

빠른 lock-on time을 위한 선택적 시작점을 갖는 DLL

김신호, 장일권, 광계달
한양대학교 전자전기공학부

전화 : 02-2290-0357 / 핸드폰 : 016-411-6365

A Fast lock-on time Delay Locked Loop with selective starting point

Sin Ho Kim, Il Kwon Chang, Kae Dal Kwack
Division of Electrical and Computer Engineering, Hanyang University
E-mail : ksinh@hymail.hanyang.ac.kr

Abstract

This paper describes a delay locked loop with selective starting point for use in a high-frequency systems. SSRDLL (selective starting point RDLL) has been simulated in a $0.25\mu\text{m}$ standard n-well CMOS process parameter to realize a fast lock-on time. This DLL is shown to be insensitive to variations in PVTL. The simulated lock time of the proposed SSRDLL is within 4 clock cycles at 333MHz clock input.

I. 서론

반도체 기술이 급속히 발전함에 따라 수백-MHz로 동작하고 있는 디지털 시스템들이 몇 년 동안 성공적으로 개발되어 왔다. 그 중에서도 마이크로프로세서와 메모리는 눈부신 발전을 거듭해 오고 있다. 고주파로 동작하는 IC들이 PCB에 집적됨에 따라서 clock skew 문제가 발생되고, 이것은 고성능 시스템에 대한 장애가 될 뿐만 아니라 동작 주파수가 증가되는 것에 큰 방해 요인이 되고 있다.

SDRAM의 경우에는 메모리 컨트롤러와 메모리 사이에서 신호 지연이 발생된다. 이는 입력 버퍼를 지나고

큰 부하 커패시턴스를 구동하기 위한 클럭 구동 드라이버를 지난 내부 클럭과 외부 클럭 사이의 skew가 발생하기 때문이다. 외부에서 SDRAM 또는 시스템에 들어오거나 나가는 신호는 데이터와 클럭 두가지라고 볼 수 있다. 외부에서 데이터는 클럭에 동기되어서 들어오는데 clock skew로 인해서 칩 내부에서는 데이터와 클럭이 동기될 수 없게 된다. 그러므로 데이터는 skew를 보상하기 위해 일정시간 이상의 값을 유지해야 하며, 이것은 입출력(I/O) 회로의 밴드 폭을 줄어둘게 한다. 즉, clock skew는 칩의 동작 속도와 성능을 저하시키는 원인이 된다. 또 다른 문제는 clock skew가 일반적으로 공정(process), 전압(voltage), 온도(temperature), 부하(load) -PVTL- 에 의해서 복잡한 문제를 만든다[1].

향후 동작 주파수가 점점 더 증가하기 위해서는 이런 장애 요소인 clock skew를 제거하여야만 한다. clock skew 문제를 해결하기 위해 클럭 동기 회로인 phase locked loop(PLL) 또는 delay locked loop(DLL)를 폭 넓게 사용하고 있다.

저전력, 고속 동작을 위해 내부 클럭 발생회로는 수 cycle 이내에 외부 클럭과 내부 클럭 위상이 동기 되어야 하고, 전력 소모가 적고, 클럭 스큐 또는 지터의 발생을 최소화하도록 하여야 한다.

본 논문에서는 II절에서 conventional digital DLL의 구조와 동작 특성에 대해서 알아보고 III절에서는 제안하는 SSR(selective starting point register)DLL의 구조

및 동작 특성에 대해서 알아본다. IV절에서는 simulation 결과를 그리고, V 절에서는 결론을 제공한다.

II. 기존의 Digital DLL

1) Synchronous Mirror Delay(SMD)

SMD는 FDA, BDA, MCC, 그리고 input buffer delay(d_1)과 clock driver delay(d_2)로 이루어진 DMC로 구성되어 있다[2]. 이를 그림1에 나타내었다.

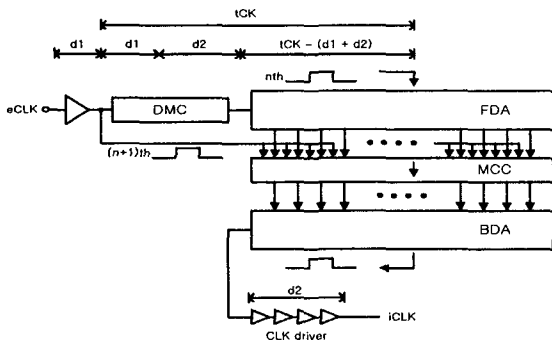


그림1. SMD의 블록도

FDA는 하나의 입력에 대해 다수의 출력을 가지고 t_{df} 의 delay를 가지는 지연 소자들로 구성되어 있고, BDA는 복수 입력에 대한 단일 출력을 가지고 t_{db} 의 delay를 가지는 지연 소자들로 구성되어 있다. FDA는 BDA에 평행하게 이루어져 있지만 서로 반대 방향으로 동작한다. FDA의 output과 BDA의 input은 MCC를 통해서 서로 서로 연결되어 있다.

DMC를 지난 클럭은 FDA로 들어가 delay 되어 지나간다. 이때 클럭 버퍼를 지난 클럭에 의해서 delay 되는 unit delay element의 수를 결정 짓는다. FDA로부터 나온 clock은 MCC를 지나 BDA로 들어가게 되고 진행 방향은 FDA의 반대이고 같은 unit delay element의 수를 지나게 된다. 그리고 나서 클럭 드라이버를 거치게 되면 내부 클럭은 외부클럭에 동기가 이루어 진다.

$$d_1 + d_1 + d_2 + (t_{ck} - d_1 - d_2) + (t_{ck} - d_1 - d_2) + d_2 = 2t_{ck}$$

SMD는 두 클럭만에 클럭을 동기 시킨다. 그러나 SMD를 거친 내부 클럭은 unit delay element에 의해 제한된 deskew 양을 가지게 된다. 때문에 좋은 phase 특성을 가지기 위해서는 FDA와 BDA에 있는 unit delay element의 지연 시간이 충분히 작아야만 한다. 이것은 minimum clock 주기에서 동작하기 위해서는 많은 수의 delay element가 있어야 하며 delay array가 커지게 되고 power 소모가 증가하게 된다. 이것을 보완하는 것으

로 delay line을 coarse와 fine으로 나누어 control하는 방식이 있다[3].

2) Register controlled DLL(RDLL)

Clock skew를 제거하기 위한 또 다른 접근 방법으로 레지스터에 delay line의 step count를 저장해 두었다가 외부 클럭과 feedback 되어 돌아온 내부 클럭의 위상 차이를 비교하여 지속적으로 register의 저장값을 변화시켜가면서 클럭 스큐를 제거하여 클럭 phase를 동기시키는 Register controlled DLL(RDLL)이다.

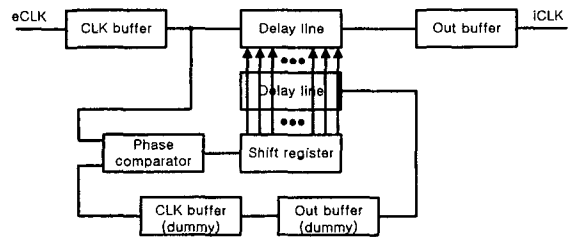


그림2. RDLL의 블록도

그림2의 RDLL은 feedback 되는 클럭이 input buffer를 지난 클럭(외부 클럭)과 일치시키기 위해서 dummy input buffer와 clock driver를 가지고 있으며, phase comparator, shift register, 그리고 delay line으로 구성되어 있다[4].

Shift register는 delay line의 길이를 조절하여 내부 clock의 위상을 보정하며, shift register의 값은 phase comparator의 출력에 따라 변화한다.

RDLL의 clock phase lock-on time이 두 클럭만에 동기되는 open loop DLL인 SMD에 비하면 수십 ~ 수백 클럭 사이클로 긴 반면, deskew 제한이 SMD가 가지는 unit delay element의 delay time skew보다 더 적다. RDLL의 성능은 unit delay element에 의해 결정되며 delay line의 총 delay 시간은 RDLL의 deskew 범위와 낮은 동작 주파수를 결정 짓는다. 낮은 동작 주파수에서 클럭 스큐를 제거하기 위해서는 delay line이 더 길어져야 하며, 이에 따라 면적을 많이 차지하는 shift register가 더 늘어나게 되므로 면적 소모가 상당히 커지게 된다 [5].

III. 제안하는 SSRDLL

RDLL은 클럭 phase lock-on time이 길기 때문에 SDRAM의 power on sequence 후에 수십~수백 clock cycle의 lock-on time이 필요하여 open loop DLL보다 상대적으로 긴 준비 시간이 요구된다. 이런 대기 시간은 불필요한 standby current의 증가를 가져오고 power 소모가 커지게 된다. fast lock-on time을 가지게 되면 이

러한 문제들은 해결이 가능하다. 이런 문제들을 해결하기 위해서 fast lock-on time과 적은 클럭 스큐를 가지는 SSRDLL을 제안하였다. RDLL이 가지는 delay line의 starting point를 외부 클럭과 내부 클럭의 phase가 input buffer와 clock driver에 의해서 발생하는 delay time을 제거하는 부분에서 선택하여 시작시킨 후 수 클럭 내에서 완전한 클럭 위상의 동기가 이루어질 수 있다. 이렇게 제안된 SSRDLL은 fast lock-on time을 갖는 closed loop DLL이다.

SSRDLL은 RDLL의 delay line starting point를 open loop 클럭 동기 회로를 사용하여 세 클럭만에 찾아내고 외부 클럭과 feedback 되어 돌아온 내부 클럭의 phase를 비교해서 shift register를 control하여 네 클럭 이후에는 정확한 위상 동기를 이룬다.

1) SSRDLL의 구조

그림3은 SSRDLL의 블록도를 나타낸다. Dummy input buffer와 clock driver는 clock의 delay 요소를 맞추어 주기 위해 사용된다. Phase comparator는 feedback clock signal과 input buffer를 지난 clock(rclk)의 상승에 지 부분을 비교해서 delay line을 선택해주는 shift register의 control signal을 발생시킨다. Phase comparator(PC)의 출력은 fast, just, slow signal을 내보낸다.

Fast lock-on time을 갖기 위해 open loop 클럭 동기 회로를 사용하여 delay time을 보상해 주고 shift register의 starting point를 선택해 준다.

Delay line controller에서는 delay time을 측정하기 위한 start와 stop 펄스를 발생시키고 coarse delay line과 fine delay line을 선택하기 위한 signal을 내보낸다. coarse delay line 선택 signal은 fine delay line이 몇 번 반복되어졌는지를 pulse generator에서 보내줌으로써 선택되고, start pulse signal에 의해 delay가 시작되고 stop pulse signal에 의해 control delay line이 reset된다.

Coarse delay line과 fine delay line으로 나누어 사용하는 hierarchical delay line scheme은 넓은 주파수 범위와 작은 layout 면적에 유용하게 적용되지만, coarse와 fine delay line 사이에 불연속성에 의한 coarse delay line의 전환점에서 jitter가 발생할 수 있다. 이것은 또한 주파수 성분을 악화할 수 있다. 이에 fine delay line을 shift register로 control함으로써 delay line을 선택하기 위해 발생된 skew 성분을 feedback clock과 input buffer를 지난 클럭 rclk을 비교함으로써 제거해 줄 수 있다. 또한 fine delay line에만 register가 필요하기 때문에 면적 효율 면에서도 상당히 유리하다.

2) 회로 구현

(a) Delay line controller 및 Coarse and Fine delay line

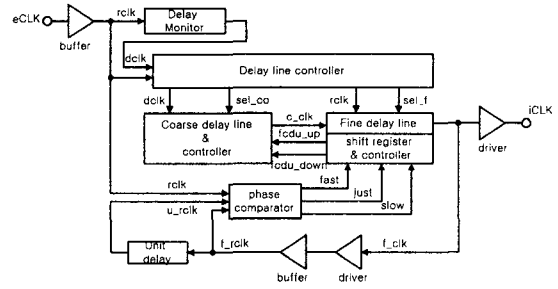


그림3. SSRDLL의 블록도

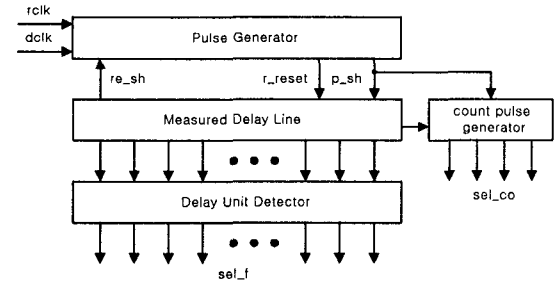


그림4. Delay line controller의 블록도

그림4의 control pulse generator는 input buffer를 지난 clock(rclk)으로 start pulse signal(p_sh)을 발생시켜서 measured delay line으로 출발시키고 monitor 회로를 지난 clock(dclk)으로 stop pulse signal(r_reset)을 발생시켜 delay line을 초기 상태로 reset 시킨다. Measured delay line을 통과한 펄스는 다시 pulse generator로 입력되어서 repeat되는 pulse signal(re_sh)을 발생시켜 다시 measured delay line에 인가 시켜준다. Reset되기 전까지 measured delay line의 반복 횟수를 count pulse generator에서 발생시켜 coarse delay line에 보내주고(sel_co), 어느 위치에서 pulse가 지나갈 때 reset 되었는지를 fine delay line의 입력 신호(sel_f)로 내보낸다. Fine delay line은 66ps의 unit delay time을 가지고 있는 delay element로 배열되어 있으며, coarse delay line은 fine delay line의 전체 delay time과 같은 시간을 갖는 하나의 block을 기본으로 하여 구성되어 있다. dclk이 coarse delay line의 입력이 되고 c_clk을 출력으로 내보내며, 이 c_clk은 fine delay line의 입력이 된다.

(b) Phase Comparator 및 Shift Register

Shift register를 제어하기 위한 control signal을 내보내 주기 위해 comparator block에서 input buffer를 지난 reference clock인 rclk와 feedback되어 돌아온 clock인 f_rclk, 그리고 unit delay element를 지난 u_rclk를 입력으로 받아서 각각의 상승 에지를 비교해서 slow, just, fast signal을 출력한다.

Feedback 되어 돌아온 내부 클럭이 외부 클럭과 비교해서 위상이 빠른 경우에는 fast signal을 내보내서 shift

register를 왼쪽으로 한 비트 이동시켜서 delay line의 delay time을 더 늘려주고, slow signal 발생 시에는 shift register를 오른쪽으로 이동시켜 delay time을 줄여 준다. just signal 발생 시에는 이전에 저장하고 있던 delay line의 delay time을 그대로 유지한다.

Shift register 마지막 단이 선택되어 있을 때, slow 신호가 발생하면 fcd_u_down 신호를 발생시키고, shift register 첫 단이 선택된 경우에 fast 신호가 발생하면 fcd_u_up 신호를 발생시킨다. fcd_u_up과 fcd_u_down 신호가 발생하지 않을 경우에는 두 개의 cdu block을 지나게 되어 있고(fine delay time의 두배), fcd_u_up 발생시에는 기존 cdu block에서 delay element가 증가되고, shift register는 마지막 단을 선택한다. fcd_u_down 발생시에는 기존 cdu block에서 delay element가 감소되어 coarse delay line이 조절되고, shift register는 첫 단이 선택되어 진행된다.

Shift register의 power 소모는 shift가 일어나지 않고 loop가 고정되어 있을 때 줄어든 것이다. Phase comparator를 디자인 할 때, flip-flops은 최소의 static phase error를 가지기 위해 setup time이 적은 빠른 flip-flops을 사용한다.

IV. Simulation 결과

제안한 SSRDLL은 0.25 μm n-well CMOS 공정 parameter를 가지고 simulation하였다. 공급 전원은 2.5V를 사용하였다.

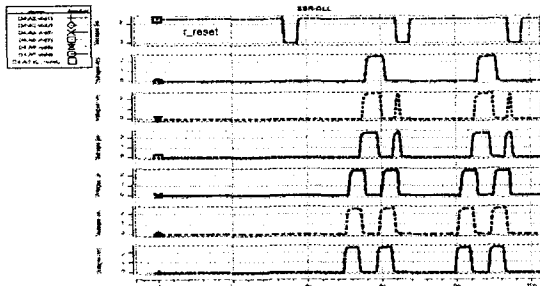


그림5. r_reset signal 및 delay pulses

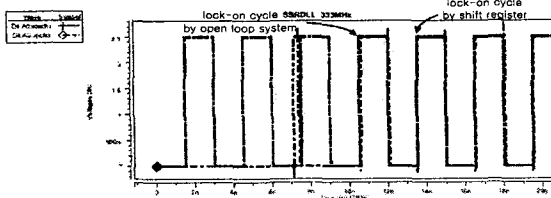


그림6. eclk 및 iclk

그림5는 pulse generator에서 발생되는 r_reset signal

과 measured delay line을 지나는 pulse들을 보여준다. 그림6은 세 번째 클럭 사이클에서 내부 클럭과 외부 클럭의 skew 성분을 제거하여 일치되고, 네 번째 클럭 사이클에서 open loop system에 의한 skew성분을 fine delay line이 shift register에 의해 제어되어 제거된다. 결과적으로 세 번째 클럭에서 92ps에 이르던 skew가 네 번째 클럭에서 27ps로 줄어든 것을 볼 수 있다. 동작 주파수는 333MHz에서 실시하였고, unit delay element의 delay time은 66ps이다.

V. 결론

본 논문에서 제안한 SSRDLL은 0.25 μm n-well CMOS 공정 parameter를 사용하여 simulation하였다. SSRDLL은 기존의 open loop DLL이 가지고 있는 장점인 fast lock-on time을 그대로 유지하면서 feedback loop를 가지고 동작하여 PVT 등 내부 환경 변화 요인에 의한 jitter 성분에 영향을 받지 않고, unit delay element보다 적은 skew를 가지고 동작한다. 333MHz 동작 주파수에서 네 클럭 사이클만에 완전히 동기를 이루어서 내부 클럭을 안정하게 발생시켜 준다. clock skew는 27ps이다. 이 값은 unit delay element의 delay time을 줄여 줌으로써 더 적게 가져갈 수 있다. 그러므로 제안한 SSRDLL은 high speed, low voltage, low power, clock deskew에 적합하다.

참고문헌

- [1] Guang-Kaai Dehng et al., "Clock-Deskew Buffer Using a SAR-Controlled Delay-Locked Loop", IEEE J. Solid-State Circuits, vol.35, pp.1128-1136, Aug. 2000.
- [2] Takanori Saeki et al., "A 2.5-ns Clock Access, 250-MHz, 256-Mb SDRAM with Synchronous Mirror Delay", IEEE J. Solid-State Circuits, vol.31, pp.1656-1665, Nov. 1996.
- [3] Seong-Jin Jang, Seon-Ho Han et al., "A Compac Ring Delay Line for High Speed Synchronous DRAM", Sym. on VLSI Circuits, pp.60-61, 1998.
- [4] A. Hatakeyama et al., "A 256-Mb SDRAM Using a Resister-Controlled Digital DLL", IEEE JSSC, vol.32, pp.1728-1734, Nov. 1997.
- [5] Feng Lin et al., "A Register-Controlled Symmetrical DLL for Double-Data-Rate DRAM", IEEE JSSC, vol.34, pp.565-568, Apr. 1999.