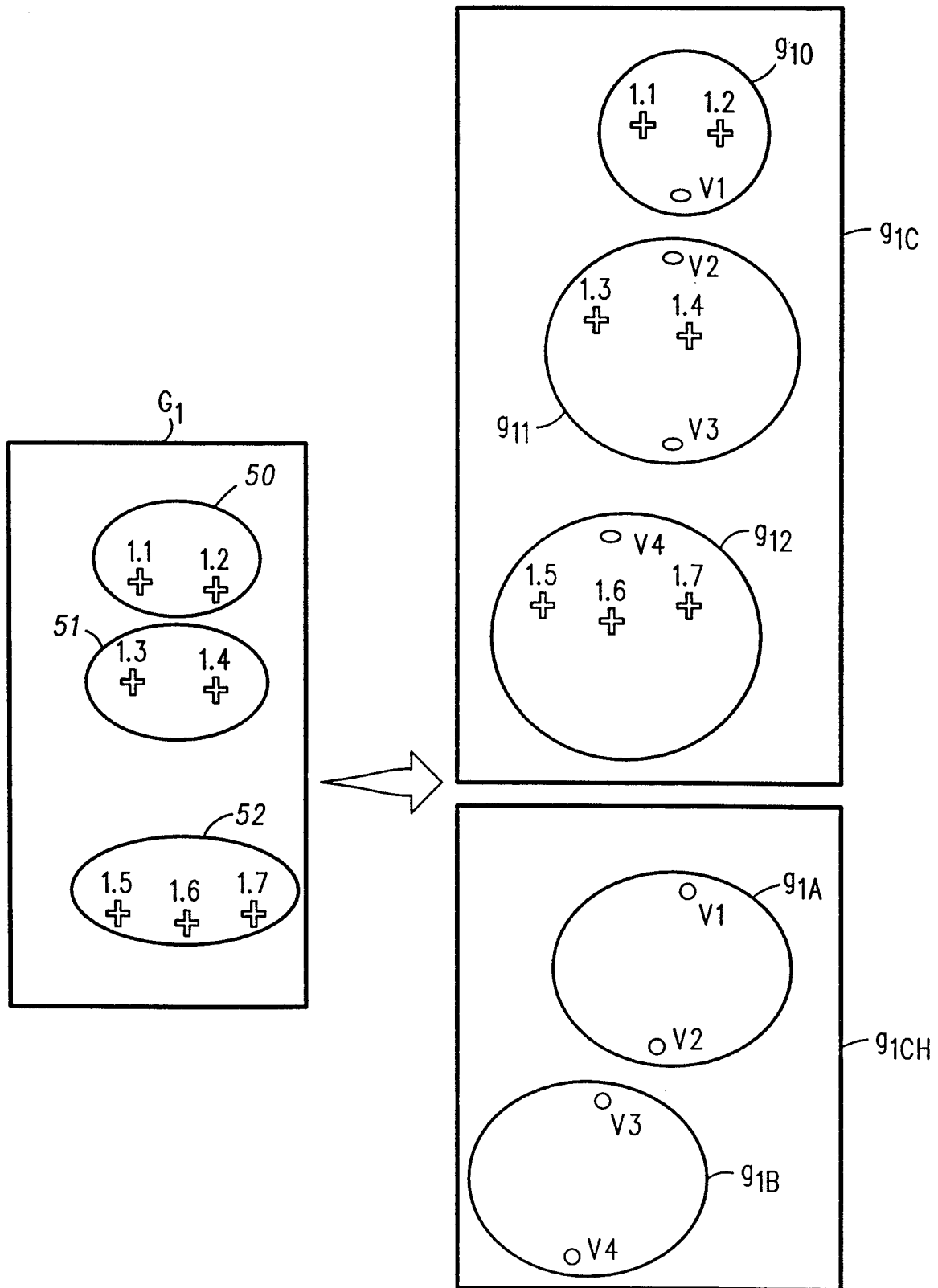
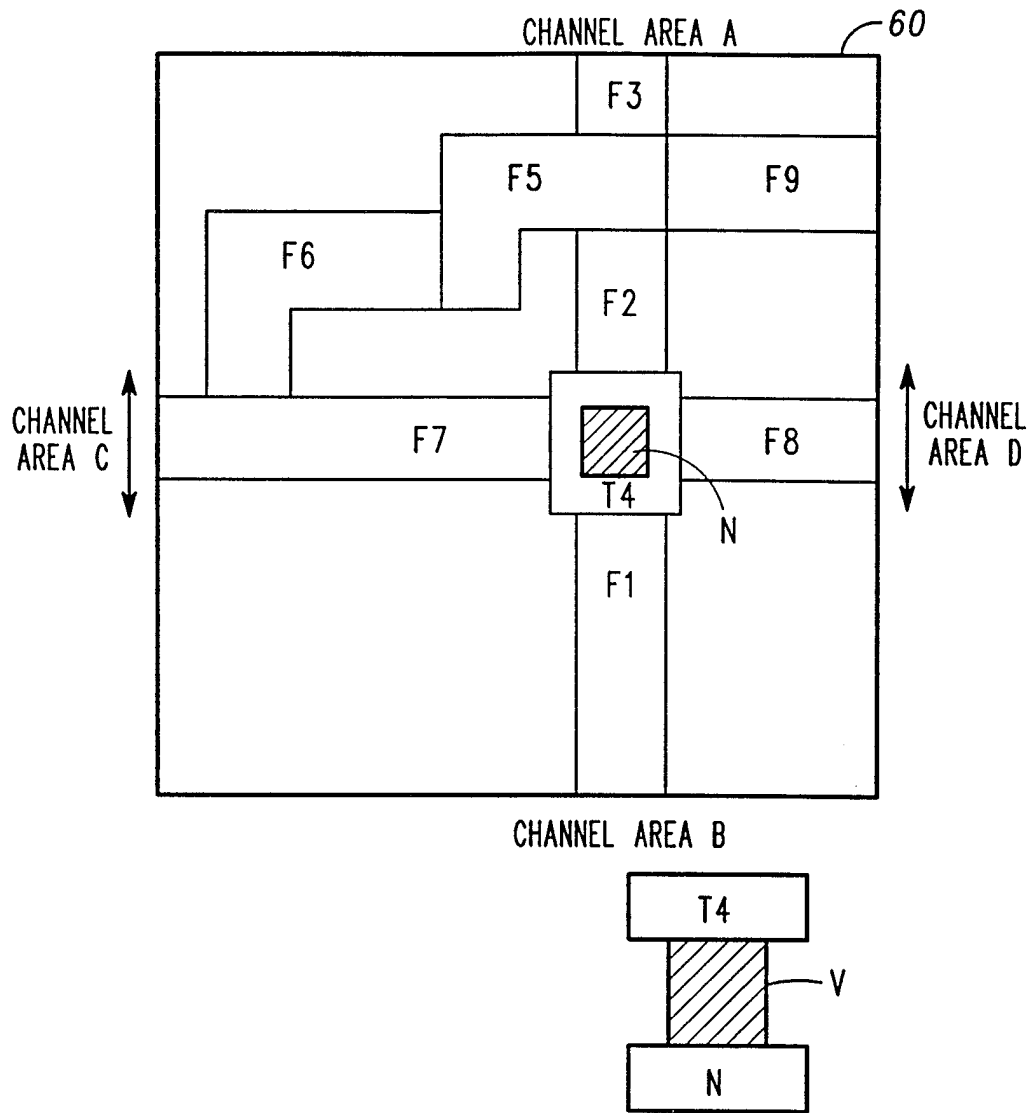
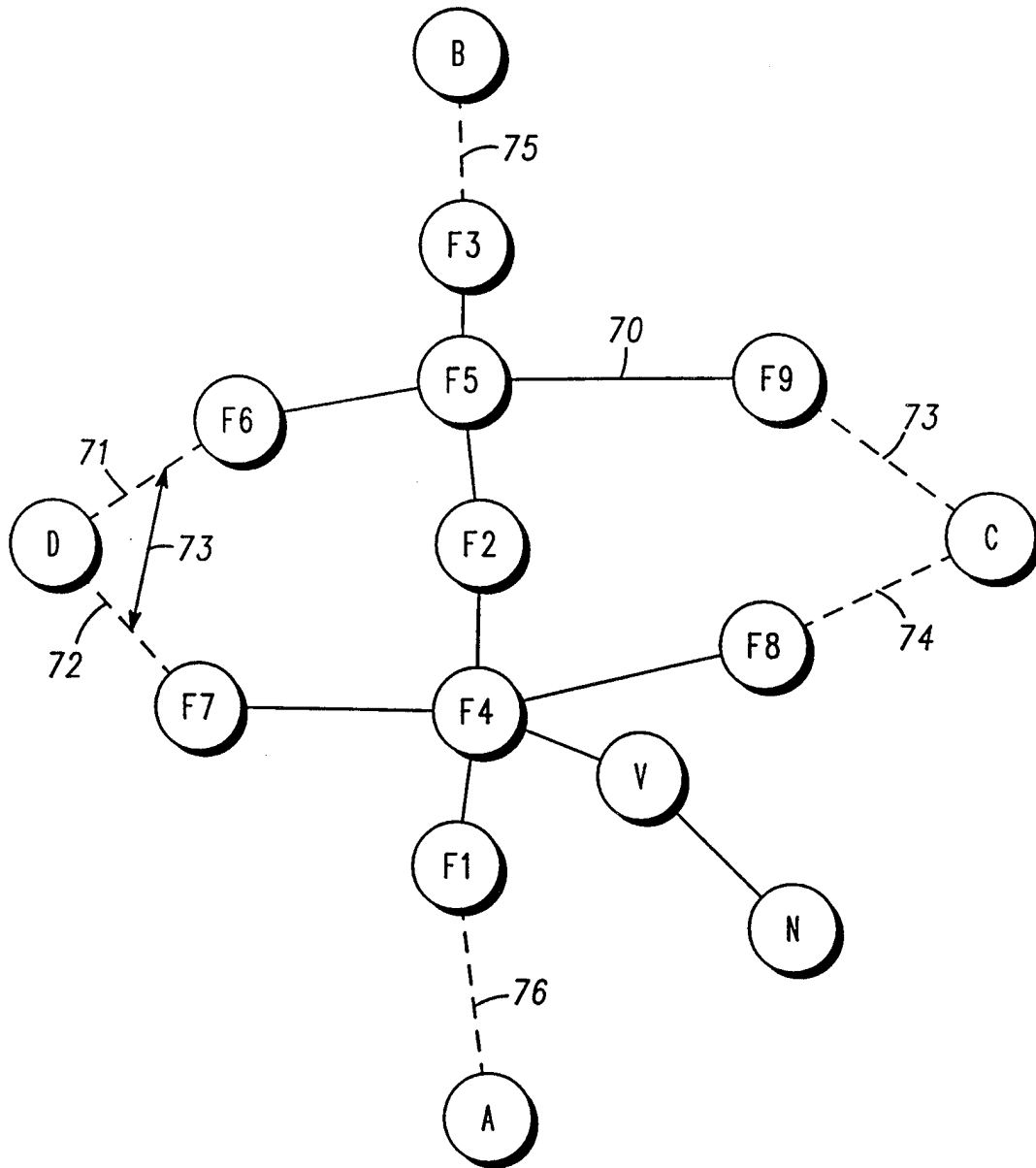


FIG. 5

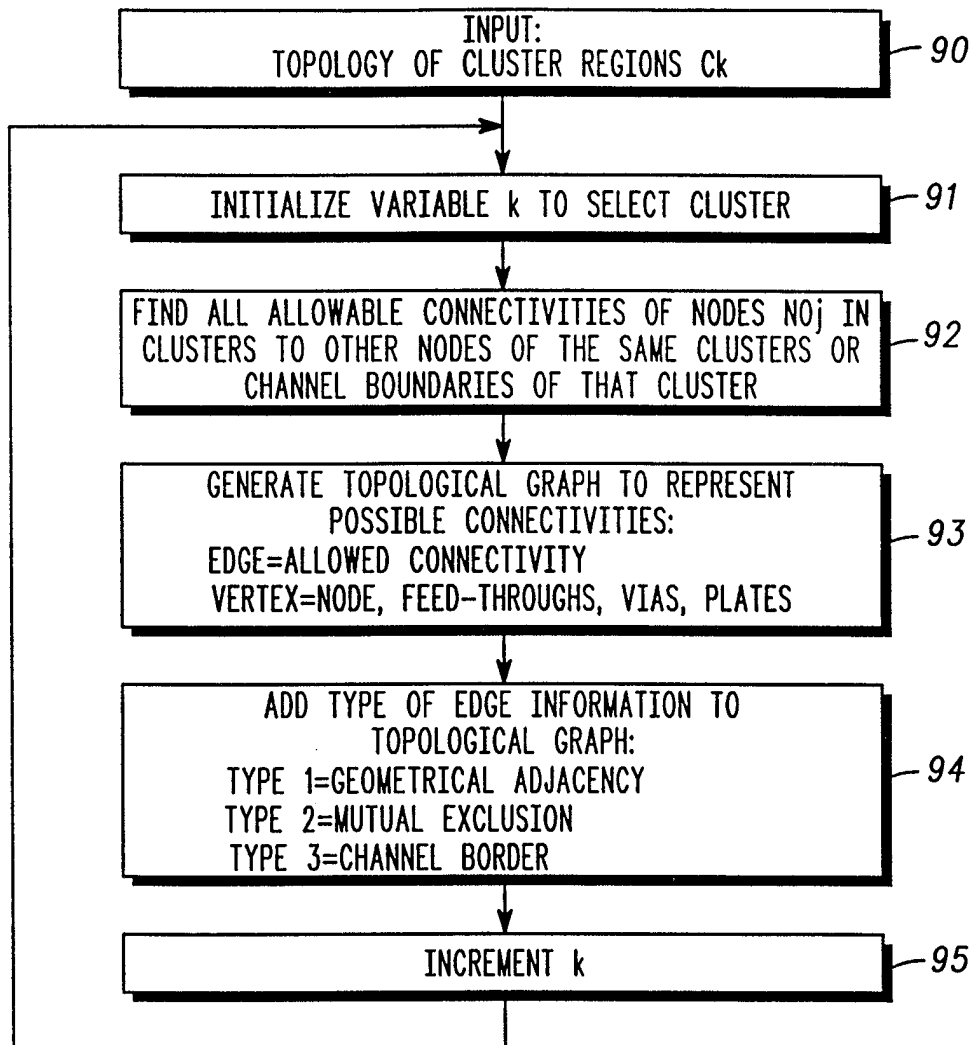


*FIG. 6*



*FIG. 7*

8/10

*FIG. 9*

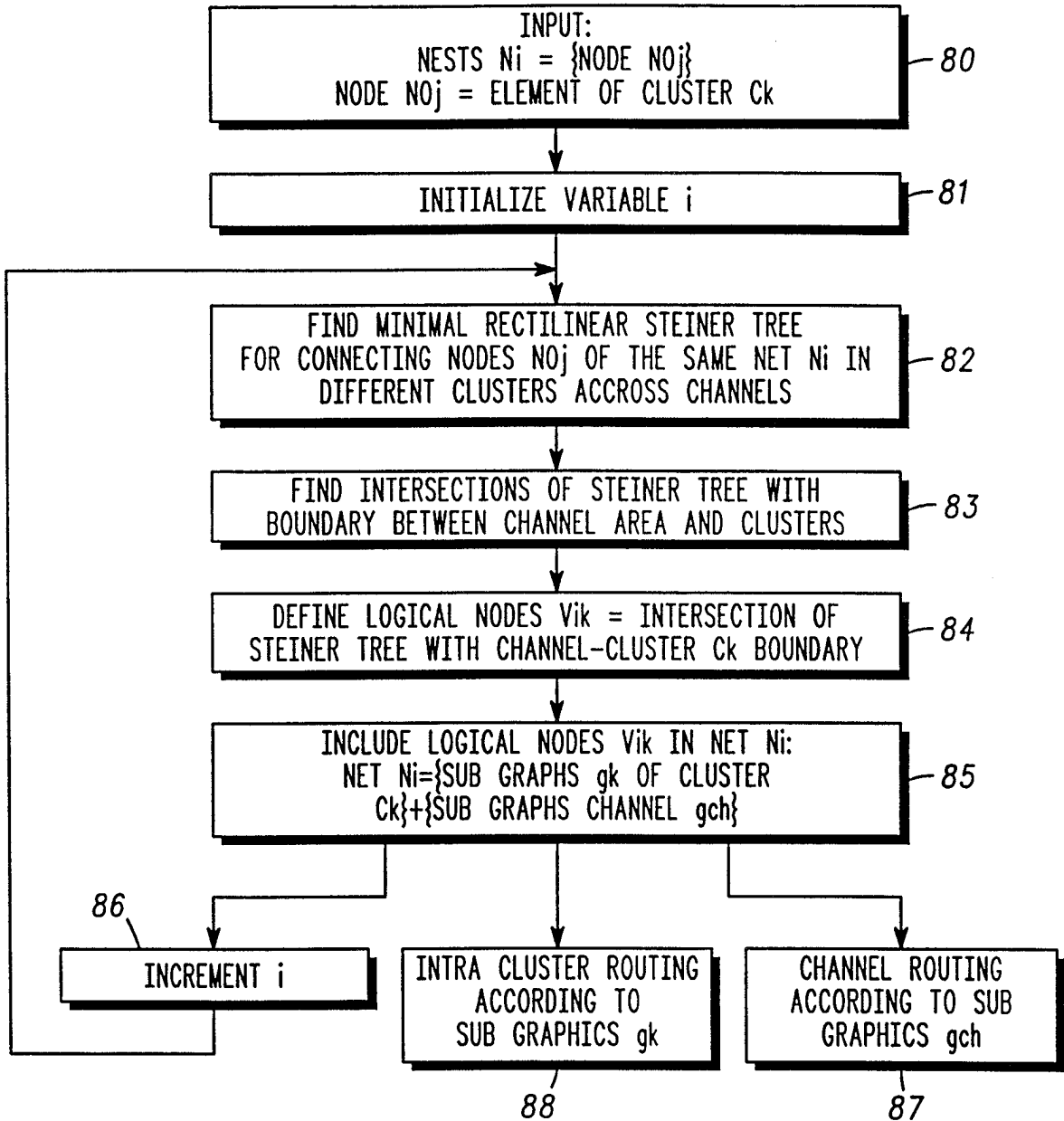


FIG. 8



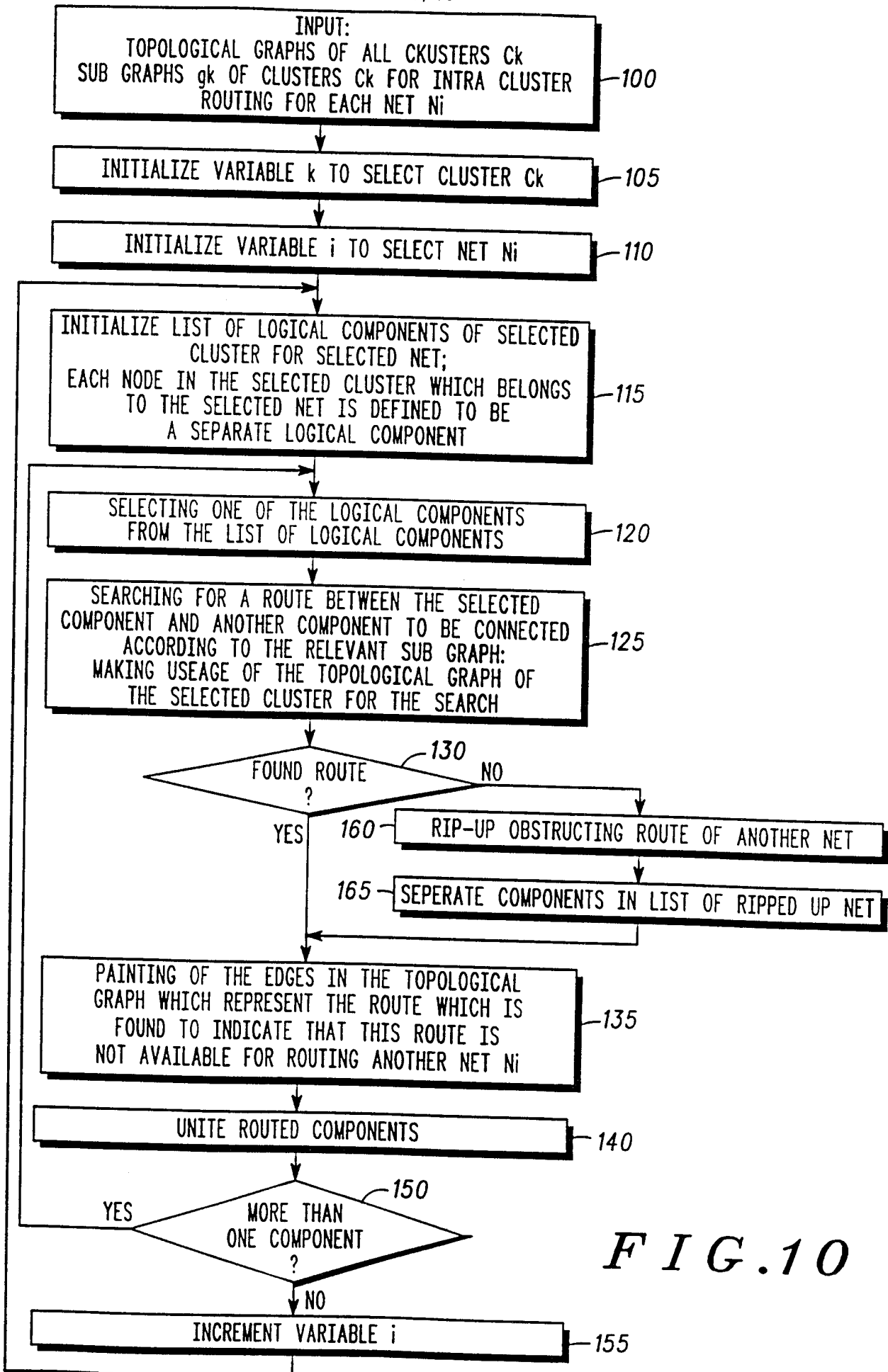


FIG. 10

# INTERNATIONAL SEARCH REPORT

International Application No

PCT/EP 97/05633

**A. CLASSIFICATION OF SUBJECT MATTER**  
IPC 6 G06F17/50

According to International Patent Classification(IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)  
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CHIANG C ET AL: "A WEIGHTED STEINER TREE-BASED GLOBAL ROUTER WITH SIMULTANEOUS LENGTH AND DENSITY MINIMIZATION" IEEE TRANSACTIONS ON COMPUTER AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, vol. 13, no. 12, 1 December 1994, pages 1461-1469, XP000491294 see the whole document ---	1,7
A	US 5 491 641 A (SCEPANOVIC RANKO ET AL) 13 February 1996 see column 1, line 26 - line 50 see column 2, line 1 - line 39; figure 2 ---	1,7
A	WO 91 06061 A (VLSI TECHNOLOGY INC) 2 May 1991 see abstract -----	1,7

Further documents are listed in the continuation of box C.

Patent family members are listed in annex.

° Special categories of cited documents :

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- "&" document member of the same patent family

Date of the actual completion of the international search

12 March 1998

Date of mailing of the international search report

23/03/1998

Name and mailing address of the ISA  
European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax: (+31-70) 340-3016

Authorized officer  
  
Guingale, A

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/EP 97/05633

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5491641 A	13-02-96	US 5615128 A	25-03-97
WO 9106061 A	02-05-91	US 5072402 A	10-12-91
		EP 0495887 A	29-07-92
		JP 5502549 T	28-04-93