

# 사이클 문제를 고려한 2층채널 배선기에 관한 연구

## (A Study on a 2 Layer Channel Router Considering Cycle Problems)

金 承 淵\*

(Seung Youn Kim)

### 要 約

본 논문에서는 사이클 문제를 고려한 채널 배선 알고리즘을 제안한다.

배선요구는 채널 상하블럭상의 핀번호로 주어지며, 출력은 등전위 핀간의 상호 연결로서 표현된다. 입력이 주어지면 알고리즘은 채널 표현그래프를 구하여 각 네트에 대한 가중치를 구하고, 사이클을 검사하며, 배선을 완료한다.

사이클이 검출되면 미로 배선으로 경로를 찾는다. 본 알고리즘은 IBM-PC/AT 상에서 C언어로 실현하였으며 사이클이 없는 경우는 근사 최적의 값을 얻으며, 사이클이 있는 경우도 배선이 가능하다.

### Abstract

In this paper, a channel routing algorithm which considers cycle problem is proposed. The requirements of routing is given by pin numbers which imply interconnections between a upper block and lower block of the channel.

Output is represented by interconnections among equipotential pins. When input is given, the algorithm constructs a channel representation graph and makes weight of each net. And then it checks cycle and finishes the routing. If the cycle is detected, it finds path with maze routing.

This algorithm have coded in C language on IBM-PC/AT. If cycle is not detected, the results are near optimal values. If it is detected, routing is possible as well.

### I. 서 론

VLSI 기술 진보와 함께 회로의 집적도가 증가함에 따라 레이아웃, 특히 배선문제가 매우 중요한 난제로 등장하였다. 배선문제는 배선이 요구되는 모듈(module)

간을 등전위로 갖는 신호선으로 연결하는 것으로서, 배선과 관련하여 주어지는 제한조건을 만족시켜야 한다는 제약이 따른다. 제한조건은 배선이 요구되는 배선 영역의 형상 및 기술(technology)의 뒷받침 등에 따라 달라진다.<sup>1)</sup>

배선과 관련한 최초의 연구는 1961년 Lee에 의해 제안된 미로법<sup>2)</sup>이며, 그후 미로법의 단점을 개선하기 위한 방안의 하나로 선분탐색법<sup>3)</sup>이 제안되는 등 많은 개선을 이룩했다. 이들 방법은 인쇄회로기판 (printed

\*正會員, 大田開放大 電子計算學科  
(Dept. of Computer Science, Nat'l Tae Jon  
Open Univ.)  
接受日字: 1987年 7月 31日

circuit board) 설계시 널리 이용하고 있다. 그러나 gate array, polycell 등 규칙적인 구조를 갖는 칩(chip)의 개발이 본격화되면서 채널배선법<sup>1)</sup>이 제안되었다. 채널배선법은 상하 양측에 균일한 형태로 배선요구가 발생하는 경우 이용하는 배선방법으로서 Bell Lab.의 LTX<sup>2)</sup>의 “Dogleg channel router”,<sup>3)</sup> 채널병합에 의한 배선<sup>4)</sup>등으로 연구 개선되었다. 이러한 연구는 모두 100% 배선율 달성을 목표로 하며 채널의 폭(width)이 가변적이도록 했다. 연구결과는 우수하여 참고문헌 7번의 알고리즘에 의해 배선을 행할 경우 Deutsch가 제시한 “Difficult example”을 최적에 가까운 해로 실현하고 있다.<sup>1)</sup> 국내의 경우 “Dogleg channel router”가<sup>8)</sup> 다양한 형태의 Dogleg를 사용하여 참고문헌7에서 Dogleg를 사용한 경우와 동일한 트랙(track)으로 배선을 완료하고 있다.

그런데 이들 배선방법은 배선요구에서 사이클(cycle)이 발생하지 않아야 한다는 제약이 있다. 그 결과 현재 활발히 연구되는 실리콘-컴파일러 등에 이용할 경우 배치단계에서 사이클을 제거해야 한다. 이러한 문제를 해결하려면 배선시 사이클을 고려하며 배선을 완료할 수 있어야 한다. 따라서 본 논문에서는 트랙이 다소 증가하는 것을 감수하고 사이클을 해결하기 위해 채널표현 그래프를 구성하고 Dogleg를 실행하며 배선을 완료하는 새로운 방법을 제안한다.

II. 수직, 수평제한 그래프

채널 배선에 있어서 배선 요구에 대한 네트리스트(netlist)는 그림 1과 같이 표현된다. 상부블럭 핀번호는 연결요구가 발생하는 등전위 핀을 의미하며, 하부블럭 핀도 동일하다. 양블럭 사이의 신호선들은 에지(edge)로 표현되며 화살표 방향의 핀과 연결되어야 함을 나타낸다.

배선시 신호선들이 놓이는 트랙을 결정하는 방법은 칩의 면적을 줄이기 위한 직접적인 변수가 된다. 채널 배선은 일반적으로 수직, 수평선분을 1층, 2층에 따로 위치시키고 갈곡점에서 via를 뚫어 등전위를 이루도록 하고 있다. 이는 동일층에서 등전위 배선을 완료하고자 할 경우 “river routing” 문제<sup>9)</sup>로 바뀌기 때문이다. 그림 1의 배선요구는 그림 2처럼 구현된다.

그림 2의 결과를 얻기 위해 트랙을 할당 할 때는 연결요구가 상부블럭핀, 하부블럭핀의 어느 것과 발생하는지가 중요한 함수가 된다. 이를 표현하는 그래프가 수직 제한 그래프(vertical constraint graph)이다. 그림 1에서는 행번호 1에서 신호선 1과 신호선 5사이에서 수직제한이 발생한다. 그러나 행번호 3에서는 수직제

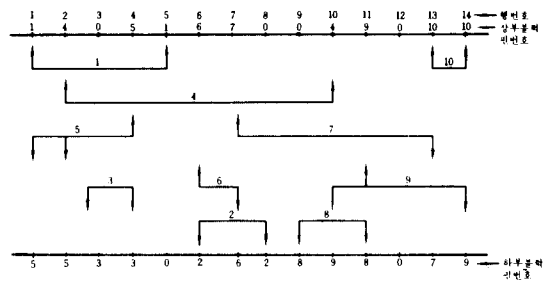


그림 1. 배선요구에 대한 네트리스트 표현  
Fig. 1. Netlist representation for routing requirement.

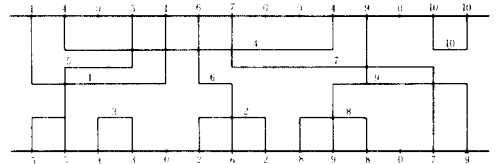


그림 2. 그림 1의 구현  
Fig. 2. A realization for the requirement in Fig. 1.

한이 발생하지 않는다. 수직제한은 특정신호선이 또 다른 신호선의 상부트랙에 놓여야 하는 제한이다. 이러한 수직제한을 종합하여 표현하면 그림 3과 같은 수직 제한 그래프를 얻는다. 이때 최상위에 위치하는 신호선으로부터 최하위의 신호선까지의 경로길이를 최대경로길이(maximum path length)라 한다. 채널배선에서는 최대경로길이 미만으로 트랙 수를 줄일 수 없다.<sup>15-17)</sup>

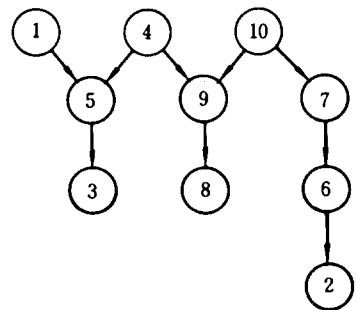


그림 3. 그림 1의 수직제한 그래프  
Fig. 3. Vertical constraint graph for the netlists in Fig. 1.

이와는 별개의 문제로서 동일트랙에 놓일 수 없는 신호선들을 구하는 문제로서 수평제한그래프(horizontal constraint graph)가 있다. 수평제한그래프는 zone의 설정 및 표현으로 나타나며, zone의 설정시에는 최대 클리크(maximum clique)의 계산이 수반된다.

그림 1에 대해 zone을 설정하면 표 1과 같고, 최대 클리크는 그림 4와 같은 인터벌그래프(interval graph)로부터 얻어진다.

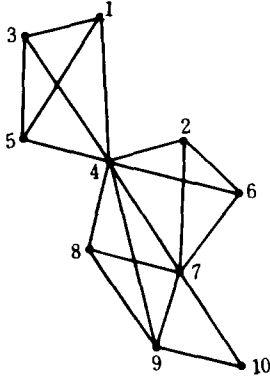


그림 4. 그림 1에 대한 인터벌그래프  
Fig. 4. Interval graph for the netlist in Fig. 1.

표 1. Zone의 설정  
Table 1. Zone construction.

행 번호	신 호 선 집 합	Zone
1	1 5	1
2	1 4 5	
3	1 3 4 5	
4	1 3 4 5	
5	1 4	
6	2 4 6	2
7	2 4 6 7	
8	2 4 7	
9	4 7 8	3
10	4 7 8 9	
11	7 8 9	
12	7 9	
13	7 9 10	4
14	9 10	

그림 4에서 최대 클리크는 1, 3, 4, 5와 2, 4, 6, 7 그리고 4, 7, 8, 9와 7, 9, 10이다. 이 값은 각 zone에서 최소로 필요로하는 트랙의 수를 결정한다. 그 결과 각 zone 내에서 최대 클리크는 해당 zone에서 최소로 필요로 하는 트랙을 결정하고 이들 중의 최대 값을

전체 트랙 zone에서 필요로 하는 최소트랙의 하한을 결정한다. 이때 이 값을 최대밀도(maximal density)라 부른다.<sup>15~17)</sup>

따라서 배선시 필요한 최소 트랙은 최대경로길이와 최대밀도 가운데 큰 값이다.

최소트랙으로 배선을 완료하면 최적의 해가 된다. 그러나 일반적으로 신호선들이 많아질 경우 최대밀도가 최대경로 길이보다 큰 값을 갖는다.

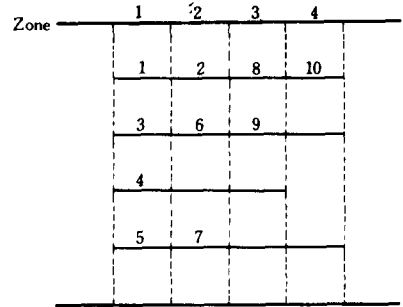


그림 5. Zone 표현  
Fig. 5. Zone representation.

III. 채널 표현 그래프

채널 표현 그래프는 수직제한그래프와 수평제한그래프를 하나로 합친 그래프이다. 이 그래프는 행번호가 노드(node), 신호선들이 에지(edge)로 각각 표현된다.

에지는 단방향, 양방향, 무방향으로 구성되며 다음과 같이 정의된다.

(1) 단방향에지 : 에지  $e_i(v_j, v_k)$ 의 j, k행 가운데 j행 (또는 k행)은 상부블럭 핀과 연결요구가 발생하고, k행 (또는 j행)은 하부블럭 핀과 연결요구가 발생한 에지(그림 6 (a)).

(2) 양방향에지 : 에지  $e_i(v_j, v_k)$ 의 j, k행 두곳에서 모두 하부블럭핀과 연결요구가 발생한 에지(그림6 (b)).

(3) 무방향에지 : 에지  $e_i(v_j, v_k)$ 의 j, k행 두곳에서 모두 상부블럭핀과 연결요구가 발생한 에지(그림6 (c)).

이때 에지  $e_i(v_j, v_k)$ 는 노드 j, k와 에지 i가 연결 관계를 갖고 있음을 의미한다.

이 개념을 사용하여 그림 1에 대한 채널표현 그래프를 구성하면 그림 7과 같다. 그래프에서 노드1에 연결요구를 갖는 에지 1과 에지 5는 트랙에 놓일 때 에지1이 에지 5보다 상부에 위치해야 함을 보이고 있다. 또한 노드 2에서 에지 5가 에지 4의 하부에 위치해야 함을 보인다.

이와함께 노드 2부터 노드 5까지는 에지 1과 에지 4

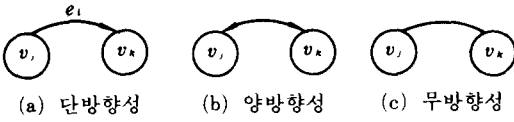


그림 6. 에지의 분류  
Fig. 6. Sorting of edges.

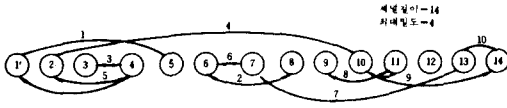


그림 7. 그림 1에 대한 채널표현그래프  
Fig. 7. A channel representation graph for the netlist in Fig. 1.

가 중복되어 동일트랙에 놓일 수 없음을 보인다. 그러므로 채널표현그래프는 수직, 수평제한을 모두 구할 수 있는 그래프임을 알 수 있다.

1. 가중치(weight) 할당

채널표현그래프를 구한 후 신호선들을 트랙에 할당하기 위한 준비로서 에지의 가중치를 구한다. 그림 8은 가중치의 할당 방법을 보인다.

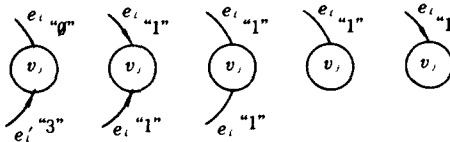


그림 8. 에지의 가중치 할당  
Fig. 8. A weight assignment of edges.

(1) 그림 8 (a)는 노드  $v_j$ 에서 상부, 하부블럭 양편에 대해 연결요구가 동시에 발생했을 때 상부블럭과 연결요구를 갖는 에지는 가중치 "0"를, 하부블럭과 연결요구를 갖는 에지는 가중치 "3"을 부가한다.

(2) 그림 8 (b)는 노드  $v_j$ 에서 (1)번 이외의 경우가 발생했을 때의 가중치 할당이다. 이때는 에지에 가중치 "1"을 부가한다.

이와같이 에지에 가중치를 부여한 이후 각 에지에 대한 총 가중치를 구한다.

2. 신호선 선택

에지로 표현된 신호선들을 트랙에 할당하기 위해 각 에지의 가중치 합을 비교한다. 에지의 총 가중치가 2 이하인 경우는 어느 특정에지에 대해 상부블럭과의 연결시 수직제한이 발생하지 않으므로 상부트랙에 할당 가능하며, 3 이상의 경우는 상부에 놓일 특정 에지를 먼저 트랙에 할당해야 하므로 현재의 트랙에는 할당이 불가능하다.

이러한 이유로 인하여 총 가중치 2 이하인 에지들을 집합 S3로 하고, 이 가운데에서 총 가중치가 1인 에지들을 S1, 2인 에지들은 S2로 나눈다. S3는 현재의 트랙에 배정가능한 에지들 집합이므로 이들 에지 중에서 수평제한을 만족하는 에지들을 구하기 위해 변화한 채널 표현 그래프를 다시 구성한다. 총 가중치가 3 이상인 에지들은 집합 S9에 넣는다. 그림 7에서 S3를 구하면  $S3 = \{1, 4, 10\}$ 이며 그림 9와 같다.

S3의 에지들로 구성된 변화된 채널 표현 그래프는 수직제한을 이미 만족한다. 따라서 수평제한을 만족하는 신호선을 선택하기 위해 S3의 처음 에지를 중심으로 수평제한을 만족하는 에지그룹(edge group)  $eg_l$ 을 구한다. 그림 9의 경우  $S3 = \{1, 4, 10\}$ 이므로 각 에

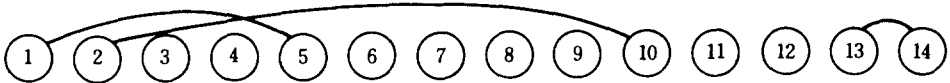


그림 9. S3에 대한 변화된 채널표현 그래프  
Fig. 9. A modified channel representation graph for a set S3.

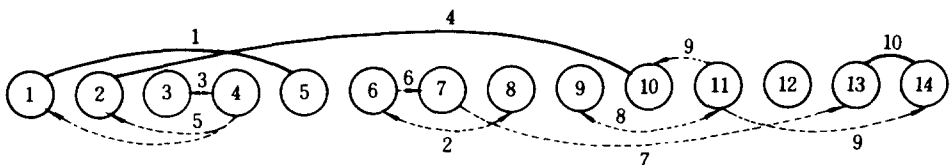


그림 10. 예제의 채널표현그래프  
Fig. 10. A channel representation graph for the example.

지그를  $eg\ell$ 은

$$eg1 = \{1, 10\}$$

$$eg2 = \{4, 10\}$$

이다. 이후  $eg1$ 과  $eg2$  가운데 어느 것을 현재의 트랙에 할당할 것인가를 결정하기 위해 유효계수를 구한다. 유효계수는 다음 방법에 의해 구한다.

에지그룹  $eg1, eg2$ 에서 공통인 에지는 유효계수 계산시 에지그룹 모두에 동일한 기여를하므로 고려대상에서 제외한다.

나머지 에지들은  $e_i(v_j, v_k)$ 에서  $j, k$ 번 노드에 대해 하부블럭으로의 연결요구가 발생하는 에지가 있는지의 여부를 검사하고 그 수를 계산한다. 에지 1이 속한  $eg1$ 은 노드1, 노드2에서 하부블럭과 연결요구를 갖는 에지를 가지므로 유효계수는 2가 된다. 에지 4는 노드2에서 에지5, 노드10에서 에지9가 하부블럭과 연결요구를 가지므로 유효계수가 2이다. 만일 예제와 같이 유효계수가 동일하면 트랙의 가장 많은 부분을 소모하는  $eg\ell$ 을 선택한다. 유효계수가 다른 경우는 그 값이 큰  $eg\ell$ 을 선택한다.

3. 사이클(cycle) 해결

채널표현 그래프에서 집합  $S3, S9$ 는 각각 총 가중치가 2 이하 3 이상의 에지들에 대한 정보를 지니고 있으며  $S9$ 는 사이클에 대한 정보를 지니고 있다. 예를 들어 그림11과 같은 채널표현그래프가 있을 경우 총 가중치는 절대로 3 미만이 될 수 없는 에지가 존재한다. 이러한 에지가 사이클형성에 기여하는 것으로 참고문헌 [5~9]의 연구들은 이 연구 배선불가능 상태로 된다.

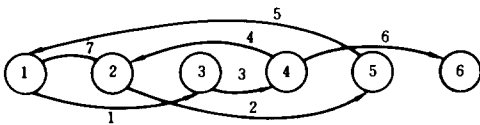


그림11. 사이클이 있는 채널표현그래프  
Fig. 11. A channel representation graph containing a cycle.

따라서 남아 있는 집합  $S9$ 에 대해 배선을 행하기 위하여 사이클을 해결해야 한다. 그림12에서 보면 사이클은 1, 2, 3, 4, 5번 에지 가운데서 어느 하나를 제거하면 가능하다. 그러므로 임의로 에지를 하나 제거하고 남은 에지들에 대해 채널표현그래프를 재구성하고 배선을 행한다. 사이클 해결을 위해 제거한 에지는 다른 배선을 종료한 후 배선을 행한다. 이때 배선에

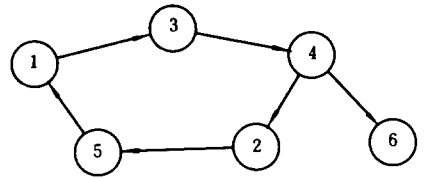


그림12. 그림11에 대한 수직제한그래프  
Fig. 12. Vertical constraint graph for the graph in Fig. 11.

필요한 트랙이 부족하면 트랙을 추가하여 배선을 완료한다.

IV. 알고리즘 요약

Step 1 : 배선요구를 기술한 네트리스트가 입력되면 다음과 같이 초기조건을 설명한다.

$$e = \{e_1, e_2, \dots, e_n, e_{n+1}, \dots, e_n\}$$

이때  $e$ 는 에지집합으로  $n$ 은 에지번호를 의미한다.  $e_i$ 는  $e_i(v_j, v_k)$ 의 특성을 갖는다. 초기 가중치는 모두 0으로 하여 가중치  $e_{wk}$ 는  $e_{wk} = 0$  for  $(1 \leq i \leq n)$ 로 한다. 사이클 해결시 제거된 에지를 저장하기 위해 사이클값  $c_v$ 는  $c_v = NiL$ 로 한다.

Step 2 : 채널표현그래프  $Gc = (v, e)$ 를 구성한다.

Step 3 :  $Gc$ 에서 각 신호선  $e_i$ 에 대한 가중치를 구한다.

Step 4 :  $e_{wk}$  값으로  $S1, S2, S3, S9$ 를 구한다.

$$S1 = \{e_i | e_{wk} < 2, 1 \leq i \leq n\}$$

$$S2 = \{e_i | e_{wk} = 2, 1 \leq i \leq n\}$$

$$S3 = S1 \cup S2$$

$$S9 = \{e_i | e_{wk} > 2, 1 \leq i \leq n\}$$

Step 5 :  $S3$ 가 0이 아니면 step 6으로 가고,  $S3$ 가 0이고  $S9$ 도 0이면 step13으로 간다. 그외의 경우는 step12로 간다.

Step 6 : 트랙수  $Tn$ 을 1 증가시킨다.

Step 7 :  $S1$ 이 0이면 트랙  $Tn$ 에 수평제한을 위배하지 않도록 에지를 할당하고 step11로 간다.

Step 8 :  $S3$ 로부터 에지그룹  $eg\ell$ 을 구성한다.  $eg\ell$ 이 1개 구성되면 step10으로 가고 그외의 경우는 에지  $e_i$ 가 2개 이상의  $eg\ell$ 에 속할 때에 step9로, 아닐때는  $i = i + 1$ 하며 step8을 반복 실행한다.

Step 9 : step8에서 에지  $e_i$ 가 속한  $eg\ell$ 을 사용하여 다음의 선택함수를 만족하는  $eg\ell$ 을 선택한다. 선택함수는 다음의 (1), (2), (3)번 순으로 적용하며  $eg\ell$ 이 선택되면 Step10으로 간다.

- (1) 유효계수값이 가장 큰  $eg\ell$
  - (2)  $\sum_{i=1}^n e_i |v_j - v_k|$  가 가장 큰  $eg\ell$
  - (3)  $eg\ell$ 의 에지 중에서  $v_j$ 가 가장 작은  $eg\ell$
- 만일  $eg\ell$ 이 2 이상 선택되면 그 다음으로 작은  $v_j$ 를 찾는 과정을 반복하여  $eg\ell$ 이 하나가 선택될 때까지 행한다.

- Step 10 : 트랙 Tn에 선택된  $eg\ell$ 의 에지들을 할당한다.  
 Step 11 : 트랙 Tn에 할당된 에지들을 네트리스트 e에서 제거하고 step2로 간다.  
 Step 12 : 임의로 S9의 에지 하나를 선택하여 사이클 값 Cv에 저장하고 그 값을 네트리스트에서 제거한 다음 step2로 간다.  
 Step 13 : Cv가 NiL이면 step14로 가고, 그외의 경우는 Cv의 내용에 대해 남은 배선을 행한다. 배선은 미로법을 응용하여 행한다. 만일 정해진 트랙 Tn내에서 경로를 찾을 수 없으면 트랙을 1증가시키고 다시 미로법에 의해 경로를 탐색한다.  
 Step14 : 각 트랙에 대한 에지시퀀스를 얻는다.  
 Step 15 : stop

V. 실험 결과

본 연구는 C언어를 사용하여 구현하였으며 실험결과는 그림 13, 14, 15에 나타났다. 그림13은 그림1의 배선요구를 최적의 해로 실현한 것이며, 그림14는 사이클이 없는 경우의 배선요구를 실현한 예이다. 배선결과는 최적의 해를 보이고 있다. 그러나 Deutsch의 "difficult example"의 경우는, 참고문헌 [7]에서 Dogleg를 사용할 때 실현한 21트랙보다 1트랙이 많은 22트랙으로 실현하고 있다. 참고문헌 [8]에서는 20트랙으로 배선을 행하여 최적의 해인 19트랙을 1트랙 초과하고 있다. 한편, 사이클이 있는 경우는 최적의 해인 4트랙 보다 3트랙이 많은 7트랙을 점유하고 있다. 따라서 본 연구에서는 배선에서 최적의 해를 얻기 위한 프로그램은 약 2200라인이며 그림15(a)예의 경우

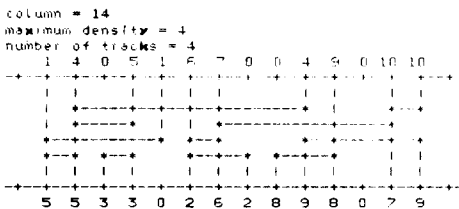


그림13. 그림 1의 배선요구 구현  
 Fig. 13. A realization of the routing requirement in Fig. 1.

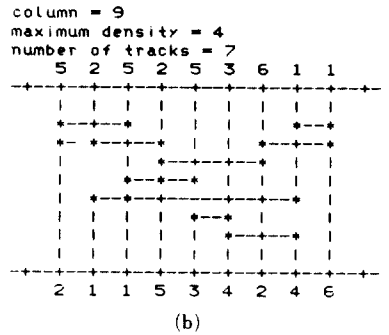
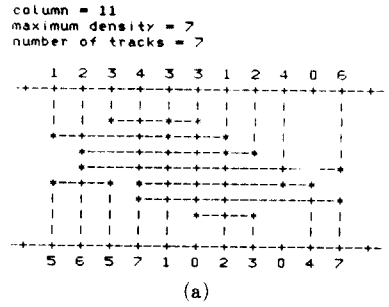


그림14. 사이클이 없는 경우의 배선 예  
 (a) 예 1  
 (b) 예 2  
 Fig. 14. Realization examples of routing requirement without cycle.  
 (a) Example 2.  
 (b) Example 1 2.

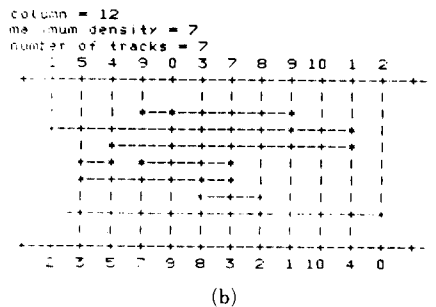
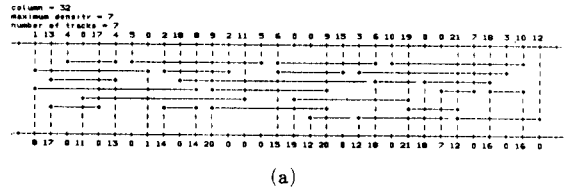


그림15. 사이클이 있는 경우의 배선 예  
 (a) 예 1  
 (b) 예 2  
 Fig. 15. A realization example of routing requirement with cycle.  
 (a) Example 1.  
 (b) Example 2.

터미널로 배선결과를 출력하는데 15초 소요되었다. 그런데 본 연구를 Deutsch의 "difficult example"에 적용했을 때는 최적의 트랙 수 19보다 3개 많은 트랙으로 구현하고 있어 이에 대한 개선이 요구된다.

## VI. 결 론

본 연구는 C언어를 사용하여 IBM-PC, SSM-16 등에서 구현하였다. 배선요구시 사이클이 발생할 경우는 채널표현그래프로부터 사이클발생에 기여한 신호선을 제거하고 채널배선을 행한 후 잔여배선을 미로법에 의해 행하는 알고리즘을 제안했다.

채널배선시 Dogleg를 일부 허용하여 트랙수 감소와 via수의 감소를 고려했다. 트랙수가 가감이 가능한 경우는 100% 배선율을 달성할 수 있다.

본 논문에서 제안한 알고리즘을 이용하여 개발한 배선기를 이용할 경우 배선영역의 최소화를 도모할 수 있고 배치단계에서 사이클 발생여부에 유의하지 않아도 될 수 있게 되며 짧은 설계시간, 배선을 완료할 수 있을 것으로 기대된다. 그러나 일부의 예에서는 배선시 트랙 수가 최적의 값을 갖지 않는 경우가 발생하여 이의 개선이 요망된다.

앞으로의 연구과제는 3층 이상의 경우를 고려하는 multi-layer 채널배선기, gridless 채널배선기 등의 연구가 필요하다.

## 參 考 文 獻

[1] T. Ohtsuki, Layout design and Verification, North-Holland: Netherlands, 1986.  
[2] C.Y. Lee, "An algorithm for path

connections and its application," *IRE Trans. on Electronic Computers*, vol. EC-10, pp. 346-365, 1961.

- [3] David W. Hightower, "A solution to line-routing problem on the continuous pane," *Proc. 6th DA Workshop*, pp. 1-24, 1969.  
[4] A. Hashimoto and J. Stevens, "Wire routing by optimizing channel assignment within large apertures," *Proc. 8th DA Workshop*, pp. 155-169, 1971.  
[5] G. Persky and D.N. Deutsch and D.G. Schweikert, "LTX-A System for the directed automatic design of LSI circuits," *Proc. of the 13th DA Conf.*, pp. 399-407, 1976.  
[6] D.N. Deutsch, "A dogleg channel router," *Proc. of the 13th DA Conf.*, pp. 425-433, 1976.  
[7] T. Yoshimura and E.S. Kuh, "Efficient algorithms for channel routing," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. CAD-1, no. 1, pp. 25-35, 1982.  
[8] 강 인, 이진배, 정정화, "Dogleg를 이용한 새로운 channel router", 대한전자공학회 CAD, 반도체 재료 및 부품연구회 합동 학술발표대회 논문집, pp. 35~39, 5, 1986.  
[9] 강 인, "VLSI의 detailed router 개발에 관한 연구", 한양대학교 공학석사 학위논문, 12, 1986.  
[10] R.Y. Pinter, "River routing: methodology and analysis," *3rd CALTECH Conf. on VLSI*, pp. 141-163, 3, 1983.