THREE-LAYER CHANNEL ROUTING

by

Y. K. Chen and M. L. Liu

Memorandum No. UCB/ERL M81/84

6 November 1981

THREE-LAYER CHANNEL ROUTING

by

Y. K. Chen and M. L. Liu

Memorandum No. UCB/ERL M81/84

6 November 1981

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

THREE-LAYER CHANNEL ROUTING*

Y. K. Chen** and M. L. Liu***

ABSTRACT

With the advent of VLSI technology, mutiple-layer routing becomes feasible. Two special types of thrae-layer channel routing are introduced in this paper. One is called HVH, the other VHV. The merging algorithm and the left edge algorithm can be extended to three-layers.

Attempts are made to compare the lower bounds of channel width among the three types of routing——two-layer, VHV, and HVH.

The algorithms were coded in PASCAL and implemented on VAX 11/780 computer. The computational results are satisfactory. Since all the results lead to a further reduction in routing area.

---

*Research supported by the National Science Foundation Grant ENG-78-24425.
**On leave from Tsinghua University, Beijing, China.
***On leave from Tianjin University, Beijing, China.

THREE-LAYER CHANNEL ROUTING

Y. K. Chen and M. L. Liu

1. Introduction

In the layout design of LSI chips, channel routing is one of the key problems. Terminals are placed along the two sides of a channel. Each terminal has a specified number. Terminals with the same number must be connected by a net. A sample problem is expressed by a net list as shown in Fig.1. Arrows indicate whether nets are to be connected to terminals in the upper or lower side of the channel.

Nowadays two-layer channel routing is used frequently, where two layers are available for routing [1],[2]. We assume horizontal segments of nets on one layer and vertical segments of nets on the other. Fig.2 shows two examples which are the two-layer channel routing realization of the problems given in Fig.1.

With the advent of VLSI technology, multiple-layer routing becomes feasible. In this paper two special types of three-layer channel routing are considered. One is called VHV, the other HVH. It will be shown in the following sections that two-layer channel routing algorithms [1],[2] can be extended to three-layers, making possible a further reduction in routing area. Attempts are made to compare the lower bounds of channel width among the three types of routing ——two-layer, VHV, and HVH.

2. Review of some basic concepts in channel routing

This paper is an extension of reference [2]. Most of the definitions and terminologies used here are the same as defined in reference [2]. Before the description of three-layer problem, let us review some of the basic concepts in channel routing . Among them, the vertical constraint graph and the zone representation are crucial in completely describing the channel routing problem.

(1) The vertical constraint graph (V.C.G), and the maximum level of V.C.G (vmax)

Because the vertical segment of one net can not overlap that of another on the same track, a constraint has been introduced on horizontal segments of the nets.

For example, in Fig.1(a), if we pay attention to the third column from the left, the horizontal segment of net 4 must be placed above that of net 5. This relation can be represented by a directed graph, so called vertical constraint graph (V.C.G) where each node corresponds to a net and a directed edge from node A to node B means that net A must be placed above net B. Fig.3 shows the V.C.G of the examples in Fig.1.

If we define the path length of a directed path on V.C.G as the number of nodes on the path, then a new term "vmax" is defined as the longest path from a source node (nodes without ancestor) to a sink node (nodes without descendent). As an example in Fig.3, the vmax in Fig.3(a) and Fig.3(b) are 3 and 4 respectively.

(2) The zone representation and the maximum density (dmax)

The horizontal relation among the nets can be expressed by the zone representation. Let S(i) be the set of nets in which the horizontal segments intersect column i. Then the channel can be divided into zones by considering the overlapping property of nets. obviously, any two nets in S(i) must not be placed on the same horizontal track. Roughly, a zone is defined in terms of the columns of a maximal set S(i). for the example of Fig.1(a) the Table below defines the four zones:

Table (1)

Column	S(i)	Zone
1	2	
2	1 2 3	
3	1 2 3 4 5	1
4	1 2 3 4 5	
5	1 2 4 5	
6	2 4 6 7	2
7	2 4 6 7	
8	2 4 7	
9	4 7 8	
10	4 7 8 9	3
11	7 8 9	
12	7 9 10	4
13	9 10	

Fig.4 gives the zone representation.

The number of elements in S(i) is called local density, and the maximum among them is called maximum density (dmax).

3. Description of three-layer channel routing

Three layers are available for routing Two special types of routing. are specified.

(1) VHV type

We assume all upward vertical segments on the first layer, all downward vertical segments on the third layer, and all horizontal segments on the second layer. Segments on each layers are isolated from those segments on the other layers. Connections between segments on the first layer and second layer, as well as segments on the third layer and second layer are made through via holes.

Fig. 5 shows an example which is the VHV routing realization of the problem given in Fig. 1(b). In Fig. 5 the solid vertical lines are assumed on the first layer, the dotted vertical lines on the third layer, and the horizontal lines on the second layer.

Because the upward vertical segments and the downward vertical segments are not in the same layer, therefore, in VHV, the vertical constraints between nets no longer exist.

(2) HVH type

All horizontal segments of nets are divided into two groups. We assume one group of horizontal segments on the first layer, the other group on the third layer, while all the vertical segments on the second layer.

Fig. 6 shows an example which is the HVH routing realization of the problem given in Fig. 1(a). In Fig. 6 the solid horizontal lines are assumed on the first layer, the dotted horizontal lines on the third layer, and the vertical lines on the second layer.

Because the horizontal segments are placed on two layers the maximum density at each layer is reduced. In an ideal case the maximum density at each layer can be reduced to $\lceil dmax/2 \rceil$ where $\lceil a \rceil$ designates the smallest integer$\geq$a.

4. The lower bounds of the realizable channel width of the three types—— two-layer, VHV, and HVH

The lower bound of the realizable channel width (i.e., the number of tracks) can be expressed in terms of the two criteria "dmax" and "vmax". Now let us discuss the lower bound of number of tracks in the following different types, and use the problems given in Fig.(1) as common examples. In Fig.1(a), we have dmax=5, vmax=3, and in Fig.1(b), we have dmax=2, vmax=4.

(1) Two-layer routing

Because in V.C.G, we can always find the longest path n1-n2-n3....-nk (k=vmax), then no two nets among n1,n2...nk can be placed on the same track. Now, the longest path in terms of the number of nodes on the path is vmax, therefore, at least vmax horizontal tracks are necessary to realize the routing. In addition, the number of tracks must also be larger than or equal to the maximum density, dmax. Therefore, in any two-layer routing the realizable channel width has a lower bound equal to max(dmax,vmax).

The lower bounds in Fig.1(a) and Fig.1(b) are 5 (max(5,3)=5), and 4 (max(2,4)=4) respectively. Fig.2(a) and Fig.2(b) are the two-layer routing realizations of Fig.1(a) and Fig.1(b). It is easy to see that they are both optimum realizations.

(2) Three-layer(VHV) routing

Vertical constraints of nets no longer exist in VHV. Therefore the channel width which is equal to the maximum density, dmax, can always be realized.

Fig.5 is the VHV routing realization of Fig.1(b). The number of tracks is 2 which is less than that in the two-layer case as shown in Fig.2(b). If we use three-layer(VHV) routing for the problem in Fig.1(a), the number of tracks is still 5, not less than that in the two-layer case. Therefore it is not worthwhile to use VHV when we have dmax>vmax.

(3) Three-layer(HVH) routing

Suppose the maximum density at each layer is equal to $\lceil dmax/2 \rceil$. Then the realizable channel width has a lower bound equal to max($\lceil dmax/2 \rceil$,vmax).

The lower bounds in Fig.1(a) and Fig.1(b) are 3 (max($\lceil 5/2 \rceil$,3)=3),and 4 (max($\lceil 2/2 \rceil$,4)=4) respectively. Fig.6 is the three-layer(HVH) routing realization of Fig.1(a). The number of tracks is 3. Therefore it is an optimum realization, and the number of tracks is less than that in the two-layer case as shown in Fig.2(a). Obviously, it is not worthwhile to use HVH when we have vmax>dmax.

Table(2) shows the comparison of the lower bounds among the above three types.

Table(2)

type	the lower bound	suitable case
two-layer	max(dmax, vmax)	in general case, with dmax>vmax
VHV	dmax	in the case with dmax<vmax
HVH	max(⌈dmax/2⌉, vmax)	in the case with dmax>2(vmax)

5. The algorithms for three-layer channel routing

Because the vertical constraint no longer exists in VHV the left edge algorithm [1] is sufficient for the VHV problem. Now we will only discuss the algorithm for the HVH problem.

In HVH the vertical constraint still exists. So there should be no directed path in V.C.G between nets which are placed at both first and third layers on the same track. The merging algorithm used in the two-layer routing can also be used for the HVH problem. The difference is that now we consider not only merging nets at the same layer between different zones (as we did in two-layer routing) but also merging nets in the same zone between different layers.

5.1 Merging of nets

Now we define two kinds of merging as follows:

(1) Serial merging(i, j)

Let net i and net j be nets such that there is no horizontal overlap in the zone representation, and no directed path between net i and net j in the V.C.G. Then place net i and net j on the same horizontal track at the same layer(either at the first layer or at the third layer).

(2) Parallel merging(i, j)

Let net i and net j be the nets such that there is horizontal overlap in the zone representation, but no directed path between net i and net j in the V.C.G. Then place net i and net j on the same horizontal track at the different layers, one at the first layer and the other on the third layer.

Fig.7 illustrates how nets are merged in HVH. Fig.7(a) shows the V.C.G and the zone representation of the given problem in Fig.1(a). We assume that, before merging no net is placed on any track, and in zone representation, thin lines represent those nets which are to be placed on tracks, and thick lines represent nets placed on tracks at the first layer and dotted lines at the third layer. First, we have parallel merging(1,4),(2,3) in zone 1 (Fig.7(b)). Second, we have serial merging(6,3),(7,5) between zone 2 and zone 1 (Fig.7(c)). Third, we have parallel merging(9,7) in zone 3 and serial merging(8,2) between zone 3 and zone 2 (Fig.7(d)). Finally, we have serial merging(10,1) between zone 4 and zone 3 (Fig.7(e)). The HVH routing realization corresponding to the zone representation in Fig.7(e) is shown in Fig.6.

5.2 The merging algorithm

As in the two-layer routing the merging procedure is the key part of the whole algorithm where two sets of nets are merged. Let us first briefly review the merging criterion as shown in reference [2]:

(1) P={n1,n2....np} and Q={m1,m2....mq} are the two sets of nets to be merged.

(2) First, among Q, find m* which maximizes f(m). Next, among P, find n* which minimizes g(n,m*), and which is neither ancestor nor descendent of m*. Merge m* and n*.

This criterion can also be used in HVH. In two-layer routing, nets are definitely specified to be in set P and set Q. If we merge nets between zone i and zone i+1, we let P be the set of nets which terminate at zone i, and let Q be the set of nets which begin at zone i+1. But in HVH we have two kinds of merging. Now, the problem is how to specify, at each step, the correspondent nets into P and Q in order to get a sufficient selection and rejection between nets. In the following this problem is explained. To make the situation precise, let us introduce several definitions. Suppose we merge nets between zone i and zone i+1, as shown in Fig.8.

(1) Let B be the set of nets which begin at zone i+1.

(2) nb is the number of nets in B.

(3) Let T be the set of nets which include:

(a) nets which terminate at zone i.

(b) nets which are placed on a track with no horizontal net on the other layer.

(4) nt is the number of nets in T.

As an example in Fig.8, we have B={6,7,8,9}, nb=4, T={3,2,5}, nt=3.

Now, in HVH we specify set P and set Q as follows:

(1) Let Q=B , nq=nb. After we find m* among Q, update Q by deleting m*, i.e., updated Q=Q-m*.

(2) If nq<=nt let P=T;

If nq>nt let P=T+Q (Q=Q-m*).

Before merging nets, all the nets in T have been placed on certain tracks, but nets in Q have not been placed yet. So if net m* which belongs to Q merges with any net in T, no new track appears. Otherwise, if net m* merges with another net in Q then a new track appears. Therefore if nq<=nt it is possible for the old tracks (where nets in T are placed) to contain all the nets in Q. In such case, in order not to increase the new track the parallel merging between nets in Q is avoided. Conversely, if nq>nt the old tracks are not enough to contain all the nets in Q, at least one new track appears. So parallel merging between nets in Q is allowed only in this case.

Merging algorithm(1)—— for merging nets between any two successive zones

```
a1   Q=B, nq=nb;

a2   while Q is not empty do
       begin

a3     among Q find m* which maximizes f(m),
       updated Q=Q-m*;

a4     if nq<=nt then P=T;
       if nq>nt  then P=T+Q;

a5     among P find n* which minimizes g(n,m*) and
       which is neither ancestor nor descendent of m*;
```

```
a6        if n* belongs to T
           then serial(or parallel) merging(m*,n*)
                updated T=T-n*,
                        nt=nt-1,
                        nq=nq-1;
          if n* belongs to Q
           then parallel merging(m*,n*)
                updated Q=Q-n*,
                        nq=nq-2;
          if n* can not be found
           then place m* on a new track,
                updated nq=nq-1,
                        nt=nt+1;

a7      end;
```

As a special case, when we start merging at a starting zone, we should first make a parallel merging between all the nets which pass through the starting zone (as we did in Fig.7(b)). Suppose zone j is the starting zone and let Pa be the set of nets which pass through zone j. Then we have merging algorithm(2).

Merging algorithm(2)——only for merging nets in the starting zone

```
a1    Q=Pa;

a2    while Q is not empty do
        begin

a3      among Q find m* which maximizes f(m);
        updated Q=Q-m*,
        P=Q;

a5      among P find n* which minimizes g(n,m*) and
        which is neither ancestor nor descendent of m*;

a6      if n* can be found
         then parallel merging(m*,n*),
              updated Q=Q-n*;
        if n* can not be found
         then place m* on a new track;

a7      end;
```

6. More constraint on HVH——the "power" and "ground" nets

In VLSI technology, the first and second layers are made of metal and the third layer is made of polysilicon. Among the nets to be routed there are "power" and "ground" nets. In order to get good conductibility the "power" and "ground" nets must be placed on the first and second layers. Because the vertical segment of each net is placed on the second layer only the horizontal segments have to be considered. The algorithm can be expressed as follows:

(1) Specify all the "power" and "ground" nets with a certain mark, and use the merging algorithm as usual. It is not allowed to have the marked nets appear on both the first and third layers on the same track.

(2) After all the nets have been placed on the tracks, check each track to see if there is any marked net on the third layer. If so, interchange the nets between the first and third layers on that track. Then it guarantees that all the "power" and "ground" nets appear on the first layer.

7. Conclusion

The merging algorithm for the HVH, and the left edge algorithm for the VHV were coded in PASCAL and implemented on the VAX 11/780 computer. We use the same examples as in reference [2]. The experimental results are quite encouraging.

In the HVH (no dogleg) the program generated optimum solutions 6 out of 7. Table(3) compares the numbers of tracks of the two-layer realization, the lower bound of the HVH, and the HVH realization.

Similar to the two-layer routing merging algorithm, doglegs can be introduced. Table(4) shows a computational result.

Table(3)

Fig.	Ex.	dmax	vmax	2-layer #track	3-layer(HVH) low. bd.	3-layer(HVH) #track	comment
9	1	12	7	12	7	7	optimum
10	3a	15	4	15	8	8	optimum
11	3b	17	9	17	9	10	
12	3c	18	6	18	9	9	optimum
13	4b	17	13	17	13	13	optimum
14	5	20	3	20	10	10	optimum
15	dif.	19	23	30	23	23	optimum

Table(4)

Fig.	Ex.	dmax	vmax	2-layer #track	3-layer(HVH) low.bd.	#track	comment
16	dif.	19	9	21	10	14	

Fig.17 gives an example of the HVH problem with regard to "power" and "ground" nets, here nets 7,8,12 are assumed to be the "power" and "ground" nets.

An example of the VHV problem is shown in Fig.18.

All the computational results lead to a further reduction in routing area.

References

[1] D. N. Deutsch, C. Persky and D. G. Schwiekert. "LTX-A System for the Directed Automatic Design of LSI Circuits" Proc. 13th Design Auto. Conf. 1976, pp.399-407.

[2] T. Yoshimura and E. S. Kuh. "Efficient Algorithms for Channel Routing" UCB/ERL M80/43, Aug. 1980.

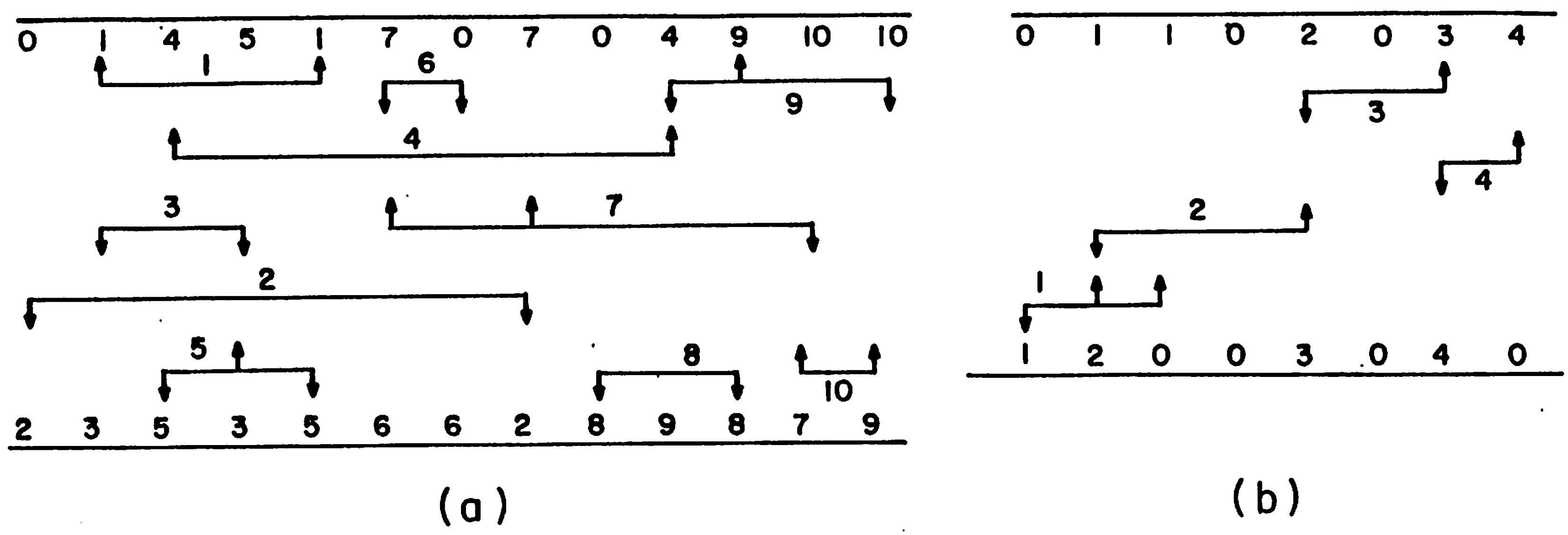


FIG. 1 NETLIST REPRESENTATIONS FOR ROUTING REQUIREMENT

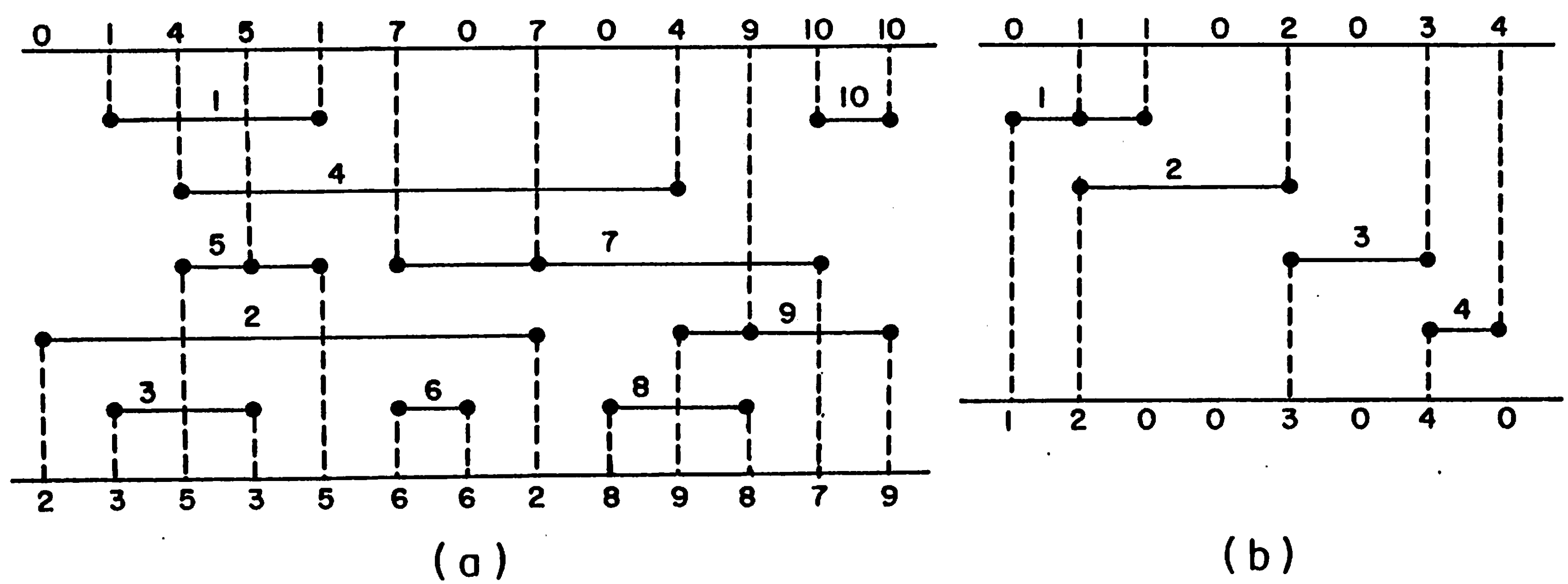


FIG. 2 TWO-LAYER ROUTING REALIZATIONS

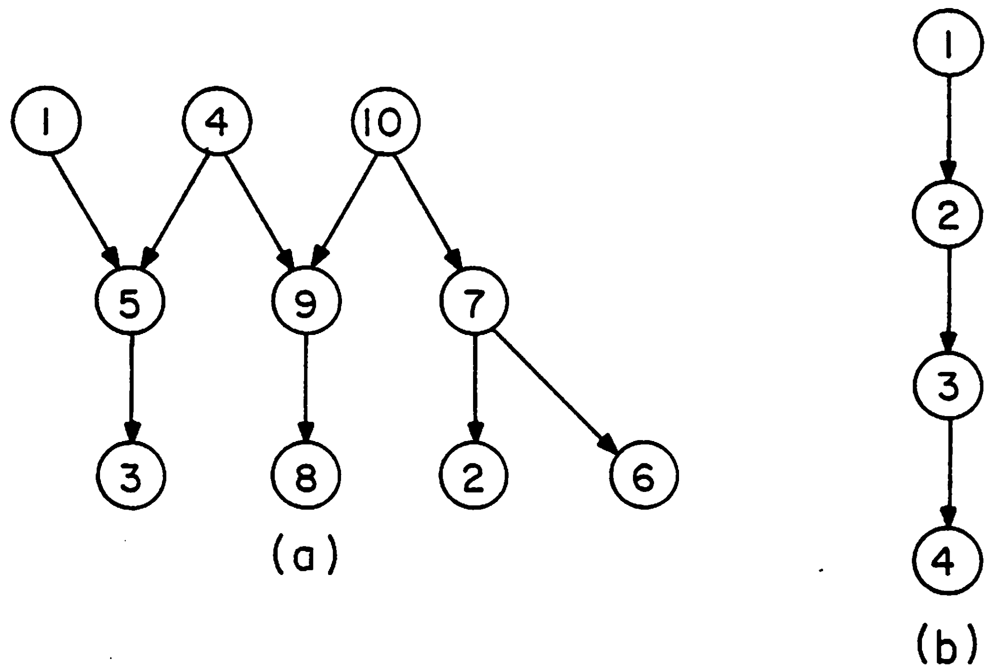


FIG. 3 VERTICAL CONTRAINT GRAPHS FOR THE NETLISTS IN FIG. 1

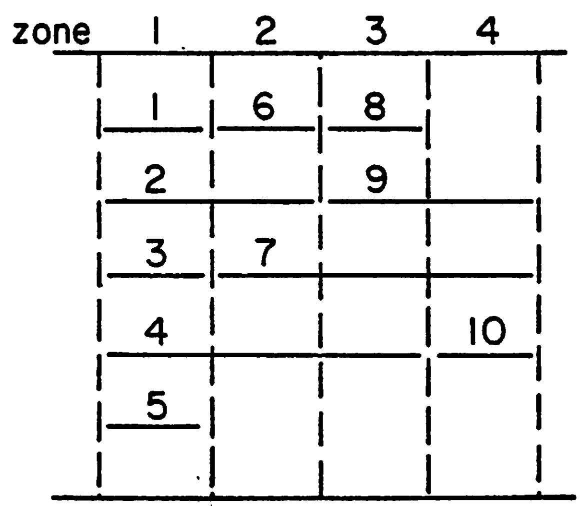


FIG. 4 ZONE REPRESENTATION FOR THE NETLIST IN FIG. 1(a)

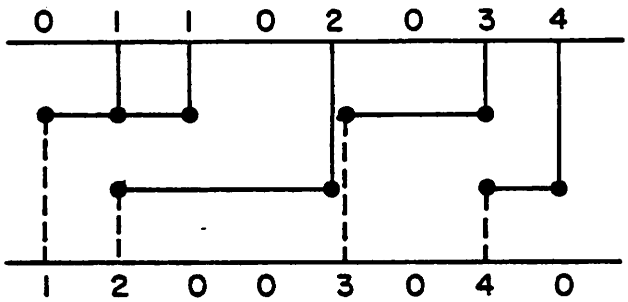


FIG. 5

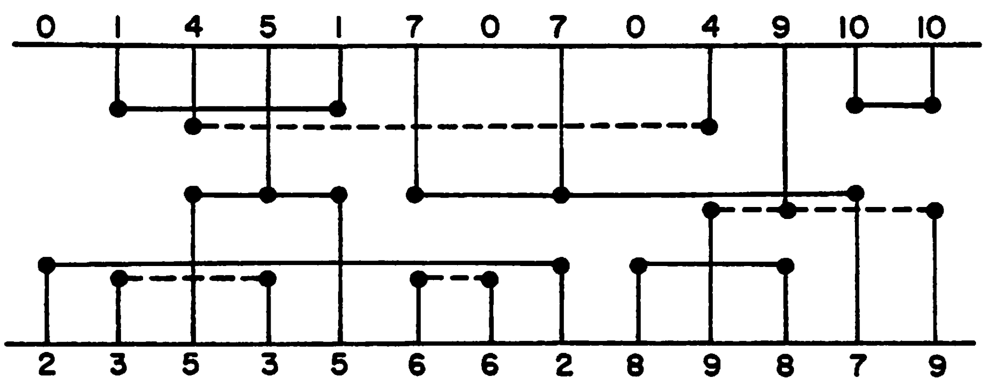
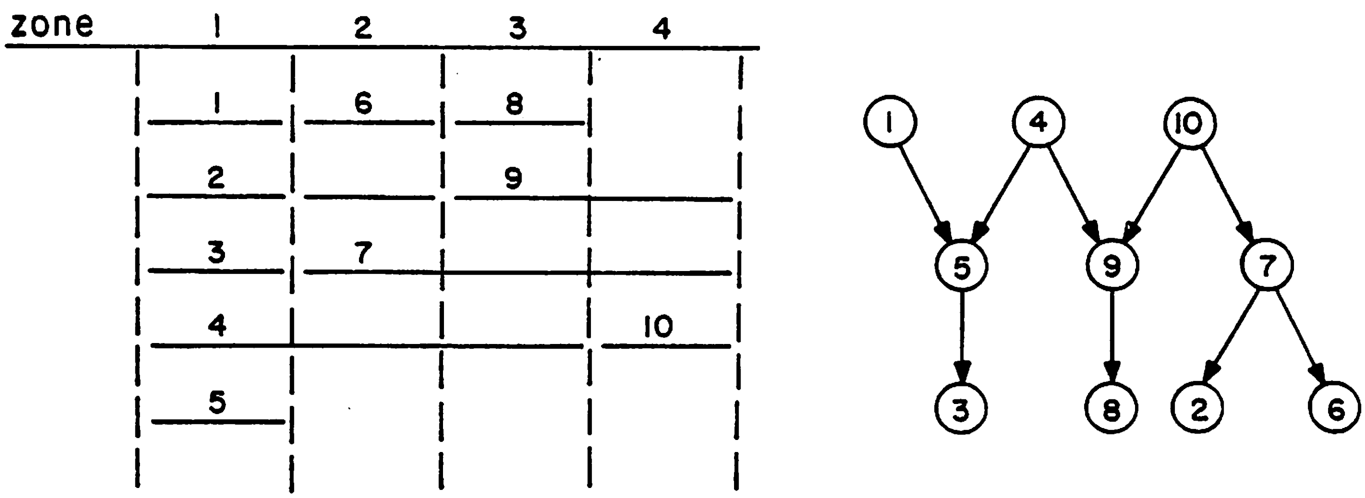
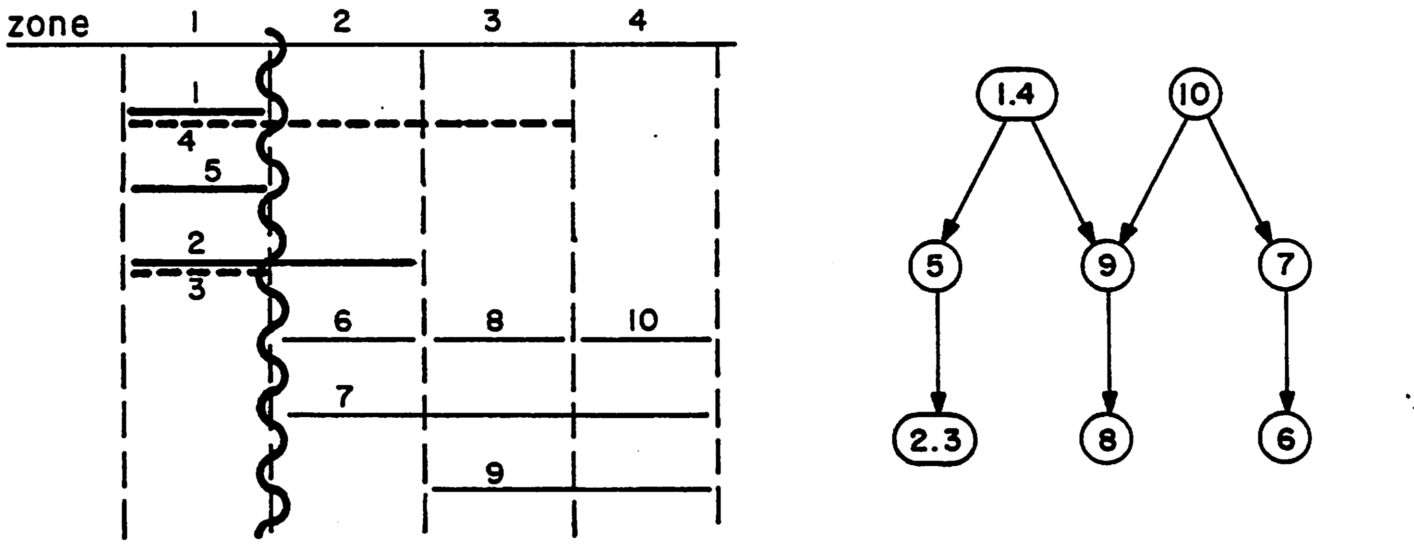


FIG. 6



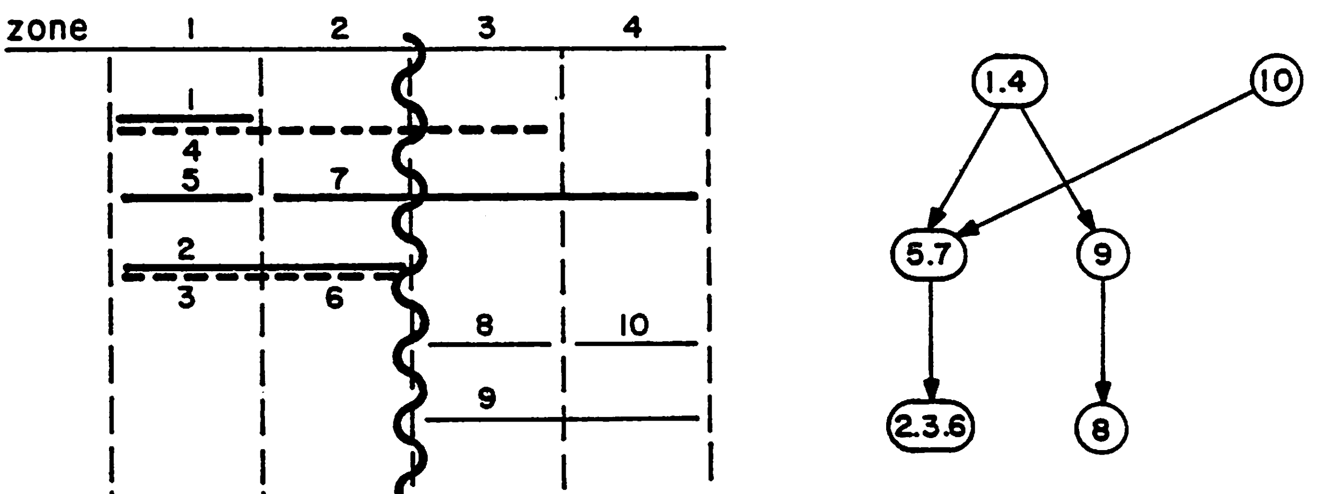
(a)

FIG. 7 THE ZONE REPRESENTATION AND THE V.C.G. OF THE GIVEN PROBLEM IN FIG. 1(a)



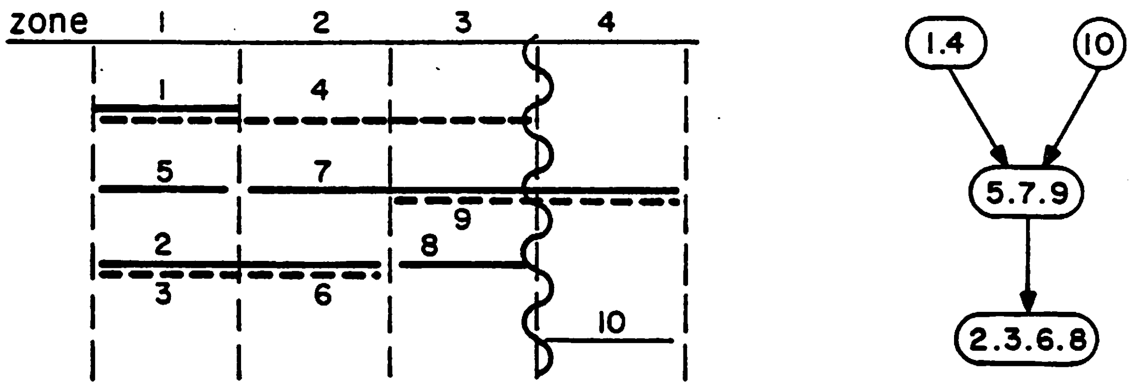
(b)

FIG. 7 AFTER MERGING NETS IN ZONE (1)



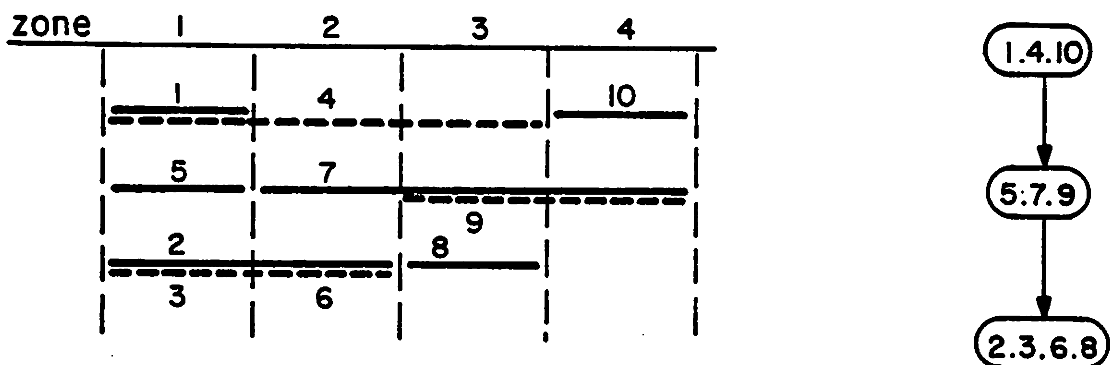
(c)

FIG. 7 AFTER MERGING NETS BETWEEN ZONE (2) & ZONE (1)



(d)

FIG. 7 AFTER MERGING NETS BETWEEN ZONE (3) & ZONE (2)



(e)

FIG. 7 THE FINAL ZONE REPRESENTATION & V.C.G

FIG. 7 HOW NETS ARE MERGED IN THE HVH

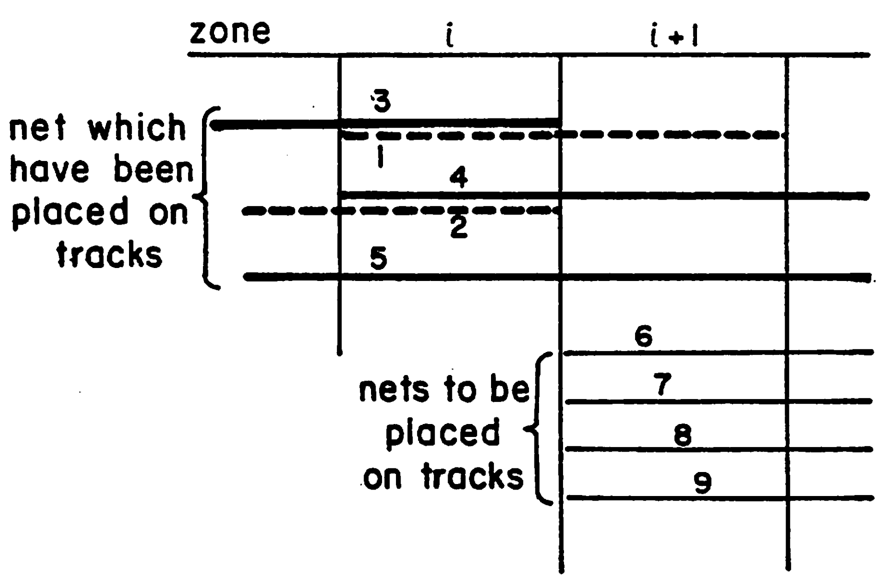
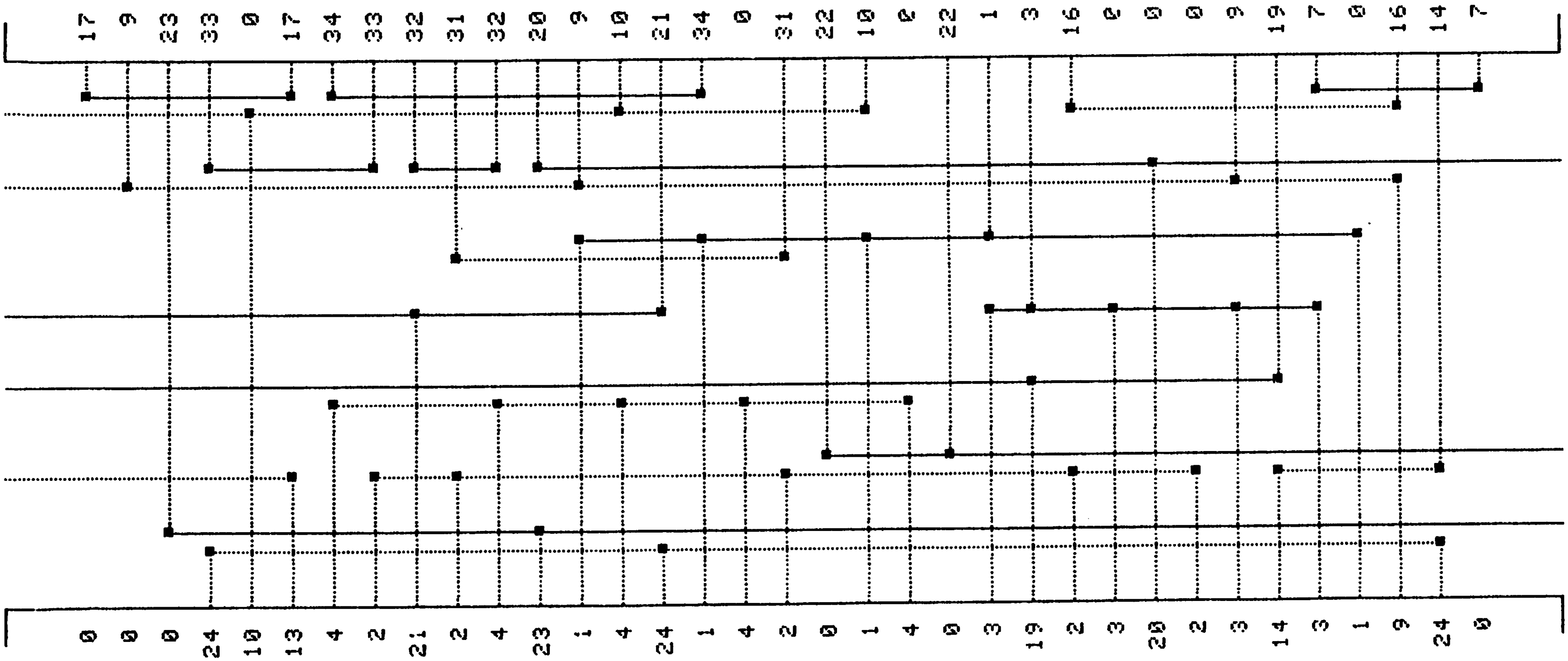


FIG. 8

****** example. 1 ******

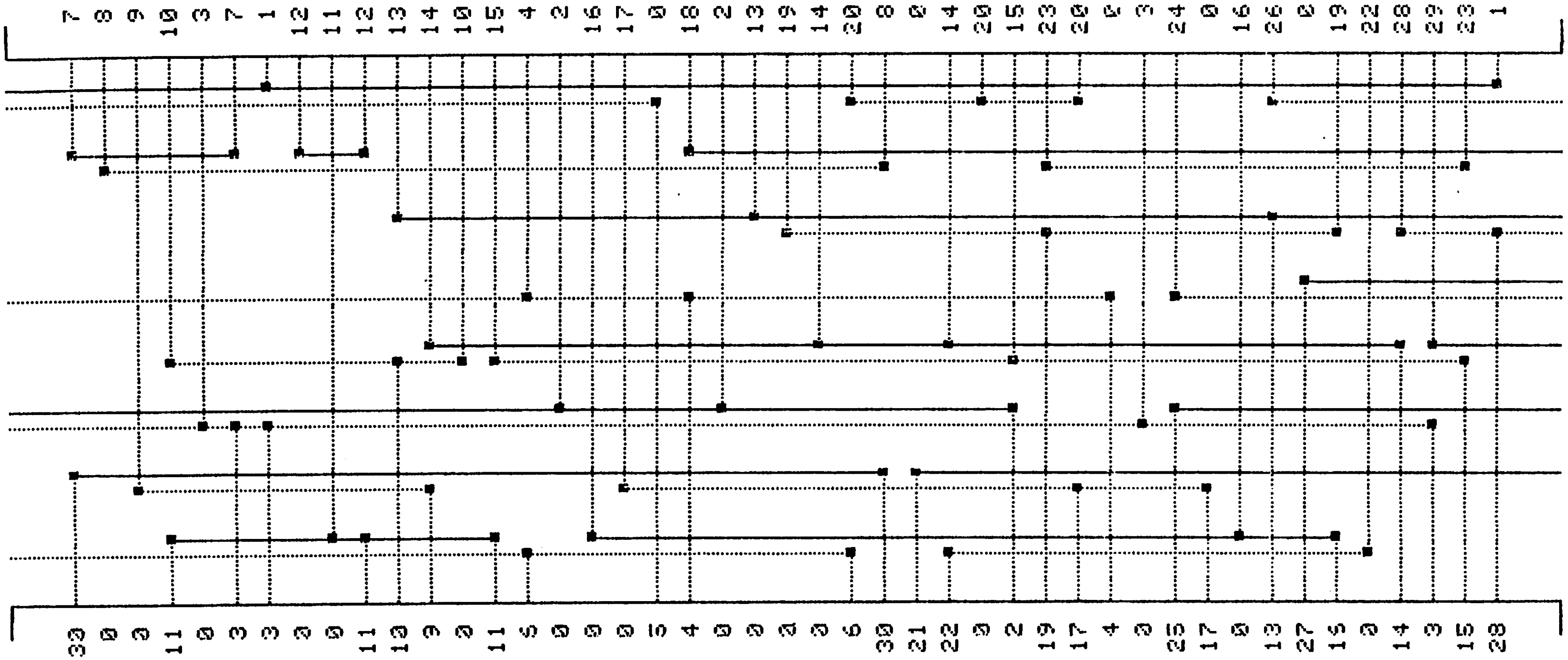


column= 43   # zone= 8   before merging vmax= 7   maximum density= 12

net= 34   start zone= 8   after merging vmax= 7   number of tracks= 7

Fig. 9

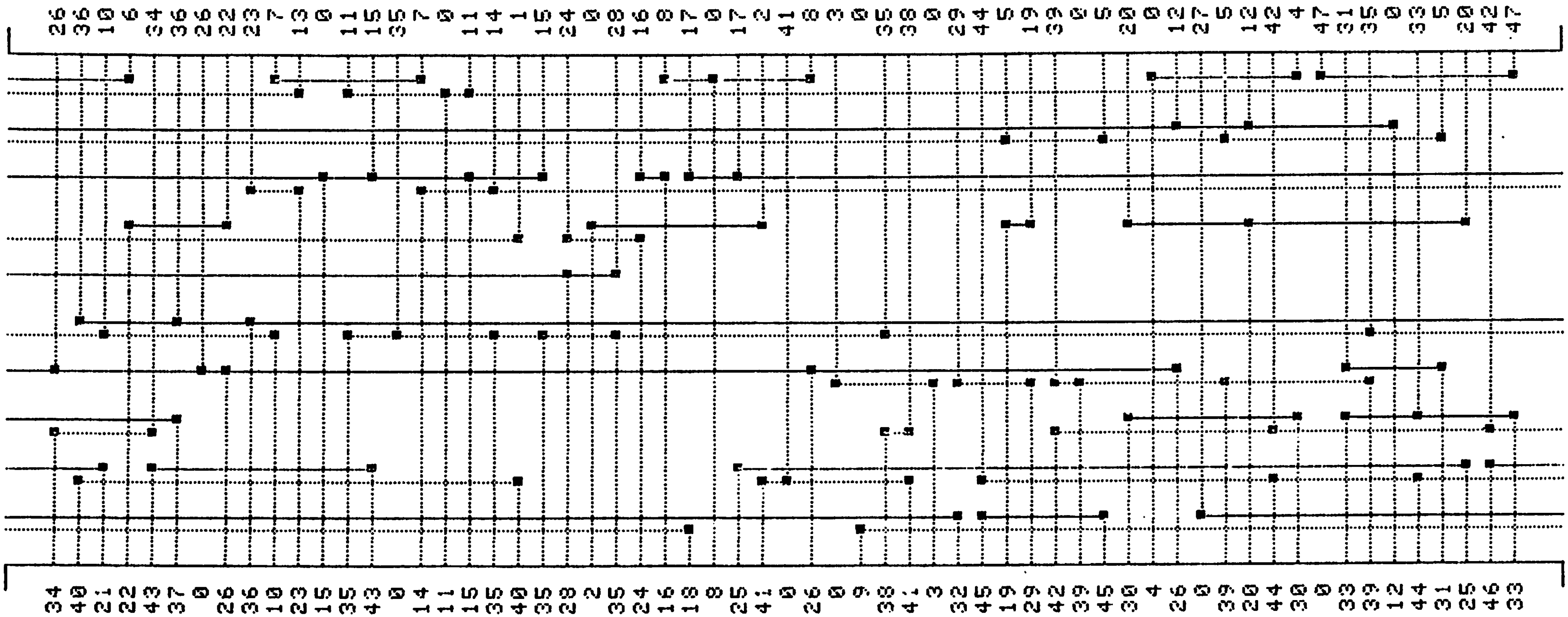
****** example 3a*******



# column= 60	# zone= 12	before merging vmax= 4	maximum density= 15
# net= 30	start zone= 16	after merging vmax= 4	number of tracks= 8

Fig. 10

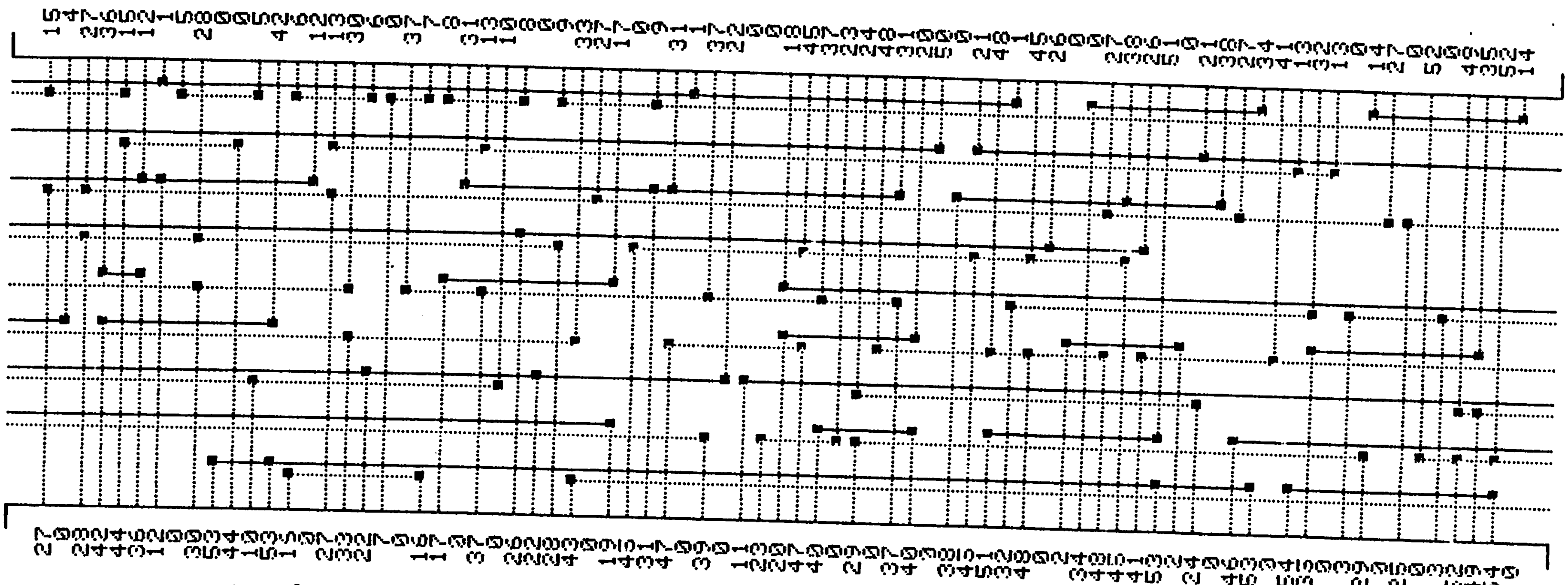
******* ex.3b *******



\# column= 84   # zone= 19   before merging vmax= 9   maximum density= 17
\# net= 47   start zone= 19   after merging vmax= 9   number of tracks= 10

Fig. 11

****** example 3c ******

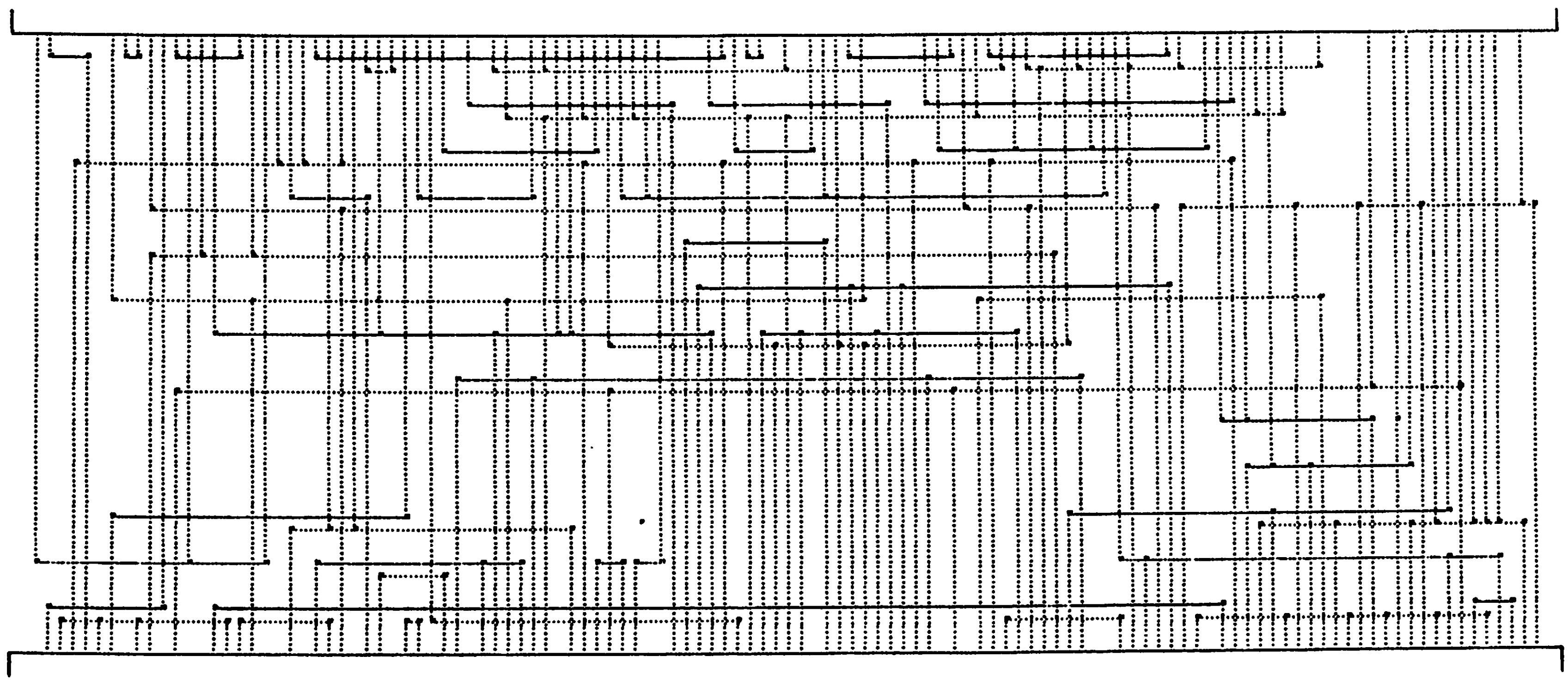


\# column=103   # zone= 20   before merging vmax= 6   maximum density= 18

\# net= 54   start zone= 2   after merging vmax= 9   number of tracks= 9

Fig. 12

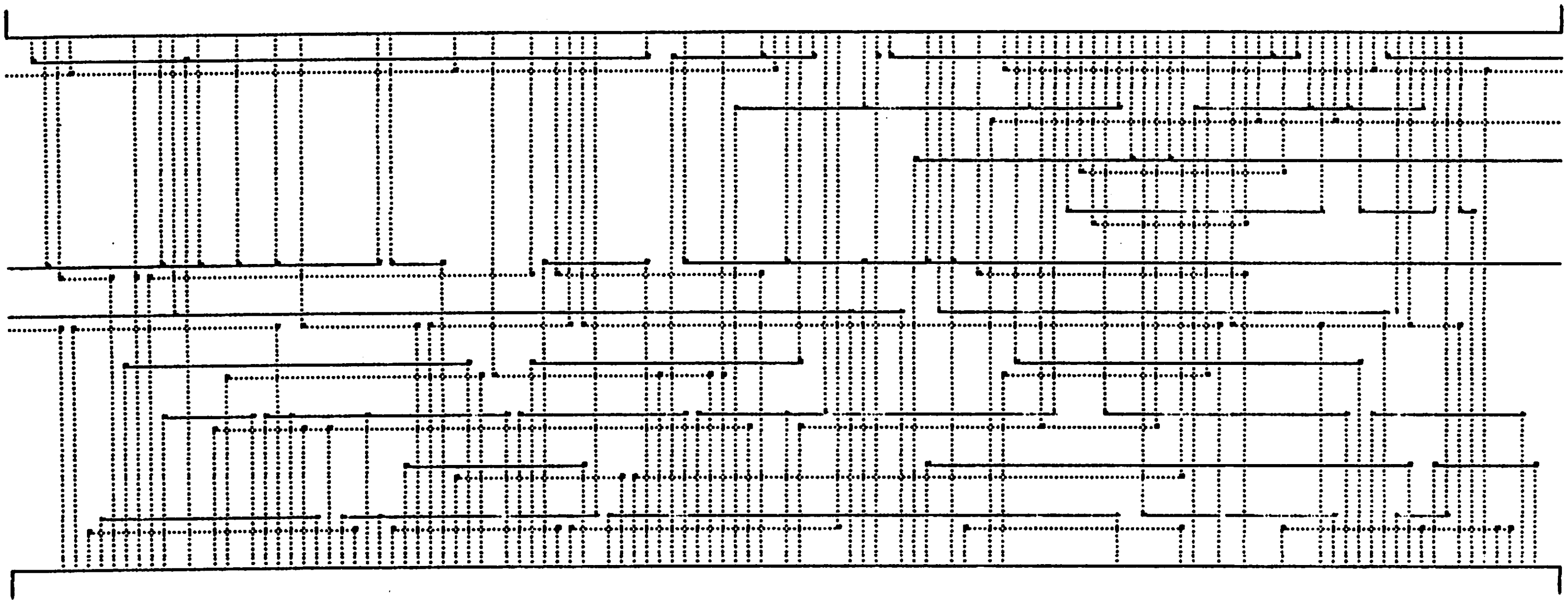
******example 4b ******



column=119   # zone= 29   before merging vmax= 13   maximum density= 17
net= 55   start zone= 18   after merging vmax= 13   number of tracks= 13

Fig. 13

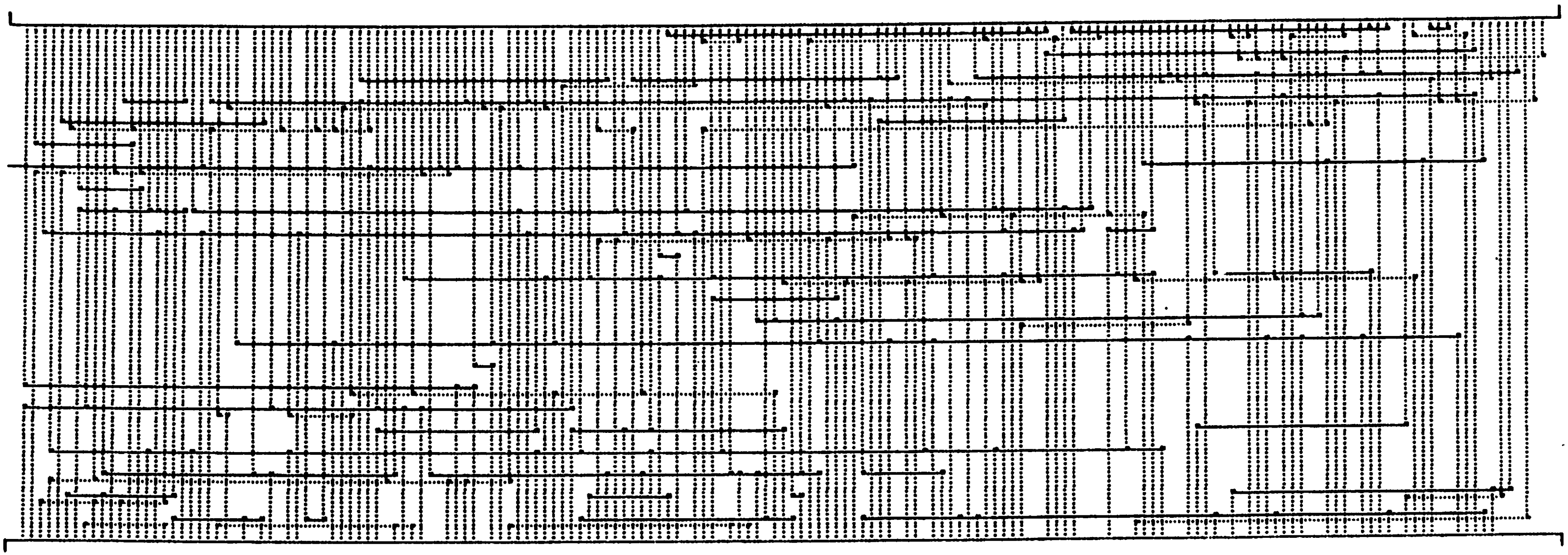
******example 5 ******



column=128 # zone= 34 before merging vmax= 3 maximum density= 20
net= 63 start zone= 1 after merging vmax= 5 number of tracks= 10

Fig. 14

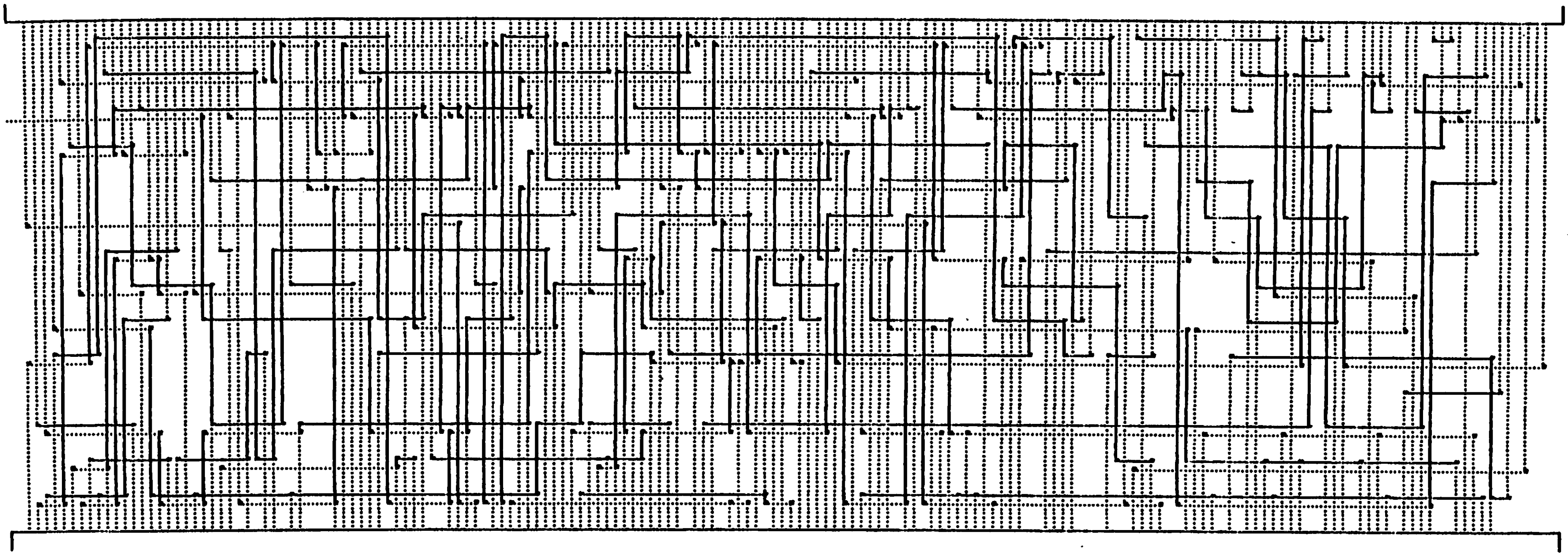
*******difficult example***************



\# column=175 # zone= 33 before merging vmax= 23 maximum density= 19
\# net= 72 start zone= 2 after merging vmax= 23 number of tracks= 23

Fig. 15

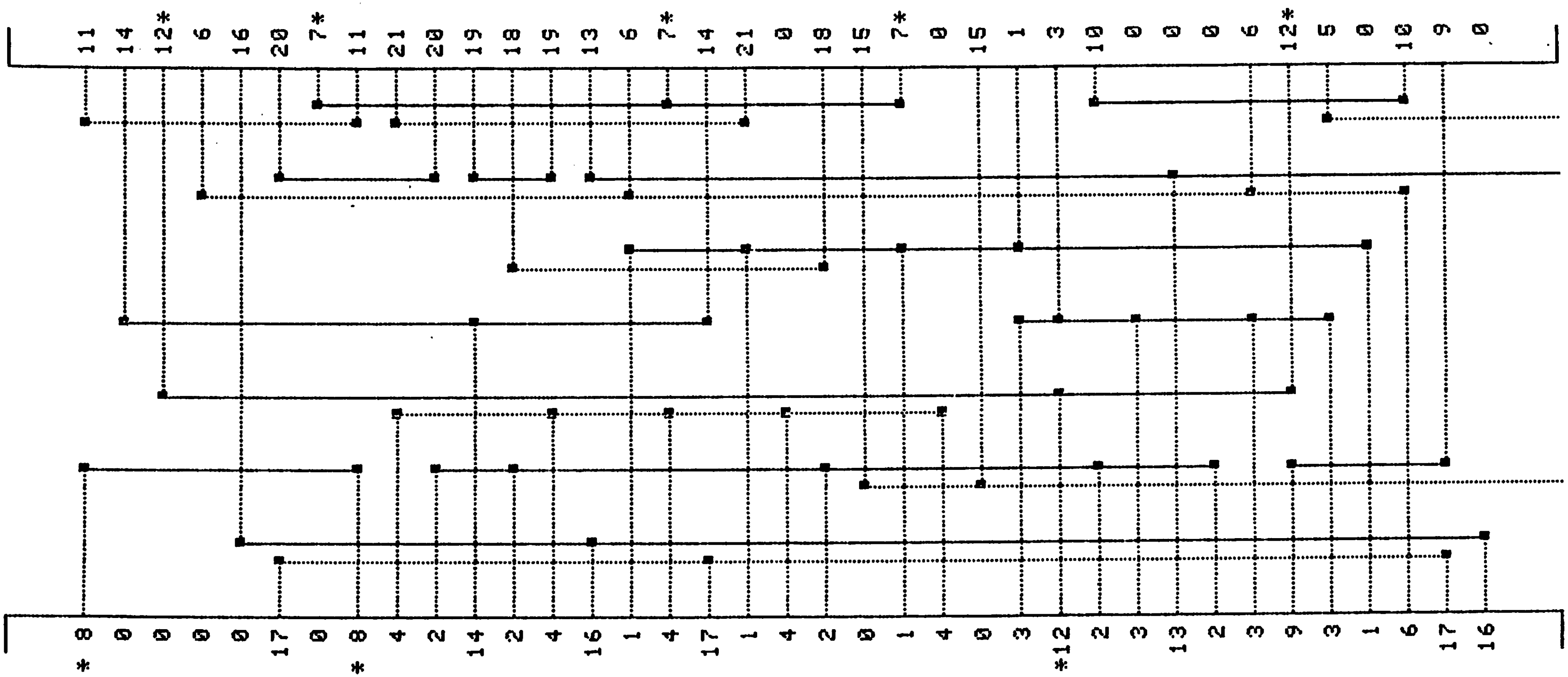
*******difficult example****************
(with dogleg)



\# column=175   # zone=129   before merging vmax= 9   maximum density= 19
\# net= 72   start zone= 58   after merging vmax= 10   number of tracks= 14

Fig. 16

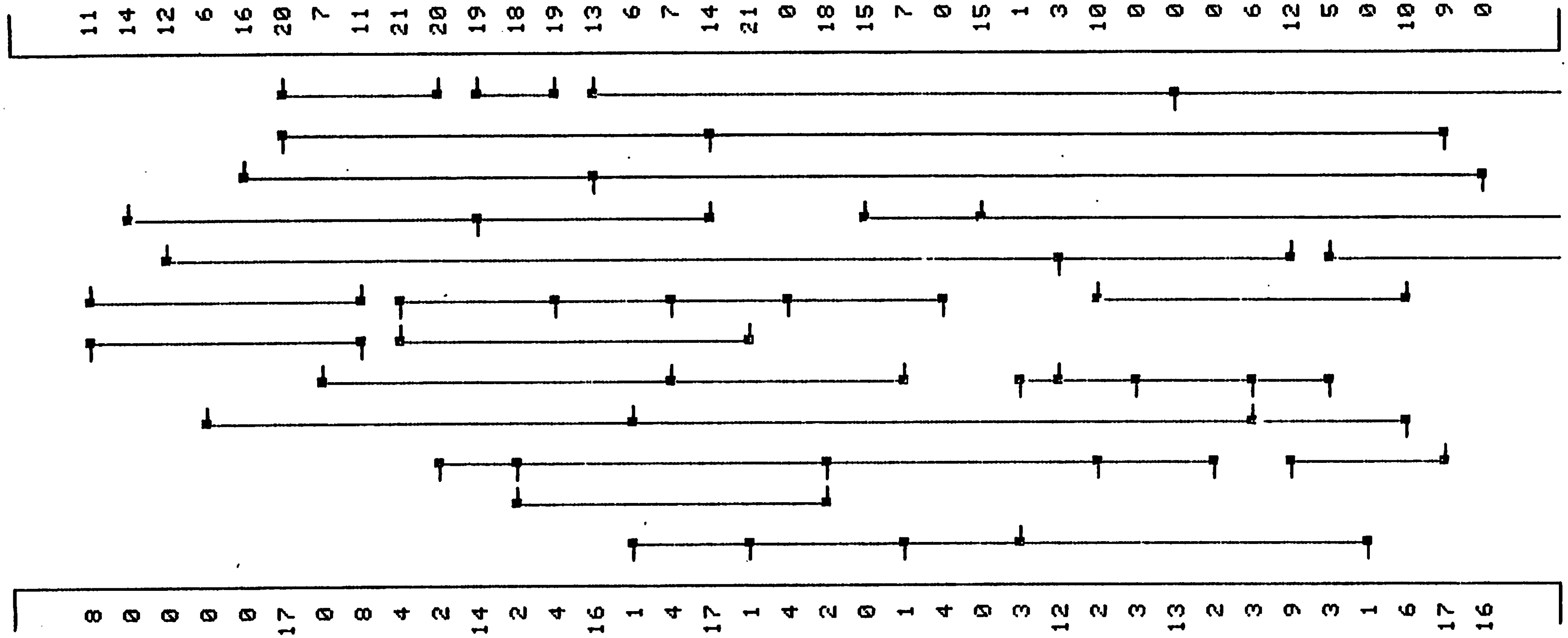
**************example 1******************* (with regard to "power" and "ground" nets)



\# column= 40 # zone= 8 before merging vmax= 7 maximum density= 12
\# net= 21 start zone= 8 after merging vmax= 7 number of tracks= 7

Fig. 17

**************example 1*******************



column= 40 # net= 21 maximum density= 12

Fig. 18