

비동기식 회로의 신호 천이 감소 방법

이원철, 이제훈, 조경록
충북대학교 정보통신공학과

Signal Transition Reducing method of Asynchronous Circuits

Wonchul Lee , Jehoon Lee and Kyoung Rok Cho
Communication & System Design Lab. , Chungbuk National University
E-mail : wclee@hbt.chungbuk.ac.kr

Abstract

본 논문은 DI(delay insensitive) 지연 모델을 적용한 비동기 회로의 데이터 전송시 발생하는 신호 천이의 수를 감소시키기 위한 새로운 데이터 인코딩 기법과 신호 천이 방법을 제시한다. DI 지연 모델을 적용한 비동기 시스템은 배선 지연에 관계없이 동작이 필요한 모듈에만 데이터와 핸드셰이크를 위한 이벤트 신호를 전송하는 장점을 갖는다. 그러나 신호의 유효성과 동작 완료 검출을 위해 듀얼레일 데이터 인코딩이 필요하며 이는 비동기 회로의 크기를 증가시키고 이로 인해 전력 소비가 증가한다. 전력 소비를 감소시키기 위해 신호 천이의 수를 줄여야 하며, 본 논문에서는 제안한 신호 천이 기법을 적용하여 실험적으로 약 21%의 전력 소비 감소 결과를 얻었다.

I. 서론

비동기 회로는 시스템 클럭에 의해 전체 모듈이 구동되는 대신 동작이 필요한 모듈로의 요구 (Request)신호와 동작 완료를 지시하는 확인 (Acknowledge)신호의 두 이벤트 신호에 따른 핸드셰이킹 프로토콜을 사용하여 구동된다[1].

비동기 회로내의 각 모듈은 request 신호에 의해 연산이 요구된 후 이를 완료하고 이를 인접 모듈에 지시해야 한다. 이를 위해 연산 시간을 미리 추정하여 지연 시간만큼의 지연 소자를 삽입하는 Delay Sensitive 설계와 듀얼 레일 데이터를 수신하여 각 모듈의 동작이

완료될 때 이를 직접 검출하는 Delay Insensitive 설계 방법을 사용한다.

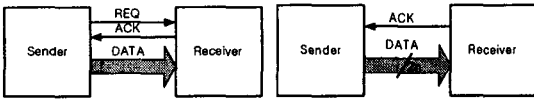
번들 데이터 (Bundled-data) 지연 모델의 경우 데이터를 전송할 때 미리 모듈내의 연산에 필요한 지연 시간을 추정하여 단선 인코딩(Single-Rail Encoding)을 사용하여 데이터를 전송하고, 추정된 지연 시간이 적용된 요청신호를 완료 신호로 전송한다. 따라서 회로의 크기는 동기식과 비교하여 거의 비슷하나 레이어아웃 후의 시스템내의 모든 모듈의 동작 시간을 추정하기가 힘들고 또 각 모듈의 최악 지연 시간을 적용해야 하기 때문에 이로 인한 성능상의 제한이 발생한다. 그러나 DI(Delay Insensitive) 지연 모델을 적용한 시스템의 경우 데이터를 통해 데이터의 유효성을 판단하는 2 선식 인코딩(Dual-Rail Encoding) 기법을 통해 전송하여 모듈의 마지막 출력 단에서 데이터의 유효성을 검출하여 완료 신호를 생성하며 회로의 크기는 크게 증가하나 각 모듈의 지연 시간을 추정할 필요가 없이 각 모듈은 자기동기식으로 동작하기에 설계가 편리하다.

본 논문은 비동기 데이터 전달 방식에 따른 데이터 인코딩에서 2 선식 인코딩 방식의 신호천이 감소 방안을 제시하였다. 이를 위해 기존 듀얼 레일 방식에서 사용되는 RZ (Return-to-Zero) 방식에 단선식에서 사용되는 NRZ(Non-Return-to-Zero)방식을 부분적으로 적용함으로써, 이전보다 더 적은 신호천이 감소를 얻을 수 있는 파이프라인 구조를 제안하였다. 또 제안된 신호 천이 기법과 기존의 신호 천이 기법의 성능 평가를 위해 덧셈기를 설계하여 소비전력에 대한 실험을 통한 성능 평가를 수행하였다.

II. 데이터 전달 채널 방식

2.1 Bundled Data channel vs. Dual-Rail Channel

비동기 회로에 있어서 데이터 전송은 미리 전송 지연 시간을 측정 한 후 데이터의 유효 여부를 지시하는 bundled data channel 과 회로의 지연 시간에 관계없이 연산 완료 시점을 검출한 후 완료 신호를 전송하는 dual-rail channel 로 구분된다[3].



(a) Bundled data channel. (b) Dual-Rail channel.

그림 1. 데이터 전달 채널 방식

그림 1 은 비동기 설계 방식에서 블록간 데이터 전송 방식을 나타낸다. 그림 1(a)는 번들 데이터(Bundled data) 전달 방식으로 1 비트 데이터를 하나의 데이터 라인으로 전달하고, 요구 신호는 데이터의 전송 지연 시간을 고려하여 같이 전송된다. 이와 달리, 그림 1(b)은 1 비트의 데이터를 두 개의 데이터 라인을 통해 전송하는 듀얼레일 인코딩 기법을 나타낸다. 이는 현재 데이터 라인의 데이터의 유효성을 수신측에 지시하기 위해 사용된다. 즉 데이터 전송시 전송할 데이터의 값과 그 보수값을 전송한다. 따라서 논리 신호 "0" 을 전송하기 위해 두 데이터 라인은 "01" 전송하고 논리 신호 "1" 을 전송하기 위해 두 데이터 라인은 "10" 을 전송한다. 또 두 데이터 라인이 모두 "1" 인 경우 데이터가 유효하지 않음을 의미하고 모두 "0" 인 경우는 사용하지 않는다.

표 1. 단선 방식

비트	코드
0	0
1	1

표 2.2선식 방식

비트	코드
0 0	Invalid
0 1	0
1 0	1
1 1	Not Use

표 1 과 2 는 단선 (single-rail) 방식과 2 선식(Dual-Rail) 방식의 데이터 표현 방식을 나타낸다.

데이터 전송시, 단선 방식과 2 선식 방식은 데이터 신호선의 성질에 따라 다른 전달 방식을 갖는다. 단선 방식은 데이터의 요구(request)신호 전송시 전송 지연을 고려되어 있기 때문에, 연속적인 데이터를 전달하는 NRZ (Non-Return-to-Zero) 방식으로 데이터를 전달한다. 2 선식 방식은 데이터의 값을 통해 유효한 값인지 판별할 수 있으므로, 데이터의 유효성을 판단하는 완료 검출(Completion Detector) 블록을 출력 데이터에 사용하여, 이를 요청신호로 대신한다. 따라서 요청신호가 있을 때는 입력 데이터를 유효한 값으로, 그렇지 않을 때는 입력 데이터를 유효하지 않은 값으로 하는 RZ (Return-to-Zero) 방식으로 블록내의 지연에 따른 올바른 동작을 수행하게 된다. 따라서 데이터 전달 방식에 따라 데이터를 표현하는 방법이 다르므로, 인코딩 방식도 다르다.

2.2 NRZ 방식 vs. RZ 방식

단선 구조에서 사용하는 NRZ 방식의 경우 블록 지연 추정이 필요하나 2 선식 구조에서는 데이터라인의 유효성을 검사하여 이를 대신한다.

그림 2(a)는 NRZ 방식과 RZ 방식을 이용한 신호 전송을 나타낸다. NRZ 방식은 요구 신호에 따라 데이터를 변화시키므로, 데이터를 표현시 한번의 신호 천이만이 요구된다. 따라서 기능 블록의 입력 데이터로 사용시 기능 블록의 지연에 의한 출력 데이터의 유효 상태를 검출해 내기 어렵다.

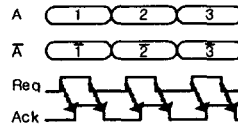


그림 2(a). NRZ 방식

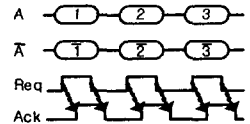


그림 2(b). RZ 방식

그러므로 기능 블록에 대한 적절한 지연 블록을 요구 신호에 적용하여 출력 데이터가 유효한 값 상태를 가질 수 있도록 해야 한다. 이 방식은 단선 방식에서 주로 사용된다. 그림 2(b)는 RZ 방식의 신호를 나타낸다. 이 방식은 요청 신호가 있을 때에만 유효한 입력 데이터로 하고, 요청신호가 없을 때에는 유효하지 않은 값의 상태로 만들어 출력 데이터가 유효한 값을 갖을 때 이를 검출하여 요청신호를 보내는 방식이다. 따라서 데이터를 표현하기 위해서는 신호선당 2 번의 신호천이가 요구되며, 출력 데이터는 기능 블록에 의해 지연을 갖은

후 유효한 값이 되기 때문에 올바른 동작을 위한 요청 신호를 검출해 낼 수 있다.

III. 제안된 2 선식 채널 구조

2 선식 구조에 있어서 요청신호가 있을 때, 각 비트의 값은 하나는 유효한 데이터 값을 나타내고, 다른 하나는 그 값의 반대의 값을 나타낸다. 하지만, 입력 데이터값이 0 이외의 유효한 값을 갖을 때, 각 비트의 유효값을 나타내는 2 개의 선중 유효값을 나타내는 선에 대해서만 0 으로 해도 전체 데이터에 대한 유효상태는 검출되지 않는다. 따라서 한쪽의 데이터 선에 대해서만 RZ 방식을 사용하고, 다른 쪽의 데이터 비트에 대해서는 NRZ 방식을 사용함으로써, 기존의 데이터 전달을 수행할 수 있으며, 전체 RZ 방식을 사용함에 있어서 발생하는 신호천이의 수를 줄일수 있게 된다. 따라서 RZ 방식에 의해 소비되는 전력도 감소 시킬 수 있다.

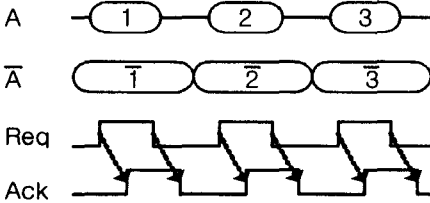


그림 3. 제안된 혼합 데이터 신호 방식

그림 3 는 제안된 방식의 신호 표현 방법이다. 신호 A 에 대해서는 RZ 방식을 사용하여 요청신호(Req)에 의해 데이터의 유효한 상태일 때는 값을 갖게하고, 그 외의 유효하지 않을 때에서 0 으로 놓는다. 그리고 신호 A-bar 에 대해서는 NRZ 방식을 사용하여 데이터 전달 시 항상 유효값을 갖도록 하여 유효하지 않은 상태(Zero)로 리턴 시킬때 발생하는 신호천이를 줄일수 있다. 따라서 예의 값인 0 일때, A-bar 는 0 이므로 각 비트 값이 0 이 되고, A-bar 신호의 비트값은 모두 1 이 되므로 요청신호에 의한 신호천이가 없다. 따라서 이때는 양 비트선의 데이터를 모두 0 으로 함으로써 유효하지 않은 상태로 만들어 기능 블록의 출력에서 적합한 완료신호 검출을 할 수 있게 한다.

일반적으로 사용되는 2 선식 채널에 대한 효율적인 전달 방식으로 HRZ/RZ 혼합 사용은 몇가지 특별한 경우의 입력에 대해서는 적용할 수 없지만, 그 외의 입력 데이터에서는 그 성질에 따라 적용되어 전력 효율을 위

한 신호 천이를 감소시키게 된다. 제안된 혼합 방식의 데이터 전달을 위한 파이프라인 구조는 다음과 같다.

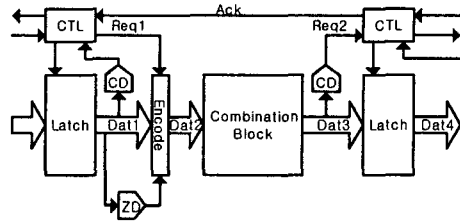


그림 4. 제안된 방식을 위한 스테이지 구조

그림 4 는 NRZ/RZ 혼합 데이터 전달방식을 위한 스테이지 구조이다. 마이크로 파이프라인에 기반한 구조로써, 기능 블록 사이에 기본 2 개의 완료 검출 블록과 예외 값 검출 블록이 있다. 첫 단의 완료 검출(CD) 블록은 래치에 저장을 확인하기 위한 블록으로 핸드셰이킹의 완료 신호를 발생하여 핸드셰이킹 컨트롤러에서 요청 신호(Req1)를 발생하게 해준다. 발생된 요청신호는 인코더 블록을 통해 RZ/NRZ 를 위해 데이터를 변화시킨다. 변화된 데이터는 기능 블록을 통해 동작을 수행하게 된다. 이렇게 수행된 결과값은 완료 검출 블록에 의해 유효상태를 검출하게 된다.

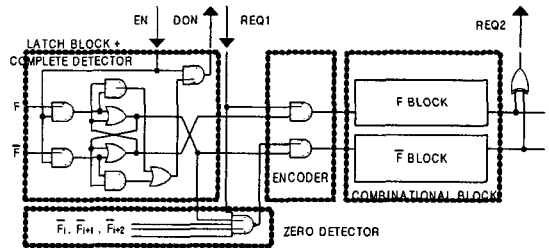


그림 5. 게이트 레벨로 표현된 데이터 전달 구조

그림 5 은 게이트 레벨로 표현된 데이터 전달 구조이다. 핸드셰이킹 제어 블록에 이전 스테이지로부터 요청 신호를 받으면 래치에 인에이블(Enable)신호를 보냄으로써 데이터를 저장하고, 저장된 데이터는 완료검출 블록에 의해 완료신호(Don)를 발생하게 된다. 래치의 저장이 완료되면 다음 스테이지를 위한 요청신호를 발생하게 되고, 이때 인코더(Encoder) 블록을 통해 데이터는 유효 값 상태로 바뀌게 된다. 이렇게 바뀐 2 선식 신호는 각각의 기능 블록의 입력 데이터로 들어가 동작을 수행하게 되고, 기능 블록의 출력 데이터에 대한 완료 검출 블록에 의해 완료신호를 발생한다. 이때 예외 값

검출 블록은 각각의 비트를 나타내는 2선 신호에서 \overline{F} 에 들어가는 입력 데이터 신호를 AND 게이트에 의해 조합함으로써, 예외 값 0 의 \overline{F} 의 입력에 해당하는 모든 비트의 1 을 검출하여, \overline{F} 에 대해 유효 값 0 일때는 RZ 동작을 그 외의 값에 대해서는 NRZ 동작을 수행하여 기능 블록에 대한 데이터의 신호전이를 감소시킨다.

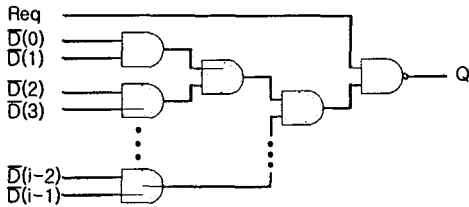


그림 6. 유효 Zero 값 판단을 위한 검출 구조

그림 6 은 유효 값 0 을 검출하기 위한 블록 구조이다. 모든 비트의 반전 신호 \overline{D} 에 대해서 각각 AND 조합에 의해 검출된 신호와 핸드셰이킹 제어 블록으로부터 받은 요청신호를 통하여 유효 값 0 일 때 출력 Q 는 0 으로, 그 외의 값에 대해서는 1 을 발생시킨다.

IV. 시뮬레이션 결과

본 논문에서 제안된 데이터 전달 방식의 신호전이 감소에 대한 전력의 감소를 검증하기 위해, 기능 블록으로 덧셈기(CLA) 블록을 사용하여 각각의 데이터 방식에 따른 전력소비를 비교하였다.

그림 7 은 덧셈기(CLA) 블록을 사용하여 각 전달 방식에 따른 전력 소비를 나타낸다. 4, 8, 16, 32 비트의 입력 데이터를 갖는 덧셈기 블록을 사용하여 각각의 경우에 대한 블록내의 전력 소비를 전송 방식별로 나타내었다. 제안된 방식은 입력 값에 따라 전력 소비가 다르므로, 모든 자리에서 자리올림비트가 발생하도록 하는 입력, 예를 들어 입력 데이터가 4 비트 일 때, "0101", "0101", 모든 자리에서 자리올림비트가 발생하지 않는 입력, 예를 들어 입력 데이터가 4 비트 일 때, "0101", "1010" 의 입력 데이터를 사용하였다. 2 선식에 RZ 를 모두 적용할 때 보다 RZ 와 NRZ 를 혼합하여 적용하였을 때 평균 21%정도의 전력이 감소되었다. 이 전력 감소는 입력 조건에 따라 최대 50%까지 전력소비를 감소시킬 수 있다.

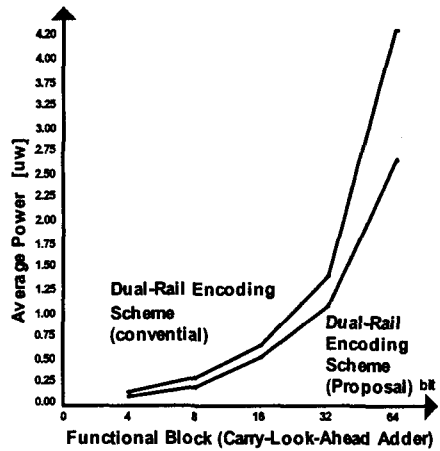


그림 7. CLA 의 입력데이터 비트에 따른 2 선식 회로구조의 소비전력 비교

V. 결론

본 논문은 2 선식 데이터 전달 방식에 대한 신호전이 감소를 위해, 2 선식 인코딩 스킴을 이용하면서 NRZ 방식의 부분적 적용을 통하여 회로내의 전체 신호전이 수를 줄여 전력 소비를 감소시키는 데에 있으며, 이는 게이트 레벨에 의해 표현되는 회로내의 비동기 회로에 대해 사용될 수 있다. 이를 위해 본 논문에서는 새로운 2 선식 인코딩 방식을 제안하였으며, 이 방식의 구현을 위해 파이프라인 구조와 예외값 처리를 위한 블록을 제안하였다. 또한 기존 2 선식 데이터 인코딩 방식과 제안된 방식과 비교하기 위해 덧셈기 블록을 사용하여 입력 데이터의 비트크기에 따른 평균 21%의 전력 감소를 보였다.

제안된 혼합 방식은 비동기 회로의 설계 방법에서 보편적으로 사용되는 2 선식 구조의 구현시, 예외값에 대한 고려를 위한 적은 구조적 변화로 효율적인 전력 감소의 결과를 얻어내어, 저전력을 지향하는 회로에 적합한 비동기 회로 설계 방법이 될 것이다.

참고문헌

- [1] Scott Hauck, "Asynchronous Design Methodologies: An Overview", 1995
- [2] Verhoeff T, "Delay-insensitive code - an overview", Distributed Computing vol 3. 1998.
- [3] D.W. Lloyd and J.D. Garside, "A Practical Comparison of Asynchronous Design Styles", 2001