

논문 2011-48SD-11-2

# 밀결합 전송선 상에서 전력 저감을 위한 코드워드 생성 기법

## ( A Codeword Generation Technique to Reduce Dynamic Power Consumption in Tightly Coupled Transmission Lines )

임 재 호\*, 김 덕 민\*, 김 석 윤\*\*

( Jae-Ho Lim, Deok-Min Kim, and Seok-Yoon Kim )

### 요 약

반도체 공정의 발달로 인해 칩의 집적도가 높아졌으며, 연결선 사이의 간격 또한 좁아지게 되었다. 그로 인해 연결선 내에 존재하는 커패시턴스와 인덕턴스가 증가하게 되었고, 특히 전역 연결선들에서는 자신의 그라운드 커패시턴스보다 인접한 다른 연결선과의 결합 커패시턴스가 더욱 커지는 경향을 보이게 되었다. 이러한 현상으로 인해 발생하는 유도성 결합과 용량성 결합은 인접한 연결선의 신호 간섭으로 심각한 문제를 야기할 수 있다. 본 논문에서는 추가적인 연결선을 이용하여 신호 무결성을 저해시키는 누화잡음을 제거하면서, 입력 데이터의 확률을 고려하여 동적 전력 소모를 최소화하는 코드워드 생성 기법을 제안하였다. 제안한 기법의 성능평가를 위해 FastCap 및 FastHenry 프로그램과 HSPICE를 이용하여 실험한 결과, 소모 전력에서 기존 기법보다 평균 15% 정도의 감소를 보임을 확인하였다.

### Abstract

As semiconductor process rapidly developed, the density of chips becomes higher and the space between adjacent lines narrows smaller. This trend increases the capacitance and inductance in interconnects and the coupling-capacitance of adjacent lines grows even bigger than the self-capacitance of themselves, especially in global interconnects. Inductive and capacitive coupling observed in these phenomena may cause serious problems in signal integrity. This paper proposes a codeword generation technique using extra interconnect lines to reduce the crosstalk caused by inductive and capacitive coupling and to reduce dynamic power consumption considering probability of input data. To estimate the performance of the proposed technique, the experimental results have been obtained using FastCap, FastHenry and HSPICE, and it has been shown that the power consumption using the proposed technique has yielded approximately 15% less than the results of the previous technique.

**Keywords :** bus-encoding , codeword, crosstalk, dynamic power consumption

## I. 서 론

반도체 공정 기술이 급격히 향상됨에 따라, IC 디바이스는 지속적으로 작아지고 있다. 이로 인해 인접한 연결선 간의 간격이 좁아지면서 내부 연결선 간의 신호

간섭을 간과할 수 없게 되었다. 신호 간섭은 연결선에서의 전력 소모, 최악의 지연 문제를 야기하기 때문에 이 문제를 해결하기 위해 많은 연구들이 진행되어 왔다. 그 중 한 연구 분야가 버스 인코딩 기법이다.

신호 간섭의 한 종류인 용량성 결합을 줄이는 대표적인 인코딩 기술은 버스 인버터 기술로 이전 데이터와 현재 데이터의 해밍 거리를 구하여 그 값이  $\frac{n}{2}$  보다 크면 현재의 데이터를 반전시켜 전송하는 기법이다<sup>[1]</sup>. 또한, 버스를 몇 개의 서브 버스로 나누어 각각 인코딩한 후 전송하는 기술이 제안되기도 하였다<sup>[5~7]</sup>. 이외에도

\* 학생회원, \*\* 정회원-교신저자,  
송실대학교 컴퓨터공학부

(School of Computer Science, Soongsil University)

※ 이 논문은 2011년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (2009-0074031)

접수일자: 2011년1월19일, 수정완료일: 2011년11월16일

데이터 전송 시 발생하는 천이를 확률적으로 분석함으로써, 누화잡음을 줄여 전력 소모를 최소화하려는 연구가 진행되었다<sup>[8-9]</sup>.

최근에는 용량성 결합뿐만 아니라, 유도성 결합을 고려한 버스 인코딩 기법들도 연구되었다. 기존 버스 인코딩 방법들은 RC회로에서의 최악의 천이 패턴을 고려하여 전력 소모나 지연을 최소화하는데 초점을 두었지만, [10], [11]에서는 RLC회로에서의 최악의 천이패턴을 RC회로에서의 최악의 천이패턴과 비교하면서 RLC회로에서의 지연 감소를 위한 버스 인버터 기법을 제안하였다. 위 기법들은 신호 결합을 줄이는데 초점을 두었지만, 이를 완벽히 제거하지는 못하여 전력 소모나 지연 측면에서는 효율적이지 못하다.

신호 결합을 완벽히 제거하기 위해서 인접한 라인에서 천이가 발생하지 않는 인코딩 구조가 제안 되었고<sup>[12]</sup>, 이를 확장한 대역폭이 큰 버스에서 확장이 용이한 코드워드 생성 기법이 제안되었다<sup>[13]</sup>. 이 두 기법은 누화잡음을 완벽히 제거하였지만, 자체(self-) 스위칭에 대해서는 고려하지 않아 전력 소모 측면에서 최적의 기법이라고 할 수는 없다.

본 논문에서는 인접한 라인에서의 신호 결합을 제거하고, 동시에 전력 소모를 최소화하기 위한 코드워드 생성 기법을 제안한다. 제안하는 기법은 [13]의 코드워드 생성 기법을 확장함으로써 확률적인 데이터 분포를 가지는 시스템에서 전력 소모를 더욱 줄일 수 있는 코드워드를 생성하는 기법이며, 연결선 모형을 통해 추출한 파라미터를 이용한 시뮬레이션을 수행하여 성능을 평가하였다.

본 논문의 순서는 다음과 같다. 서론에 이어 II장에서는 관련 연구로서 연결선의 특성 및 기존 연구에 대한 문제점을 분석하고, III장에서는 누화잡음 제거 및 전력 소모 저감을 위한 입력 데이터의 확률에 따른 코드워드 생성 기법을 제안한다. IV장에서는 실험을 통해 제안한 기법의 성능을 평가하고, V장에서 결론을 맺는다.

## II. 관련 연구

### 1. 천이 신호

코드워드 생성 기법을 설명하기 전에 몇가지 사항을 정의하기로 한다.

첫째, 연결선에서의 천이 방향 “↑”는 신호가 0에서 1로 천이되었음을 의미하며, “↓”는 신호가 1에서 0으로

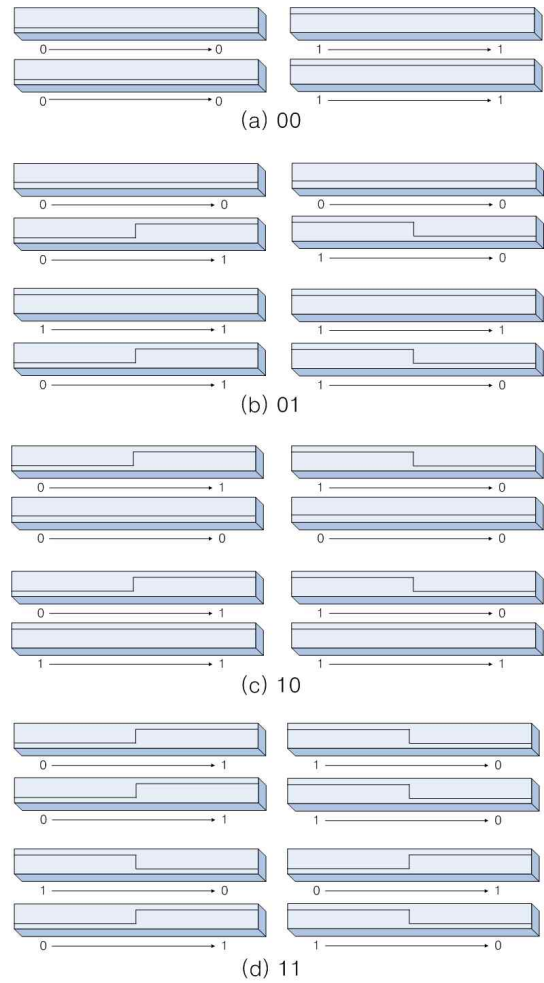


그림 1. 두 연결선에서 나타나는 천이 패턴  
Fig. 1. Transition patterns in two interconnects.

로 천이되었음을 의미한다. 또한 “-”는 천이가 일어나지 않았음을 의미한다.

둘째, 전송되는 코드워드에 나타난 “1”은 해당 라인의 신호가 천이되었음을 의미하며, “0”은 해당 라인에서 천이가 발생하지 않았음을 의미한다. 따라서 코드워드 “00100”은 가운데 연결선에서 천이가 발생되었음을 의미하며, “--↑--”과 같이 상승천이 하였거나, “--↓--”과 같이 하강천이 하였음을 의미한다.

위에서 정의하는 천이 신호를 생성하기 위해 XOR 게이트를 사용하여 인코더와 디코더를 설계하도록 한다.

두 연결선에서 나타날 수 있는 천이 패턴은 총 4가지로 그림 1처럼 표현할 수 있다.

그림 1에 나타난 것처럼 누화잡음을 발생시키는 천이 패턴은 “11”로 인접한 라인에서 천이가 발생하는 경우이다. 용량성 결합이 존재하는 두 연결선에서 신호가 같은 방향으로 천이하는 경우 최대 전압에 큰 영향을

주며, 서로 다른 방향으로 천이할 경우 지연과 전력 소모에 영향을 주게 된다. 따라서 누화잡음과 전력 소모의 저감을 위해서는 인접한 연결선에서의 천이가 발생하지 않는 코드워드를 생성하는 것이 중요하다.

### 2. 코드워드의 상관관계

1비트로 나타낼 수 있는 코드워드는 “0”과 “1”이다. 각각의 의미를 살펴보면, “0”은 전송 신호에서 천이가 발생하지 않았음을 의미하고(0→0, 1→1), “1”은 천이가 일어나는 경우를 의미한다(0→1, 1→0).

이 데이터를 기준으로 인접한 연결선에서 천이가 발생하지 않은 2비트 코드워드를 구해보면 다음과 같다. 1비트로 표현할 수 있는 코드워드 “0”의 최상위 비트에 “0”과 “1”을 추가하게 되면 “00”, “10”이라는 코드워드를 얻을 수 있으며, “1”의 최상위 비트에 “0”을 추가하게 되면 “01”이라는 코드워드를 얻을 수 있다. 1비트 코드워드 “1”의 최상위 비트에 “1”을 추가하지 못하는 것(즉, 코드워드 “11”)은 인접한 라인에서 천이되는 패턴

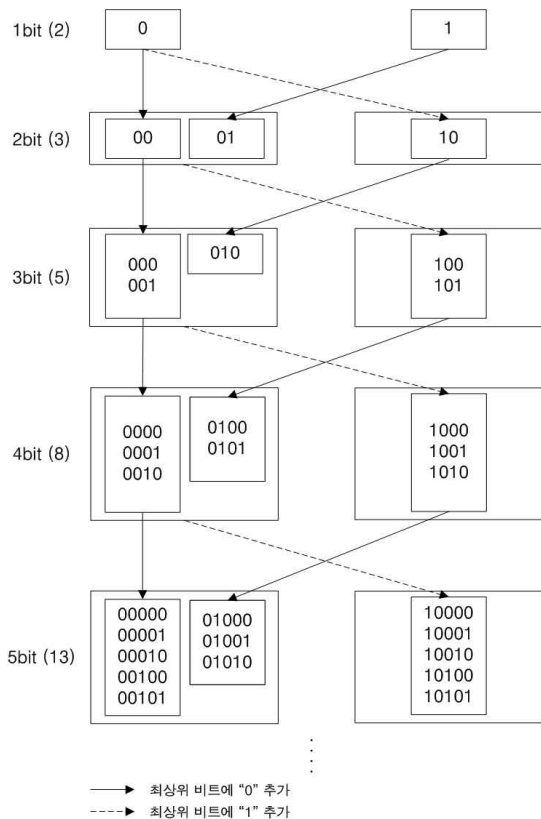


그림 2. 인접한 연결선에서 천이가 발생하지 않는 코드워드 생성 패턴

Fig. 2. Codeword generation pattern preventing transitions on adjacent interconnects.

을 제거하는 것이 본 논문의 목표이기 때문이다.

위와 같은 기준으로 2비트의 코드워드를 구해보면 “00”, “01”, “10”이 될 수 있다. 이러한 방식으로 가능한 3비트의 코드워드를 구해보면 “00”과 “01”의 최상위 비트에 “0”과 “1”을 추가하여 “000”, “001”, “100”, “101”을, “10”의 최상위 비트에 “0”을 추가하여 “010”의 코드워드를 생성한다.

생성 규칙을 살펴보면 N비트 코드워드의 최상위 비트가 “0”인 경우는 “0”과 “1”을 최상위 비트로 추가할 수 있으며, “1”일 경우는 “0”만을 최상위 비트로 추가하여 (N+1)비트 코드워드를 생성하게 된다.

그림 2는 코드워드 생성 규칙을 그림으로 나타낸 것이다.

그림 2와 같이, 인접한 연결선에서 천이가 발생하지 않는 코드워드는 위에서 설명한 규칙을 통해 생성할 수 있다. 이를 정리하면, n-비트에서 사용 가능한 코드워드의 개수가 피보나치 수열을 따르는 것을 확인할 수 있다.

$$F(1) = 2 \tag{1}$$

$$F(2) = 3$$

$$F(n) = F(n-1) + F(n-2) \quad (n > 3, n = \text{codeword의 bit})$$

예를 들면, 4비트의 입력 데이터(경우의 수 = 16)에 대해서 인접한 라인에서 스위칭이 일어나지 않도록 하기 위해서는 6비트의 코드워드로 인코딩 되어야 함을 알 수 있다(F(5) = 13, F(6) = 21).

위 내용을 기반으로 n비트의 입력 데이터가 인접 연결선에서 천이가 발생하지 않는 코드워드로 인코딩될 때, 몇 비트의 코드워드로 인코딩 되는지를 수식으로 나타내면 식 (2)와 같다.

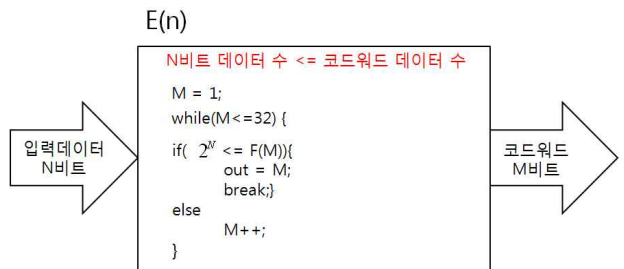


그림 3. 입력 데이터 N비트와 인코딩된 코드워드 M비트 관계

Fig. 3. Relation between N-bit input data and M-bit encoded codeword.

$$E(n) = \left. \begin{cases} n + \left\lfloor \frac{n}{2} \right\rfloor & (1 < n \leq 11) \\ n + \left\lfloor \frac{n-1}{2} \right\rfloor & (11 < n \leq 20) \\ n + \left\lfloor \frac{n}{2} \right\rfloor - 1 & (20 < n \leq 29) \\ n + \left\lfloor \frac{n-1}{2} \right\rfloor - 1 & (29 < n \leq 32) \end{cases} \right\} \quad (2)$$

식 (1)과 식 (2)의 관계를 그림 3으로 표현하였다.

### 3. 연결선의 전력 소모 및 최대 전압

연결선에서의 동적 전력 소모량을 알아보기 위해서 버스 구조를 모델링하고 각 천이 패턴에 대하여 전력 소모량을 측정하였다.

0.13um 공정에서 글로벌 연결선을 대상으로 하나의 연결선 너비가 0.6um, 높이가 0.45um, 길이가 1mm인 연결선 5개를 모형화하였으며, 연결선 간의 간격을 0.6um로 모형화하였다. FastCap과 FastHenry를 이용하여 RLC 값을 추출하고, HSPICE를 이용하여 아래의 3 가지 경우를 실험해 보았다.

- ▶ 한 연결선만 천이하는 경우("00100"): 상승천이 하는 경우("00↑00")와 하강천이 하는 경우("00↓00")에 대해 전력 소모량 측정
- ▶ 한 연결선만 천이할 경우: "10000", "01000", "0010", "00010", "00001"의 전력 소모량 측정
- ▶ 여러 연결선에서 천이가 발생할 경우, 천이하는 라인의 개수에 따른 전력 소모량 측정

위 3가지의 실험을 통해 다음과 같은 관찰을 얻을 수 있었다.

- ▶ 특정한 연결선이 상승 천이할 경우(0→1)와 하강 천이할 경우의 전력 소모량은 거의 동일하다.
- ▶ 한 연결선만 천이 할 경우, 안쪽 연결선에서 천이가 발생할 때보다 바깥쪽 연결선에서 천이가 발생할 때의 전력 소모가 더 적다.
- ▶ 천이하는 연결선의 수가 많을수록 전력 소모는 더 많다.

### 4. 기존 연구의 문제점

표 1은 기존 연구<sup>[13]</sup>에서 제안한 기법을 기반으로 실제 입력 데이터와 맵핑되는 코드워드와 맵핑되지 않는

표 1. 기존 연구 기법에 의해 맵핑된 코드워드와 맵핑되지 않은 코드워드

Table 1. Mapped and unmapped codewords in the previous technique

비트 수	입력 데이터	맵핑 가능한 코드워드		
		비트 수	맵핑되는 코드워드	맵핑되지 않는 코드워드
1bit	{0, 1}	2bit	{00, 01}	{10}
2bit	{00, 01, 10, 11}	3bit	{000, 001, 010, 100}	{101}
3bit	{000, 001, 010, 011, 100, 101, 110, 111}	4bit	{0000, 0001, 0010, 0100, 0101, 1000, 1001, 1010}	-
4bit	{0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111}	6bit	{000000, 000001, 000010, 000100, 000101, 001000, 001001, 001010, 010000, 010001, 010010, 010100, 010101, 100000, 100001, 100010}	{100100, 100101, 101000, 101001, 101010}
...	...	...	...	...
Nbit		Mbit		

코드워드를 표현하였다. 표 1을 살펴보면 인접한 연결선 사이에서의 천이는 완벽히 제거하였지만 천이하는 횟수에 있어서는 최적이지 못하다는 사실을 확인할 수 있다.

간단한 예로서, 4비트의 입력 데이터를 6비트로 인코딩하였을 때의 코드워드를 살펴보면 맵핑된 코드워드 중에서 "010101"을 확인할 수 있고, 맵핑되지 않은 코드워드 중에서 "101000"의 코드워드를 확인할 수 있다. "010101"의 코드워드를 101000"의 코드워드로 바꿔준다면 연결선에서 발생하는 천이 횟수를 줄여서 전력 소모를 최소화할 수 있게 된다.

또한 입력되는 데이터의 패턴을 확률적으로 알 수 있는 시스템에서는 빈번하게 들어오는 입력 데이터를 천이 발생이 적은 코드워드로 맵핑시켜 준다면 동적 전력 소모를 더 줄일 수 있을 것이다.

## III. 코드워드 생성을 위한 코딩 구조

### 1. 제안하는 코드워드 생성 기법

가. 코드워드 생성

천이가 발생하는 횟수를 기준으로 코드워드를 그룹으로 나누면 표 2와 같다. 각 비트별 코드워드는 10진수로 표현하며 마지막 단계에서 각 비트별 2진수로 변경하기로 한다.

천이가 N번 일어나는 코드워드 그룹을 Class N으로 정의한다. 천이가 발생하는 횟수를 Class별로 나누어 생성되는 사용 가능한 코드워드를 살펴보면, 식 (4)와 같은 규칙을 확인할 수 있다.

Class 0은 천이가 발생하지 않는 경우로, Class 0에 포함되는 코드워드를 10진수로 변환하면 {"0"}이 된다. Class 1은 천이가 1번 발생하는 경우로, Class 1에 포함되는 코드워드 집합을 10진수로 표시하면 초항이 1이고, 공비가 2인 등비수열의 집합이 된다.

Class 0과 Class 1의 경우 위 설명으로 간단히 정의할 수 있지만, Class 2이상의 경우 그림 1의 규칙을 잘 살펴보면 재귀적 방법을 이용해 쉽게 구할 수 있다.

N비트의 코드워드를 이용하여 (N+1)비트의 코드워드를 생성한다고 가정하면, N비트 코드워드의 최상위 비트에 "0"과 "1"을 각각 추가하여 인접한 연결선 사이에 천이가 발생하지 않는 (N+1)비트 코드워드를 생성할 수 있다. 아 사실을 근거로 천이가 M번 발생하는 (N+1)비트의 코드워드를 생성하기 위해서는 천이가 M

표 2. 천이 발생 횟수에 따른 2진수 및 10진수 코드워드 집합  
Table 2. Binary and decimal codeword depending on transition numbers

코드워드 비트 수	2진수 코드워드	10진수 코드워드
1bit	{0}, {1}	{0}, {1}
2bit	{00}, {01, 10}	{0}, {1, 2}
3bit	{000}, {001, 010, 100}, {101}	{0}, {1, 2, 4}, {5}
4bit	{0000}, {0001, 0010, 0100, 1000}, {0101, 1001, 1010}	{0}, {1, 2, 4, 8}, {5, 9, 10}
6bit	{000000}, {000001, 000010, 000100, 001000, 010000, 100000}, {000101, 001001, 001010, 010001, 010010, 010100, 100001, 100010, 100100}	{0}, {1, 2, 4, 8, 16, 32}, {5, 9, 10, 17, 18, 20, 33, 34, 36}
n-bit	...	...

번 발생하는 N비트 코드워드의 최상위 비트에 "0"을 추가하거나 천이가 (M-1)번 발생하는 N비트 코드워드의 최상위 비트에 "1"을 추가하면 된다. 최상위 비트에 "1"을 추가할 때 고려해야 할 사항은 인접한 연결선 사이에 천이가 발생하지 않아야 하므로 (M-1)번 발생하는 N비트의 코드워드 중 최상위 비트가 "1"인 경우는 제외되어야 한다.

특정 비트의 Class N의 집합을 수식적으로 표현하기 위해서, 함수  $CD(n,x)$ 를 n비트로 인코딩된 코드워드 로 Class x에 해당하는 집합으로 정의한다.

위에서 설명한 규칙을 이용하여 각 Class를 표현하면 식 (3)과 같이 나타낼 수 있다.

$$\begin{aligned}
 \text{Class 0: } CD(n,0) &= \{0\}, (n \geq 1) \\
 \text{Class 1: } CD(n,1) &= \{x|2^{k-1} \in x, (1 \leq k \leq n, k \text{는 정수})\} \\
 \text{Class 2: } CD(n,2) &= \left\{ \left\{ \begin{aligned} &\{x|x = k+2^{n-1}, k \in CD(n-2,1)\}, (n \geq 3) \\ &\cup CD(n-1,2) \end{aligned} \right\} \right\} \\
 \text{Class 3: } CD(n,3) &= \left\{ \left\{ \begin{aligned} &\{x|x = k+2^{n-1}, k \in CD(n-2,2)\}, (n \geq 5) \\ &\cup CD(n-1,3) \end{aligned} \right\} \right\} \\
 \text{Class 4: } CD(n,4) &= \left\{ \left\{ \begin{aligned} &\{x|x = k+2^{n-1}, k \in CD(n-2,3)\}, (n \geq 7) \\ &\cup CD(n-1,4) \end{aligned} \right\} \right\} \\
 &\vdots \\
 \text{Class } m: CD(n,m) &= \left\{ \left\{ \begin{aligned} &\{x|x = k+2^{n-1}, k \in CD(n-2,m-1)\}, (n \geq 2m-1) \\ &\cup CD(n-1,m) \end{aligned} \right\} \right\}
 \end{aligned}
 \tag{3}$$

위 수식을 사용하기 위해서는 코드워드의 각 비트별 최고 Class를 확인할 수 있어야 한다. 인접한 라인에서 천이가 발생하지 않는 코드워드를 생성하기 때문에, 천

표 3. 코드워드 비트별 나타낼 수 있는 최고 Class  
Table 3. The highest class available for codeword bit number.

코드워드 비트수	최고 Class를 갖는 코드워드	Class
1	1	1
2	10 or 01	1
3	101	2
4	1010 or 0101	2
5	10101	3
6	101010 or 010101	3
7	1010101	4
8	10101010 or 01010101	4
N	...	...

표 4. 4비트로 인코딩된 코드워드의 Class별 집합  
Table 4. Class set of codewords encoded in 4-bits.

코드워드	Class 0	Class 1	Class 2
4bit	0000(0)	0001(1), 0010(2), 0100(4), 1000(8)	0101(5), 1001(9), 1010(10)

이가 발생한다는 의미의 “1”이 존재할 경우 인접한 라인에서는 “1”이 올수 없다는 사실을 근거로 각 비트별 가능한 최고 Class를 구할 수 있다.

$$Class(n) = \left\{ x \mid 0 \leq x \leq \left\lfloor \frac{n+1}{2} \right\rfloor \right\}, \quad (4)$$

$n = \text{codeword bit,}$   
 $\lfloor n \rfloor = n\text{보다크지않는가장큰정수}$

식 (2)와 식 (3) 그리고 식 (4)를 이용하여 입력 데이터 3비트를 인코딩하여 코드워드를 구하여 보자. 우선, 식 (2)를 이용하여 3비트 입력 데이터가 몇 비트의 코드워드로 변환 가능한지를 알아보면  $E(x) = 4$ 이므로 4비트 코드워드로 인코딩 되어야 한다는 사실을 알 수 있다.

4비트 코드워드가 가질 수 있는 최고 Class의 집합이 몇인지 확인하기 위해 식 (4)를 이용하면,  $Class(4) = \left\lfloor \frac{4+1}{2} \right\rfloor = \lfloor 2.5 \rfloor = 2$ 이므로 Class 0, Class 1, Class 2의 집합을 가진다. 마지막으로 식 (3)을 이용하여 각각의 Class 집합을 표시하면 표 4와 같이 나타낼 수 있다.

나. 천이 패턴에 따른 상대적 전력 소모비

연결선의 특성을 알아보기 위해 수행한 모의실험을 통해 상대적으로 Class가 높은 집합의 코드워드가 전력 소모량이 크다는 사실과 동일 Class 내(동일한 천이 수를 발생하는 코드워드)에서도 천이 패턴에 따라 소모되는 전력이 다르다는 것을 확인할 수 있었다.

위 두 사실을 근거로 코드워드에 대한 상대적인 전력 소모비를 구해본다.  $x_n$ 은 최하위 비트를 기준으로 n번째 위치하는 연결선의 값이라고 하면,  $x(x_n, x_{n-1}, x_{n-2}, \dots, x_1, x_0)$ 으로 표현할 수 있으며, 아래의 수식을 이용하여 n비트 코드워드 천이패턴에 따른 전력 소모량의 상대적인 비를 구할 수 있다.

$$PLR(x, n) = \quad (5)$$

$$a \cdot x_n + (a+1) \cdot x_{n-1} + (a+2) \cdot x_{n-2} + \dots$$

$$+ \dots + (a+1) \cdot x_1 + a \cdot x_0, \quad \left( a = \left\lfloor \frac{n}{2} \right\rfloor \right)$$

표 5. 생성된 코드워드의 패턴에 따른 상대적 전력 소모비

Table 5. Relative power consumption ratio for generated codeword patterns.

Class	코드워드	수식	PLR(x,n)
0	0000(0)	2·0+3·0+3·0+2·0	0
1	0001(1)	2·0+3·0+3·0+2·1	2
	0010(2)	2·0+3·0+3·1+2·0	3
	0100(4)	2·0+3·1+3·0+2·0	3
	1000(8)	2·1+3·0+3·0+2·0	2
2	0101(5)	2·0+3·1+3·0+2·1	5
	1001(9)	2·1+3·0+3·0+2·1	4
	1010(10)	2·1+3·0+3·1+2·0	5

위 수식을 4비트 코드워드에 적용시켜 표 5로 표현하면 다음과 같다.

입력 데이터를 확률적으로 계산하여 빈번하게 입력되는 데이터를 기준으로 내림차순으로 정렬한다. 또한, 식 (5)로부터 구해진 코드워드의 상대적인 전력 소모비를 오름차순으로 코드워드를 정렬하면 가장 빈번하게 입력되는 데이터가 인접한 연결선에서 천이가 발생하지 않으며 전력 소모가 최소가 되는 코드워드와 맵핑시킬 수 있다.

IV. 실험 및 결과

본 논문에서는 0.13um 공정에서의 글로벌 연결선을 기준으로 하나의 연결선 너비를 0.6um, 높이 0.45um, 길이 1mm, 간격을 0.6um로 하여 다양한 비트의 버스 구조를 모형화하였다. FastCap을 이용하여 그라운드 커패시턴스와 커플링 커패시턴스를 추출하였으며, 셀프 인덕턴스와 커플링 인덕턴스는 FastHenry를 이용하여 추출하였다.

이렇게 추출된 저항(63.8585Ω), 커패시터(3.112e-15F), 인덕터(1.60969e-9H) 값을 이용하고 전원 전압 1v, transition time 1ps, segment는 1개로 가정하여, 이것을 넷리스트 형태로 만들어 각각의 입력 데이터 패턴에 대한 전력소모를 측정하였으며, 제안한 코드워드 생성 기법의 성능을 평가하기 위해서 다양한 확률 분포를 가지는 입력 데이터를 이용하였다. 각 비트별 입력 데이터를 무작위로 500개 생성하였다. 생성된 데이터를 바탕으로 기존 연구 기법<sup>[13]</sup>과 제안한 연구 기법을 적용하여 해당 코드워드를 맵핑 시켰으며, 코드워드 간의 천이 패턴 및 전력 소모량을 분석하였다.

3비트 입력 데이터 500개를 사용하여 코드워드를 생

표 6. 3비트 입력 데이터의 패턴에 따른 코드워드 생성

Table 6. Codeword generation for 3-bit input data patterns

입력 데이터	Case I			Case II			Case III		
	발생 횟수	기존 기법	제안 기법	발생 횟수	기존 기법	제안 기법	발생 횟수	기존 기법	제안 기법
000	68	0000	0010	60	0000	1001	60	0000	0010
001	78	0001	0000	50	0001	0101	63	0001	0100
010	65	0010	1000	63	0010	0100	50	0010	1010
011	52	0100	1001	75	0100	0000	70	0100	0001
100	60	0101	0100	73	0101	0001	80	0101	0000
101	58	1000	0101	70	1000	1000	65	1000	1000
110	67	1001	0001	45	1001	1010	55	1001	0101
111	52	1010	1010	64	1010	0010	57	1010	1001

표 7. 3비트 입력 데이터에 대한 기존의 기법과 제안하는 기법의 전력 소모 차이

Table 7. Power consumption comparison of previous technique and proposed technique for 3-bit input data

Case	기존 기법으로 생성된 코드워드	제안하는 기법으로 생성된 코드워드	전력 감소 비율
I	5.7159E-03	5.5299E-03	3.25%
II	5.9170E-03	5.3694E-03	9.25%
III	5.9136E-03	5.8498E-03	1.07%
평균	5.8488E-03	5.5830E-03	4.54%

성한 결과를 표 6에 나타냈다.

이 코드워드를 바탕으로 천이 패턴을 분석하고, 시뮬레이션을 통해 평균 전력 소모를 구할 수 있었으며, 실험 결과를 표 7에 나타냈다.

생성된 3비트 입력 데이터에 대한 전력 소모의 감소량은 최대 10%에서 최소 1%로, 평균 약 4~5%정도 감소되었음을 확인할 수 있다.

또한, 좀 더 일반적인 경우를 위해 비트 수에 따라 기존 기법으로 생성된 코드워드와 제안하는 기법으로 생성된 코드워드 전송에 대한 전력 소모량을 측정하였다. 각 비트별 500개의 랜덤 데이터를 생성하여 테스트한 결과 표 8과 같은 전력 소모량을 확인할 수 있었다.

표 8을 비교 분석해 보면, 기존에 제안된 기법으로 생성된 코드워드를 전송하는 방법 보다 제안한 기법으로 생성된 코드워드를 전송하는 방법이 전력 소모 측면에서 더 효율적임을 확인할 수 있다. 또한, 전송되는 비트 폭이 클수록 제안하는 기법의 성능이 더욱 효율적임

표 8. 각 비트별 입력 데이터에 대한 기존의 기법과 제안하는 기법의 전력 소모 차이 (500개의 입력 데이터)

Table 8. Power consumption comparison of previous technique and proposed technique for number of input bits (500 input data)

입력 데이터 비트수	기존 기법으로 생성된 코드워드	제안하는 기법으로 생성된 코드워드	전력 감소 비율
2	3.2785E-03	3.2139E-03	1.97%
3	5.8488E-03	5.5830E-03	4.54%
4	7.8667E-03	6.4314E-03	18.24%
5	1.1172E-02	9.3880E-03	15.96%
6	1.1748E-02	8.4541E-03	28.03%
7	1.4359E-02	1.1146E-02	22.37%
8	1.8455E-02	1.3667E-02	25.94%

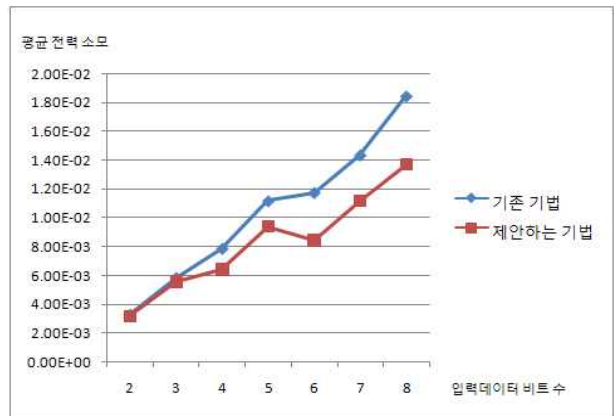


그림 4. 비트 수에 따른 평균 전력 소모량 비교 (입력 데이터 수 : 500)

Fig. 4. Comparison of average power consumption for number of bit lines (number of input data : 500).

을 확인할 수 있다. 표 8을 그래프로 나타내면 그림 4와 같다.

## V. 결 론

본 논문은 밀결합된 연결선에서 데이터 전송 시에 발생하는 누화 잡음 및 전력 소모를 저감하는 기법을 제안하였다. 기존 연구에 의해 개발된 기법은 추가적인 비트라인을 이용하여 누화잡음을 최소화시켰지만, 전력 소모 측면에서 최적이지 못하였다. 본 논문에서는 이를 보완하기 위하여 연결선의 천이 특성을 고려함으로써 누화잡음 및 전력 소모를 최소화하는 코드워드 생성 기법을 제안하였다.

제안된 코드워드 생성기법은 입력 데이터의 발생 빈

도를 고려할 수 있도록 고안되었으며, 무작위로 생성된 입력 데이터에 대해 제안한 기법으로 코드워드를 생성하여 실험하였고, 코드워드간의 천이 패턴을 분석하였다. MIT에서 개발한 FastCap과 FastHenry를 이용하여 저항, 커패시턴스, 인덕턴스 등의 연결선의 특성치를 추출하였으며, HSPICE로 시뮬레이션을 수행하였다.

실험을 통하여 제안하는 기법이 기존 연구 기법에 비해 전력 소모 측면에서 효율적이며, 이는 입력 데이터 비트 폭이 넓을수록 더욱 향상됨을 확인하였다. 본 논문에서 제안하는 밀결합 전송선에서 전력 저감을 위한 코드워드 생성 기법은 버스 인코딩 분야 뿐 아니라, 신호의 무결성을 보장해야 하는 모든 분야에 널리 적용될 수 있을 것으로 예상된다.

### 참 고 문 헌

- [1] Mircea R. Stan and Wayne P. Burleson, "Bus-Invert Coding for Low-Power I/O," IEEE Transactions on Large Scale Integration Systems Vol. 3, No.1, pp. 49-58, 1995.
- [2] Mahdi Moradinasab, Siamak Mehrnami and Rasul Yousefi, "A modified bus invert method for the submicron technology," in Proc. Design and Test Workshop, pp. 1-3, 2009
- [3] B. Victor and K. Keutzer, "Bus encoding to prevent crosstalk delay," in Proc. ICCAD, pp. 57-63, 2001.
- [4] Kuang-Chin Cheng, Jing-Yang Jou, "Crosstalk-avoidance coding for low-power on-chip bus," in Proc. IEEE International Conference on Electronics, Circuits and Systems, pp. 1051-1054, 2008.
- [5] K. S. Sainarayanan, C. Raghunandan and M. B. Srinivas, "Delay and Power Minimization in VLSI Interconnects with Spatio-Temporal Bus-Encoding Scheme," in Proc. IEEE Computer Society Annual Symposium on VLSI, pp. 401-408, 2009
- [6] Youngsoo Shin, Soo-Ik Chae and Kiyoun Choi, "Partial Bus-Invert Coding for Power Optimization of Application-Specific Systems," IEEE Transactions on very large scale integration systems, Vol. 9 No. 2, pp. 377-383, 2001.
- [7] Sunpack Hong, Unni Narayanan, Ki-Seok Chung, and Taewhan Kim, "Bus-Invert Coding for Low-Power I/O - A Decomposition Approach," in Proc, 43rd IEEE Midwest Symp. on Circuits and Systems, pp. 750-753, 2000
- [8] Avnish R. Brahmbhatt, Jingyi Zhang, Qinru Qiu, Qing Wu, "Adaptive Low-Power Bus Encoding Based on Weighted Code Mapping," in Proc. IEEE International Symposium on Circuits and Systems, pp. 1739-1742, 2006.
- [9] 여준기, 김태환, "저전력과 크로스톡 지연 제거를 위한 버스 인코딩," 정보과학회논문지 제29권 제12호, pp. 680-686, 2002.
- [10] Shang-Wei Tu, Yao-Wen Chang and Jing-Yang Jou, "RLC Coupling-Aware Simulation and On-Chip Bus Encoding for Delay Reduction," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 25, No. 10, pp. 2258-2264, 2006.
- [11] Jiun-Sheng Hwang, Shang-Wei Tu and Jing-Yang Jou, "ON-CHIP BUS ENCODING FOR LC CROSS-TALK REDUCTION," in Proc. IEEE VLSI-TSA International Symposium on VLSI Design, Automation and Test, pp. 233-236, 2005.
- [12] P. Subrahmanya, R. Manimegalai and V. Kamakoti, "A Bus Encoding Technique for Power and Cross-talk Minimization," in Proc. IEEE International Conference on VLSI Design, pp. 443-448, 2004.
- [13] Kedar Karmarkar and Spyros Tragoudas, "Scalable Codeword Generation for Coupled Buses," in Proc. Design, Automation & Test in Europe Conference & Exhibition, pp. 729-734, 2010.



저 자 소 개



임 재 호(학생회원)  
2009년 숭실대학교 컴퓨터학부  
학사 졸업  
2011년 숭실대학교 컴퓨터학부  
석사 졸업  
2011년~현재 LG 전자 연구원

<주관심분야 : VLSI, 회로해석 및 설계, Signal Integrity, 설계 자동화>



김 석 윤(정회원)-교신저자  
1980년 서울대학교 공대  
전기공학과 학사 졸업  
1990년 University of Texas at  
Austin 전기, 컴퓨터학과  
석사 졸업  
1993년 University of Texas at  
Austin 전기, 컴퓨터학과  
박사 졸업

1982년~1987년 한국전자 통신연구소 연구원  
1993년~1995년 Motorola Inc. Senior Staff  
Engineer

1995년~현재 숭실대학교 컴퓨터학부 교수  
<주관심분야 : 설계 자동화, VLSI 회로해석 및  
설계>



김 덕 민(학생회원)  
2008년 수원대학교 컴퓨터학과  
학사 졸업  
2010년~현재 숭실대학교 컴퓨터  
학부 석사

<주관심분야 : VLSI, 회로 해석  
및 설계, 설계 자동화>