(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2012년03월16일
(11) 등록번호 10-1116943
(24) 등록일자 2012년02월08일

(51) 국제특허분류(Int. Cl.)
*H03K 19/177* (2006.01)
(21) 출원번호 10-2006-7022720
(22) 출원일자(국제) 2005년02월28일
심사청구일자 2009년12월14일
(85) 번역문제출일자 2006년10월30일
(65) 공개번호 10-2007-0024520
(43) 공개일자 2007년03월02일
(86) 국제출원번호 PCT/US2005/006583
(87) 국제공개번호 WO 2005/104375
국제공개일자 2005년11월03일
(30) 우선권주장
10/814,943 2004년03월30일 미국(US)
(56) 선행기술조사문헌
US05880597 A
US06163168 A
US06670825 B1
전체 청구항 수 : 총 126 항

(73) 특허권자
**어드밴티지 로직, 인코포레이티드**
미국 캘리포니아 마운틴 뷰 아나 프리바다 1110 (우: 94040)
(72) 발명자
**패니, 페터, 엠.**
미국 94040 캘리포니아 마운틴 뷰 아나 프리바다 1110
**팅, 벤자민, 에스.**
미국 95070 캘리포니아 사라토가 설리반 웨이 21120
(74) 대리인
**남상선**

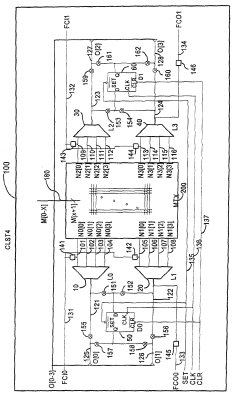
심사관 : 강현일

(54) 발명의 명칭 **스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한, 집적 회로, 전자 시스템, 방법, 상기 집적 회로의 접속 방법 및 상기 전자 시스템을 제공하는 방법**

## (57) 요 약

확장성 넌-블럭킹 스위칭 네트워크(SN)는 각각의 상호접속 리소스들 제약들내에서 일반적으로 제한되지 않는 방식으로 다른 다수 세트들의 도선들에 다수의 제 1 도선들을 접속시키는데 사용되는 스위치들 및 중간 스테이지들의 도선들을 갖는다. 상기 SN은 네트워크, 라우터들, 및 프로그래머블 로직 회로들에 사용되는 대형 스위치 네트워크를 제공하도록 직렬로 또는 계층적으로 광범위한 애플리케이션들에 적용될 수 있다. 상기 SN은 SN을 통해 제 1 세트의 도선들을 주어진 로직 회로 계층의 다수의 세트들의 도선들에 접속시키는데 사용됨으로써, 각각의 다수 세트들의 도선들은 동일하거나 상호교환가능하고, 즉 구조화에 의해 다음 레벨의 회로 계층에 사용될 때 동일한 제 1 세트의 도선들을 형성할 수 있다. 상기 SN은 큰 크기의 도선들의 세트들에 대해 확장될 수 있고 큰 크기의 회로들 사이에서 프로그래머블 상호접속들을 인에이블시키기 위해 계층적으로 사용될 수 있다.

*대 표 도* - 도1



## *특허청구의 범위*

### 청구항 1

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로로서,

다수의 M 수(number)의 제 1 도선들(conductors);

다수의 k 세트들의 N 수의 제 2 도선들 ? 제 2 도선들의 수는 k×N 임 ?; 및

스위칭 네트워크(SN: switching network) ? 여기서 M은 적어도 k+N 이고, k는 적어도 2 이며, N은 적어도 2 임 ?

를 포함하고, 상기 SN은,

다수의 스위치들;

i=[1-NO]에 대해 NO 세트들의 $IO_i$ 수의 도선들을 포함하는 다수의 IO 수의 제 3 도선들 ? 여기서 NO는 적어도 2이고, N은 적어도 NO이며, IO는 적어도 M과 같고 M이 (k×N) 미만일 때 M과 (k×N) 사이임 ?;

다른 도선의 통과(traversal)를 요구함이 없이 상기 다수의 스위치들을 통해 p, q=[1-NO]에 대해 적어도 2개의 상이한 $IO_p$ 수의 도선들 및 $IO_q$ 수의 도선들로부터 각각 선택되는 적어도 2개의 상이한 도선들에 선택적으로 연결하기 위한 상기 다수의 M 수의 제 1 도선들 중 임의의 2개의 도선들;

상기 SN을 통해 제 1 세트의 (k×(M/NO)) 이하의(at most) 수의 도선들에 선택적으로 연결하기 위한 상기 적어도 2개의 상이한 도선들 중 제 1 도선 ? 상기 제 1 세트의 (k×(M/NO)) 이하의 수의 도선들은 상기 k 세트들의 N 수의 도선들 각각으로부터 (M/NO) 이하의 수의 도선들을 가짐 ?; 및

상기 SN을 통해 제 2 세트의 (k×(M/NO)) 이하의 수의 도선들에 선택적으로 연결하기 위한 상기 적어도 2개의 상이한 도선들 중 제 2 도선 ? 상기 제 2 세트의 (k×(M/NO)) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터, 상기 제 1 세트의 (k×(M/NO)) 이하의 수의 도선들과 상이한, (M/NO) 이하의 수의 도선들을 가짐 ?

을 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

### 청구항 2

제 1 항에 있어서,

상기 $IO_i$ 수의 도선들의 각각의 도선은 다른 도선의 통과를 요구함이 없이 상기 SN을 통해 k 수의 도선들에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

### 청구항 3

제 2 항에 있어서,

상기 $IO_i$ 수는 (k×N/NO) 수 및 (M/NO) 수 중에서 보다 큰 수 이하이고,

상기 다수의 스위치들은 적어도 [(M-N+1)×N + M×k] 수의 스위치들이고 M×(k+N) 수 이하의 스위치들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

### 청구항 4

제 2 항에 있어서,

상기 집적 회로는 k 수의 프로그래머블 로직 셀들을 더 포함하고,

각각의 상기 k 수의 프로그래머블 로직 셀들은 N 수의 입력 도선들을 가지며,

각각의 상기 k 수의 도선들은 상기 k 수의 프로그래머블 로직 셀들의 상기 N 수의 입력 도선들 중 하나의 도선에 대응하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 5**

제 1 항에 있어서,

상기 M은 R×(k×N)과 같으며 여기서 R은 적어도 0.5이고 1.0 이하인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 6**

제 1 항에 있어서,

상기 I0i 수의 도선들에 선택적으로 연결하기 위한 다수의 $I1_i$ 수의 제 4 도선들 ? 상기 $I1_i$ 수의 도선들은 j=[1-N1]에 대해 $I1_{ij}$ 수의 도선들의 N1 그룹들을 포함하고, 여기서 N1은 적어도 2임 ?

을 더 포함하고, 상기 $IO_i$ 수의 도선들 중 임의의 2개의 도선들은, 다른 도선의 통과를 요구함이 없이 상기 다수의 스위치들을 통해 상기 $I1_i$ 수의 도선들 중 2개의 상이한 도선들에 선택적으로 연결되며,

상기 2개의 상이한 도선들 중 제 3 도선은 $I1_{ir}$ 수의 도선들로부터 나오고, 상기 2개의 상이한 도선들 중 제 4 도선은 $I1_{is}$ 수의 도선들로부터 나오며, 상기 r의 수는 r,s=[1-N1]에 대한 s의 수와 상이하고,

상기 2개의 상이한 도선들 중 상기 제 3 도선은 상기 SN을 통해 상기 k 세트들의 N 수의 도선들로부터 제 3 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들과 선택적으로 연결되며, 상기 제 3 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 (M/(NO×N1)) 이하의 수의 도선들을 갖고,

상기 2개의 상이한 도선들 중 상기 제 4 도선은 상기 SN을 통해 상기 k 세트들의 N 수의 도선들로부터 제 4 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들과 선택적으로 연결되며, 상기 제 4 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 상기 제 3 세트의 (k×(M/(NO×N1))) 수의 도선들과 상이한 (M/(NO×N1)) 이하의 수의 도선들을 갖는,

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 7**

제 6 항에 있어서,

상기 $I1_{ij}$ 수의 도선들의 각각의 도선은 다른 도선의 통과를 요구함이 없이 상기 SN을 통해 k 수의 도선들과 선택적으로 연결하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 8**

제 7 항에 있어서,

상기 집적 회로는 k 수의 프로그래머블 로직 셀들을 더 포함하고,

각각의 상기 k 수의 프로그래머블 로직 셀들은 N 수의 입력 도선들을 가지며, 각각의 상기 k 수의 도선들은 상기 k 수의 프로그래머블 로직 셀들의 상기 N 수의 입력 도선들 중 하나의 도선에 대응하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 9**

제 7 항에 있어서,

상기 $I1_{ij}$ 수는 (k×N/(NO×N1)) 수 및 (M/(NO×N1)) 수 중에서 보다 큰 수 이하이고, 상기 다수의 스위치들은 [M×(NO+N1+k)] 이하의 수의 스위치들로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 10**

제 1 항에 있어서,

각각의 상기 다수의 스위치들은 적어도 하나의 프로그램 제어되는 패스게이트(passgate)로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 11**

제 1 항에 있어서,

각각의 상기 다수의 스위치들은 적어도 하나의 프로그램 제어되는 드라이버들/리시버(receiver)들로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 12**

제 1 항에 있어서,

상기 다수의 스위치들은 프로그램 제어되는 패스게이트들 및 프로그램 제어되는 드라이버들/리시버들 중 적어도 하나로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 13**

제 1 항에 있어서,

상기 다수의 스위치들 중 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 14**

제 1 항에 있어서,

상기 집적 회로는 메모리 소자들을 통합하는(incorporating) 프로세스 기술을 이용하여 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 15**

제 1 항에 있어서,

상기 집적 회로는 비휘발성 메모리 소자들을 통합하는 프로세스 기술을 이용하여 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 16**

제 1 항에 있어서,

상기 집적 회로는 퓨즈 소자들을 통합하는 프로세스 기술을 이용하여 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 17**

제 1 항에 있어서,

상기 집적 회로는 퓨즈-방지 소자(anti-fuse device)들을 통합하는 프로세스 기술을 이용하여 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 18**

제 1 항에 있어서,

상기 집적 회로는 강유전체(ferro-electric) 소자들을 통합하는 프로세스 기술을 이용하여 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 19**

제 1 항에 있어서,

상기 M은 $R\times(k\times N)$으로 결정되고, 여기서 R은 적어도 0.5이고 1.0 이하인, 스위칭 네트워크를 사용하여 프로그

래머블 로직을 구축하기 위한 집적 회로.

**청구항 20**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법으로서,

상기 집적 회로는 다수의 M 수의 제 1 도선들, 다수의 k 세트들의 N 수의 제 2 도선들 ? 상기 제 2 도선들의 수는 k×N 임 ?, 및 스위칭 네트워크(SN)를 포함하며, 여기서 M은 적어도 k+N이고, k는 적어도 2이며, N은 적어도 2이고,

상기 SN은 다수의 스위치들 및 i=[1-NO]에 대해 NO 세트들의 $IO_i$ 수의 도선들을 포함하는 다수의 IO 수의 제 3 도선들을 포함하며, 여기서 NO는 적어도 2이고, N은 적어도 NO이며, IO는 적어도 M과 같고 M이 (k×N) 미만일 때 M과 (k×N) 사이이며, 상기 접속 방법은,

다른 도선의 통과를 요구함이 없이 상기 다수의 스위치들을 통해 p,q=[1-NO]에 대해, 적어도 2개의 상이한 $IO_p$ 수의 도선들 및 $IO_q$ 수의 도선들 각각으로부터 선택된 적어도 2개의 상이한 도선들에 상기 다수의 M 수의 제 1 도선들 중 임의의 2개의 도선들을 선택적으로 각각 연결하는 단계;

상기 SN을 통해 제 1 세트의 (k×(M/NO)) 이하의 수의 도선들에 상기 적어도 2개의 상이한 도선들 중 제 1 도선을 선택적으로 연결하는 단계 ? 상기 제 1 세트의 (k×(M/NO)) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 (M/NO) 이하의 수의 도선들을 가짐 ?; 및

상기 SN을 통해 제 2 세트의 (k×(M/NO)) 이하의 수의 도선들에 상기 적어도 2개의 상이한 도선들 중 제 2 도선을 선택적으로 연결하는 단계 ? 상기 제 2 세트의 (k×(M/NO)) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 상기 제 1 세트의 (k×(M/NO)) 이하의 수의 도선들과 상이한 (M/NO) 이하의 수의 도선들을 가짐 ?

를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 21**

제 20 항에 있어서,

다른 도선의 통과를 요구함이 없이 상기 SN을 통해 상기 $IO_i$ 수의 도선들의 각각의 도선을 k 수의 도선들에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 22**

제 21 항에 있어서,

상기 $IO_i$ 수의 도선들은 (k×N/NO) 수 및 (M/NO) 수 중에서 보다 큰 수 이하이고, 상기 다수의 스위치들은 적어도 [(M-N+1)×N + M×k] 수의 스위치들 및 M×(k+N) 수의 스위치들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 23**

제 21 항에 있어서,

k 수의 프로그래머블 로직 셀들을 더 포함하고,

각각의 상기 k 수의 프로그래머블 로직 셀들은 N 수의 입력 도선들을 가지며, 각각의 상기 k 수의 도선들은 상기 k 수의 프로그래머블 로직 셀들의 상기 N 수의 입력 도선들 중 하나의 도선에 대응하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 24**

제 20 항에 있어서,

상기 M은 R×(k×N)과 같고 여기서 R은 적어도 0.5이고 1.0 이하인, 스위칭 네트워크를 사용하여 프로그래머블

로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 25**

제 20 항에 있어서,

다수의 $I1_i$ 수의 제 4 도선들을 상기 $IO_i$ 수의 도선들에 선택적으로 연결하는 단계 ? 상기 $I1_i$ 수의 도선들은 j=[1-N1]에 대해 N1 그룹들의 $I1_{ij}$ 수의 도선들을 포함하고 여기서 N1은 적어도 2임 ?;

다른 도선의 통과를 요구함이 없이 상기 다수의 스위치들을 통해 상기 $I1_i$ 수의 도선들 중 2개의 상이한 도선들에 상기 $IO_i$ 수의 도선들 중 임의의 2개의 도선들을 선택적으로 연결하는 단계 ? 상기 2개의 상이한 도선들 중 제 3 도선은 $I1_{ir}$ 수의 도선들로부터 나오고, 상기 2개의 상이한 도선들 중 제 4 도선은 $I1_{is}$ 수의 도선들로부터 나오며, 상기 r은 r,s=[1-N1]에 대한 s와 상이함 ?;

상기 SN을 통해 상기 k 세트들의 N 수의 도선들로부터 제 3 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들에 상기 2개의 상이한 도선들 중 상기 제 3 도선을 선택적으로 연결하는 단계 ? 상기 제 3 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 (M/(NO×N1)) 이하의 수의 도선들을 가짐 ?; 및

상기 SN을 통해 상기 k 세트들의 N 수의 도선들로부터 제 4 세트의 (k×(M/(NO×N1))) 수의 도선들에 상기 2개의 상이한 도선들 중 상기 제 4 도선을 선택적으로 연결하는 단계 ? 상기 제 4 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들은 각각의 상기 k 세트들의 N 수의 도선들로부터 상기 제 3 세트의 (k×(M/(NO×N1))) 이하의 수의 도선들과 상이한 (M/(NO×N1)) 이하의 수의 도선들을 가짐 ?

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 26**

제 25 항에 있어서,

다른 도선의 통과를 요구함이 없이 상기 $I1_{ij}$ 수의 도선들의 각각의 도선을 k 수의 도선들에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 27**

제 26 항에 있어서,

k 수의 프로그래머블 로직 셀들을 더 포함하고,

각각의 상기 k 수의 프로그래머블 로직 셀들은 N 수의 입력 도선들을 가지며, 각각의 상기 k 수의 도선들은 상기 k 수의 프로그래머블 로직 셀들의 상기 N 수의 입력 도선들 중 하나의 도선에 대응하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 28**

제 26 항에 있어서,

상기 $I1_{ij}$ 수는 (k×N/(NO×N1)) 수 및 (M/(NO×N1)) 수 중에서 보다 큰 수 이하이고, 상기 다수의 스위치들은 [M×(NO+N1+k)] 이하의 수의 스위치들을 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로의 접속 방법.

**청구항 29**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로로서,

k 이하의 수의 기능 블록들; 및

확장성(scalable) 스위칭 네트워크(SN)

를 포함하고, 상기 SN은,

다수의 N 세트들의 제 1 스위치들 ? 여기서 N은 1보다 더 큼 ?; 및

M 수의 핀들을 갖는 제 1 포트 ? 여기서 M은 (k×N) 이하임 ?

를 포함하며, 상기 SN은 제 1 세트의 I 수의 도선들을 더 포함하고, 여기서 I는 M 내지 (k×N) 범위이며, 상기 I 수의 도선들은 j={1 내지 N}에 대해 N 세트들의 $I_j$ 수의 도선들로 분할되고, 여기서 $I_j$는 (1/N)과 같으며, 상기 제 1 포트의 (M-N+1) 내지 M 수의 핀들은 다른 도선의 통과를 요구함이 없이 상기 다수의 N 세트들의 제 1 스위치들의 각 세트의 스위치들을 통해 상기 제 1 세트의 I 수의 도선들 중 각각의 상기 N 세트들의 $I_j$ 수의 도선들과 선택적으로 연결되고, 여기서 상기 각 세트의 스위치들은 (M-N+1) 내지 M 범위의 X 수의 스위치들을 가지며,

상기 SN은 다수의 N 세트들의 제 2 스위치들 및 다수의 N 수의 제 2 포트들을 더 포함하고, 각각의 상기 다수의 N 수의 제 2 포트들은 k 이하의 수의 핀들을 가지며, k는 적어도 j={1 내지 N}에 대한 가장 큰 $I_j$와 같고,

각각의 상기 N 세트들의 $I_j$ 수의 도선들 중 상기 $I_j$ 수의 도선들은 다른 도선의 통과를 요구함이 없이 상기 다수의 N 세트들의 제 2 스위치들의 각 세트의 스위치들을 통해 상기 다수의 N 수의 제 2 포트들의 각 포트의 k 이하의 수의 핀들에 선택적으로 연결하도록 구성되며,

상기 각 세트의 스위치들은 $I_j \times k$ 이하의 수의 스위치들을 갖고, 상기 다수의 N 수의 제 2 포트들의 각 포트에 있는 각각의 상기 k 이하의 수의 핀들은 상기 k 이하의 수의 기능 블록들의 N 수의 핀들의 각각의 핀에 물리적으로 접속되는,

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 30**

제 29 항에 있어서,

각각의 상기 기능 블록들은 실질적으로 논리적으로 동일하며, 각각의 상기 기능 블록들은 k에 대해 N 이하의 수의 핀들을 갖는 포트를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 31**

제 29 항에 있어서,

상기 SN의 제 1 포트의 핀들은 단지 순서대로(exclusively in sequence), 상기 SN의 다수의 N 세트들의 제 1 스위치들, 상기 제 1 세트의 I 수의 도선들 및 상기 다수의 N 세트들의 제 2 스위치들을 통하여 상기 다수의 N 수의 제 2 포트들의 핀들과 선택적으로 연결하도록 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 32**

제 29 항에 있어서,

상기 스위치들은 프로그램 제어되는 패스게이트들을 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 33**

제 29 항에 있어서,

상기 스위치들은 프로그램 제어되는 드라이버들/리시버들을 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 34**

제 29 항에 있어서,

상기 스위치들은 프로그램 제어되는 패스게이트들 및 프로그램 제어되는 드라이버들/리시버들을 포함하는, 스위

칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 35**

제 29 항에 있어서,

상기 스위치들 중 적어도 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 36**

제 29 항에 있어서,

상기 스위치들 중 적어도 하나는 단일의 퓨즈-방지 소자로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 37**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법으로서,

k 이하의 수의 기능 블록들을 제공하는 단계 ? 각각의 상기 기능 블록들은 실질적으로 논리적으로 동일하고, 각각의 기능 블록은 N 이하의 수의 핀들을 갖는 포트를 포함하며, 여기서 k와 N은 각각 1보다 더 큼 ?;

다수의 N 세트들의 제 1 스위치들 및 (k×N) 이하와 같은 M에 대해 M 수의 핀들을 갖는 제 1 포트를 확장성 스위칭 네트워크(SN)에 제공하는 단계;

제 1 세트의 I 수의 도선들을 상기 SN에 제공하는 단계 ? 여기서 I는 M 내지 k×N 범위이고, 상기 I 수의 도선들은 j={1 내지 N}에 대해 N 세트들의 $I_j$ 수의 도선들로 분할되며, 여기서 $I_j$는 (1/N)과 같음 ?;

다른 도선의 통과를 요구함이 없이 상기 다수의 N 세트들의 제 1 스위치들의 각 세트의 스위치들을 통해 상기 제 1 세트의 I 수의 도선들 중 각각의 상기 N 세트들의 $I_j$ 수의 도선들에 상기 제 1 포트의 (M-N+1) 내지 M 수의 핀들을 선택적으로 연결하는 단계 ? 여기서 상기 각 세트의 스위치들은 (M-N+1) 내지 M 범위의 X 수의 스위치들을 가짐 ?;

다수의 N 세트들의 제 2 스위치들 및 다수의 N 수의 제 2 포트들을 상기 SN에 제공하는 단계 ? 각각의 N 포트들은 k 이하의 수의 핀들을 갖고, 여기서 k는 j={1 내지 N}에 대한 가장 큰 $I_j$와 적어도 같음 ?;

다른 도선의 통과를 요구함이 없이 상기 다수의 N 세트들의 제 2 스위치들의 각 세트의 스위치들을 통해 상기 다수의 N 수의 제 2 포트들의 각 포트의 k 이하의 수의 핀들에 각각의 상기 N 세트들의 $I_j$ 수의 도선들의 $I_j$ 수의 도선들을 선택적으로 연결하는 단계 ? 상기 각 세트의 스위치들은 $I_j \times k$ 이하의 수의 스위치들을 가짐 ?; 및

상기 k 이하의 수의 다수의 제 1 기능 블록들의 상기 k 이하의 수의 기능 블록들의 N 수의 핀들의 각 핀에 상기 다수의 N 수의 제 2 포트들의 각 포트의 각각의 k 이하의 수의 핀들을 물리적으로 접속하는 단계

를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 38**

제 37 항에 있어서,

단지 순서대로 상기 SN의 상기 다수의 N 세트들의 제 1 스위치들, 상기 제 1 세트의 I 수의 도선들 및 상기 다수의 N 세트들의 제 2 스위치들을 통하여 상기 다수의 N 수의 제 2 포트들의 핀들에 상기 SN의 제 1 포트의 핀들을 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 39**

제 37 항에 있어서,

상기 스위치들 중 적어도 하나는 드라이버/리시버 및 프로그램 제어되는 패스게이트를 포함하는, 스위칭 네트워

크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 40**

제 37 항에 있어서,

상기 스위치들 중 적어도 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 41**

제 37 항에 있어서,

상기 스위치들 중 적어도 하나는 단일의 퓨즈-방지 소자로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 42**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템으로서,

M 수의 핀들을 갖는 제 1 포트, 각각의 N 수의 포트들에 k 이하의 핀들을 갖는 다수의 N 수의 제 2 포트들, j 수의 정수(integer)들 Nj, 다수의 j 수의 스위치들 및 (j+1) 세트들의 도선들을 포함하는 확장성 스위칭 네트워크(SN)

를 포함하며, 여기서 N1은 1과 같고, Nj는 1보다 더 큰 j에 대해 1보다 더 크며, 각각의 j 세트들은 2보다 더 큰 j에 대해 M 내지 (k×N) 범위의 $I_j$에 대한 $I_j$ 수의 도선들을 갖고, 상기 다수의 N 수의 제 2 포트들의 각각의 핀은 상기 (j+1) 세트들의 도선들 및 상기 다수의 j 수의 스위치들을 통해서만 상기 제 1 포트의 적어도 (M-N2+1) 핀들에 선택적으로 연결되며,

상기 제 1 포트의 각각의 핀은 I1 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들 중 제 1 세트의 도선들의 각 도선에 물리적으로 접속되고, 상기 (j+1) 세트들의 도선들 중 (j+1)번째 세트의 도선들은 k×N 수의 도선들을 가지며, I1은 M과 같고,

I2 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들 중 제 2 세트의 도선들은 i = {1 내지 N2}에 대해 각각의 세트가 I2(i) 수의 도선들을 갖는 N2 세트들의 도선들로 분할되며, 상기 제 1 세트의 도선들 중 (I1-N2+1) 내지 I1 수의 도선들은 상기 다수의 j 수의 스위치들의 다수의 제 1 스위치들 중 X 수의 스위치들을 통해 상기 제 2 세트의 I2 수의 도선들 중 각각의 I2(i) 수의 도선들에 선택적으로 연결되고, 여기서 X는 (I1-N2+1) 내지 I1 범위이며,

상기 (j+1) 세트들의 도선들 중 제 3 세트의 도선들은 q = {1 내지 N2}에 대해 N2 세트들의 I3(q) 수의 도선들로 분할되고, 여기서 각각의 I3(q)는 I3/N2이며, 각각의 I3(q) 수의 도선들은 s = {1 내지 N3}에 대해 N3 세트들의 I3(q,s) 수의 도선들로 추가로 분할되고, 여기서 각각의 I3(q,s)는 I3/N2/N3이며, 상기 제 2 세트의 I2 수의 도선들 중 (I2(i)-N3+1) 내지 I2(i) 수의 도선들은 상기 다수의 j 수의 스위치들의 다수의 제 2 스위치들 중 Y 수의 스위치들을 통해 r = {1 내지 N3}에 대해 상기 제 3 세트의 도선들 중 각각의 I3(i,r) 수의 도선들에 선택적으로 연결되고, Y는 (I2(i)-N3+1) 내지 I2(i) 범위인,

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 43**

제 42 항에 있어서,

N2 및 N3은 N의 정수 약수(integer divisor)들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 44**

제 43 항에 있어서,

각각의 상기 I3(i,r) 수의 도선들은 상기 j 세트들의 스위치들의 제 3 세트의 스위치들 중 I3(i,r)× k 이하의 수의 스위치들을 통해 상기 (j+1) 세트들의 도선들의 제 4 세트의 도선들 중 각각의 k 이하의 수의 도선들에 선

택적으로 연결하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 45**

제 44 항에 있어서,

각각의 상기 k 이하의 수의 도선들은 상기 다수의 N 수의 제 2 포트들 중 하나의 포트의 각각의 k 이하의 핀들에 물리적으로 접속되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 46**

제 45 항에 있어서,

k 이하의 수의 기능 블록들을 더 포함하고, 각각의 기능 블록은 각각의 포트의 N 이하의 핀들을 가진 적어도 하나의 포트를 가지며, 상기 다수의 N 수의 제 2 포트들 중 하나의 포트의 k 이하의 수의 핀들의 각각의 핀은 상기 k 이하의 수의 기능 블록들 중 하나의 상기 적어도 하나의 포트의 각각의 핀에 물리적으로 접속되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 47**

제 46 항에 있어서,

상기 전자 시스템은 집적 회로로 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 48**

제 47 항에 있어서,

상기 스위치들 중 적어도 하나는 드라이버/리시버 및 프로그램 제어되는 패스게이트를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 49**

제 47 항에 있어서,

상기 스위치들 중 적어도 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 50**

제 47 항에 있어서,

상기 스위치들 중 적어도 하나는 단일의 퓨즈-방지 소자로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 51**

제 42 항에 있어서,

I4 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들의 후속하는 제 4 세트의 도선들은 t = {1 내지 N4}에 대해 N2×N3×N4 세트들의 I4(q,s,t) 수의 도선들로 순차적으로 분할되고, 각각의 I4(q,s,t)는 I4/N2/N3/N4이며, 상기 제 3 세트의 I3 수의 도선들 중 (I3(i,r)-N4+1) 내지 I3(i,r) 수의 도선들은 상기 다수의 j 수의 스위치들의 다수의 제 3 스위치들 중 Z 수의 스위치들을 통해 v = {1 내지 N4}에 대해 상기 제 3 세트의 도선들 중 각각의 I4(i,r,v) 수의 도선들에 선택적으로 연결되고, Z는 (I3(i,r)-N4+1) 내지 I3(i,r) 범위인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 52**

제 51 항에 있어서,

N2, N3 및 N4는 N의 정수 약수들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스

템.

**청구항 53**

제 52 항에 있어서,

상기 전자 시스템은 집적 회로로 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 54**

제 53 항에 있어서,

상기 스위치들 중 적어도 하나는 드라이버/리시버 및 프로그램 제어되는 패스게이트를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 55**

제 53 항에 있어서,

상기 스위치들 중 적어도 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 56**

제 53 항에 있어서,

상기 스위치들 중 적어도 하나는 단일의 퓨즈-방지 소자로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템.

**청구항 57**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법으로서,

M 수의 핀들을 갖는 제 1 포트, 각각의 N 수의 포트들에 k 이하의 핀들을 갖는 다수의 상기 N 수의 제 2 포트들, j 수의 정수들 Nj, 다수의 j 수의 스위치들 및 (j+1) 세트들의 도선들을 구비한 확장성 스위칭 네트워크(SN)를 제공하는 단계 ? N1은 1과 같고, Nj는 1보다 더 큰 j에 대해 1보다 더 크며, 각각의 j 세트들은 2보다 더 큰 j에 대해 M 내지 k×N 범위의 Ij에 대해 Ij 수의 도선들을 가짐 ?;

상기 (j+1) 세트들의 도선들 및 상기 다수의 j 수의 스위치들을 통해서만 상기 제 1 포트의 적어도 (M-N2+1) 핀들에 상기 다수의 N 수의 제 2 포트들의 각각의 핀을 선택적으로 연결하는 단계;

I1 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들 중 제 1 세트의 도선들의 각각의 도선에 상기 제 1 포트의 각각의 핀을 물리적으로 접속하는 단계 ? 상기 (j+1) 세트들의 도선들 중 (j+1)번째 세트의 도선들은 k×N 수의 도선들을 갖고, I1은 M과 같음 ?;

I2 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들 중 제 2 세트의 도선들을 각각의 세트가 i = {1 내지 N2}에 대해 I2(i) 수의 도선들을 갖는 N2 세트들의 도선들로 분할하는 단계 ? 각각의 I2(i)는 I2/N2이고, 상기 제 1 세트의 도선들 중 (I1-N2+1) 내지 I1 수의 도선들은 상기 다수의 j 수의 스위치들의 다수의 제 1 스위치들 중 X 수의 스위치들을 통해 상기 제 2 세트의 I2 수의 도선들의 각각의 I2(i) 수의 도선들에 선택적으로 연결됨 ?; 및

I3 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들 중 제 3 세트의 도선들을 q = {1 내지 N2}에 대해 N2 세트들의 I3(q) 수의 도선들로 먼저 분할하고, 각각의 I3(q) 수의 도선들을 N3 세트들의 I3(q,s) 수의 도선들로 추가로 분할하는 단계 ? 여기서 각각의 I3(q)는 I3/N2이고, 각각의 I3(q,s)는 I3/N2/N3이며, 상기 제 2 세트의 I2 수의 도선들 중 (I2(i)-N3+1) 내지 I2(i) 수의 도선들은 상기 다수의 j 수의 스위치들의 다수의 제 2 스위치들 중 Y 수의 스위치들을 통해 r = {1 내지 N3}에 대해 상기 제 3 세트의 도선들의 각각의 I3(i,r) 수의 도선들에 선택적으로 연결되고, 여기서 Y는 (I2(i)-N3+1) 내지 I2(i)임 ?

를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 58**

제 57 항에 있어서,

N2 및 N3는 N의 정수 약수들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 59**

제 58 항에 있어서,

각각의 상기 I3(i,r) 수의 도선들은 상기 j 세트들의 스위치들의 제 3 세트의 스위치들 중 I3(i,r)×k 이하의 수의 스위치들을 통해 상기 (j+1) 세트들의 도선들의 제 4 세트의 도선들 중 각각의 k 이하의 수의 도선들에 선택적으로 연결하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 60**

제 59 항에 있어서,

각각의 상기 k 이하의 수의 도선들은 상기 다수의 N 수의 제 2 포트들 중 하나의 포트의 각각의 k 이하의 수의 핀들에 물리적으로 접속되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 61**

제 60 항에 있어서,

k 이하의 수의 기능 블록들을 더 포함하고, 각각의 기능 블록은 각각의 포트에 N 이하의 수의 핀들을 갖는 적어도 하나의 포트를 가지며, 상기 다수의 N 수의 제 2 포트들 중 하나의 포트의 k 이하의 수의 핀들의 각각의 핀은 상기 k 이하의 수의 기능 블록들 중 하나의 적어도 하나의 포트의 각각의 핀에 물리적으로 접속되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 62**

제 61 항에 있어서,

상기 전자 시스템은 집적 회로로 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 63**

제 62 항에 있어서,

상기 스위치들 중 적어도 하나는 드라이버/리시버 및 프로그램 제어되는 패스게이트를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 64**

제 62 항에 있어서,

상기 스위치들 중 적어도 하나는 프로그램 제어되는 온 상태 및 오프 상태를 갖는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 65**

제 62 항에 있어서,

상기 스위치들 중 적어도 하나는 단일의 퓨즈-방지 소자로 구성되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 66**

제 57 항에 있어서,

I4 수의 도선들을 갖는 상기 (j+1) 세트들의 도선들의 후속하는 제 4 세트의 도선들을 제공하여 t = {1 내지 N4}에 대해 N2×N3×N4 세트들의 I4(q,s,t) 수의 도선들로 순차적으로 분할하는 단계 ? 여기서 각각의 I4(q,s,t)는 I4/N2/N3/N4 임 ?; 및

상기 다수의 j 수의 스위치들의 다수의 제 3 스위치들 중 Z 수의 스위치들을 통해 v = {1 내지 N4}에 대해 상기 제 3 세트의 도선들의 각각의 I4(i,r,v) 수의 도선들에 상기 제 3 세트의 I3 수의 도선들 중 (I3(i,r)-N4+1) 내지 I3(i,r) 수의 도선들을 선택적으로 연결하는 단계 ? 여기서 Z는 (I3(i,r)-N4+1) 내지 I3(i,r) 범위임 ?

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 67**

제 66 항에 있어서,

N2, N3 및 N4는 N의 정수 약수들인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 68**

제 67 항에 있어서,

상기 전자 시스템은 집적 회로로 구현되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 전자 시스템을 제공하는 방법.

**청구항 69**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로로서,

상기 집적 회로는 제 1 스위칭 네트워크(SN)를 포함하고,

상기 제 1 SN은,

다수의 M 수의 제 1 도선들;

다수의 N 세트들의 k 수의 제 2 도선들 ? 여기서 N은 2보다 더 크고 k는 2보다 더 큼 ?;

다수의 N 세트들의 q 수의 제 4 도선들을 포함하는 다수의 I 수의 제 3 도선들 ? 여기서 q는 M/N과 같음 ?;

다수의 제 1 스위치들 ? 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 상기 다수의 I 수의 제 3 도선들에 선택적으로 연결되고, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 각각의 상기 다수의 N 세트들의 q 수의 제 4 도선들의 q 수의 도선들 중 하나에 선택적으로 연결됨 ?; 및

다수의 제 2 스위치들 ? 상기 다수의 제 2 스위치들을 통하여, 상기 다수의 N 세트들의 q 수의 제 4 도선들의 각 세트의 각각의 q 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 다수의 N 세트들의 k 수의 제 2 도선들의 각 세트 중 하나의 세트의 모든 도선들에 선택적으로 연결됨 ?

을 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 70**

제 69 항에 있어서,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 각각의 상기 다수의 N 세트들의 q 수의 제 4 도선들의 q 수의 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 71**

제 69 항에 있어서,

상기 다수의 I 수의 제 3 도선들의 다수의 D1 세트들의 I1 수의 제 5 도선들을 더 포함하고,

D1은 적어도 2이며,

상기 다수의 N 세트들의 q 수의 제 4 도선들은 q 수의 도선의 N/D1 세트들 중 D1 세트들로 분할되고,

N/D1은 적어도 2이며,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D1 세트들의 I1 수의 제 5 도선들의 D1 세트들의 각 세트의 각각의 상기 I1 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 q 수의 도선들의 N/D1 세트들 중 D1 세트들의 상기 q 수의 도선들의 N/D1 세트들의 각 세트의 q 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 72**

제 71 항에 있어서,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 각각의 상기 다수의 D1 세트들의 I1 수의 제 5 도선들의 I1 수의 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 73**

제 71 항에 있어서,

상기 다수의 I 수의 제 3 도선들의 다수의 D2 세트들의 I2 수의 제 6 도선들을 더 포함하고,

D2는 적어도 2이며,

상기 다수의 D1 세트들의 I1 수의 제 5 도선들은 I1 수의 도선들의 D1/D2 세트들 중 D2 세트들로 분할되고,

D1/D2는 1보다 더 크며,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D2 세트들의 I2 수의 제 6 도선들의 D2 세트들 중 각 세트의 각각의 I2 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 I1 수의 도선들의 D1/D2 세트들 중 D2 세트들의 I1 수의 도선들의 D1/D2 세트들의 각 세트의 I1 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 74**

제 73 항에 있어서,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 각각의 상기 다수의 D2 세트들의 I2 수의 제 6 도선들의 I2 수의 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 75**

제 73 항에 있어서,

상기 다수의 I 수의 제 3 도선들의 다수의 D3 세트들의 I3 수의 제 7 도선들을 더 포함하고,

D3은 적어도 2이며,

상기 다수의 D2 세트들의 I2 수의 제 6 도선들은 I2 수의 도선들의 D2/D3 세트들 중 D3 세트들로 분할되고,

D2/D3는 1보다 더 크며,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D3 세트들의 I3 수의 제 7 도선들의 D3 세트들 중 각 세트의 각각의 I3 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 I2 수의 도선들의 D2/D3 세트들 중 D3 세트들의 I2 수의 도선들의 D2/D3 세트들 중 각 세트의 I2 도선들 중 하나에 선택적으로 연결되는, 스위

칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 76**

제 75 항에 있어서,

상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들은 다른 도선의 임의의 선택가능한 접속을 요구함이 없이 각각의 상기 다수의 D3 세트들의 I3 수의 제 7 도선들의 I3 수의 도선들 중 하나에 선택적으로 연결되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 77**

제 70 항에 있어서,

N은 4보다 더 크고 k는 4보다 더 큰, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 78**

제 70 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 79**

제 72 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워트(SN)들을 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 80**

제 74 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 81**

제 76 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 82**

제 78 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 제 1 영역 내에 실질적으로 근접하게 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 83**

제 79 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 84**

제 80 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 85**

제 81 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 86**

제 82 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 87**

제 83 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 88**

제 84 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 89**

제 85 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 90**

제 70 항에 있어서,

상기 제 1 SN과 실질적으로 유사한 제 2 스위칭 네트워크를 더 포함하고,

각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각 세트의 k 수의 포트들을 형성하며,

상기 세트의 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선들은 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 접속되는,

집적 회로.

**청구항 91**

제 72 항에 있어서,

상기 제 1 SN과 실질적으로 유사한 제 2 스위칭 네트워크를 더 포함하고,

각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각 세트의 k 수의 포트들을 형성하며,

상기 세트의 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선들은 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 접속되는,

집적 회로.

**청구항 92**

제 74 항에 있어서,

상기 제 1 SN과 실질적으로 유사한 제 2 스위칭 네트워크를 더 포함하고,

각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각 세트의 k 수의 포트들을 형성하며,

상기 세트의 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선들은 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 접속되는,

집적 회로.

**청구항 93**

제 76 항에 있어서,

상기 제 1 SN과 실질적으로 유사한 제 2 스위칭 네트워크를 더 포함하고,

각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각 세트의 k 수의 포트들을 형성하며,

상기 세트의 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선들은 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 접속되는,

집적 회로.

**청구항 94**

제 90 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 95**

제 91 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 96**

제 92 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 97**

제 93 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 집적 회로.

**청구항 98**

스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법으로서,

집적 회로에 제 1 스위칭 네트워크(SN)를 제공하는 단계;

상기 제 1 SN에 다수의 M 수의 제 1 도선들을 제공하는 단계;

상기 제 1 SN에 다수의 N 세트들의 k 수의 제 2 도선들을 제공하는 단계 ? 여기서 N은 2보다 더 크고 k는 2보다 더 큼 ?;

다수의 N 세트들의 q 수의 제 4 도선들을 제공하는 단계를 포함하는, 상기 제 1 SN에 다수의 I 수의 제 3 도선들을 제공하는 단계 ? 여기서 q는 M/N 임 ?;

상기 제1 SN에 다수의 제 1 스위치들을 제공하는 단계;

상기 다수의 제 1 스위치들을 통해 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 상기 다수의 I 수의 제 3 도선들에 선택적으로 연결하는 단계;

상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 각각의 상기 다수의 N 세트들의 q 수의 제 4 도선들의 q 수의 도선들 중 하나에 선택적으로 연결하는 단계;

상기 제 1 SN에 다수의 제 2 스위치들을 제공하는 단계; 및

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 각 세트의 상기 다수의 N 세트들의 q 수의 제 4 도선들의 각각의 q 수의 도선들을 상기 다수의 N 세트들의 k 수의 제 2 도선들의 각 세트 중 하나의 세트의 모든 도선들에 선택적으로 연결하는 단계

를 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 99**

제 98 항에 있어서,

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 각각의 상기 다수의 N 세트들의 q 수의 제 4 도선들의 q 수의 도선들 중 하나에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 100**

제 98 항에 있어서,

상기 제 1 SN에 상기 다수의 I 수의 제 3 도선들의 다수의 D1 세트들의 I1 수의 제 5 도선들을 제공하는 단계 ? 여기서 D1은 적어도 2 임 ?;

상기 다수의 N 세트들의 q 수의 제 4 도선들을 q 수의 도선의 N/D1 세트들 중 D1 세트들로 분할하는 단계 ? 여기서 N/D1은 적어도 2 임 ?; 및

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D1 세트들의 I1 수의 제 5 도선들의 D1 세트들의 각 세트의 각각의 I1 수의 도선들을 상기 q 수의 도선들의 N/D1 세트들 중 D1 세트들의 q 수의 도선들의 N/D1 세트들의 각 세트의 q 수의 도선들 중 하나에 선택적으로 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 101**

제 100 항에 있어서,

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 각각의 상기 다수의 D1 세트들의 I1 수의 제 5 도선들의 I1 수의 도선들 중 하나에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 102**

제 100 항에 있어서,

상기 다수의 I 수의 제 3 도선들의 다수의 D2 세트들의 I2 수의 제 6 도선들을 상기 제 1 SN에 제공하는 단계 ? 여기서 D2는 적어도 2 임 ?;

상기 다수의 D1 세트들의 I1 수의 제 5 도선들을 I1 수의 도선들의 D1/D2 세트들 중 D2 세트들로 분할하는 단계 ? 여기서 D1/D2는 1보다 더 큼 ?; 및

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D2 세트들의 I2 수의 제 6 도선들의 D2 세트들의 각 세트의 각각의 I2 수의 도선들을 상기 I1 수의 도선들의 D1/D2 세트들 중 D2 세트들의 I1 수의 도선들의 각각의 D1/D2 세트들의 I1 수의 도선들 중 하나에 선택적으로 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 103**

제 102 항에 있어서,

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 각각의 상기 다수의 D2 세트들의 I2 수의 제 6 도선들의 I2 수의 도선들 중 하나에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 104**

제 102 항에 있어서,

상기 다수의 I 수의 제 3 도선들의 다수의 D3 세트들의 I3 수의 제 7 도선들을 상기 제 1 SN에 제공하는 단계 ? 여기서 D3은 적어도 2 임 ?;

상기 다수의 D2 세트들의 I2 수의 제 6 도선들을 I2 수의 도선들의 D2/D3 세트들 중 D3 세트들로 분할하는 단계 ? 여기서 D2/D3는 1보다 더 큼 ?; 및

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 D3 세트들의 I3 수의 제 7 도선들의 D3 세트들의 각 세트의 각각의 I3 수의 도선들을 상기 I2 수의 도선들의 D2/D3 세트들 중 D3 세트들의 I2 수의 도선들의 각각의 D2/D3 세트들의 I2 수의 도선들 중 하나에 선택적으로 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 105**

제 104 항에 있어서,

다른 도선의 임의의 선택가능한 접속을 요구함이 없이 상기 다수의 제 1 스위치들을 통하여, 상기 다수의 M 수의 제 1 도선들의 각각의 M 수의 도선들을 각각의 상기 다수의 D3 세트들의 I3 수의 제 7 도선들의 I3 수의 도선들 중 하나에 선택적으로 연결하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 106**

제 99 항에 있어서,

N은 4보다 더 크고 k는 4보다 더 큰, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 107**

제 99 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 제공하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 108**

제 101 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 제공하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 109**

제 103 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 제공하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 110**

제 105 항에 있어서,

각각 상기 제 1 SN과 실질적으로 유사한 다수의 스위칭 네트워크(SN)들을 제공하는 단계를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 111**

제 107 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 제 1 영역 내에 실질적으로 근접하게 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 112**

제 108 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 113**

제 109 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 114**

제 110 항에 있어서,

각각의 상기 다수의 스위칭 네트워크들에 대한 상기 다수의 제 1 스위치들, 상기 다수의 제 2 스위치들 및 상기 다수의 I 수의 제 3 도선들은 근접한 제 1 영역 내에 실질적으로 위치되는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 115**

제 111 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 116**

제 112 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 117**

제 113 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 118**

제 114 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 119**

제 99 항에 있어서,

상기 제 1 SN과 실질적으로 유사한 제 2 스위칭 네트워크를 제공하는 단계 ? 각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각각의 k 수의 포트들을 형성함 ?; 및

상기 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선을 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 120**

제 101 항에 있어서,

제 2 스위칭 네트워크를 제공하는 단계 ? 각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각각의 k 수의 포트들을 형성함 ?; 및

상기 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선을 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 121**

제 103 항에 있어서,

제 2 스위칭 네트워크를 제공하는 단계 ? 각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각각의 k 수의 포트들을 형성함 ?; 및

상기 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선을 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 122**

제 105 항에 있어서,

제 2 스위칭 네트워크를 제공하는 단계 ? 각각의 N 세트들의 도선들의 각각의 k 수의 도선들은 각각의 k 수의 포트들을 형성함 ?; 및

상기 k 수의 포트들 중 제 1 포트의 각각의 k 수의 도선을 상기 제 2 스위칭 네트워크의 상기 다수의 M 수의 제 1 도선들에 연결하는 단계

를 더 포함하는, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 123**

제 119 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 124**

제 120 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 125**

제 121 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

**청구항 126**

제 122 항에 있어서,

상기 집적 회로는 필드 프로그래머블 게이트-어레이인, 스위칭 네트워크를 사용하여 프로그래머블 로직을 구축하기 위한 방법.

# *명 세 서*

## *기 술 분 야*

[0001] 본 발명의 실시예들은 스위칭 네트워크들에 관한 것으로서, 특히 프로그래머블 로직 회로(programmable logic circuit)들에 사용되는 스위칭 네트워크들에 관한 것이다.

## *배 경 기 술*

[0002] 필드 프로그래머블 게이트 어레이(FPGA)로도 지칭되는 프로그래머블 로직 회로는 로직 기능들을 수행하도록 사용자에 의해 프로그래밍될 수 있는 종래의 집적된 로직 회로이다. 회로 설계자들은 목표된 로직 기능들을 정의하고 이에 따라 회로는 신호들을 처리하도록 프로그래밍된다. 로직 밀도 요구조건들과 제조 볼륨(volume)들에 따라, 프로그래머블 로직 회로들은 시간과 비용 면에서 시장의 우수한 대안들이다. 통상적인 프로그래머블 로직 회로는 각각의 로직 셀들이 그 입력 변수들에서 로직 기능들을 수행하도록 프로그래밍될 수 있는 로직 셀들로 구성된다. 부가적으로, 상호접속 리소스들은 사용자 스펙(specification)에 따라 로직 셀들의 출력들로부터 로직 셀들의 입력들로 신호들을 전달하도록 프로그래밍될 수 있는 프로그래머블 로직 회로를 통해 제공된다.

[0003] 보다 대형으로 보다 지능화된 프로그래머블 로직 회로들을 허용하기 위해 기술이 진보함에 따라, 회로에서 로직 셀들의 수 및 요구되는 상호접속 리소스들이 증가된다. 회로 사이즈를 작게 유지하기 위해 로직 셀들의 수 및 상호접속 리소스들을 증가시키는 것이 필요하다. 요구되는 회로 사이즈를 최소화하는 한가지 방법은 일정 레벨의 접속성(connectivity)을 유지하면서 상호접속 리소스들을 최소화하는 것이다. 따라서, 칩에서 구현되는 기능이 많아짐에 따라 다수의 신호들을 접속시키는데 요구되는 상호접속 리소스들이 빨리 소모될 수 있다는 것을 알 수 있다. 트레이드-오프(trade-off)들은 회로 사이즈를 작게 유지하면서 회로의 로직 셀들의 사용을 줄이거나 회로 사이즈를 급격히 증가시킬 수 있는 더 많은 라우팅 리소스들을 제공하는 것이다.

[0004] 프로그래머블 로직 회로 분야에서 지난 40년간 매우 복잡한 커넥션 스타일들이 개발되어 왔다. 1965년 L.M. Spandorfer는 근접 상호접속을 이용하는 프로그래머블 로직 회로의 가능한 구현, 및 Clos 네트워크의 스위치들을 이용하는 다수의 도선(conductor)들을 통한 접속들을 기술한다. 1970년 박사 논문에서 R.G. Shoup는 근접 상호접속의 사용 및 더 긴 거리의 상호접속을 위한 버스의 사용을 기술한다.

[0005] 1989년의 미국특허 제4,870,302호에서 Freeman은 근접 상호접속들, 근거리(길이 1, 싱글로 지칭됨) 상호접속들, 및 클럭들과 같은 신호들을 위한 광역 라인들을 이용하여 FPGA의 상업적 구현을 기술한다. 근거리 상호접속들은 로직 셀들의 입력들 및 출력들과 상호작용하고, 그 각각의 입력은 스위치들을 통해 로직 셀에 인접한 모든 짧은 배선에 접속되며, 수평 및 수직의 짧은 배선들은 접합부의 스위치 박스를 통해 접속된다. 미국특허 제4,758,745호의 El Gamal 외는 세그먼트로 나뉜 라우팅을 소개하고, 여기서 로직 셀들의 입력들 및 출력들은 일 영역의 상이한 길이들의 라우팅 세그먼트들과 상호작용한다.

[0006] 미국특허 제5,260,610호의 Peterson 외 및 미국특허 제5,260,611호의 Cliff 외는 로직 엘리먼트들의 세트와 인터페이싱하는 도선들의 로컬 세트를 소개하고, 여기서 로직 엘리먼트들의 모든 입력은 스위치들을 통해 상기 세트의 모든 로컬 도선에 접속되며; 부가적인 칩 길이 도선들은 수평 및 수직으로 도입되고, 수평 도선은 수직 도선들에 접속될 수 있으며, 수평 도선들은 다수의 로컬 도선들에 접속된다. 미국특허 제4,870,302호, 미국특허 제4,758,745호, 미국특허 제5,260,610호 및 미국특허 제5,260,611호에서, 로직 셀의 입력 도선은 로컬 도선들의 세트에 대한 전체 커넥션들을 갖는다(예, n-입력들 및 k-로컬 도선들에 대해, 입력들을 로컬 도선들에 연결하는 n×k 스위치들이 있다). 멀티플렉서(MUX) 수단이 트랜지스터들의 수를 감소시키기 위해 사용될 수 있다. 미국특허 제4,870,302호, 미국특허 제4,758,745호, 미국특허 제5,260,610호, 및 미국특허 제5,260,611호에서, 일반적인 상호접속 리소스들은 하나 또는 둘의 상이한 길이들로 제한되거나(즉, 미국특허 제4,870,302호의 신호들, 미국특허 제5,260,610호 및 미국특허 제5,260,611호의 로컬 길이 및 칩 길이), 일 영역으로 제한된다(즉, 미국특허 제4,758,745호의 수평으로 상이한 길이들, 미국특허 제5,260,610호 및 미국특허 제5,260,611호에서 수직의 로컬 길이들).

[0007] 미국특허 제5,144,166호의 Camarota 및 미국특허 제5,469,003호의 Kean은 도선들의 영향으로 제한되는 두 영역들의 둘 이상의 상이한 길이들을 갖는 라우팅 수단을 소개한다. 미국특허 제5,144,166호는 각각의 배선이 하나 이상의 가능한 구동 소스에 의해 선택적으로 구동될 수 있도록 하는 반면, 미국특허 제5,469,003호는 각각의 배선이 MUX 출력에 배선 접속(hardwired)된다는 점에서 단방향으로 제한된다. 미국특허 제5,144,166호 및 미국특허 제5,469,003호에 제공되는 접속성은 커넥션들이 인접하거나 상대적으로 로컬화되거나, 로직 셀들 자체는 로직 기능들을 수행하는 대신 상호접속 리소스들로서 사용될 수 있다는 전제하에서 매우 낮다. 미국특허 제5,457,410호, 미국특허 제6,507,217호, 미국특허 제6,051,991호, 미국특허 제6,597,196호에서 Ting은 다중 레벨 아키텍쳐를 기술하고, 여기서 다중 길이들의 도선들이 로직 셀들의 계층에서 스위치들을 통해 상호접속된다.

[0008] 미국특허 제2001/0007428호 및 미국특허 제5,914,616호의 Young 외는 두 영역들에서(각각의 영역에 3개) 다중 길이들의 배선들을 갖는 아키텍쳐를 기술하고, 여기서 근거리 로컬 접속들을 위해, 근접 크로스-바 수단(near cross-bar scheme)이 사용되며, 로직 셀들 출력들의 세트는 출력 포트들의 감소된 세트로 멀티플렉싱된 다음 다른 상호접속 리소스들에 인터페이스된다. 더 긴 배선들은 일반적으로 각각의 영역에서 더 짧은 길이의 배선들로 팬 인(fan-in)된다. 미국특허 제6,417,694호의 Reddy 외는 인터-슈퍼-영역, 인터-영역 및 로컬 도선들이 사용되는 다른 아키텍쳐를 개시한다. 크로스-바 수단은 로컬 배선들이 로직 엘리먼트들의 입력들에 대한 범용 액세스를 갖도록 최하위 레벨에서 사용된다(MUXs을 이용하여). 미국특허 제5,883,526호의 Reddy 외는 로컬 크로스-바에서 회로 감소 기술들을 가진 다양한 수단들을 개시한다.

[0009] 회로 계층의 베이스 레벨에서, 4-입력 룩업 테이블(LUT) 로직 셀들이 통상 사용된다. 베이스 로직 셀로서 LUT를 이용함에 있어서 2가지 장점들이 있다. 한가지 장점은 회로가 프로그래머블 제어들과 함께 임의의 4-입력, 1출력 부울 기능들을 허용할 수 있다는 것이다. 다른 장점은 4 입력들이 교환가능하고 논리적으로 동일하다는 것이다. 따라서, 이러한 4개의 신호들이 LUT의 4 입력들에 접속되면 LUT가 정확히 기능하도록 하기 위해 LUT의 입력 핀에 어떤 신호가 접속되는지는 중요하지 않는다.

[0010] 임의의 프로그래머블 로직 회로에서 해결되어야 하는 통상의 문제는 상호접속성 문제, 즉 신호들을 전달하는 제 1 세트의 도선들이 다수의 세트들의 도선들에 접속되는 방법이고, 신호들을 수신하는 로직 셀들은 집적 회로의 넓은 영역에 대해 분산된다(즉, M 로직 셀들의 M 출력들, 여기서 각각의 출력은 다수의 로직 셀들의 입력들에 접속됨). 상당히 바람직하지만 대부분의 경우들에서 비실용적인 솔루션은 제 1 세트의 모든 도선이 스위치를 통해 직접 다수의 세트들의 도선들의 모든 도선에 접속가능한 크로스 바 스위치를 사용하는 것이다. 종래의 솔루션들은 로컬 커넥션들의 연장부들을 통해 또는 보다 장거리 커넥션들을 이용하여, 로직 셀들의 로컬 클러스터들이 상호접속되고 로직의 다른 클러스터들로 확장되는 분할정복식(divide and conquer) 전략을 이용하여 접속성 문제를 다중 피이스(piece)들로 어느 정도 분할하려고 시도한다. 이러한 종래의 상호접속 수단들은 애드혹(ad hoc)이고 대부분 실험적 경험들을 기반으로 한다. 목표된 라우팅 모델 또는 상호접속 아키텍쳐는 항상 회로의 많은 부분에 대해 다수의 세트들의 도선들에 접속되는(프로그래머블 상호접속 도선들을 통해) 다수의 입력

들 및 출력들에 대해 완전한 접속력을 보장해야 한다.

[0011] 복잡한 소프트웨어는 상호접속 리소스들을 추적하는데 필요하고, 알고리즘들은 프로그래머블 로직 회로를 이용하여 주문 설계를 구현하는 배치 및 배선(place and route) 스테이지 동안 상호접속력을 개선시키는데 사용된다. 따라서, 프로그래머블 로직 회로들을 위한 새로운 상호접속 수단을 갖는 것이 바람직하며, 상기 새로운 상호접속 수단은 라우팅 능력(routability) 또는 상호접속력은 보다 광역 스케일로 보장될 수 있는 동시에 상호접속 비용은 요구되는 스위치들 면에서 낮게 유지되고, 주문 설계 구현을 위한 배치 및 배선을 결정함에 있어서 소프트웨어 노력들은 간소화되어야 한다.

[0012] 본 발명의 목적들, 특징들 및 장점들은 이하의 상세한 설명으로부터 명백해질 것이다.

## *발명의 상세한 설명*

[0013] 다수의 제 1 도선들을 다수의 세트들의 도선들에 연결하는 도선들의 중간 스테이지(들)를 포함하고 스위치들을 이용하는 혁신적인 확장성(scalable) 넌-블럭킹(non-blocking) 스위칭 네트워크(SN)가 먼저 기술되며, 상기 다수의 제 1 도선들의 각각의 도선은 SN을 통해 다중 세트들의 도선들 각각으로부터 하나의 도선에 연결될 수 있다. 확장성 넌-블럭킹 스위칭 네트워크는 광범위한 애플리케이션들에 적용될 수 있고, 단일 스테이지에서 사용되거나, 다중 스테이지들에서 계층적으로 사용되어 스위칭, 라우터들 및 프로그래머블 로직 회로들에 사용되는 많은(large) 스위치 네트워크를 제공한다. 확장성 넌-블럭킹 스위칭 네트워크가 SN을 통해 제 1 세트의 도선들을 다수의 세트들의 도선들에 연결하는데 사용됨으로써, 다수의 세트들 각각의 도선들이 동일하거나 상호교환가능하고, 예를 들어 다중 세트들 중 하나의 도선들은 로직 셀의 입력들이다(로직 셀들의 계층에 대한 입력들 또는 LUT의 입력들일 수 있음). 본 발명에서 확장성 넌-블럭킹 스위칭 네트워크는 제 1 세트의 도선들의 임의의 서브세트가 SN을 통해 다수의 제 2 세트들의 도선들에 연결될 수 있도록 함으로써, 서브세트의 각각의 도선이 다수의 세트들의 도선들의 각각의 세트로부터 하나의 도선에 접속될 수 있다.

[0014] 이하의 상세한 설명에서, 설명을 위하여, 많은 특정한 세부사항들이 본 발명의 철저한 이해를 제공하기 위해 기술된다. 본 발명의 실시예들은 이러한 특정 세부사항들 없이 실시될 수 있다는 것은 통상의 당업자에게 명백할 것이다. 다른 예들에서, 공지된 구조들 및 회로들은 본 발명을 불필요하게 방해하는 것을 방지하기 위해 블럭도 형태로 도시된다. 설명을 위하여, 반대되는 것으로 특정되지 않으면, 프로그램 제어되는 스위치 및 스위치란 용어들은 본 상세한 설명의 범주에서 상호교환될 수 있다: 프로그램 구성된(configured) 로직 셀, 로직 셀, 셀, 룩업 테이블(LUT), 프로그래머블 로직 셀이란 용어들은 본 상세한 설명의 범주에서 상호교환될 수 있고; 도선, 신호, 핀, 포트, 라인이란 용어들은 본 상세한 설명의 범주에서 상호교환될 수 있다. 또한, 본 발명은 사용되는 스위치들의 상태들을 설정하기 위한 프로그램 제어 수단을 사용하는 실시예들을 기술하고, 이러한 제어 수단은 퓨즈/퓨즈-방지 기술들과 같이 1회일 수 있거나, SRAM(휘발성), FLASH(비휘발성), 강유전체(Ferro-electric)(비휘발성) 등과 같이 재프로그래밍될 수 있다는 것을 유의해야 한다. 따라서, 본 발명은 이에 제한됨이 없이, 정적 랜덤 액세스 메모리(SRAM), 동적 랜덤 액세스 메모리(DRAM), 퓨즈/퓨즈-방지, 소거가능한 프로그래머블 리드-온리 메모리(EPROM), FLASH와 같은 전기적으로 소거가능한 프로그래머블 리드-온리 메모리(EEPROM), 및 강유전체 프로세스들을 포함하는 다양한 프로세스들과 관련된다.

[0015] 본 발명에서 기술되는 프로그래머블 로직 회로에 사용되는 확장성 넌-블럭킹 스위칭 네트워크들은 커넥션 요구조건들이 이용가능한 도선들을 초과하지 않는 한, 일반적으로 다수의 세트들의 도선들에서 다수의 도선들 간에 제한 없는 커넥션들을 허용하기 위해 적용될 수 있다.

[0016] 프로그램 제어되는 스위치가 하나의 도선을 다른 도선에 상호접속시키는데 사용될 때, 이러한 도선들을 통과하는 신호의 속도를 개선하기 위해 드라이버 회로가 스위치에 연결될 수 있다. 부가적으로, 다수의 도선들(신호들)이 프로그램 제어되는 스위치들을 통해 도선으로 팬-인(fan-in)되면, 사용되는 프로세스 기술에 따라 도선의 부하를 감소시키거나 회로 사이즈를 감소시키거나 도선 부하 및 회로 사이즈 모두를 감소시키기 위해, 바람직하다면 MUX 수단을 사용할 수 있다. MUX가 사용되는 경우, 다중 스위치들은 새로운 스위치 메커니즘으로 변환되고, 유효 상태들의 수는 스위치들의 수와 동일하며, 2개의 도선들을 연결함에 있어 특정 상태(다중 스위치들이 사용된 경우 스위치에 해당함)를 선택함으로써 접속이 인에이블되고, 상기 상태들은 프로그래머블 제어에 의해 결정된다.

[0017] 다양한 타입의 확장성 넌-블럭킹 스위칭 네트워크들은 이에 제한됨이 없이 다음을 포함하는 것으로 기술된다: 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크(0-SN), 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크(1-SN), 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크(2-SN) 및 멀티-스테이지 확장성 넌-블럭킹 스위칭 네트워크들로의

확장부들, 및 프로그래머블 로직 회로들에 대한 상호접속을 제공함에 있어서 계층적으로 확장성 넌-블럭킹 스위칭 네트워크들의 사용.

## 실 시 예

[0031] 도 1은 확장성 넌-블럭킹 스위칭 네트워크(200)를 포함하고 k개의 4-입력 로직 셀들(10, 20, 30, 40)(본 실시예에서 k=4) 및 2개의 플립플롭들(50, 60)을 포함하는 클러스터(CLST4) 회로(100)의 일 실시예를 도시한다. 로직 셀들(10-40)은 각각 셀(10)에 대해 4 입력들(101-104)(N0[0-3])을 갖고, 셀(20)에 대해 4 입력들(105-108)(N1[0-3])을 가지며, 셀(30)에 대해 4 입력들(109-112)(N2[0-3])을 갖고, 셀(40)에 대해 4 입력들(113-116)(N3[0-3])을 가지며, 셀들(10-40) 각각에 대해 4 출력들로서 4개의 도선들(121-124)을 갖는다. 스위치들(151-156 및 159, 160)은 로직 셀이 플립플롭을 구동하거나 로직 셀이 회로 출력들(125-128)로 직접 출력할지 여부를 제어하는데 사용된다. 플립플롭들(50, 60)은 스위치들(157, 158, 161, 162)을 이용하여 회로(100) 출력들(125-128)로 출력한다. 부가적으로, 도선(131)은 스위치(141)를 통해 셀(10)의 도선(101)을 구동하고 스위치(142)를 통해 셀(20)의 도선(105)을 구동할 수 있다. 유사하게, 도선(132)은 스위치들(143, 144)을 통해 각각 셀들(30, 40)을 구동할 수 있다. 셀(20)은 도선(133)에 대한 스위치들(145)을 이용하여 출력(122)을 통해 인접한 CLST4 회로(도 1에 미도시)를 구동할 수 있다. 셀(40)의 출력(124)은 도 1의 스위치(146)를 통해 도선(134)으로 구동된다. 3개의 다른 신호들(135-137)은 SET, CLOCK 및 CLEAR로서 각각 플립플롭들을 제어하는데 사용된다. 부가적으로, 도 1은 스위치 네트워크 MTX(200)를 이용하여 16 입력들(101-116)을 구동하도록 팬-인되는 (X+1) 도선들(180)(M[0-X])을 갖는다. 도선들 M[0-X](180)는 M 도선들로 지칭되고, 여기서 M은 도 1의 실시예에서 도선들(X+1)의 수와 동일하다. i=[0-(k-1)]에 대한 입력 도선들 Ni[0-3](101-116)는 Ni 도선들로 지칭되며, Ni는 도 1의 실시예에서 4인 입력들의 수와 동일하다. 예시를 목적으로, 크기 Ni=N=4는 도 1에 도시된다. 선택적으로, 각각의 Ni는 본 발명에서 기술된 접속 특성을 변화시키지 않으면서 상이한 크기를 가질 수 있다.

[0032] 도 2는 도 1의 MTX(200)가 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크(0-SN)(300)에 의해 나타낸다; 각각의 N 도선(101-116)은 M 도선들(예, 도 1의 도선들(180))(201-211)(M[0-10])의 (M-N+1) 도선들에 접속가능하고, 이에 따라 도선들(101-116)의 각각의 입력 도선에 대해 도 2에 도시된 스위치들의 수는 도 2의 0-SN(300)에 대해 (M-N+1)=8이다. 커넥션들의 수가 이용가능한 상호접속 리소스들을 초과하지 않는 한(즉, 임의의 로직 셀들의 입력들을 구동시키는 M 도선들의 수는 로직 셀의 입력들의 수를 초과할 수 없음), 스위칭 네트워크(0-SN)(300)는 임의의 서브세트의 M 도선들(201-211)이 임의의 블럭킹 없이 스위칭 네트워크(300)의 스위치들을 이용하여 로직 셀들(10-40) 각각의 1 입력 도선을 구동할 수 있도록 한다. 도 2의 수단은 M×(k×N)= 11×(4×4)= 176 스위치들을 포함하는 전체 스위치 매트릭스를 대신하는 크로스 바 커넥션에 대한 개선이며, 그 스위치들의 수는 (M-N+1)×(k×N)= 128이다. 도 2의 0-SN(300)은 각각의 로직 셀들에 대한 4 입력들을 상호교환가능한 또는 논리적으로 동일한 것으로 가정함으로써(즉, 도 1의 셀(10)의 도선들(101-104)은 동일하거나 상호교환가능함) 상술한 접속성이 가능하므로, 커넥션 요구조건이 특정 M 도선을 주어진 로직 셀에 연결하는 것이면 특정한 M 도선(즉, M[4] 도선(205))을 주어진 로직 셀의 임의의 입력 핀(즉, 스위치(222)를 이용하여 도 1의 셀(10)의 도선들(101-104) 외부의 도선(101))에 연결하는 것만이 필요하다.

[0033] 프로그래머블 회로들에 사용되는 기술에 따라, 일부 영역의 최소화가 달성될 수 있다. 예를 들어, 패스게이트(passgate)를 이용하여 구현되는 각각의 스위치에 대한 프로그램 제어로서 6개의 트랜지스터들로 SRAM 메모리 셀을 이용하여, 입력 라인(101)당 도 2의 8 스위치들(221-228)은 56 트랜지스터들을 요구할 것이다. 대신, 3개의 메모리 비트들을 이용하는 8 입력 MUX는 8 SRAM 비트들 및 8 스위치들을 효과적으로 대체하도록 8개의 상태들을 제어하는데 사용될 수 있다. MUX 수단에서, 3 비트들, 14 패스게이트들 및 1 인버터(신호를 재생성하기 위한)는 각각의 스위치에 대한 프로그램 제어로서 8 SRAM 메모리 셀들에 사용되는 56 트랜지스터들로부터 크게 감소되는 34 트랜지스터들을 사용한다. 도선(101)의 부하는 MUX 구현을 이용하여 감소되는 반면, 8 대 1 MUX로 인해 부가적인 지연들이 있다.

[0034] 도 3은 서브그룹들 Ma=[A0-A4]=5 도선들(301-305) 및 Mb=[B0-B4]=5 도선들(306-310)로 구성되는 M=Ma+Mb=10 도선들(301-31-)을 갖는 2개의 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크들(330, 320)을 이용함으로써 도 1의 MTX(200)를 나타내는 일 실시예를 도시한다. 각각의 4 로직 셀들의 상위 2 입력 도선들(셀(10)에 대한 도선들(101-102), 셀(20)에 대한 도선들(105-106), 셀(30)에 대한 도선들(109-110) 및 셀(40)에 대한 도선들(113-114)로 구성됨)에 대해 각각 Na=2 및 k=4 로직 셀들 각각의 하위 2 입력 도선들(셀(10)에 대한 도선들(103-104), 셀(20)에 대한 도선들(107-108), 셀(30)에 대한 도선들(111-112) 및 셀(40)에 대한 도선들(115-116)로 구성됨)에 대해 Nb=2. 도 3의 전체 크기의 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크는 입력 도선당 (M-

N+1)=10-4+1=7 프로그램 제어되는 스위치들을 갖는다. 그 대신, 도 3의 실시예에서, 입력 스위치들의 수는 개별 Ma 도선들 및 Mb 도선들(Ma=Mb=5를 가짐) 때문에 단지 4이고, 개수 N은 두 부분들로 나뉜다(Na=Nb=2). 이와 같이, 네트워크(330)에서 입력 도선당 프로그램 제어되는 스위치들의 수는 Ma-Na+1=5-2+1=4이고, 네트워크(320)에서 입력 도선당 프로그램 제어되는 스위치들의 사용은 Mb-Nb-1=4이다. 네트워크(330)를 이용하여 4 로직 셀들의 상위 2 입력들에 연결되는 Ma(301-305) 도선들은 도 2에 도시된 접속을 유지하는 것은 사실이지만(네트워크(320)를 이용하여 4 로직 셀들의 하위 2 입력들에 연결되는 Mb 도선들(306-310)에 대해서도 유사함), 4 로직 셀들로 팬-인되는 [A0-A4], [B0-B4]의 임의적인 사용은 상기 접속을 유지하지 않는다. 이러한 제약은 도 3의 2개의 0-SNs(320, 330)을 통해 N 도선들에 연결되는 M 도선들의 임의의 할당들을 방지한다. 그러나, 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크들(320, 330)은 프로그래머블 로직 회로를 위한 양호한 접속성을 제공하는 경제적인 구현일 수 있는 반면에, 허용가능한 M 도선들 사용을 추적하고 부기(book-keeping)하는 소프트웨어 노력들은 도 2의 수단보다 더 복잡하다. 도 3은 4 로직 셀들의 입력들에 임의로 접속되는 10개 중 적어도 8개 M 도선들을 허용하고, 각각 하나의 도선은 네트워크들(320, 330)을 이용하여 4 로직 셀들 각각에 대한 하나의 입력에 연결되며, 여기서 제약은 10개의 도선들이 도 2의 경우에서와 같이 임의로 할당될 수 없다는 것이다.

[0035] 본 발명의 실시예들에서, 제 1 그룹의 도선들은 스위치 네트워크를 이용하여 다수 그룹들의 동일한 도선들에 접속된다. 따라서, 0-SN이 존재하는 한, 여기서 다수의 k 세트들의 N 도선들에 대한 제 1 세트의 M 도선들 사이의 제한되지 않는 커넥션들을 제공하도록 (M-N+1)×N×k 스위치들이 존재하고, 임의의 서브세트의 M 도선들은 임의의 차단(blockage) 없이 0-SN을 이용하여 k 세트들의 N 도선들 각각에 하나의 도선을 접속시킬 수 있다.

[0036] 도 4는 스위치 네트워크에 사용되는 스위치들의 수가 0-SN의 접속 특성을 변경시키지 않으면서 크게 감소될 수 있는 선택적 실시예를 도시한다. 도 4는 도 1의 MTX(200)가 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크(1-SN)를 이용함으로써 나타낸 실시예를 도시한다. 1-SN(400)은 1-SN(400)의 2개의 스위치들과 하나의 중간 도선을 이용하는 도선들(101-116)의 N 도선에 도선들(401-411)의 M 도선을 접속시킨다. 128 스위치들이 사용되는 도 2의 네트워크(300)를 통해 M 도선들(201-211)을 k 세트의 N 도선들(101-116)에 직접 접속하는 대신에, 도 4의 1-SN(400)은 스위치(437)를 통해 중간 I 도선(454)에 먼저 접속한 다음, 서브-네트워크(450)의 스위치(441)를 통해 N 도선(109)에 접속시킴으로써 M 도선(407)(M[6])을 N 도선(109)에 접속시킨다. 유사하게, 동일한 M 도선(407)은 스위치들(442, 443, 444) 각각을 통해 동일한 중간 도선(454)을 통해 N 도선들(101, 105, 113)에 접속될 수 있다. 도 4의 1-SN(400)은 도 2의 0-SN(300)과 비교하여 스위치들의 수에서 25% 감소되는 96개의 스위치들을 갖는다. 중간 스테이지(들)의 상호접속부로 확장성 넌-블럭킹 스위칭 네트워크를 생성함으로써 0-SN에 요구되는 스위치들의 수를 감소시킬 수 있으며, 여기서 각각의 M 도선들은 각각의 k 세트들의 N 도선들로부터 도선에 임의로 접속될 수 있다. 확장성 넌-블럭킹 스위칭 네트워크는 각각의 k 세트들의 N 도선들로부터 하나 이상의 도선에 M 도선을 접속할 수 있지만, 논리적으로 각각의 N 도선들에서 하나 이상의 도선에 접속될 필요는 없다.

[0037] 도 4는 i=[1-N]에 대해 N 세트들의 중간 도선들 $I_i$를 구비한 1-SN(400)을 도시하며, 여기서 11 M 도선들(401-411), 4 세트들의 N 도선들(101-104, 105-108, 109-112, 113-116)이 있고, k는 4이다. 제 1 중간 도선들 $I_1$은 예를 들어, 각각의 N 도선들에 대한 제 1 입력 및 이에 따라 도선들(101, 105, 109, 113)과 연동되는 4개의 도선들(451-454)이다. 유사하게, 도선들(461-464)은 도선들(104, 108, 112, 116)과 연동되는 $I_4$ 도선들이다. 0-SN에서 N 도선들의 각각의 도선에 대한 (M-N+1) 스위치들이 도 4의 해당 $I_i$ 도선들 사이에서 분배된다. 예를 들어, M 도선들(401-408)을 연결시키는 8개 스위치들(431-438)은 $I_1$ 도선들(451-454)에 분배되며, 여기서 각각의 $I_1$ 도선들은 2개인 $[(M-N+1)/I_1]$ 스위치들에 연결된다. 도 4의 예에서, 각각의 $I_i$ 도선들의 중간 도선들의 수는 4이다. 일반적으로, 상이한 $I_i$는 일정한 수(이하에서 기술되는 것처럼)일 필요는 없다. 도 4의 1-SN(400)은 $[(M-N+1)\times N + \text{sum}_{i=[1-N]}(I_i\times k)]=32+64=96$ 스위치들을 갖고, 여기서 $I_i$는 각각의 N 세트들의 $I_i$ 중간 도선들에서 중간 도선들의 수이다. 도 4의 1-SN(400)은 도 2의 각각의 0-SN(300)과 동일한 접속 특성을 허용하고, 2개 스위치들과 1-SN(400)의 하나의 중간 도선을 통해 각각의 k 세트들의 N 도선들 중 하나의 도선에 M 도선들의 임의의 도선을 접속시킨다.

[0038] 도 4의 1-SN(400)에서, 임의의 N-터플(tuple)의 M 도선들은 상이한 N 세트들의 $I_i$ 도선들에 대한 스위치들의 적절한 선택을 갖는다. 예를 들어, 도선들(401, 404, 405, 410)은 M 도선들 중 4-터플(N=4)이고, 여기서 도선(401)은 스위치(431)를 통해 도선(451)($I_1$ 도선들 중)에 접속되며; 도선(404)은 스위치(446)를 통해 도선

(467)($I_3$ 도선들 중)에 접속되고; 도선(410)은 스위치(427)를 통해 도선(464)($I_4$ 도선들 중)에 접속된다. M 도선들의 N-터플의 임의의 서브세트는 중간 도선들에 접속되는 동일한 특성을 갖는다. 부가적으로, $I_i$ 도선들의 각각의 중간 도선은 각각의 k 세트들의 N 도선들에서 하나의 N 도선에 접속가능하다. 예를 들어, 도선들(451-454)의 임의의 도선은 서브-네트워크(450)의 스위치들을 통해 도선들(101, 105, 109, 113)에 접속가능하다. 유사하게, 도선들(461-464)의 임의의 도선은 서브-네트워크(420)의 스위치들을 통해 도선들(104, 108, 112, 116)에 접속가능하다.

[0039] 도 5는 도 1의 MTX(200)를 나타내는 1-SN의 선택적 실시예를 도시한다. 1-SN(500)에서, 12 M 도선들(501-512), 4 세트들의 N 도선들(101-116), 및 N 세트들의 중간 $I_1$ 도선들(521-523), $I_2$ 도선들(524-526), $I_3$ 도선들(527-529), 및 $I_4$ 도선들(530-532)이 있고, 여기서 $M=I_1+I_2+I_3+I_4$ 또는 $I_i=M/N=3$이다. 도 5의 스위치들의 수는 $[(M-N+1)\times N + sum_{i=[1-N]}(I_i\times k)]=36+48=84$이다. 해당 0-SN은 144 스위치들을 갖고, 크로스 바는 192개의 스위치들을 갖는다. 도 5의 1-SN(500)의 접속 특성은 더 적은 중간 도선들 및 스위치들을 갖는 도 4의 1-SN(400)에 대해 이전에 논의된 것들과 동일하다. 도 4 및 도 5의 예시들은 도 4의 서브-네트워크(450)의 스위치들 및 도 5의 서브-네트워크의 스위치들 각각을 통해, 도 1의 4개 로직 셀들(10-40)의 각각의 제 1 입력인 도선들(101, 105, 109, 113)에 접속되는 제 1 세트의 중간 $I_1$ 도선들(도 4의 도선들(451-454) 및 도 5의 도선들(521-523))을 갖는다. 동일하게 효과적인 대안은 본 예에서 특정 로직 셀의 4 입력들 각각이 상이한 세트의 $I_i$ 도선들에 의해 커버되면 4개의 로직 셀들 각각으로부터 임의의 하나의 도선($i^{th}$ 인 하나 대신에)에 각각의 세트의 $I_i$ 도선들을 접속시키는 것이다.

[0040] 도 6은 도 5의 1-SN(500) 보다 더 강한 접속 특성을 갖는 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크의 상이한 버전의 일 실시예를 도시한다. 많은 스위치들이 요구되지만, 1-SN(600)의 12 M 도선들(601-612)(M[0]-M[11])은 각각의 N 세트들의 $I_i$ 중간 도선들(621-623, 624-626, 627-629, 630-632)에서 모든 도선들에 접속가능하다. 이것은 도 4 및 도 5의 M 도선들의 (M-N+1) 도선들에 대한 연결과 대조적이다. 1-SN(600)에서, 도선들(601-612)은 서브-네트워크(620)의 스위치들을 통해 $I_1$ 도선들(621-623)에 접속가능하다. 도선들(601-612)은 서브-네트워크(640)의 스위치들을 통해 $I_2$ 도선들(624-626)에 접속가능하다. 도선들(601-612)은 서브-네트워크(650)의 스위치들을 통해 $I_3$ 도선들(627-629)에 접속가능하다. 도선들(601-612)은 서브-네트워크(660)의 스위치들을 통해 $I_4$ 도선들(630-632)에 접속가능하다. 도 6의 12 M 도선들(601-612)은 도 5의 1-SN(500)에 비해 더 강력한 접속 특성을 가지며, 여기서 $M/I_i$ 도선들 중 하나의 도선은 임의의 k 세트들의 특정 N 도선들에 접속되게 선택되도록 프로그래밍될 수 있다. 일 예로서, 도 6의 실시예에서, 임의의 N-터플들의 도선들(601-604, 605-608, 609-612)(M 도선들 중에서)은 1-SN을 이용하여 임의의 4(k=4) 세트들의 N 도선들의 임의의 특정 입력 도선에 접속될 수 있지만, 각각의 4-터플들내에서 도선들은 특정 입력 도선에 대해 상호 배타적이다. 도 6의 이러한 1-SN(600)에서 요구되는 스위치들의 수는 $[M\times N + sum_{i=[1-N]}(I_i\times k)]=48+48=96$ 스위치들이다.

[0041] 요구되는 스위치들 면에서 0-SN 및 1-SN 사이의 차는 도 5의 경우에 $[(M-N+1)\times N\times k]$ 및 $[(M-N+1)\times N + sum_{i=[1-N]}(I_i\times k)]$ 사이의 차이고, 여기서 (M-N+1)의 M 도선들은 $I_i$ 도선들의 각각의 N 세트들에서 $I_i$ 도선들에 1-SN을 통해 접속가능하다. 요구되는 스위치들 면에서 0-SN 및 1-SN 사이의 차는 도 6의 경우에 $[M\times N\times k]$ 및 $[M\times N + sum_{i=[1-N]}(I_i\times k)]$ 사이의 차이다. 각각의 $I_i=k$라고 간략화하면, M은 도 5의 경우에 대해 적어도 [k+N+1/(k-1)]이고, M은 적어도 [k+1+1/(k-1)]이며, 도 5의 수단은 M이 상술한 수 미만이 되도록 여전히 작용한다는 것을 유의할 필요가 있다. 부가적으로, 1-SN의 수단이 작용하도록 하기 위해, 중간 도선$[(M-N+1)I_i]$당 스위치들의 수는 SN의 넌-블럭킹 특성들을 잃지 않으면서 N보다 더 크지 않을 수 있다. 수 $[(M-N+1)/I_i]$는 정수가 아닐 수 있으며, 이 경우 정수 $P_i$는 수 $(M-N+1)/I_i$ 주변에서 반올림 또는 반내림시킴으로써 사용되고, $sum_{i=[1-N]}P_i=(M-N+1)$이다. 유사하게, 도 6의 경우에 대해, M은 (M-N+1) 대신에 사용되어 $P_i$는 $(M/I_i)$를 반올림 또는 반내림시키는 정수이고, $sum_{i=[1-N]}P_i=M$이다. 더욱이, 도 4 및 도 5의 예들에서, 중간 도선들의 수 $sum_{i=[1-N]}I_i$는 M 또는 k×N일 수 있거나 이들 사이의 일정 수일 수 있으며; 각각의 개별 $I_i$는 M/N, k 또는 이들 사이의 일정 수로 제한되고, M/N은 분할가능한 정수가 아닐 수 있기 때문에, $I_i$는 M/N을 반올림 또는 반내림시킴으로써 정수이며, 이

에 따라 개별 $I_i$는 i=[1-N]에 대한 모든 i 중에서 일정하지 않을 수 있다고 이해할 수 있다.

[0042] 도 7은 도 6의 실시예에서 스위치들의 수가 1-SN의 접속 특성에 대한 많은 변경 없이 감소되는 일 실시예를 도시한다. 도 7은 그러한 감소를 나타내며, 도 6에서, 도선(601)이 도선(621)에 대해 쇼트되고, 도선(602)는 도선(624)에 대해 쇼트되며, 도선(603)은 도선(627)에 대해 쇼트되고, 도선(604)는 도선(630)에 대해 쇼트되며; 여기서 도 6의 서브-네트워크(670)의 16 스위치들은 소거되고 도 6의 96개 대신에 도 7에서 스위치들의 수는 80개이다. M 도선들(605-612)을 갖는 도 7에서 1-SN(700) 마이너스 서브-네트워크들(710, 720, 730, 740)은 도 6에서 기술된 것과 동일한 1-SN(600)의 더 강력한 접속 특성을 가지며, M=8을 갖는 1-SN이다. 예를 들어, 더 많은 M 도선들을 중간 도선들에 쇼트시킴으로써 스위치들의 수를 추가로 감소시킬 수 있지만, 접속 특성이 더 많이 변화(reduce)되고 접속 패턴을 결정하는 소프트웨어 노력들(software effors)이 더욱더 많이 복잡해진다.

[0043] 도 8은 i=[1-16]에 대해 M=48, k=4, N=16 및 $I_i$=3을 갖는 1-SN의 일 실시예를 도시한다. 1-SN(800)에서 720개의 스위치들이 있기 때문에, 0-SN은 2112개의 스위치들을 요구하고 크로스 바는 3072개의 스위치들을 요구한다. 각각의 N(=16) 세트들의 $I_i$ 중간 도선들, 예를 들어 $I_{16}$은 3개의 도선들(서브-네트워크(810) 내부에)을 가지며, 여기서 $I_{16}$ 도선들은 도 8의 (M-N+1)=33 M 도선들에 연결되고, 각각의 중간 도선들은 서브-네트워크(811)의 11 스위치들을 통해 11 M 도선들에 연결된다. 접속 경로에 중간 도선과 여분의 스위치를 도입함으로써, 1-SN(800)은 0-SN에 비해 요구되는 스위치들의 수에 있어 큰 감소를 제공한다.

[0044] 다양한 실시예들에서, M이 k×N 미만인 예들이 사용되고, M 도선들은 팬-인 신호들을 전달하는 도선들이며, k 세트들의 N 도선들은 이러한 팬-인 신호들을 수신하는 도선들이다. 이것은 반드시 그럴 필요는 없다. 간단히, M이 k×N 보다 더 큰 SN을 가질 수 있다. 선택적으로, 예를 들어, 로직 셀들의 4 클러스터들로부터의 16 출력들로서 도 6의 도선들(101-104, 105-108, 109-112, 113-116)을 고려하고, 16에서 12로의 출력 감소를 목적으로 1-SN을 이용하는 것을 고려할 수 있으며, 여기서 16 출력들 중에서 임의의 서브세트의 12 출력들은 1-SN을 이용하여 선택될 수 있다. 부가적으로, 다양한 도면들에서 도선들(101-104, 105-108, 109-112, 113-116)은 로직 셀들의 입력들 또는 출력들일 필요는 없지만 다수의 동일한 도선들일 수 있으며, 여기서 하나의 동일한 다수의 도선들에서 임의의 도선에 대한 접속은 다수의 동일한 도선들에서 특정 도선에 대한 접속과 반대가 되기에 충분하다.

[0045] 프로그래머블 로직 회로들을 위한 상호접속 아키텍처를 설계함에 있어서, 특정 설계 주문을 위한 회로 사이즈, 속도 및 배치 및 배선에 대한 소프트웨어의 용이성과 같은 엔지니어링 트레이드-오프들을 기초로 합리적인 접속성과 적절한 상호접속을 제공하는 것은 중요할 수 있다. M 도선들과 k 세트들의 N 도선들 사이의 비율 R이 있고, 여기서 R=M/(k×N); R이 너무 작으면, 접속성은 더 큰 R 보다 더 제한된다. 예를 들어, 도 6의 회로에서, R=0.75이다. 확장성 넌-블럭킹 스위칭 네트워크들을 이용하여 회로들의 계층을 형성함에 있어서 R을 확장 컴포넌트로 지칭한다. 확장성 넌-블럭킹 스위칭 네트워크들을 이용하여 프로그래머블 로직 회로들의 설계를 위해 통상 사용되는 확장 컴포넌트는 0.5 내지 1.0 범위이고, 그 선택은 엔지니어링 설계 트레이드-오프들(즉, 로직 사용, 회로 면적 최소화, 소프트웨어 배치 및 배선의 용이성 등), 사용되는 기술(즉, SRAM, 퓨즈-방지 등) 등과 같은 팩터들에 의존한다. 예를 들어 출력 감소와 같이, 회로들의 부품들의 범위를 초과하는 것이 때때로 바람직하며, 여기서 많은 수의 출력들은 SN을 이용하여 더 적은 수로 감소된다.

[0046] 프로그래머블 로직 셀들의 상호접속을 위해 회로 계층을 형성하는데 사용될 수 있는 0-SN 및 1-SN을 이용함으로써 이전의 논의가 처리되고, 이에 의해 계층의 각 레벨은 다양한 확장성 넌-블럭킹 스위칭 네트워크들을 이용하여 회로들을 통해 다양한 도선들에 접속되기 위해 연동되는 0-SN 및/또는 1-SN을 갖는 몇가지 프로그래머블 로직 회로들을 포함한다. 이전에 기술된 수단들은 상호접속 리소스들 및 로직 용량들이 이용가능하게 유지되는 한 0-SNs 및 1-SNs를 이용하여 임의의 레벨의 회로 계층에서 임의의 신호에 대한 접속이 계층내에서 임의의 로직들의 입력에 도달할 수 있도록 한다.

[0047] 스테이지-1 및 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크들을 계층적으로 이용하여 프로그래머블 로직 회로를 형성하는 수단이 이하에서 기술된다. 도 9는 16 M 도선들(901-916), N=4인 4 세트들의 N 도선들(101-104, 105-108, 109-112, 113-116), 및 각각의 $I_i$=M/N=4인 i=[1-N]에 대해 N 세트들의 $I_i$ 도선들(931-934, 935-938, 939-942, 943-946)을 갖는 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크를 이용하는 도 1의 CLST4 회로(100)의 MTX 회로(200)의 일 실시예를 도시하며; 도 9의 실시예에서 확장 컴포넌트 R은 1.0이다.

[0048] 예를 들어 도 9의 1-SN(900)을 이용하여 프로그래머블 회로를 형성하는 구성에 의해, 임의의 서브세트의 M 도선들(901-916)은 1-SN(900)을 통해 각각의 k 세트들의 N 도선들의 하나의 도선에 개별적으로 접속될 수 있다. 그

다음, 이러한 M 도선들은 자체로 논리적으로 동일해진다. 도 1의 4 로직 셀들(10-40) 각각으로부터(도선들(101-104)로부터 하나, 도선들(105-108)로부터 하나, 도선들(109-112)로부터 하나, 및 도선들(113-116)로부터 하나) 4 입력들까지 연결시키기 위해 도 1의 CLST4 회로(100) 외부의 일부분으로부터 시작되는 임의의 신호에 대해, M 도선들 중 하나에 접속시키는 것만이 필요하다. 따라서, 이러한 M 도선들(901-916)은 N 도선들(여기서, N=16)로서 계층적으로 처리될 수 있으며, 1-SN과 함께 4 로직 셀들과 2개의 플립플롭들을 포함하는 회로를 각각 구비하는 다수의 새로운 k 세트들의 새로운 N 도선들은 새로운 세트의 M 도선들에 의해 SN과 같은 새로운 스위치 네트워크를 통해 선택적으로 접속되어야 한다. 이러한 프로세스는 목표된 회로 사이즈에 도달할 때까지 반복될 수 있고, 목표된 회로는 상술한 것처럼 제한되지 않는 접속성을 허용한다.

[0049] 도 10은 도 1의 4 세트들의 CLST4(100)를 이용하는 다음 레벨의 회로 계층 CLST16(1000)의 블럭도 실시예(도 10의 CLST(1010), CLST(1020), CLST(1030), CLST(1040))를 도시하며, 여기서 회로 MTX(200)는, 48 도선들(1055)(M[0-47])과 16 도선들(1056)(OW[0-7], OE[0-7])을 가진 64 M 도선들 및 4 세트들의 N 도선들(1060, 1070, 1080, 1090)을 구비한 회로 MTX16(1050)의 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크, 및 도 9의 1-SN(900)을 이용하여 구현되고, 여기서 각각의 N 도선들은 도 9의 16 M 도선들(901-916)에 대응하는 16 도선들을 갖는다. 도 10에서, 64 M 도선들(1055, 1056)의 16 도선들(1056)은 4개의 CLST4(100) 회로들(1010, 1020, 1030, 1040)의 4 출력들(1065, 1075, 1085, 1095)에 바로 접속된다. 16 도선들(1056)(OW[0-7], OE[0-7])은 4 세트들의 4 도선들을 갖고, 각각의 4 도선들은 도 1의 CLST4(100) 회로의 4 출력들(125-128)(O[0-3])에 해당한다. 확장 컴포넌트 R은 이러한 회로(1000)에서도 1.0이다.

[0050] 많은 수의 도선들을 다수 세트들의 도선들에 접속시키는 이러한 다음 레벨의 회로 계층에서 확장성 넌-블럭킹 스위칭 네트워크들의 사용은 도 11a에 도시된다. 도 11a는 도 10의 회로 MTX16(1050)의 블럭도 형태의 일 실시예를 도시하고, 여기서 64 M 도선들(1101)(M[0-47], OW[0-7], OE[0-7])은 도 10의 도선들(1055, 1056)에 해당한다. 중간 도선들의 제 1 스테이지는 N0(N0=4) 세트들의 16 $IO_i$ 도선들(1150, 1160, 1170, 1180)(여기서, i=[1-N0]에 대해 $IO_i$=M/N0=16)로 구성된다. M 도선들(1101)은 서브-네트워크들(1110, 1120, 1130, 1140)의 스위치들을 이용하여 첫 번째 4 세트들의 중간 스테이지 $IO_i$ 도선들(1150, 1160, 1170, 1180)에 인터페이스된다. 도 11b는 서브-네트워크(1120)를 통해 도선들(1101)이 도선들(1160)에 접속되는 수단을 도시한다. 도선들(1101)이 서브-네트워크(1110)를 통해 도선들(1150)에 접속되고, 서브-네트워크(1130)를 통해 도선들(1170)에 접속되며, 서브-네트워크(1140)를 통해 도선들(1180)에 접속되는 접속 수단은 도 11b의 서브-네트워크(1120)와 동일하다. 본 실시예에서 4 세트들의 제 1 스테이지 중간 도선들(1150, 1160, 1170, 1180)에 대해 M 도선들(1101) 사이에 사용되는 스위치들의 수는 MxN0=256이다. 도 5와 연계하여 기술된 것처럼, 대안적인 구현은 (M-N0+1)×N0 스위치들을 갖는 것이다.

[0051] 도 12는 도선들(1160)이 제 2 N0 세트의 $IO_i$ 도선들인 회로 TA1(1165)의 일 실시예를 도시하며, 여기서 i=2, $IO_i$=16이고; 중간 도선들(1201-1216)(도 11a의 도선들(1160)에 해당함)은 16 도선들(1241-1256)(도 11a의 도선들(1161-1164)에 해당함)과 인터페이스한다. 도 11a의 서브-네트워크들(1155, 1175, 1185)은 도선들(1150, 1170, 1180)을 도 11a의 도선들(1151-1154, 1171-1174, 1181-1184)에 각각 상호 접속시키기 위한 서브-네트워크(1165)와 동일한 회로이다.

[0052] 도 12에서, 회로 TA1은 도 11a의 1-SN(1165)이고, 여기서 M 도선들(1201-1216)은 도 11a의 16 중간 $IO_2$ 도선들(1160)(I1_1[0-15])이며; 16 중간 도선들(1221-1236)은 i=2, j=[1-N1]에 대해 $I1_{2j}$($I1_{2j}$=M/N1=4) 도선들의 N1(=4) 세트들로 구성된다: 도선들(1221-1224, 1225-1228, 1229-1232, 1233-1236). $I1_{2j}$ 도선들은 j=[1-N1]에 대해 4 세트들의 목적지 도선들(1241-1244, 1245-1248, 1249-1252, 1253-1256)에 각각 접속된다. 도 12의 1-SN(1165)은 도 9의 동일한 1-SN(900)을 사용한다. 그러나, 1-SN(1165)은 도 11a의 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크(2-SN)(1050)의 제 2 부분에서 4개(서브-네트워크들(1155, 1165, 1175, 1185)) 중 하나이고, 여기서 2-SN의 도선들(151-1154, 1161-1164, 1171-1174, 1181-1184)은 도 10의 각 CLST4 회로들(1010, 1020, 1030, 1040)의 M 도선들(1060, 1070, 1080, 1090)이다. 각각의 CLST4 회로들(1010, 1020, 1030, 1040)은 도 9의 1-SN(900)과 함께 도 1의 CLST4 회로(100)에 해당한다.

[0053] 도 12의 TA1 회로(1165)는 도선들(1201-1216)을 도선들(1241-1256)에 선택적으로 접속시키고; 도선들(1241, 1245, 1249, 1253)은 도 10의 CLST4(1010)의 16 M 도선들 중 4 M 도선들(C0[0-15] 중 C0[4-7])에 해당하는 도 11a의 도선들(1161)(N0[4-7])이다. 유사하게, 도선들(1242, 1246, 1250, 1254)은 도 10의 CLST4(1030)의 16 M 도선들(1080) 중 4 M 도선들(C1[0-15] 중 C1[4-7])에 해당하는 도 11a의 도선들(1162)(N1[4-7])이다. 도선들

(1243, 1247, 1251, 1255)은 도 10의 CLST4(1020)의 16 M 도선들(1070) 중 4 M 도선들(C2[0-15] 중 C2[4-7])에 해당하는 도 11a의 도선들(1163)(N2[4-7])이다. 도선들(1244, 1248, 1252, 1256)은 도 10의 CLST4(1020)의 16 M 도선들(1090) 중 4 M 도선들(C3[0-15] 중 C3[4-7])에 해당하는 도 11a의 도선들(1164)(N3[4-7])이다.

[0054] 도 11a의 MTX(1050) 회로의 1-SN 구현에서, M=64, k=4, 및 N=16이고, 2-SN 구현에서, 각 스테이지의 중간 도선들의 세트들의 수는 N0=4 및 N1=4이고, 여기서 곱 N0×N1은 N과 동일하다. 도 6 및 도 9에서 기술된 더 강력한 접속성의 SN을 이용하는 도 10의 2-SN(1050)에서 스위치들의 수는 $M\times N0 + sum_{i=[1-N0]}[(IO_i\times N1) + sum_{j=[1-N1]}[I1_{ij}\times(IO_i/N1))]$, 여기서 i=[1-N0]에 대해 $IO_i=M/N0$, 네트워크(1050)에서 i=[1-N0], j=[1-N1]에 대해 $I1_{ij}=IO_i/N1$이므로, $IO_i=16$, $I1_{ij}=4$이고 네트워크(1050)의 2-SN은 768 스위치들을 갖는다. 1-SN 구현은 1280 스위치들을 요구하고, 전체 크로스 바 스위치는 4096 스위치들을 요구한다. 각각의 $IO_i$ 도선들은 M 도선들의 M 대신에 (M-N0+1)에 인터페이스하고, 각각의 $I1_{ij}$ 도선들에 대해 $IO_i$ 도선들의 $IO_i$ 대신에 ($IO_i$-N1+1)에 인터페이스하며, 스위치들의 수는 $(M-N0+1)\times N0 + sum_{i=[1-N0]}[(IO_i-N1+1)\times N1)+ sum_{j=[1-N1]}[I1_{ij}\times(IO_i/N1))]$이다. 도 10의 경우에, $N=N0\times N1$, $IO_i=M/N0$, $I1_{ij}=M/N=k$이므로, 2-SN에 대한 경우 스위치들의 수는 $[M\times(N0+N1+k)]$이다.

[0055] 이전에 언급한 것처럼, 상이한 SNs에서 k 세트들의 N 도선들의 각각의 N 도선들은 일정한 크기일 필요는 없다. SN은 상이한 크기의 $N_i$로 구성될 수 있고, 여기서 최대 크기의 $N_i$는 일정한 크기의 새로운 N으로 사용되고, 가상 도선들과 스위치들이 더 작은 크기의 $N_i$에 부가되어 $N_i$가 크기 N인 것처럼 보이게 한다. 상호접속 스펙은 $N_i$ 이상의 접속들을 갖기 위해 더 작은 크기의 $N_i$를 요구하지 않기 때문에, SN의 접속 특성에서 변화는 없다. 일 예로서, 도 1에서 로직 셀들(10-40)에 대한 입력들로서 각각 4 세트들의 N 도선들(101-104, 105-108, 109-112, 113-116) 대신에, 도 1의 로직 셀(10)은 3개의 입력들(101-103)만을 갖는다. M 도선들(601-612)을 갖는 도 6의 SN에서, 도 6의 스위치들과 중간 도선들(621-632)은 서브-네트워크(680)의 3개의 스위치들과 도선(104)이 "가상"이고 도 6의 SN을 없앨 수 있다는 것을 제외하고 동일하게 유지된다.

[0056] 다중 스테이지들의 확장성 넌-블럭킹 스위칭 네트워크들은 상술한 수단들을 이용하여 형성될 수 있고, 예를 들어 도 10의 MTX(1050)는 i=[1-N0], j=[1-N1] 및 k=[1-N2]에 대해 제 1 중간 $IO_i$ 도선들 $IO_i=M/N0$, $I1_{ij}=IO_i/N1$ 및 $I2_{ijk}=I1_{ij}/N2$를 갖는 N0=2, N1=2 및 N2=4를 이용하는 스테이지-3 확장성 넌-블럭킹 스위칭 네트워크로서 구현될 수 있으며, 여기서 도 10의 4 CLST4 회로들(1010, 1020, 1030, 1040) 각각에 대한 입력들의 수인 $N0\times N1\times N2=N=16$이다. 유사하게, SN(1050)은 스테이지-4 SN으로 구현될 수 있고, 여기서 M 도선들을 N 도선들로 접속시키는 4 중간 스테이지들의 도선들을 갖는 N0=2, N1=2, N2=2 및 N3=2이다. 도 10의 SN(1050)에서 1-SN 구현에 대한 2-SN 구현에서 $N\times M=16M$ 및 $(N0+N1)\times M=(4+4)\times M=8M$ 사이의 차에 의해 스위치들의 수가 감소되고; (N0+N1+N2)=(2+2+4)=8 및 (N0+N1+N2+N3)=(2+2+2+2)=8인 각각의 3-SN 및 4-SN은 (N0+N1)=(4+4)=8인 2-SN에 비해 개선점이 없다. 이와 같이, Ni의 합, 각 스테이지에 대한 중간 도선들의 세트들의 수가 이전 스테이지 멀티-스테이지 SN 보다 적게 증가할 때에만 이해될 수 있다. 따라서, N=64에 대해, (N0+N1+N2)=12인 N0=N1=N2=4를 이용하는 3-SN은 (N0+N1)=16을 갖는 N0=N1=8을 이용하는 2-SN에 비해 스위치 감소에 있어서 매우 효과적이고, N=64인 1-SN에 비해 2-SN에 대해서도 유사하다는 것이 이해될 수 있다.

[0057] 따라서, 2-SN을 통해, 그 다음 1-SN을 통해 16 4-입력 로직 셀들에 접속시키기 위해 64 M 도선들이 팬-인되는 확장성 넌-블럭킹 스위칭 네트워크들을 이용하는 2가지 레벨들의 회로 계층을 기술하였다. 64 M 도선들 중 16은 4 CLST4(도 1의 100의 125-128) 회로들 각각의 16 출력들에 직접 접속되고, 임의의 출력으로부터 모든 16 로직 셀들에 제한되지 않는 접속들을 제공한다. 제 1 레벨의 회로 계층은 도 9의 1-SN(900)으로서 구현되는 MTX(200)을 갖는 도 1의 회로 CLST4(100)를 포함하고, 여기서 CLST4(100)는 도 1에 도시된 바와 같은 4개의 4-입력 로직 셀들(10-40) 및 2개의 플립플롭들(50, 60)을 갖는다. 더 상위의 다음 레벨의 회로 계층은 도 10에 도시된 것과 같은 2-SN MTX16(1050)을 갖는 4 CLST4(100) 회로들을 갖는 도 10의 CLST16(1000) 회로들이고, 여기서 네트워크(1050) 구현은 도 11a, 도 11b, 및 도 12에 도시된다. CLST16(1000)에서, 각각의 16 출력들(1065, 1075, 1085, 1095)(도선들(1056)에 직접 접속됨)은 CLST16(1000) 회로에서 모든 로직 셀에 대해 제한되지 않는 접속을 가지며, 도 10의 다른 48 M 도선들(1055)은 다음 레벨의 회로 계층을 형성할 때 CLST16(1000)의 N 도선들로서 처리될 수 있다. 도 10의 각각의 4 CLST4 회로들(1010, 1020, 1030, 1040)에 대해 도 1의 CLST4(100)의 16 출력들(125-128)은 16 M 도선들(1056)로 직접 배선되고, 그 출력들은 SN을 통해 형성 블럭들로서 CLST16(1000) 회로들을 이용하여 회로 계층의 다음 제 3 레벨에 추가로 접속될 수 있으며, 다른 48 M 도선들은 프로그래머블 로직 회로에서 연속되는 높은 접속성을 제공하기 위해 CLST(1000) 회로들에 대해 동일한 핀

들 또는 입력 도선들이다.

[0058] 도 10의 CLST(1000) 회로는 다양한 SNs 중에서 총 1280 스위치들을 이용하여 4개의 그룹들(1010, 1020, 1030, 1040)에서 구조화된 64 M 도선들(1055, 1056) 및 16개의 4-입력 로직 셀들을 갖는 4개의 1-SNs이 연속되는 2-SN을 이용하여 도시된다: 도 10의 각각의 그룹(1010-1040)에 대해 도 10의 SN(1050) 및 도 1의 SN(200). 도 10의 CLST(1000) 회로는 64 M 도선들, 도 9에서 기술된 방법들을 이용하는 다수의 k(예, 16)의 N(예, 4) 도선들을 갖는 1-SN을 이용하는 선택적 구현을 가질 수 있다. 스위치들의 수는 본 발명에서 기술된 분석을 이용하여 M×(N+k)=1280이다. 이 경우, 1-SN 구현 및 도 10의 실시예는 동일한 수의 스위치들을 갖는 것이 입증된다.

[0059] 어떤 구현이 보다 적합한지를 결정하는 것은 다음과 같은 엔지니어링 고려요소들에 의존할 것이다: 도 10의 실시예에서 더 많은 중간 스테이지들의 도선들을 갖는 4-입력 MUX 구현 또는 1-SN 구현에서 16-입력 MUX 및 더 적은 수의 중간 스테이지들의 도선들이 SRAM 기술을 이용하여 보다 바람직한지 여부, 레이아웃 구현 등에서 하나의 스타일이 보다 적합한지 여부 등. 상기한 분석을 기초로, SN을 통해 접속되는 로직 셀들의 합리적인 크기의 베이스 어레이를 가짐으로써, 다음 레벨의 회로 계층에서 다른 SN을 이용하여 로직 셀들의 몇가지 베이스 어레이들을 형성할 때 전체 스위치 계산 경비가 로직 셀들의 더 큰 크기의 베이스 어레이를 구현하는 것을 초과하지 않도록 하는 것이 바람직하는 것을 유의하는 것이 중요하다. 대부분의 프로그래머블 로직 회로들에서, 베이스 로직 셀(SN을 갖는 로직 셀 어레이의)은 일반적으로 3개의 입력들 또는 4개의 입력들을 갖고, 상술한 예시된 예들로부터 베이스 로직 어레이에서 로직 셀들의 수 k는 작은 수이여서는 안되며, 오히려 N의 크기에 따라, k×N은 상호접속 네트워크로서 효율적으로 사용되는 SN에 대해 합리적인 크기(예, 도 1의 CLST4(100))이여야 한다는 것을 알 수 있다.

[0060] 많은 실시예들 및 예시들을 이용하여, 다양한 확장성 넌-블럭킹 스위칭 네트워크들을 구축하는데 대한 상세한 설명이 제공되고, 입력들 및 출력들, 프로그래머블 로직 회로들을 위한 상호접속을 제공하기 위해 다양한 조합들로 사용된다. 기술과 엔지니어링 고려요소들에 따라, 이에 제한됨이 없이 메모리 제어들의 수, 스위치 수등을 감소시키기 위한 MUX의 사용을 포함하는 확장성 넌-블럭킹 스위칭 네트워크들의 구현에서 변화들이 사용될 수 있다.

## *도면의 간단한 설명*

[0018] 도 1은 확장성 넌-블럭킹 스위칭 네트워크(SN)를 이용하는 4개의 4-입력 로직 셀들 및 2개의 플립플롭들을 구비한 회로의 일 실시예를 도시한다.

[0019] 도 2는 4개 세트들의 4 N 도선들에 액세스하는 11 M 도선들을 갖는 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크(0-SN)를 이용하는 회로의 일 실시예를 도시한다.

[0020] 도 3은 4개 세트들의 2 N 도선들에 액세스하는 5 M 도선들을 갖는 각각의 0-SN을 구비한 2 스테이지-0 확장성 넌-블럭킹 스위칭 네트워크들을 이용하는 회로의 일 실시예를 도시한다.

[0021] 도 4는 N 세트들의 4 중간 도선들을 통해 4개 세트들의 4 N 도선들에 액세스하는 11 M 도선들을 구비한 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크(1-SN)를 이용하는 회로의 일 실시예를 도시한다.

[0022] 도 5는 더 적은 중간 도선들을 통해 4개 세트들의 4 N 도선들을 액세스하는 12 M 도선들을 구비한 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크를 이용하는 회로의 일 실시예를 도시한다.

[0023] 도 6은 더 강한 접속 특성을 가진 4개 세트들의 4 N 도선들에 액세스하는 12 M 도선들을 구비한 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크를 이용하는 회로의 일 실시예를 도시한다.

[0024] 도 7은 더 적은 스위치들을 갖는 감소된 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크의 일 실시예를 도시한다.

[0025] 도 8은 더 큰 사이즈의 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크의 일 실시예를 도시한다.

[0026] 도 9는 16 M 도선들을 갖는 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크의 일 실시예를 도시한다.

[0027] 도 10은 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크(2-SN) 및 도 9의 확장성 넌-블럭킹 스위칭 네트워크를 각각 이용하는 도 1의 4개 로직 회로들을 구비한 회로의 일 실시예를 나타내는 블럭도이다.

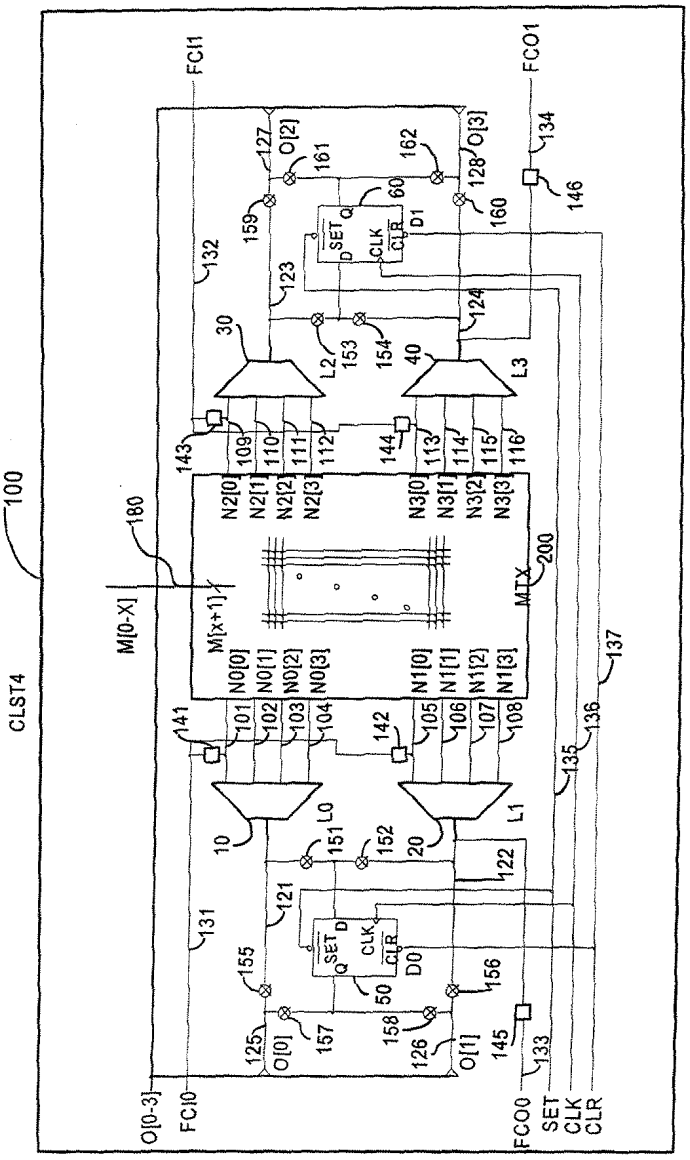
[0028] 도 11a는 도 10의 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크의 블럭도 실시예를 도시한다.

[0029] 도 11b는 도 11a의 스테이지-2 확장성 넌-블럭킹 스위칭 네트워크의 제 1 부분의 일 실시예를 도시한다.

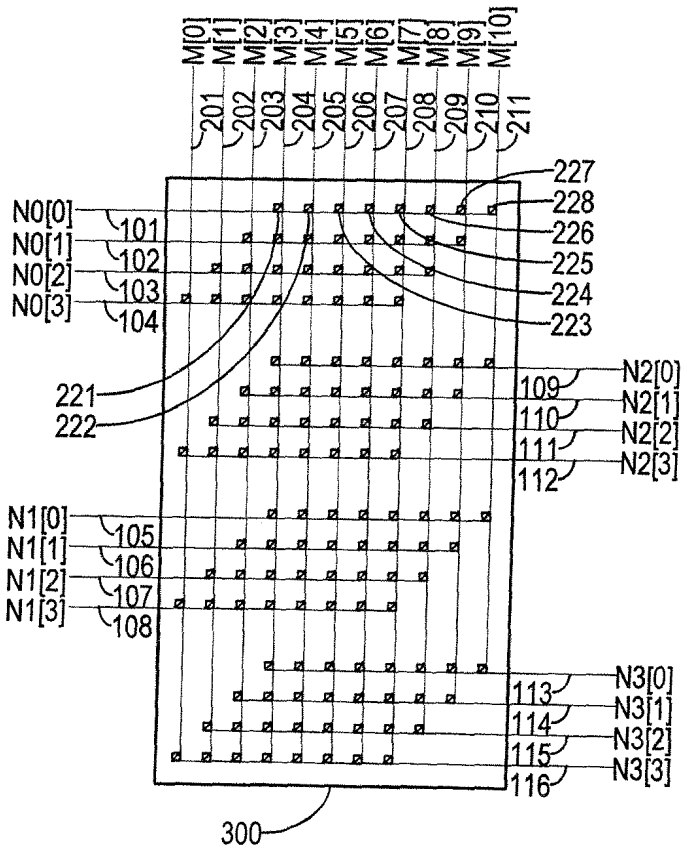
[0030] 도 12는 도 11a의 2-SN의 제 2 부분을 구현하는 스테이지-1 확장성 넌-블럭킹 스위칭 네트워크의 일 실시예를 도시한다.

## 도면

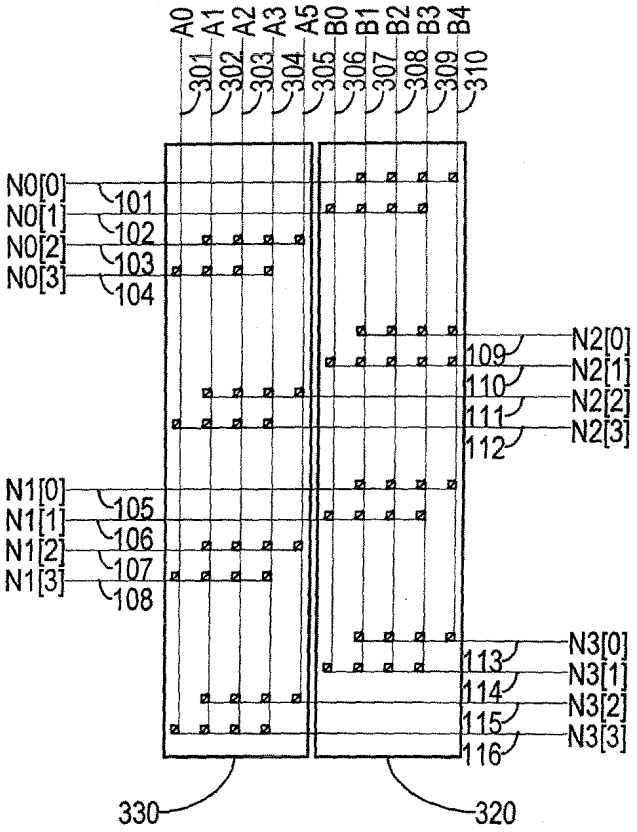
### 도면1



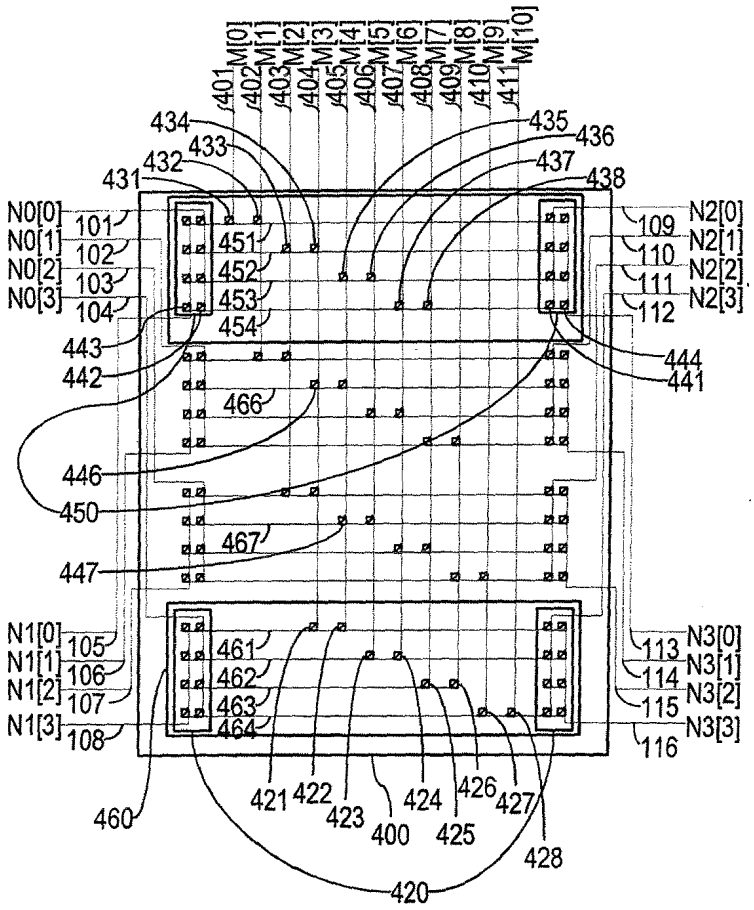
도면2


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
201
202
203
204
205
206
207
208
209
210
211
227
228
226
225
224
223
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
221
222
N2[0]
N2[1]
N2[2]
N2[3]
109
110
111
112
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
N3[0]
N3[1]
N3[2]
N3[3]
113
114
115
116
300

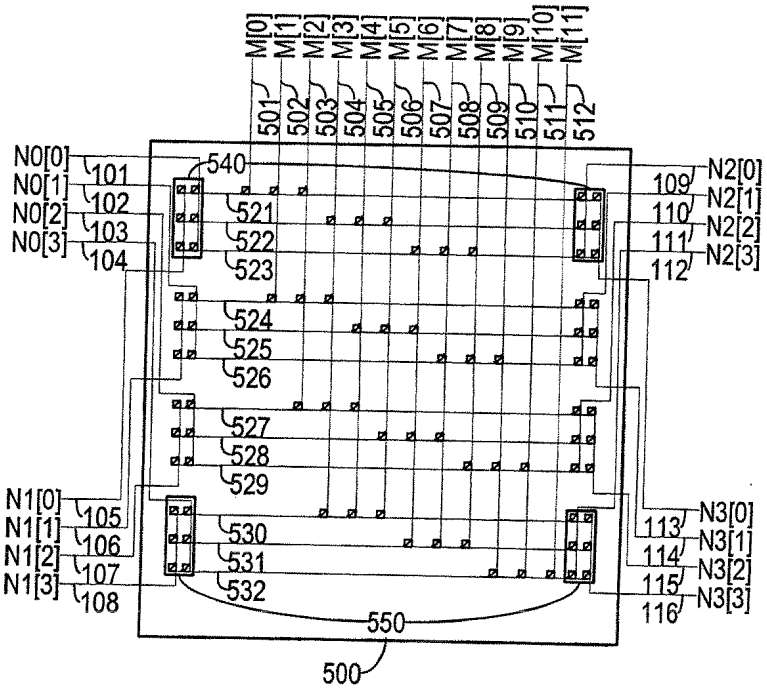
도면3


A0
A1
A2
A3
A5
B0
B1
B2
B3
B4
301
302
303
304
305
306
307
308
309
310
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
N2[0]
N2[1]
N2[2]
N2[3]
109
110
111
112
N3[0]
N3[1]
N3[2]
N3[3]
113
114
115
116
330
320

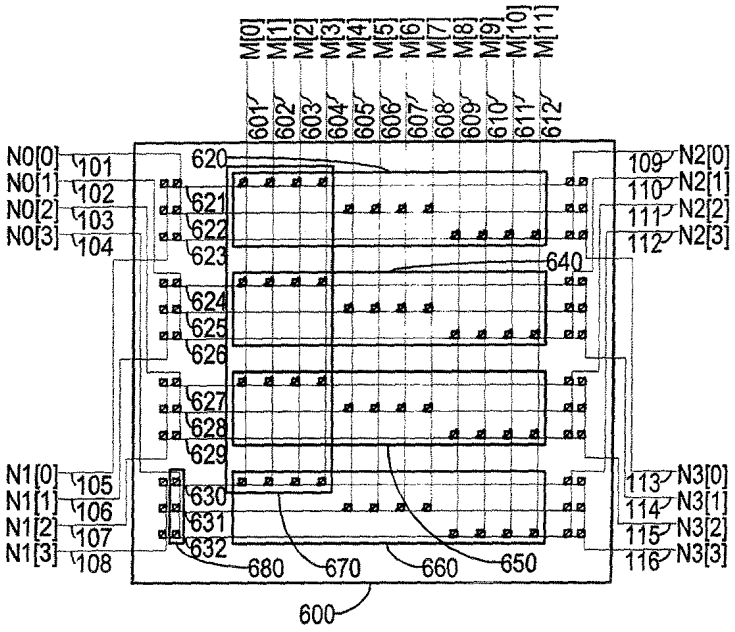
도면4


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
401
402
403
404
405
406
407
408
409
410
411
434
433
432
431
435
436
437
438
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
451
452
453
454
443
442
466
446
450
467
447
N2[0]
N2[1]
N2[2]
N2[3]
109
110
111
112
444
441
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
461
462
463
464
N3[0]
N3[1]
N3[2]
N3[3]
113
114
115
116
460
421
422
423
400
424
425
426
427
428
420

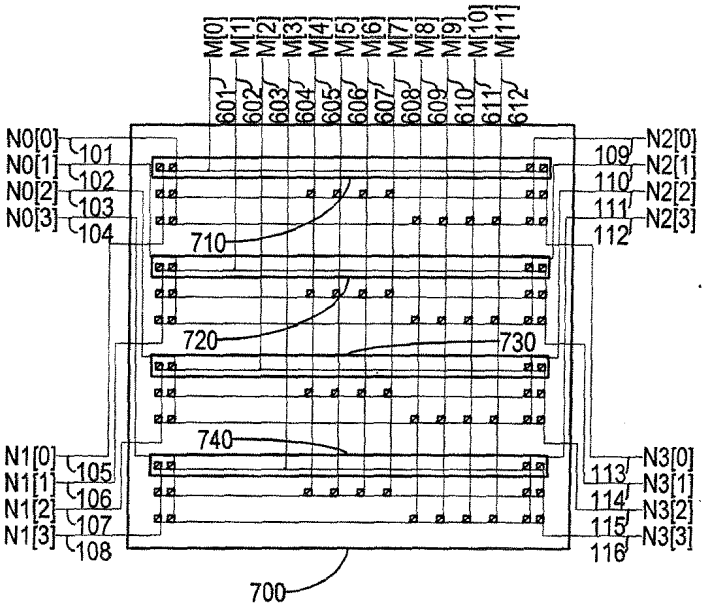
도면5


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
M[11]
501
502
503
504
505
506
507
508
509
510
511
512
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
540
521
522
523
524
525
526
527
528
529
530
531
532
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
550
500
109
110
111
112
N2[0]
N2[1]
N2[2]
N2[3]
113
114
115
116
N3[0]
N3[1]
N3[2]
N3[3]

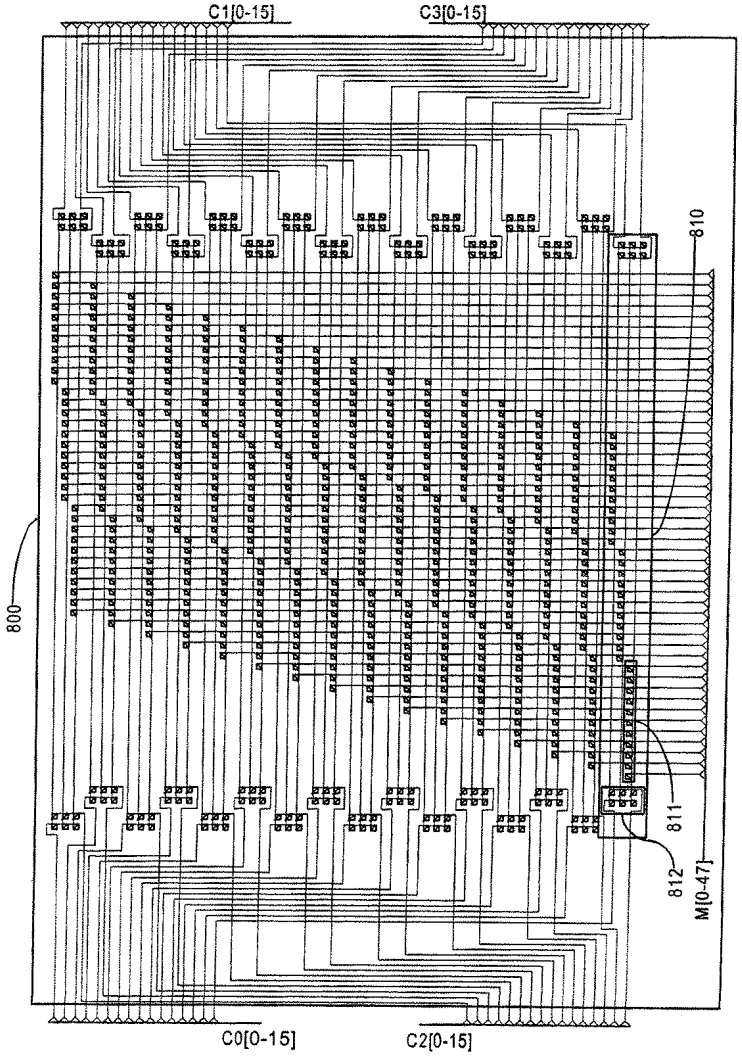
도면6


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
M[11]
601
602
603
604
605
606
607
608
609
610
611
612
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
620
621
622
623
624
625
626
627
628
629
630
631
632
640
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
680
670
660
650
600
109
110
111
112
N2[0]
N2[1]
N2[2]
N2[3]
113
114
115
116
N3[0]
N3[1]
N3[2]
N3[3]

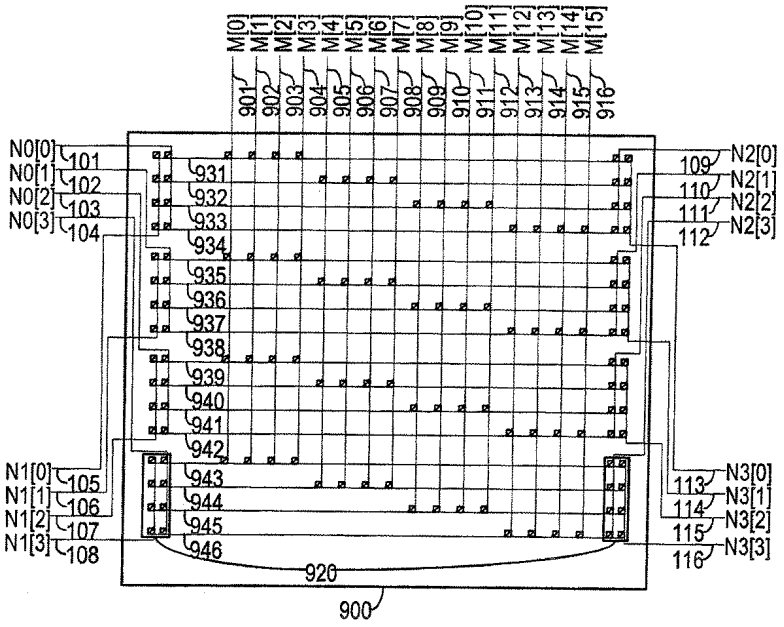
도면7


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
M[11]
601
602
603
604
605
606
607
608
609
610
611
612
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
710
720
730
740
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
700
N2[0]
N2[1]
N2[2]
N2[3]
109
110
111
112
N3[0]
N3[1]
N3[2]
N3[3]
113
114
115
116

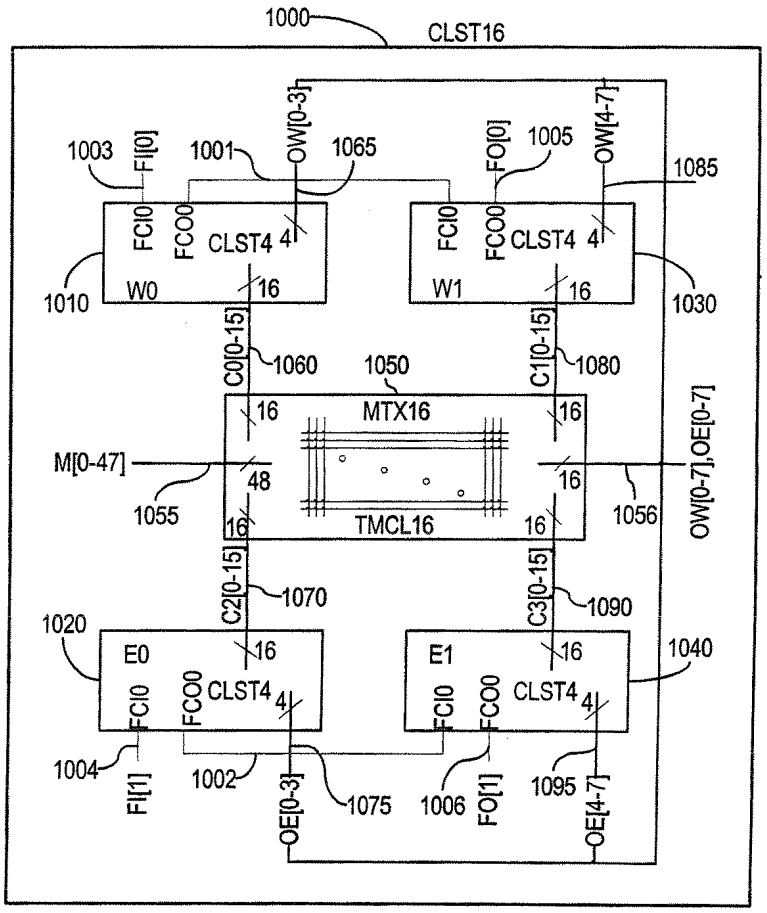
도면8



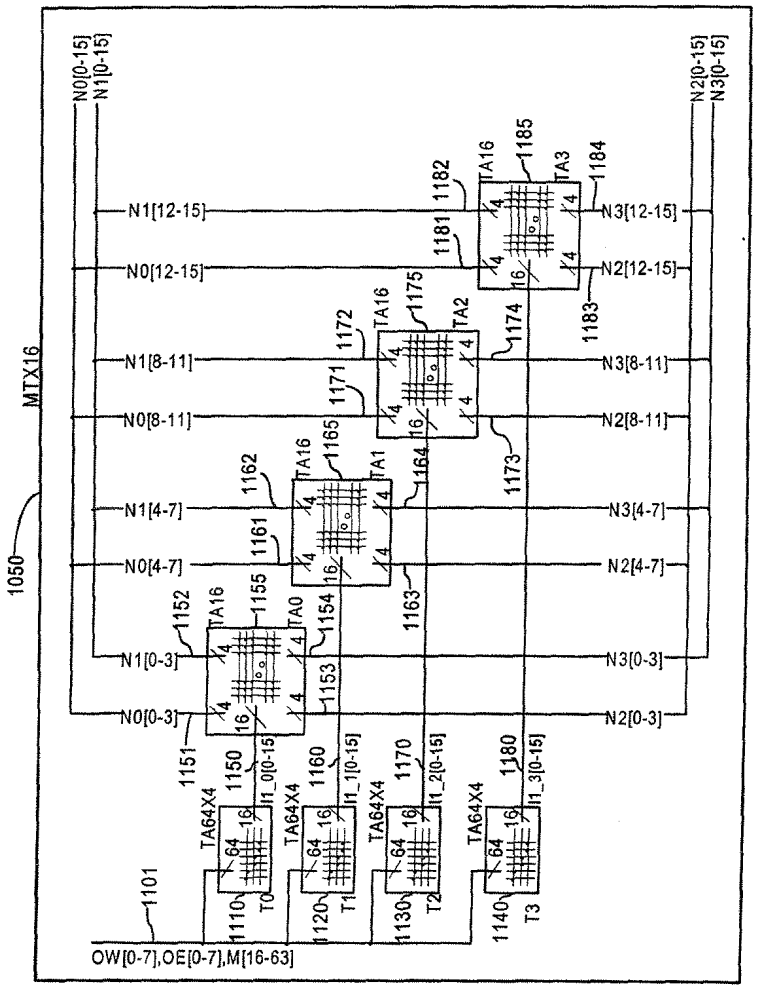
도면9


M[0]
M[1]
M[2]
M[3]
M[4]
M[5]
M[6]
M[7]
M[8]
M[9]
M[10]
M[11]
M[12]
M[13]
M[14]
M[15]
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
N0[0]
N0[1]
N0[2]
N0[3]
101
102
103
104
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
N1[0]
N1[1]
N1[2]
N1[3]
105
106
107
108
109
110
111
112
N2[0]
N2[1]
N2[2]
N2[3]
113
114
115
116
N3[0]
N3[1]
N3[2]
N3[3]
920
900

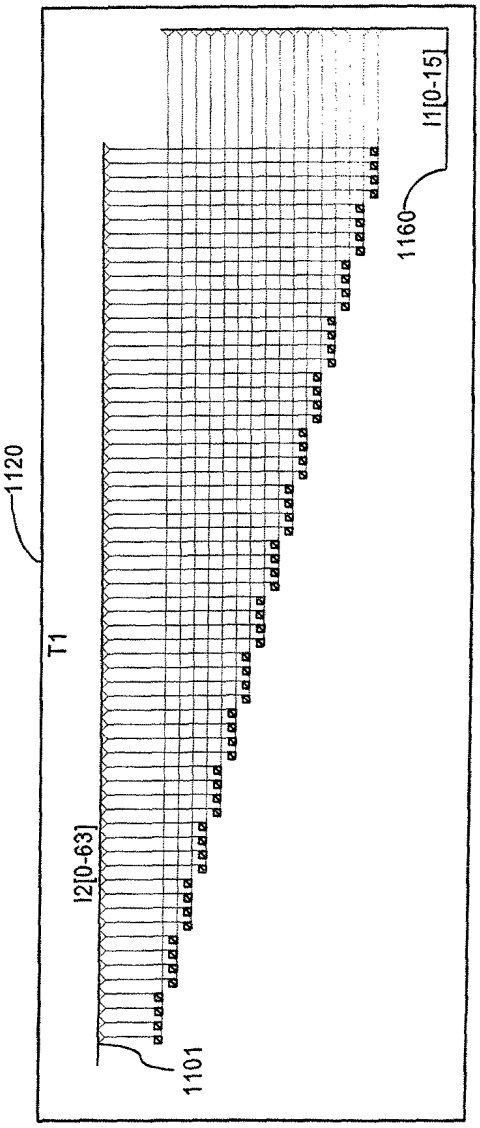
도면10


1000
CLST16
1003
FI[0]
1001
OW[0-3]
1065
FO[0]
1005
OW[4-7]
1085
FCI0
FCO0
CLST4
4
W0
W1
16
1010
1030
C0[0-15]
1060
1050
C1[0-15]
1080
MTX16
TMCL16
M[0-47]
48
1055
1056
OW[0-7],OE[0-7]
C2[0-15]
1070
C3[0-15]
1090
1020
E0
E1
1040
1004
FI[1]
1002
OE[0-3]
1075
1006
FO[1]
1095
OE[4-7]

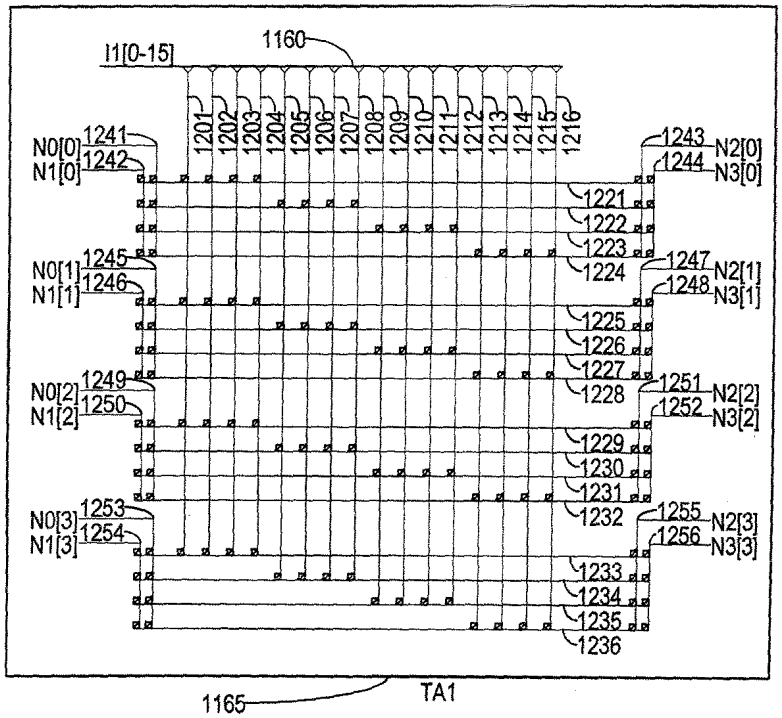
도면11a


MTX16
1050
N0[0-15]
N1[0-15]
N2[0-15]
N3[0-15]
N1[12-15]
N0[12-15]
N3[12-15]
N2[12-15]
N1[8-11]
N0[8-11]
N3[8-11]
N2[8-11]
N1[4-7]
N0[4-7]
N3[4-7]
N2[4-7]
N1[0-3]
N0[0-3]
N3[0-3]
N2[0-3]
TA16
TA3
TA2
TA1
TA0
TA64X4
OW[0-7],OE[0-7],M[16-63]

도면11b



도면12


I1[0-15]
1160
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
N0[0]
1241
N1[0]
1242
1243
N2[0]
1244
N3[0]
1221
1222
1223
1224
N0[1]
1245
N1[1]
1246
1247
N2[1]
1248
N3[1]
1225
1226
1227
1228
N0[2]
1249
N1[2]
1250
1251
N2[2]
1252
N3[2]
1229
1230
1231
1232
N0[3]
1253
N1[3]
1254
1255
N2[3]
1256
N3[3]
1233
1234
1235
1236
1165
TA1