

고속 메모리 동작을 위한 내부 클럭 발생 회로의 기술동향

박 부 응

LG 반도체 ULSI 연구소 선행제품설계Gr.

I. 서 문

SDRAM(Synchronous Dynamic Random Access Memory)은 1990년대 중반에 들어서 16Mb DRAM 세대에 본격적으로 나타나게 되었는데 본 기고에서는 SDRAM의 핵심 기술인 내부 클럭 발생 회로에 대해서 기술적인 내용을 정리하여 보고자 하며, 이에 앞서 SDRAM의 출현 배경과 내부 클럭 발생 회로를 사용하게 된 배경에 대해 서술한다.

반도체 테크놀러지의 발전은 VLSI 회로의 집적도와 성능을 빠른 속도로 향상 시키고 있다. 그 중에서도 마이크로프로세서와 메모리는 눈부신 발전을 거듭해 왔으며, 메모리 중에서도 DRAM은 칩 집적도가 3년마다 4배로 증가하면서 칩의 크기는 1.5배의 증가에 머물렀다. 이는 DRAM의 비트 당 가격이 다른 RAM(random access memory)과 비교하여 우위에 있을 수 있었고 DRAM이 현존하는 여러 가지 컴퓨터 시스템에서 주 메모리로서의 자리를 확고히 하게 하였다.

또한 메모리 가격이 지속적으로 낮아짐에 따라 컴퓨터 시스템에서의 평균 메모리 사용이 늘기도 하였지만, 개인용 컴퓨터(PC)에서 Microsoft Windows와 응용 소프트웨어의 출현은 주 메모리 용량을 2배로 늘려야 하는 결과로 나타나게 되었다. 결국 시스템 설계자들은 시스템 성능을 만족하면서 적절한 가격을 유지할 수 있도록 비트 당 가격이 낮은, 즉 칩 집적도가 높은 DRAM을 채택하게 되는데, 이는 컴퓨터 메모리 시스템이 필요로 하는 최소 DRAM 용량(Granularity)보다 많아져 사용이 제한 되고, DRAM 시장의 새로운 세대로의 이동을 방해하는 요인이 되었다.

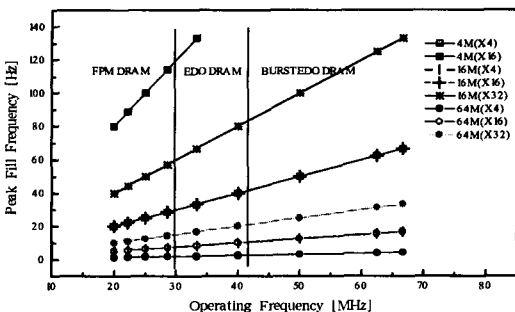
한편 DRAM은 매년 25%(column access time ; row access time은 7%)의 비율로 속도 개선이 이루어져 pin 당 bandwidth 및 bit 당 bandwidth가 증가하였다. 그러나 마이크로프로세서는 매년 50% 정도의 속도 향상이 이루어져 DARAM이 메모리 시스템이 요구하는 bandwidth를 만족하지 못하는, 즉 마이크로프로세서와

DRAM의 성능 차이로 인해 컴퓨터 시스템 성능의 저하를 초래하게 되었다. 따라서 메모리 시스템에서 필요한 bandwidth를 만족하기 위해서는 Granularity를 초과하더라도 메모리 용량을 늘려야 하는 딜레마에 빠지게 되었다. 이에 시스템에서 DRAM 사용의 최적화를 피하기 위해 Granularity와 Bandwidth를 동시에 비교할 수 있는 새로운 척도로서 Fill Frequency (FF = MBps/MB)가 제시되었다¹⁾. Fill Frequency는 칩 집적도, 즉 DRAM 크기가 증가하면 감소하게 된다. 예를 들어 동작 주파수가 25MHz인 1Mx4, 4Mx4 Fast-Page Mode(FPM) DRAM의 FF는 각각 25Hz에서 6.25Hz로 감소한다. 즉 DRAM 세대가 나아갈수록 컴퓨터 시스템이 요구하는 Fill Frequency를 만족할 수 없게 된다. 따라서 DRAM 설계자들은 새로운 DRAM 인터페이스를 모색하게 되었으며, [그림.1]에서 보듯이 FPM(Fast-Page Mode) DRAM에서 EDO(Extended-Data Out) DRAM, BEDO(Burst-Extended-Data Out) DRAM으로 column 동작 방식을 개선 하였다. 이렇게 하여 column cycle time을 개선하면서 DRAM width도 넓혀 메모리 시스템 버스에 적용 가능한 주파수, 즉 현실적인 최대 전송 주파수를 증가시킨다. 따라서 시스템이 요구하는 FF를 만족하도록 하여 새로운 DRAM 세대가 메모리 시장을 형성 할 수 있도록 하였다.

하지만 최근에 100~200MHz 이상에서 동작하는 마이크로프로세서의 사용은 시스템 동작 주파

수가 66MHz 이상이 필요하게 되어 BEDO DRAM의 사용도 한계에 부딪힌다. 그래서 DRAM의 읽기/쓰기 동작을 시스템 클럭에 동기시켜 고속 데이터 처리 속도를 가능하게 한 SDRAM이 연구 되고 있다. SDRAM은 고속 데이터 전송을 위하여 Pipeline[2] 및 Prefetch[3] 방식의 아키텍처를 사용하고, 아울러 multi-bank 구조를 사용하여 row-access time을 줄일 수 있는 bank interleave 방식을 채택하고 있다. 그러나 SDRAM과 메모리 컨트롤러 사이에서의 데이터와 컨트롤 신호의 지연은, 66MHz 동작 주파수에 100MHz SDRAM을 사용해야 하는 비효율성이 존재한다. 이런 문제를 해결하기 위해 LVTTL (Low-Voltage TTL) 인터페이스보다 신호 swing 폭이 작고, 고속 신호 전달에 적합한 새로운 인터페이스(Stub Series Transceiver Logic 등)로의 전환이 이루어 지고 있다. 이와 함께 LVTTL 혹은 SSTL 인터페이스에서 데이터 전송 속도를 보다 향상시키기 위하여 SDRAM의 클럭에서 데이터 출력 까지의 시간(tAC; clock access time)을 줄이려는 연구가 진행되어 왔다. 데이터는 입력 버퍼를 지난 내부 클럭에 동기 되어 출력 되며, 입력 버퍼를 거친 내부 클럭은 큰 부하 커패시턴스를 구동하여야 하므로 클럭 구동 드라이버가 필요하다. 이런 입력 버퍼와 클럭 구동 드라이버에서 발생하는 신호 지연은 데이터 액세스 시간의 손해로 나타나게 된다. 따라서 내부 클럭 지연(Clock Delay)을 제거하기 위해 내부 클럭 에지를 입력 클럭 에지와 위상이 일치하도록 하는 방법을 사용하여야 한다. 최근에 개발 되는 SDRAM은 대부분 이런 문제를 해결하기 위하여 내부 클럭 발생 회로를 채택하고 있다.

이상 SDRAM과 SDRAM에서의 내부 클럭 발생 회로의 사용 배경에 대한 서술을 마치고 다음은 내부 클럭 발생 회로의 종류에 대해 설명하기로 한다.



〈그림 1〉 DRAM의 Density 및 Width에 따른 Peak Fill Frequency

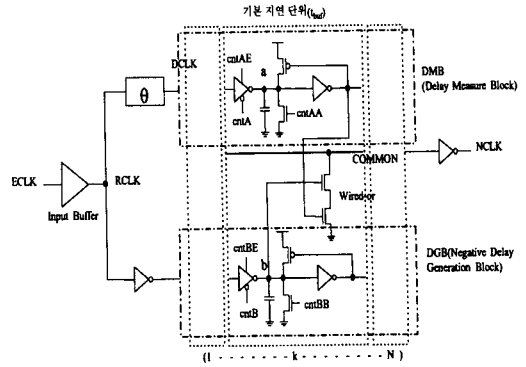
II. 내부 클럭 발생 회로(On-chip clock generation circuit)

대부분의 동기식 회로에서는 내부 클럭 지연 (Clock Delay or Clock Skew)을 제거하기 위하여 PLL(Phase-Locked Loop)이나 DLL(Delay-Locked Loop) 기법을 사용하여 왔다. Feedback loop를 사용하는 PLL/DLL은 두 클럭의 위상을 일치시키는데 수 십에서 수 백 사이클이 필요하다. 하지만 SDRAM에서는 전원을 제한하여야 하는 저전력 모드(Standby Mode)가 있어 많은 전력 소모를 하는 PLL/DLL의 사용을 어렵게 한다. 즉 저전력 모드에서 전력 소비를 제한하기 위하여 PLL/DLL 동작을 정지하도록 하면 다시 정상 동작 모드로 전환시 클럭의 위상을 동기화 하는데 걸리는 시간이 길어 SDRAM의 불안정한 동작을 유발하기 때문이다. 따라서 클럭의 동기화를 빨리 이룰 수 있는 방법에 대한 연구가 활발히 수행되어 왔다. Feedback 기법을 사용하지 않는 클럭 발생 회로는 수 사이클 내에 위상 동기화가 이루어 지므로 SDRAM의 저전력 모드에서 동작을 제한 하더라도 정상 모드로 전환시 내부 클럭 위상이 빨리 안정되어 저전력 모드에서 전원을 효율적으로 제한할 수 있다. 이에feedback를 사용하지 않는 고속 위상 동기화 방법을 살펴보고, feedback 이 있는 DLL로 고속 위상 동기를 실현할 수 있는 방법도 함께 설명하고자 한다.

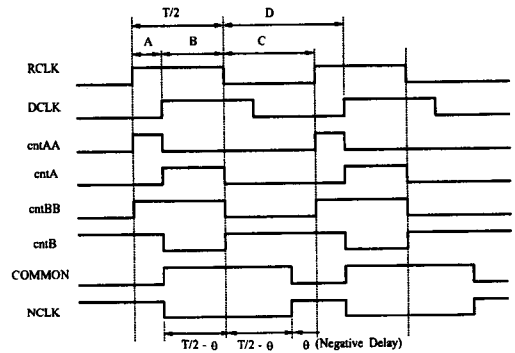
1. Feedback loop가 없는 고속 동기 회로

1) 부 지연 회로(Negative Delay Circuit [NDC])

NDC 회로는 SDRAM에 입력되는 외부 클럭의 triggering edge보다 앞서는 내부 클럭의 triggering edge를 구현하여 데이터 출력 버퍼에 공급함으로써 클럭 액세스(tAC) 특성을 개선하기 위하여 사용한다. NDC는 [그림 2] 에서처럼 기본 지연 단위가 직렬로 반복되어 구성된다. 기본 지연 단위는 지연 동작을 수행하는 두 개의 버퍼와 부(negative) 지연된 클럭의 edge를 결정하는



(그림 2) NDC 구성 및 기본 지연 단위 회로



(그림 3) NDC의 동작 파형

wired-or 회로로 구성된다. wired-or 회로의 common 노드는 모든 기본 지연 단위에 연결되어 부(negative) 지연된 클럭의 펄스 폭을 결정하게 된다. [그림 2]에서 원하는 부 지연 시간을 측정하기 위한 지연 측정 블럭(DMB)과, 부 지연 클럭을 발생 시키기 위한 지연 클럭 발생 블럭(DGB)을 보였다. 입력 버퍼를 지난 RCLK을 원하는 부 지연 시간만큼(θ) 지연시킨 클럭을 DCLK, 부 지연된 클럭을 NCLK이라고 했을때, NDC 기본 지연 단위 회로의 동작을 결정하는 제어 신호는 RCLK과 DCLK을 이용하여 생성하고 이 때 NDC의 동작을 [그림 3]에 보였다.

구간 A($0 < t < \theta$)에서 DMB는 방전 동작을 행하며 모든 기본 지연 단위 회로에서 노드 a는 로우(low)가 되고, DGB는 전 주기 신호의 버퍼 동작을 계속 수행한다. 이때 제어신호 cntBB도 하

이(high)이므로 방전 동작을 행하지만 제어신호 cntB가 제어하는 트랜지스터의 구동 능력이 더욱 크기 때문에 버퍼 동작을 하게 된다.

구간 B($\theta < t < T/2$)에서 DMB는 버퍼 동작을 하게 되어 입력 신호를 지연 시킨다. 이때 단위 버퍼의 지연 시간이 t_{but} 라 하면, $T/2 - \theta = k t_{but}$ 가 될 때 $k-1$ 번째까지의 기본 지연 단위 회로의 노드 a들은 입력 신호의 지연된 형태이고 k 이후의 기본 지연 단위 회로의 노드 a는 로우(low) 상태를 계속 유지한다. DGB는 방전 동작을 수행하고 모든 노드 b는 로우(low)가 된다.

구간 C($T/2 < t < T$)에서 DMB는 전 상태를 계속 유지하게 되고 기본 지연 단위 k 번째까지 지연 동작을 행하는 버퍼가 된다.

구간 D($T/2 < t < T + \theta$)에서 DGB는 버퍼 동작을 수행하며 k 번째 기본 지연 단위 회로의 노드 b가 하이(high)가 될 때부터 common 노드가 로우(low)가 되고, 이 신호를 반전 시킨 것이 부지연된 NCLK이 된다.

이러한 NDC의 동작 영역은 $T/2 - \theta > 0$, $T/2 - \theta < Nt_{but}$ 를 충족해야 하므로 $T/2 - Nt_{but} < \theta < T/2$ 가 되고, 부지연 클럭 펄스 폭(NPW)은 2θ 가 된다.

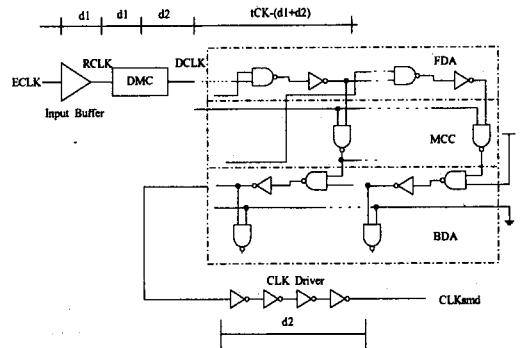
Toshio Yamada가 제안한 이와 같은 NDC[4]는, 첫째 θ 에 클럭 구동 버퍼 지연 시간(d_2)가 포함되지 않고, $\theta < d_2$ 라면 NCLK은 ECLK보다 뒤서게 된다. 즉 $\theta = d_2 + \alpha$ 가 되어야 하며, 이때 α 가 진정 원하는 부지연 시간이 된다. 그렇지만 여전히 입력 버퍼 지연 시간(d_1)은 고려되지 않았다. 입력 버퍼는 LVTTTL이나 SSTL 신호를 받아서 CMOS 레벨로 전환하기 때문에 NCLK의 jitter 요인이 될 수 있다. 따라서 θ 에 d_1 까지 고려한다면 $\theta = d_1 + d_2 + \alpha$ 가 되고, NDC의 동작 영역은 $T/2 - (d_1 + d_2) > 0$, $T/2 - (d_1 + d_2 + \alpha) > 0$, $T/2 - (d_1 + d_2 + \alpha) < Nt_{but}$ 를 모두 충족해야 하므로 $T/2 - Nt_{but} < (d_1 + d_2 + \alpha) < T/2$ 가 되며, 이상으로부터 NDC가 허용할 수 있는 부지연 시간(t_{NDC})의 영역은 $0 < t_{NDC} < T/2 - (d_1 + d_2)$ 이고 클럭 사이클 시간의 영역은 $2(d_1 + d_2 + \alpha) < T < 2((d_1 + d_2 + \alpha) + Nt_{but})$ 이다. 이때의 부

지연 클럭 펄스 폭(NPW)은 $2(d_1 + d_2 + \alpha)$ 이 된다. NPW가 클럭 주기와 같아지면 NCLK이 생기지 않을 수도 있어 현실적으로 데이터 출력 버퍼의 triggering 펄스로 사용하기에는 어렵게 된다. 두 번째는 지연 시간을 측정하는 체인과 지연 시간 후에 펄스를 발생하는 체인의 특성이 정확하게 일치하지 않을 경우 오차가 생기게 된다. 세 번째 지연 시간의 측정 및 발생이 모두 $T/2$ 에 의존하게 되므로 정확히 50% 듀티 사이클(duty cycle)을 갖는 입력 클럭을 사용하여야 하며 듀티비가 50% 이하이면 지연 시간 θ 보다 빠른 클럭이 발생하고, 50% 이상이면 지연 시간 θ 보다 늦은 클럭이 발생할 수 있다.

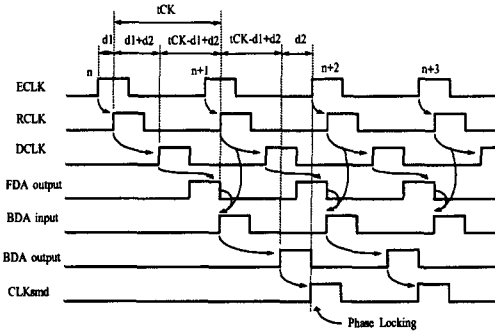
2) Synchronous Mirror Delay (SMD)

SMD는 NDC와 마찬가지로 feedback이 없는 회로로서 두 사이클 만에 클럭 skew를 제거 한다. Takanori Saeki가 발표한 SMD[5]를 [그림 4]에서 보였다. SMD의 구성은 Forward Delay Array (FDA), Backward Delay Array(BDA), Mirror Control Circuit(MCC), 입력 버퍼와 클럭 구동 버퍼를 복사한 Delay Monitor Circuit(DMC)으로 이루어져 있다. 입력 버퍼의 지연 시간을 d_1 , 클럭 구동 버퍼의 지연 시간을 d_2 라고 했을때, SMD의 동작을 [그림 5]에 보였다.

임의의 n 번째 클럭의 DMC 출력 상승에지에서부터 $n+1$ 번째 클럭의 입력 버퍼 출력 상승에지까지 ($t_{CK} - d_1 - d_2$)를 측정하고, SMD 회로에서 이 시간



(그림 4) SMD의 구성 및 개념도



〈그림 5〉 SMD의 위상 동기 동작파형

을 지연시켜 $n+2$ 번째 외부 클럭과 동기 된 CLKsmd를 발생시키게 된다. 즉 $n+1$ 번째 클럭의 입력 버퍼 출력이 MCC로 입력 될 때 n 번째 클럭의 FDA출력이 BDA로 입력되고, $tCK-d1-d2$ 만큼 시간이 지연된 후 n 번째 클럭의 BDA 출력은 $n+2$ 번째 클럭 보다 출력 구동 드라이버 지연 시간 만큼 앞서고, 출력 구동 드라이버를 거친 CLKsmd는 외부 클럭 ECLK와 위상이 동일해진다.

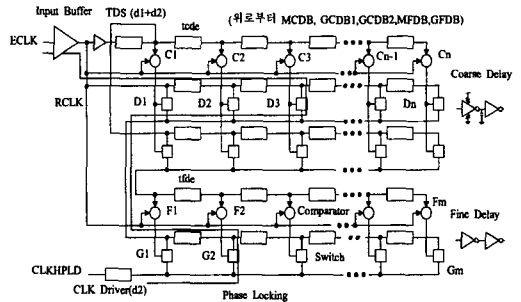
SMD에서 사용 가능한 지연 시간의 합 $tDMC (=d1+d2)$ 는 $tCK-tDMC > 0, tCK-tDMC < N t_{but}$ 를 만족해야 하므로 $tCK-N t_{but} < tDMC < tCK$ 가 된다. 그리고 SMD가 허용할 수 있는 Locking 영역($tLOCK$)은 $tDMC < tLOCK < (tDMC + N t_{but})$ 가 된다.

SMD는 정확한 동작을 위하여 FDA 와 BDA의 지연 단위 셀들의 지연 시간이 일치하도록 설계되어야 하며, 특히 BDA에 배열 된 load를 MCC의 패턴과 같도록 주의 하여야 한다. 또 [그림 4]에서처럼 간단한 MCC를 사용한다면 i 번째 클럭의 FDA 출력 하강에지와 $i+1$ 번째 클럭의 입력 버퍼 출력 상승에지사이 시간을 측정하여 CLKsmd를 발생시키므로 입력 클럭의 펄스 폭에 관계되는 출력을 만들게 되어 입력 클럭 펄스 폭이 $d1+d2$ 보다 크면 초기 오류가 발생 할 수 있다.

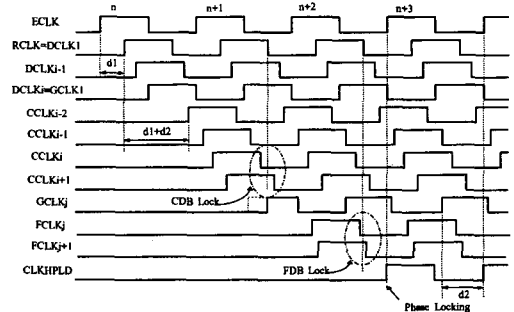
3) Hierarchical Phase Locking Delay (HPLD)

일반적으로 입력 클럭을 지연시켜 위상을 동기 시키는 회로는 주파수 대역이 낮아지면 외부 클럭 과 위상이 일치한 내부 클럭을 얻기 위하여 더욱

많은 delay cell을 필요로 하게 된다. 이것은 위상 동기 회로가 전체 칩 에서 차지하는 면적이 커지게 되므로 다른 로직 회로나 메모리 셀 효율을 나쁘게 한다. 이러한 문제를 극복하면서 넓은 주파수 영역에서 동작하도록 한 클럭 발생 회로가 J. H Han이 발표한 HPLD[6]이다. HPLD는 두 개의 위상 동기화 단계를 갖는 것으로 [그림 6]에서처럼 큰 지연 시간 단위 블럭(CDB; Coarse Delay Block)과 작은 지연 시간 단위 블럭(FDB; Fine Dealy Block) 및 Tracking Delay Stage(TDS)로 구성된다. TDS는 SMD회로의 DMC(Dealy Monitor Circuit)과 같은 역할을 하며, CDB 및 FDB는 앞서 설명한 SMD와 유사한 동작을 수행하기 위하여 각각 동일한 지연 소자로 구성된 지연 시간 측정 체인 MCDB(Measure CDB) 및 MFDB와 지연 클럭 발생 체인 GCDB(Generate



〈그림 6〉 HPLD의 구성 및 개념도



〈그림 7〉 HPLD의 위상 동기 동작파형

CDB) 및 GFDB를 포함한다. GCDB는 MFDB 및 GFDB에 동일한 부하 조건으로 1차 위상 동기 신호를 전달하기 위하여 GCDB1, GCDB2의 두 개 체인을 갖는다. HPLD의 동작은 [그림 7]에서 보였다. CDB에서 입력 버퍼를 지난 임의의 RCLKn+1의 하강 에지에 대하여 MCDB를 지난 CCLKn의 하강 에지를 CDB의 각 지연 단계에서 비교하여 위상 검출 영역에 있는 CDBi에서 tapping한다. 이때 GCDB1 및 GCDB2에서 tapping된 DCLKn은 각각 GFDB 및 MFDB로 입력되고, FDB에서는 RCLKn+2의 하강 에지에 대하여 MFDB를 지난 FCLKn의 하강 에지를 각 지연 단계에서 비교하여 위상 검출 영역에 있는 FDBj에서 tapping한다. 이때 tapping된 FCLKn은 GFDB를 지난 뒤 ECLKn+3에 대해 클럭 구동 버퍼 지연 시간 d_2 만큼 앞서서 GCLKn+1이 된다. 따라서 클럭 구동 버퍼를 지난 GCLKn+1은 ECLKn+3와 위상이 같은 CLKHPLD가 된다. 즉 4 사이클 만에 외부 클럭과 내부 클럭의 위상이 같아지게 된다.

HPLD에서 사용 가능한 지연 시간의 합 t_{TDS} ($=d_1+d_2$)는 $t_{CK}-t_{TDS} > 0$, $t_{CK}-t_{TDS} < It_{cde} + Jt_{fde}$ 를 만족해야 하므로 $t_{CK}-It_{cde}-Jt_{fde} < t_{TDS} < t_{CK}$ 가 된다. 그리고 HPLD가 허용할 수 있는 Locking 영역(t_{LOCK})은 $t_{TDS} < t_{LOCK} < (t_{TDS} + It_{cde} + Jt_{fde})$ 이 된다.

HPLD는 NDC나 SMD에 비해 위상 동기까지 두 사이클이 더 필요하나 로우 액세스(Row Access)에서 부터 CL(cas latency) 3 이후의 동작에서는 저전력 모드(standby mode)시 전력 소모를 줄이기 위하여 동작하지 않아도 된다. 또한 SDRAM의 모드 레지스터 셋(set) 주기 동안에 미리 위상 동기화를 하였다면, edge-triggered register 형태의 위상 비교기가 정보를 갖고 있기 때문에 1 사이클 위상 동기가 가능하다. 그러나 외부 클럭이 스큐를 가지면 동기 정보가 풀리게 되어, 다시 동기화를 수행해야 하므로 동작 모드에 주의 하여야 한다. 다른 클럭 동기화 구성들(SMD, NDC)은 레이아웃 면적의 증가 없이 비교적 낮은 주파수에서도 동작하도록 하려면 기본

지연 단위 버퍼의 지연 시간을 크게 설정해야 하므로 출력 클럭의 jitter를 크게 만들지만, HPLD는 이런 단점을 해결하였다. 그러나 CDB에서 FDB로의 신호 전달시 발생하는 지연 시간보다 CDB에서의 위상 비교기 셋업 시간이 항상 커야 하고, 따라서 두 클럭(CCLK, DCLK)의 샘플링 마진을 줄이기 위하여 비교기의 셋업 시간을 줄이는데 제한이 있으며, 이는 CDB의 최소 단위 지연 시간 간격도 제한하게 된다. 그리고 CDB의 큰 지연 시간을 위해 폴리(poly) 저항을 사용하였을 때, 온도 및 전원 전압, 공정 프로세스에 의한 지연 시간 변화는 tapping 지점이 움직여 jitter가 발생할 수 있다. CDB 단위 지연 시간($1 > ns$) 만큼의 jitter는 SDRAM의 불안정한 동작을 유발할 수 있다.

2. Feedback Loop가 있는 고속 위상 동기 회로

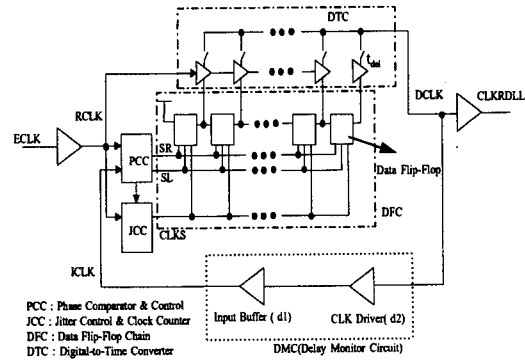
앞 절에서 설명한 방법들은 온도, 전원 전압, 공정 프로세스 등의 환경 변화에 대해 입력 버퍼와 클럭 구동 버퍼에서의 지연 시간 변화는 반영이 되어 클럭 스큐에 영향이 없지만, 단위 지연 시간 블록에서의 지연 시간 변화는 feedback loop가 없기 때문에 환경 변화에 따라 클럭 스큐가 발생할 수 있다. 따라서 본 절에서는 지연 체인을 사용하고 feedback loop를 통해 환경 변화에 대한 jitter를 최소화 하도록 하며 아울러 SDRAM의 standby 동작 시 전력 소모를 줄이기 위하여 고속 위상 동기를 하는 방법에 대해 서술한다.

1) Register controlled DLL(RDLL)

DLL은 PLL과 같이 아날로그 회로인 오실레이터가 없으므로 SDRAM의 저전력 모드 동작 시 전력 소모를 줄일 수 있다. 하지만 전형적인 DLL들은 여전히 아날로그적인 전압 조절 회로를 이용하여 시간 지연 동작을 수행하므로 전원 전압의 변동에 영향을 받으며 고속 위상 동기 동작을 하기가 어렵다. 따라서 지연 시간 추출을 위해 디지털 지연 소자를 이용한 DLL이 연구되었다^[7]. 디지털 DLL은 데이터 플립-플롭과 디지털 시간격 추출 체인을 이용해 매 클럭마다 지연 시간을 늘려 나가는 방법으로 입력 클럭과 내부 클럭의 위

상을 동기 시키고 난 후, 두 클럭의 위상을 주기적으로 비교하여 환경 변화에 의한 jitter를 보상하도록 한다.

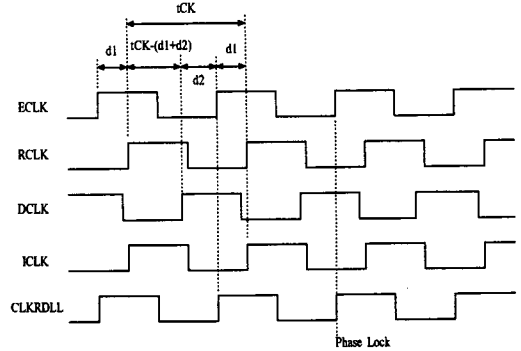
RDLL은 [그림 8]에서 처럼 디지털 값의 시 간격 변환 체인(DTC), 데이터 플립-플롭 체인(DFC), 위상 검출기 및 위상 조절기로 구성된 회



〈그림 8〉 RDLL의 구성 및 개념도

로(PCC), 클럭 카운터 및 선택기로 구성된 jitter 조절 회로(JCC), 지연 시간 측정 회로(DMC)로 구성된다. RDLL의 동작은 초기 모든 데이터 플립-플롭은 리셋되어 0 값을 갖고 있으며, PCC에서 위상 검출기는 입력 버퍼를 지난 RCLK과 DMC를 지나 feedback되는 ICLK를 비교하여 그 값을 위상 조절 회로를 통해 DFC와 JCC로 보낸다. 이때 i번째 데이터 플립-플롭은 지연 시간의 tapping지점을 결정하기 위해 i번째 와 i+1번째 지연 시간 소자를 함께 조절한다. JCC는 DLL 자체 jitter 발생을 억제하고자 RCLK과 ICLK의 위상이 동기화 되기 전까지는 단순히 RCLK을 지연시켜 데이터 플립-플롭 트리거 클럭으로 사용되는 CLKS를 만들고, 위상이 동기 된 이후에는 CLKS를 차단시키고 클럭 카운터에서 128 사이클마다 CLKS를 동작시킨다. 따라서 ICLK가 동기화 검출 영역에 올 때까지 매 사이클 마다 위상이 비교되면서 DTC의 지연 시간 단계를 늘려 간다. 임의의 n번째 단계에서 위상이 동기화 되면 더 이상 지연 시간의 증가는 일어나지 않으며, 이때 DFC가 위상 동기 정보를 계속 유지하고, JCC가 128

사이클마다 위상을 재조절 한다. 이때 RDLL의 자체 jitter폭은 $\pm t_{del} / 2$ 가 되는데, 위상 검출기는 환경 변화에 의한 클럭 스쿠가 t_{del} 값을 넘기 전에는 위상 재조절 시 발생하는 jitter를 억제하도록 설계 된다. [그림 9]에 RDLL의 위상 동기시각 클럭의 위상을 나타내었다.



〈그림 9〉 RDLL의 위상 동기시 동작과형

RDLL의 동작 조건은 $tCK - (d1 + d2) > 0$, $tCK - (d1 + d2) < N t_{del}$ 만족해야 하므로 Locking영역은 $(d1 + d2) < tLOCK < (d1 + d2) + N t_{del}$ 이 된다.

RDLL의 위상 동기화를 SDRAM의 power-up 후 부터 모드 레지스터 셋(set) 사이에 이루고 나서 정상 동작 시(Active mode)에만 이용하면 2~4 사이클 만에 동기화를 이룰 수 있으므로 고속 위상 동기 및 저전력 모드에서 전력 소모를 막을 수 있다. RDLL의 설계시 주의 해야 할 것은 위상 검출기의 셋 업 및 홀드 시간이 t_{del} 보다 작게 하여야 한다. 그리고 데이터 플립-플롭에서 위상 조절 신호가 CLKS에 대해 충분한 셋 업 시간을 갖도록 해야 한다.

2) Self Measure-Controlled DLL (SMDLL)

SMDLL은 NDC, SMD, HPLD 와 같은 고속 위상 동기 동작을 RDLL 자체적으로 수행하여 위상 동기화를 이룬 이후에 RDLL의 동작을 따르게 한 것이다. 이를 위해 SMDLL에는 RDLL의 DTC와 동일한 시간격의 디지털 값 변환 체인(TDC)이 필요하다. 즉 TDC에서 두 사이클에 클럭 지연 시

간을 측정하여 위상 동기를 이루고 난 뒤, 변환된 디지털 값을 데이터 플립-플롭에 저장하면 TDC 경로는 차단되고 DTC 경로를 통해 동기화 된 클럭이 출력 된다. 위상 동기 이후에 카운터에 의해 원하는 사이클마다 위상 비교를 하여 환경 변화에 의한 클럭 스큐를 제거한다. SMDLL은 네 사이클 이내에 위상 동기가 가능하므로 저전력 모드 동작 시 문제가 없으며 negative feedback을 통해 내부 클럭의 위상을 외부 클럭보다 앞서게 만들어 데이터 출력 버퍼에서의 클럭 대 데이터 시간(tAC)를 더욱 개선 할 수 있다. 하지만 SMDLL은 자체 지연 시간 측정 동작에서 루프(loop) 동작으로의 전환 동작에서 정확한 지연 시간 측정을 보장하지 못하면 불안정 할 수 있다.

III. 내부 클럭 발생 회로의 개발 방향

SDRAM의 저전력 및 고속 동작을 위해 내부 클럭 발생 회로는 첫째, 내부 클럭 위상을 4 사이클 이내에 외부 클럭 위상과 동기 되게 하여야 하며 둘째, 전력 소모가 적도록 설계 해야 하며, 셋째 클럭 스큐(skew) 혹은 지터(jitter)의 발생을 최소화 하도록 해야 한다. 앞 절에서 서술한 기술들은 처음 두 가지 사항에 대해서는 대부분 만족하고 있으나 셋째 사항은 각 기술별로 차이를 가지고 있다. 그러나 공통적으로 지연 시간 소자에 의해 존재하는 클럭 스큐는 좁은 지연 시간 간격을 가지는 소자 개발을 통해 감소 할 수 있으며, 클럭 드라이버에서의 긴 배선 패달로 인한 저항 성분 때문에 생기는 클럭 스큐에 대한 대책도 레이아웃 및 floorplan등에서 함께 고려 되어야 한다. 또한 데이터 출력 버퍼(off-chip driver)에서의 데이터 지연 시간도 DMC, TDS 혹은 feedback loop에 모델링 하여 데이터 path에서의 신호 지연을 앞당기도록 해야 한다.

한편 데이터의 입,출력만 내부 동작 주파수의 2배(DDR; Double Data Rate) 혹은 4배(QDR; Quadruple Data Rate)의 주파수로 동작하게 하여

bandwidth를 크게 하려는 요구가 있다. 이를 위해 50% 듀티 사이클을 가지는 클럭과 differential 클럭 혹은 주파수2배 클럭의 발생을 위상 동기화와 연관 지어 고려해야 할 것이다.

짧은 기간에 준비하여 다소 내용이 미흡한 점이 있으나 끝까지 읽어 주신 독자 제위께 심심한 감사를 드리며, 고속DRAM 개발에 도움이 되었으면 한다. 끝으로 많은 조언을 해 주신 엘지 반도체 안진홍 연구위원 및 서울 시립 대학교 최중호 교수께 감사 드린다.

참 고 문 헌

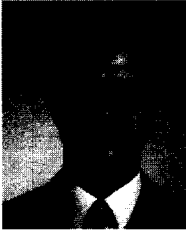
- [1] Steven A. Przybylski, New DRAM Technologies : A Comprehensive Analysis of the New Architectures. Second Edition, MicroDesign Resources, Sebastopol, Calif., 1996.
- [2] Y. Takai et al., "250 Mbyte/s Synchronous DRAM Using a 3-Stage-Pipelined Architecture," IEEE J. Solid-State Circuits, vol.29, pp. 426-431, Apr.1994.
- [3] Y. H. Choi et al., "16-Mb Synchronous DRAM with 125-Mbyte/s Data Rate," IEEE J. Solid-State Circuits, vol.29, pp. 529-533, Apr. 1994.
- [4] T. Yamada et al., "Capacitance coupled Bus with Negative Delay Circuit for High speed and Low Power (10GB/s < 500mW) Synchronous DRAMs," Symp. VLSI circuits, pp. 112-113, June. 1996.
- [5] T. Sakei 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. 1994.
- [6] J. M. Han et al., "Skew Minimization

Techniques for 256M-bit Synchronous DRAM and beyond," Symp. VLSI circuits, pp. 192-193, June. 1996.

[7] A. Hatakeyama et al., "A 256Mb SDRAM

Using a Register-Controlled Digital DLL," IEEE International Solid-State Circuits Conf., pp. 72-73, 1997.

저 자 소 개



朴 富 用

1968年 7月 12日生

1992年 2月 한양 대학교 전자공학과 졸업(공학사)

1994年 2月 한양 대학교 대학원 전자공학과 졸업(공학석사)

1994年 1月~현재 LG반도체 ULSI 연구소 선행제품설계Gr. 주임 연구원

주관심 분야: 고속 DRAM 및 인터페이스 설계, Double Data Rate SDRAM 설계