

클록 게이팅을 이용한 저전력 UART 설계

*오태영, 송승완, 김희석
청주대학교 전자공학과

e-mail : tyoh@cju.ac.kr, parilos@cju.ac.kr, khs8391@cju.ac.kr

A Low Power UART Design by Using Clock-gating

*Tae-Young Oh, Sung-Wan Song, Hi-Seok Kim
Department of Electronic Engineering
Chongju University

Abstract

This paper presents a Clock-gating technique that reduces power dissipation of the sequential circuits in the system. The Master Clock of a Clock-gating technique is formed by a quaternary variable. It uses the covering relationship between the triggering transition of the clock and the active cycles of various flip-flops to generate a slave clock for each flip-flop in the circuit. At current RTL designs flip-flop is acted by Master clock's triggering but the Slave Clock of Clock-gating technique doesn't occur trigger when external input conditions have not matched with a condition of logic table. We have applied our clocking technique to UART controller of 8bit microprocess

I. 서론

최근 들어 저전력 설계에 대한 필요성은 지속적인 소자의 스케일링과 휴대용 기기의 확산으로 인해서 꾸준히 늘어나고 있고 이에 따라서 저전력 설계기술의 지속적인 연구가 진행되고 있다.

현재 저전력 기술이 주로 적용될 것으로 예측되는

차세대 휴대 단말기는 통신 모듈과 멀티미디어 모듈에서 대부분의 전력소모를 점유한다. 또한 유비쿼터스 컴퓨팅의 핵심 어플리케이션인 분산 센서 네트워크, 통합 물류 시스템 등에서도 저전력 통신이 핵심 기술의 하나로 여겨지고 있다.

일반적으로 클록은 사이클마다 클록 스위칭(Clock switching)을 하고 회로 동작을 유발하기 때문에 시스템 전력의 상당부분을 차지한다.

따라서, 본 논문에서는 순차회로에서 클록의 4가지 상태를 기준으로 한 클록 게이팅 기술을 분석하고 8비트 마이크로프로세서(8bit microprocessor)의 비동기 송신 컨트롤러에 적용하여 저전력으로 설계하고자 한다.

II. 본론

2.1 클록 전력

현재 시스템의 전력손실을 감소시키려는 연구가 다방면으로 이뤄지고 있다. 제안한 클록 게이팅에 의한 전력 손실 최소화 방법이 미래에 최고급 마이크로프로세서(High-end microprocessors)들에 충분히 적합한 기술이 아닐지라도 부수적인 디바이스(devices) 전력 손실 예방에 효과적일 것으로 본다.

시스템의 순차회로는 클록에 의한 전력 낭비가 크

며, 클록은 스위칭 과정을 통해 많은 불필요한 게이트 활동(Gate activity)을 초래하며 높은 부하를 갖는 경향이 있다. 그러므로 클록을 구분하고 제어하기 위한 클록 버퍼를 지닌 클록 네트워크가 필요하다. 현재 디지털 구조에서 클록 신호가 시스템 전력의 15% - 45% 전력을 사용한다고 연구되고 있다.[1] 그러므로 클록의 제어를 통해 순차회로의 전력낭비를 줄일 수 있다.[2]

일반적으로 클록에 의한 전력 손실을 줄이는 방법은 다음과 같다.

- I) 마스터 클록(Master clock)의 부하를 줄이고 클록 네트워크에서 요구하는 버퍼 수를 줄인다.
- II) 슬레이브 클록(Slave clock)을 수신하는 플립플롭이 아이들 상태에서 트리거하지 않는다.
- III) 슬레이브 클록에 의해 트리거 되는 플립플롭이 슬레이브 클록에 의해 트리거 되지 않을 때의 사이클 조건을 무의조건(Don't care) 처리하여 동작 조건을 단순화한다.[4]

2.2 클록 동작과 클록 게이팅에 의한 묘사

동기 시스템에서 플립플롭은 클록 신호의 특정 천이 동작에 의해서 트리거(trigger) 된다. 또한 마스터 클록보다 오히려 다른 신호들에 의존하는 클록인 슬레이브 클록도 플립플롭의 트리거를 위해서 마스터 클록과 같은 특정 천이 동작을 제공해야 하며 그 신호는 마스터 클록에 동기된다.

\tilde{CLK}	$CLK(t) \rightarrow CLK^+(t)$	Behavior
0	0 0	0 - holding
α	0 1	α - transition
β	1 0	β - transition
1	1 1	1 - holding

표 1. 신호의 동작을 위한 4가지 상태

표 1에서 클록의 4가지 상태를 표현하였다. 이 상태는 $CLK(t)$ 와 $CLK^+(t)$ 으로써 로직의 천이(transition) 전과 후의 값을 나타냄에 의해서 구분되어진다. 여기서 4가지 상태 값은 (0, α , β , 1) 이다.

α , β 는 2개의 트리거 상태를 나타내고 0, 1은 2개의 홀딩 상태를 나타낸다.

순차회로에서 n개의 플립플롭이 있고, 그들의 출력과 클록 입력을 Q_i 와 clk_i 로 표현한다. 여기서 $i = 0,$

1, ..., n-1 이다. 동기 순차회로에서 플립플롭마다의 클록 clk_i 가 마스터 클록 clk 에 의해서 트리거 된다. 그런데 몇몇 아이들 사이클(idle cycle)에서 마스터 클록으로 부터의 영향을 받지 않기 위해선 슬레이브 클록을 사용해야만 한다. 주의할 점은 이 슬레이브 클록도 순차회로의 동기를 유지하기 위해서 마스터 클록에 동기 되어야 한다는 점이다.

2.3 저전력 설계

기존의 설계에서 클록은 주기적인 상태 천이를 통하여 플립플롭의 동작에 영향을 끼치고 있다. 이런 클록의 무의미한 동작은 전력 낭비를 가져온다. 본 논문은 클록의 무의미한 동작에 따른 전력 낭비를 줄이기 위해서 슬레이브 클록을 통한 플립플롭의 아이들 사이클에서 트리거 동작을 유발하지 않도록 하였다. 이는 클록에 의한 전력손실을 줄이는 효과를 가져왔다. 본 제안을 8비트 마이크로 프로세서의 비동기 송신 컨트롤에 적용하였다.

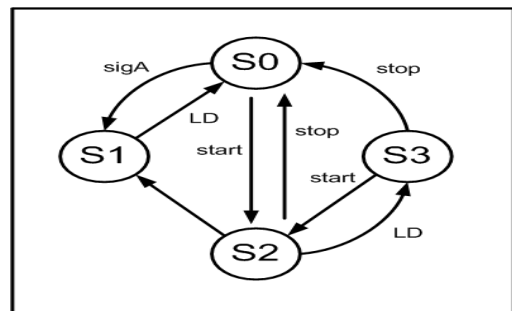


그림 1. 비동기 송신단의 상태천이도

그림 1에서 S0, S1, S2, S3는 비동기 송신단의 상태를 나타낸다. S0는 통신을 위한 포트가 설정되지 않아서 통신이 이뤄질 수 없는 초기 상태이다. 또한 전송 블록에 데이터를 로드(load)하고 start 신호를 기다리고 있는 상태기도 하다.

S1은 대기상태를 나타낸다. 이 상태는 통신을 위한 포트가 설정된 단계고 전송할 데이터의 로드를 기다리는 상태를 말한다. S2는 현재 전송상태에서 다음에 전송할 데이터가 전송블록에 로드되었음을 알려주는 상태이다. 이는 데이터의 연속 송신상태를 나타낸다. S3의 경우 S2와 마찬가지로 전송상태를 나타내지만, S2와의 차이는 현재 전송중인 데이터 이후의 데이터 전송이 없다는 것이다.

각각의 상태 천이는 그림에서 표현된 sigA, LD, start, stop 신호들에 의해서 이루어진다. sigA는 통신 가능 여부를 알려주는 신호이고, LD는 전송블록에 데이터가

실렸음을 알려주는 신호이다. start와 stop는 전송단계에서 한 프레임의 시작과 완료를 알려주는 신호이다. 이 두 신호는 전송하는 데이터와 함께 전송된다.

아래의 수식들은 비동기 송신 모드에서 송신 컨트롤러 내부의 동작 상태에 영향을 주는 외부 입력 신호들에 의한 플립플롭의 동작과 이를 간략화하고 공통항이 생기도록 전개한 식이다.

$$\begin{aligned} & \overline{A_8 A_7 A_6 A_5 (A_4 Q_1 + \bar{A}_4 \bar{Q}_1)} \cdot \{A_8 A_7 A_6 A_5\} Q_1 \\ & + A_8 A_7 A_6 A_5 (A_4 Q_1 + \bar{A}_4 \bar{Q}_1) \cdot \bar{Q}_1 \\ & + \overline{A_8 A_7 \{ \bar{A}_4 \bar{Q}_1 (A_3 Q_0 + \bar{Q}_0) + A_4 Q_1 \}} \cdot (\bar{A}_6 + \bar{A}_5 + A_6 A_5) \cdot A_8 A_7 Q_1 \\ & + A_8 A_7 \{ \bar{A}_4 \bar{Q}_1 (A_3 Q_0 + \bar{Q}_0) + A_4 Q_1 \} \cdot (\bar{A}_6 + \bar{A}_5) \cdot \bar{Q}_1 \\ & + (A_4 Q_1 + \bar{A}_4 \bar{Q}_1) \cdot (\bar{A}_8 + \bar{A}_7) \cdot (\bar{A}_8 + \bar{A}_7) \cdot Q_1 \\ & + (A_4 Q_1 + \bar{A}_4 \bar{Q}_1) \cdot (\bar{A}_8 + \bar{A}_7) \cdot \bar{Q}_1 \end{aligned}$$

식(1). 플립플롭 Q_1 의 논리식

$$\begin{aligned} & A_8 A_7 A_6 A_5 \bar{Q}_0 + \overline{A_8 A_7 A_6 A_5 \bar{Q}_0} \cdot (A_8 A_7) Q_0 \\ & + A_8 A_7 \bar{A}_4 Q_1 (\bar{A}_6 + \bar{A}_5) \bar{Q}_0 + \overline{A_8 A_7 \bar{A}_4 Q_1 \bar{Q}_0 (\bar{A}_6 + \bar{A}_5)} \cdot (A_8 A_7) Q_0 \\ & + A_8 A_7 \bar{Q}_1 (\bar{A}_6 + \bar{A}_5) (A_4 A_3 \bar{A}_2 + \bar{A}_4) \bar{Q}_0 \\ & + \overline{A_8 A_7 \bar{Q}_1 \bar{Q}_0 (\bar{A}_6 + \bar{A}_5) (A_4 A_3 \bar{A}_2 + \bar{A}_4)} \cdot (A_8 A_7) Q_0 \end{aligned}$$

식(2). 플립플롭 Q_0 의 논리식

$$\begin{aligned} & \bar{A}_4 Q_0 [A_8 A_7 \{ \bar{A}_6 A_3 \bar{Q}_1 + \bar{A}_5 A_3 \bar{Q}_1 + A_6 A_5 + Q_1 \} + \bar{A}_6 + \bar{A}_7] \\ & + \bar{A}_4 \bar{Q}_0 [A_8 A_7 \{ \bar{A}_6 \bar{Q}_1 + \bar{A}_5 \bar{Q}_1 + A_6 A_5 + Q_1 \} + \bar{A}_6 + \bar{A}_7] \end{aligned}$$

식(3). 플립플롭 Q_1 의 공통항 생성

$$\begin{aligned} & A_8 A_7 Q_1 [\bar{A}_5 \bar{A}_4 \bar{Q}_0 + \bar{A}_6 \bar{A}_4 \bar{Q}_0 + A_6 A_5 + Q_0] \\ & + A_8 A_7 \bar{Q}_1 \left[\begin{array}{l} \bar{A}_6 A_4 A_3 \bar{A}_2 \bar{Q}_0 + \bar{A}_6 \bar{A}_4 \bar{Q}_0 + \bar{A}_5 A_4 A_3 \bar{A}_2 \bar{Q}_0 \\ + \bar{A}_5 \bar{A}_4 \bar{Q}_0 + A_6 A_5 + Q_0 \end{array} \right] \end{aligned}$$

식(4). 플립플롭 Q_0 의 공통항 생성

위의 식(1)과 (2)는 비동기 송신 컨트롤러 블록에서 송신 동작 상태를 나타내는 플립플롭 Q_1 과 Q_0 에 영향을 주는 외부 신호들과의 논리 관계를 나타냈다. 또한

식(3)과 (4)는 제시된 논리식 식(1)과 (2)를 공통된 항을 찾아 간략화한 것이다. 위의 식에서 보듯 플립플롭 Q_1 과 Q_0 는 각각 \bar{Q}_4 와 $Q_8 Q_7$ 를 공통항으로 가지고 있다. 공통항은 마스터 클록에 동기하여 슬레이브 클록을 만든다. 이렇게 생성된 슬레이브 클록은 idle cycle에서 플립플롭에 제공하는 클록을 제어할 수 있게 된다. 공통항으로 표현된 논리식으로부터 그림 2와 같이 플립플롭을 이용한 회로를 표현할 수 있다.

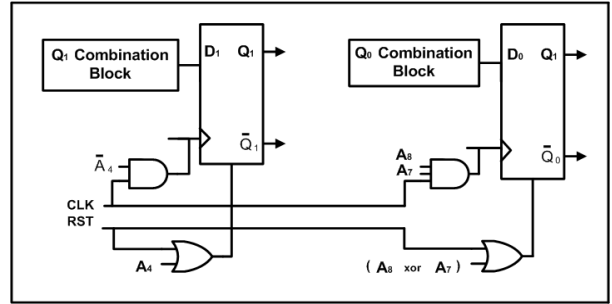


그림 2. 슬레이브 클록에 의한 플립플롭 동작

그림 2를 통해 우선적으로 각각의 플립플롭에 Q_1 Combination Block과 Q_0 Combination Block은 식(3)과 (4)에서 표현한 논리식이다. 또한 식(3)과 (4)에서 표현된 공통항은 마스터 클록과 and 논리에 의해서 슬레이브 클록을 생성하고 있다. 슬레이브 클록은 주어진 입력조건이 아닐 경우 플립플롭에 트리거를 발생시키지 않는다. 또한 이 경우 리셋과 같은 비동기처리를 하였다. 이러한 과정들을 통해서 클록 트리거에 의한 플립플롭 동작을 제어하였고 손실되는 시스템 전력의 낭비를 최소화 하였다.

III. 구현

Sort (mW)	Models	
	standard	Ideal
Dynamic Thermal Power Dissipation	15.39	15.32
Static Thermal Power Dissipation	187.50	187.50
Total Thermal Power Dissipation	202.89	202.82

표 2. 기존의 설계와 제안된 이론의 전력비교

표 2는 기존의 RTL 설계와 제안한 클록 게이팅의 슬레이브 클록에 의해 동작하는 플립플롭을 8비트 마이크로프로세서의 비동기 송신 컨트롤러에 적용하여 얻은 두 모델간의 전력비교를 나타낸다. 표에서 standard는 기존의 RTL 설계의 결과치를 나타내고, Ideal은 제안한 방식을 채택한 결과를 나타낸다. 두 방

식의 전력 차이는 크지 않다. 하지만, 제안한 이론이 적용된 플립플롭이 증가하면 전력 차이도 증가할 것이다.

본 실험치는 Quartus II Version 4.2를 통한 결과이고, Stratix-EP1S10F484C5를 타겟으로 하여 얻은 실험치이다.

IV. 결론

본 논문에서 클록 게이팅 기술을 통한 전력낭비를 감소시키는 방안을 제시하였다. 클록은 플립플롭에서 트리거링(triggering) 동작을 분석하기 위한 기초이다. 제안한 기술을 8비트 마이크로프로세서의 비동기 송신 컨트롤러에 적용하여 약 1% 전력손실이 감소됨을 보였다. 실험치로 평가할 때 이 기술을 시스템의 더 많은 플립플롭에 적용하여 감소하는 전력량은 증가할 것으로 본다.

참고문헌

- [1] M. Pedram, "Power minimization in IC Design: Principles and applications," ACM Transactions on Design Automaton, vol. 1, no. 1, pp.3-56, Jan. 1996
- [2] G. Friedman, "Clock distribution design in VLSI circuits: an overview," in Proc. IEEE ISCAS, San Jose, pp.1475-1478, May 1994.
- [3] E. Tellez, A. Farrah and M. Sarrafzadeh, "Activity-driven clock design for low power circuits," in Proc. IEEE ICCAD, San Jose, pp.62-65, Nov. 1995.
- [4] Qing Wu, Massoud Pedram, Xunwei Wu, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," IEEE CUSTOM INTEGRATED CIRCUITS CONFERENCE, pp.479-482, 1997.
- [5] Hans Jacobson, Pradip Bose, Zhigang Hu, Alper Buyuktosunoglu, Victor Zyuban, Rick Eickemeyer, Lee Eisen, John Griswell, Doug Logan, Balaram Sinharoy, Joel Tandler, Proceedings of the 11th Int'l Symposium on High-Performance Computer Architecture, 2005