DLL DESIGN IN SYNCHRONOUS DRAM

ARMANI ALVAREZ

UNIVERSITY OF NEVADA, LAS VEGAS

ECG 721 MEMORY CIRCUIT DESIGN

DEFINITION OF SDRAM

- Asynchronous DRAM uses timing while Synchronous DRAM uses a synchronous clock to execute commands.
- In synchronous DRAM all the commands and data are executed on the edges that are synchronous with the system clock.
- SDRAM first started as Single Data Rate (SDR), then later it was turned to Double Data Rate (DDR). Later it progressed from DDR2, DDR3, DDR4, and now it has progressed to DDR5.
- Single Data Rate meant data was only executed on the rising edge of the clock. DDR transfers data on the rising and falling edges of the clock, in turn doubling the data bandwidth. This means twice the data execution without changing the frequency. [1]

DEFINITION OF DELAY LOCKED LOOP

- The delay-locked loop (DLL) is a circuit fed by a reference clock that attempts to find the period of that reference clock by adjusting the delay of a variable delay buffer in a feedback loop. The loop is locked when the delayed clock signal matches the incoming clock signal.
- The DLL is part of the dedicated DDR SDRAM.
- Problems with PLL output Jitter resulting from VCO output frequency changing with a constant input voltage has led to the concept of DLL. [2,3]

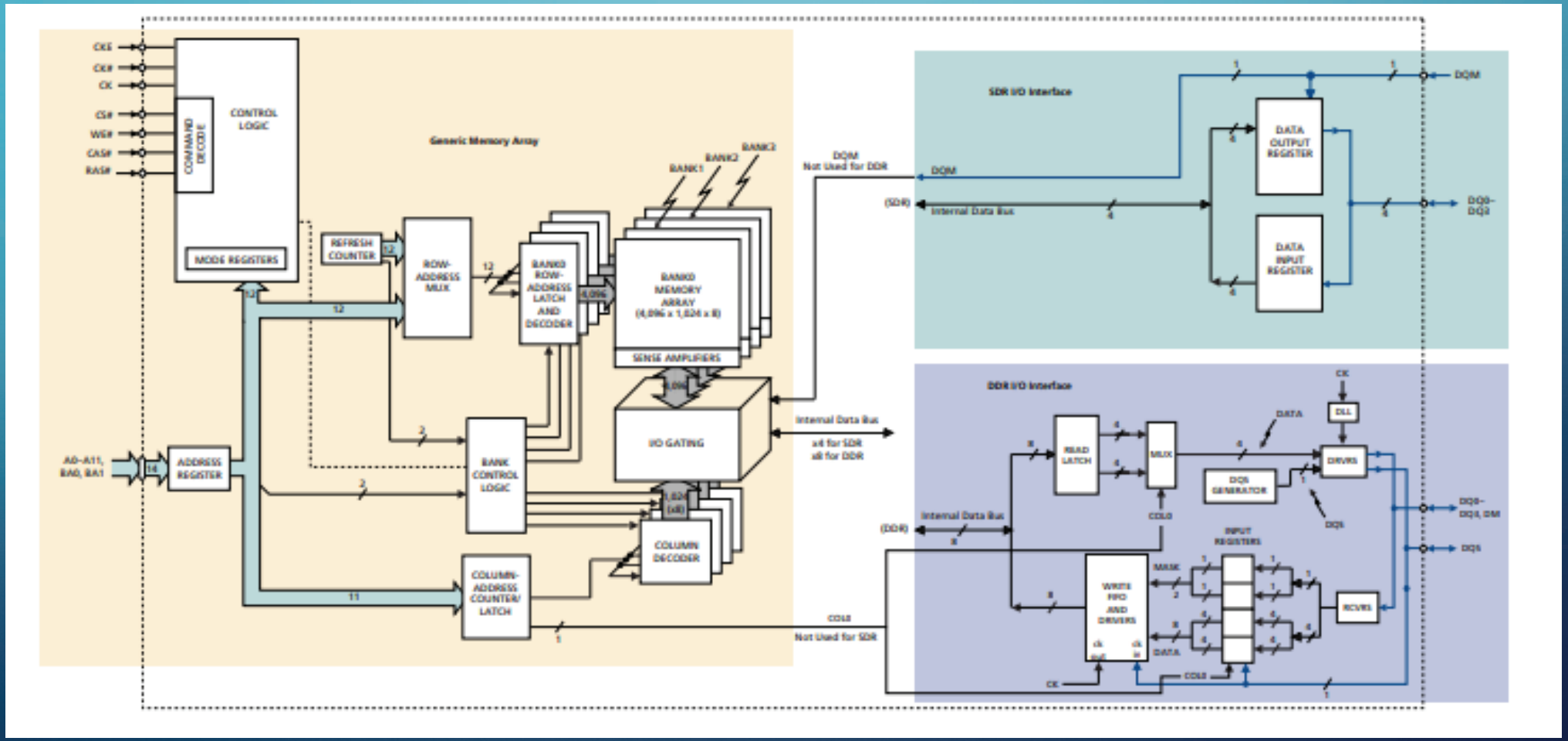
WHY DLL IN SDRAM ?

- DLL is used in Synchronous DRAM because the generated clock can be used to sync the output data with the system clock which allows a high degree of predictability when the output data is accessible which allows an increase in data transfer.

ADVANTAGES OF DIGITAL DLL

- Zeroth order transfer function makes DLL easier to stabilize than other alternatives, meaning that whatever the error is, it has the same exact response.
- Minimum Delay known
- Low clock Skew
- Short Locking Time

DDR SRAM ARCHITECTURE – [4]

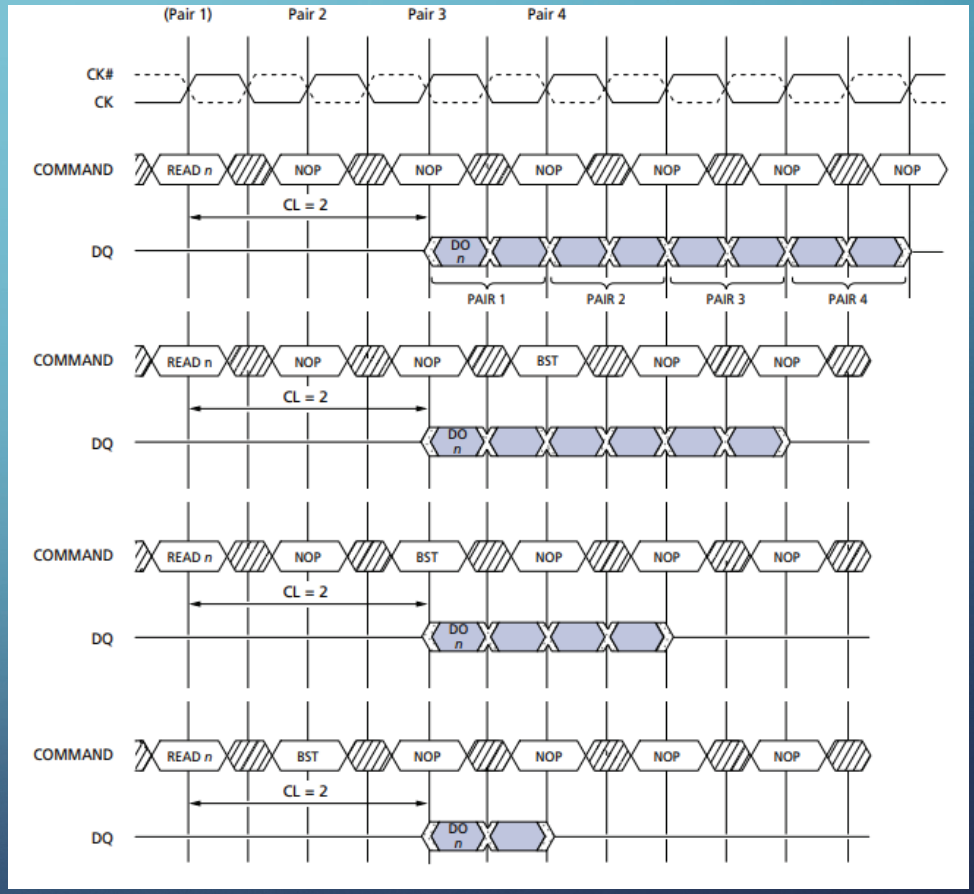


DDR SDRAM ARCHITECTURE

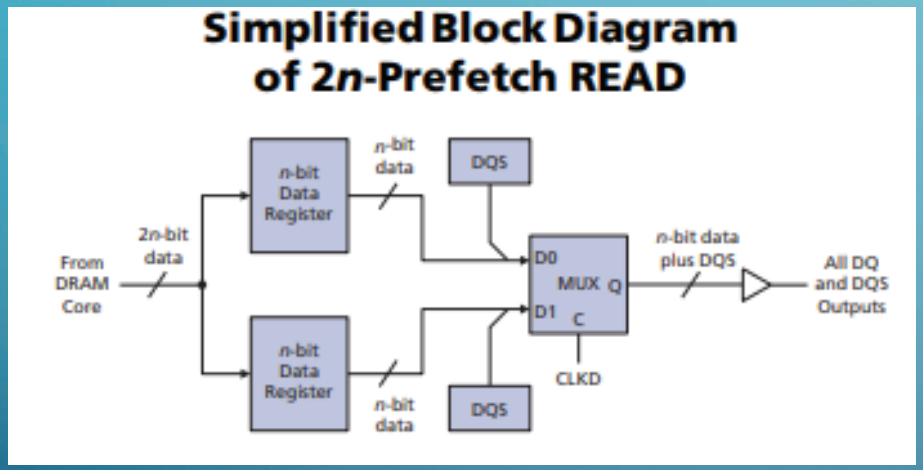
- From the figure of the previous slide:
- The yellow section – Generic Memory Array
- The green section – additional circuitry to make a SDR Array
- The purple section – additional circuitry to make a DDR Array. In this array it can be seen that there is a DQS generator. In addition to the DQS generator, there is also a pre-fetch buffer that allows a single address request to fetch multiple words in bursts that can be data or it can be masked. [4]

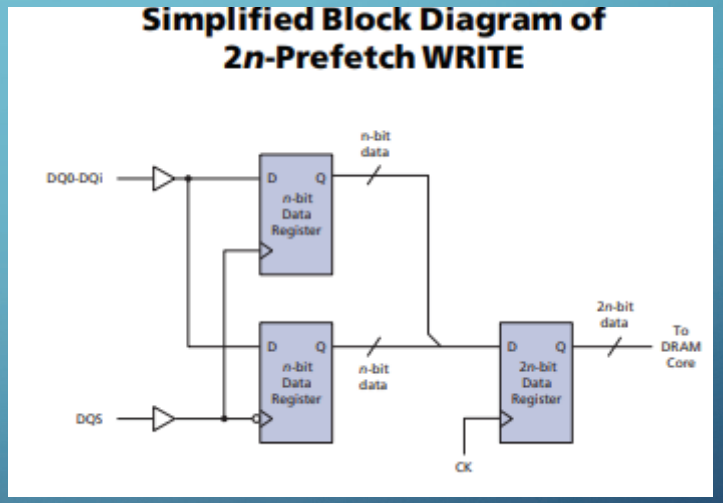
DDR READ

- For READs, the data strobe signals are edge-aligned with the data signals, meaning that all data and data strobes are clocked out of the device by the same internal clock signal, and all will transition at the outputs at nominally the same time. The controller will internally delay the received strobe to the center of the received data eye.
- For READs, the controller can choose to ignore either of the two words, but the time slots for both will be occupied. [4]



SDRAM 2N PREFETCH READ AND WRITE – [4]





SDRAM 2N PREFETCH READ AND WRITE

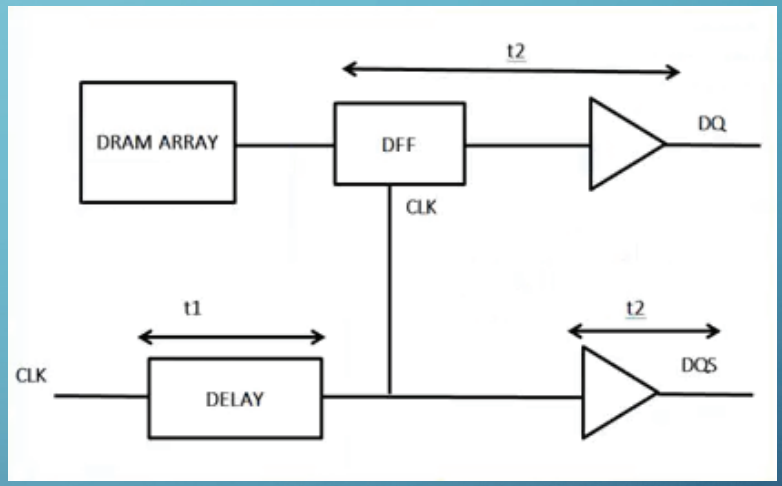
- To the DRAM vendor, 2*n*-prefetch means that the internal data bus can be twice the width of the external data bus, and therefore, the internal column access frequency can be half of the external data transfer rate.
- To the user, from a high-level view, 2*n*-prefetch means that data accesses occur in pairs. For example a single read access fetches two data words; and for a single write access, two data words (and/or data mask bits) must be provided. This affects both the minimum burst size and nonminimum burst interruptions. The minimum burst size of a 2*n*-prefetch architecture is two external data transfers. [4]

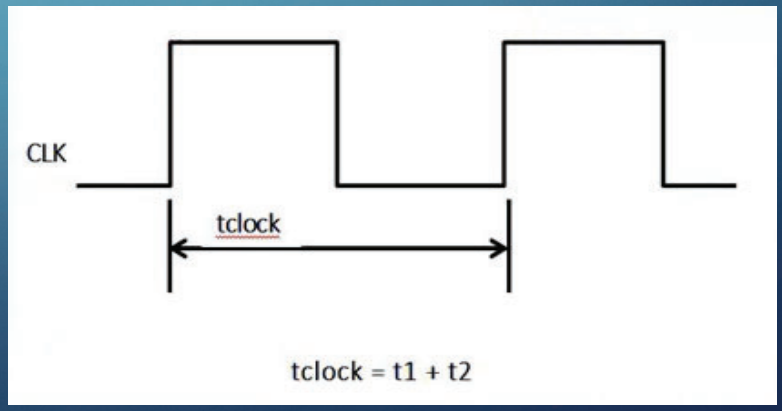
DQS

- Data pins in the DDR circuitry are labeled DQ and the strobe pin is labeled DQS. RAM that use DDR are designed to transfer two data words per clock cycle, using both the positive and the negative edge of the clock.

CLK TO DQS PART 1

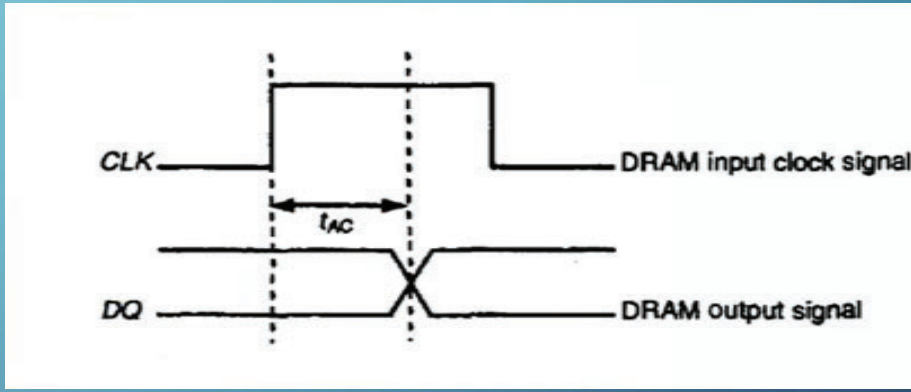
- DRAM is about to be read out through DQ some delay through T2 through flip flop and driver, there is a clock with an added delay, T1, and a DQS buffer.
- The sum of the delays T1 and T2 should match the delay of the clock, Tclock. We need T1 because of PVT that can cause clock skew.





CLK TO DQS PART 2

- The figure on this slide is showing how clock relates to the data, with PVT. The goal of the DLL is to drive access time Tac = 0 so that data syncs with the clock. [5]

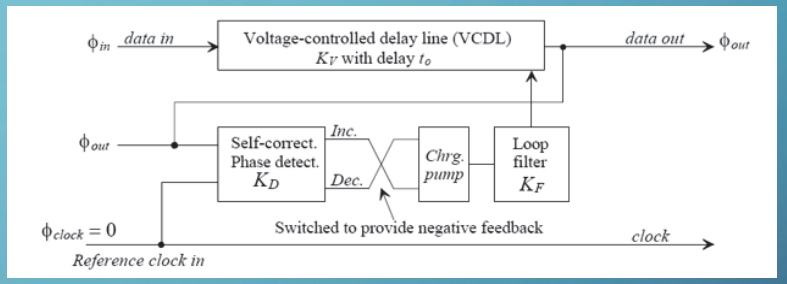


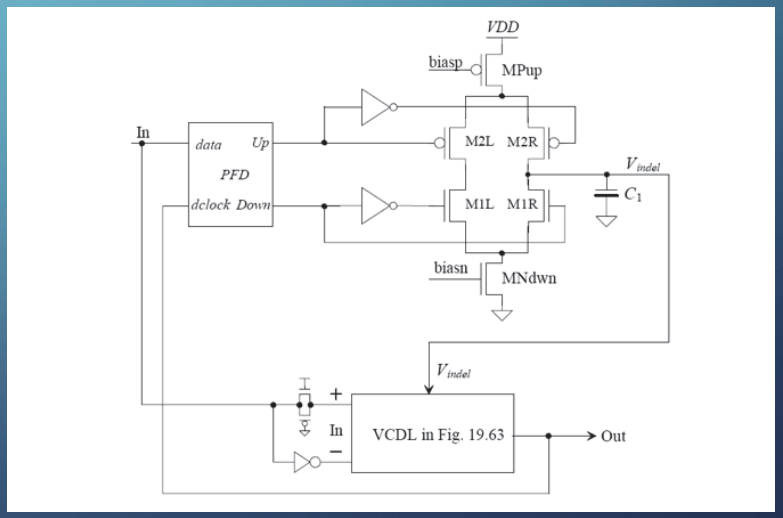
THE DLL

- The basics of the DLL are input buffers, delay lines, feedback with delay model, phase detector, and a shift register. [7]

Important equations for DLL basics:

- PD loop delay = N * Tclock
- Tclock = t(ib) + t(d) (which are subject to PVT variations)
- Delay Line = N * Tclock - [t(ib) + t(d)]
- Ideal forward delay = t(ib) + N * Tclock - [t(ib) + t(d)] + t(d) = N * Tclock



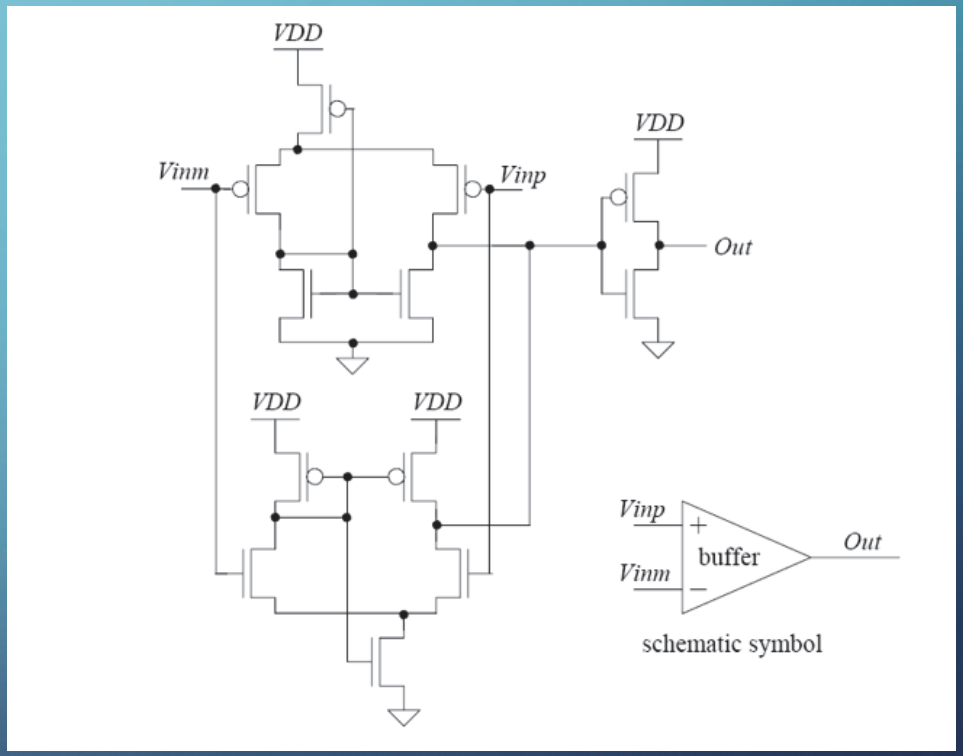


DESIGN IMPORTANCE

- Jitter - Clock jitter in DDR2/3 systems is bound by clock jitter specifications in positive and negative directions. The negative direction corresponds to a smaller clock period and the positive direction corresponds to a larger clock period. If the clock jitter is not Gaussian in nature, then the clock violates the timing specifications. [6]
- Duty Cycle at 50%
- PVT variations
- False Lock

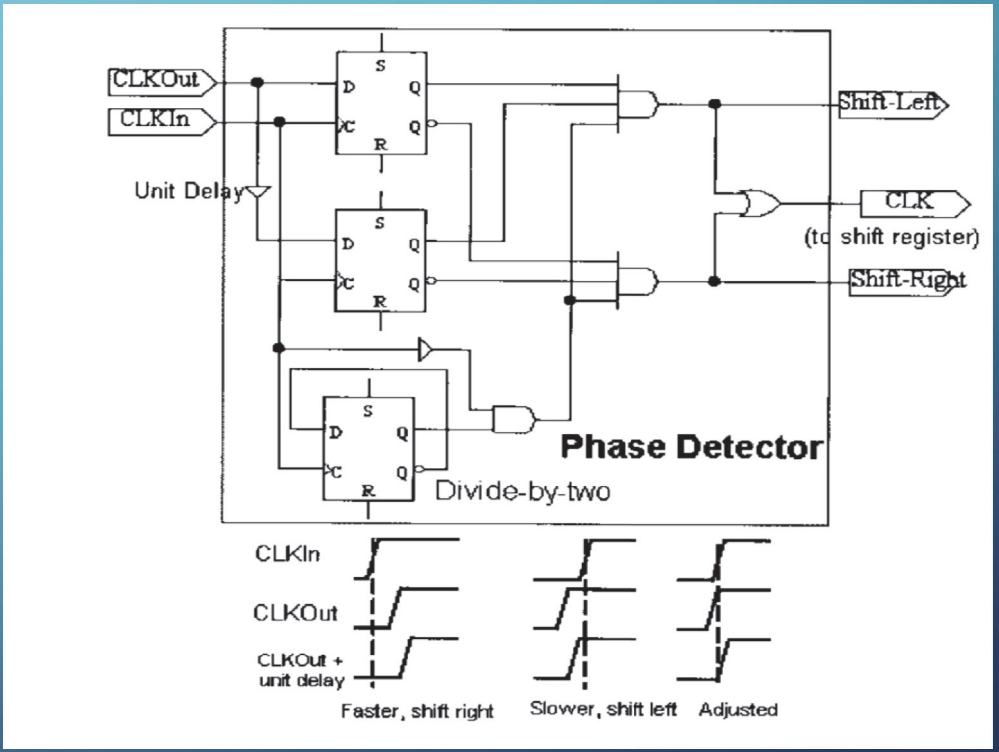
INPUT/OUTPUT BUFFER – [7]

- One of the designs we can use for the input buffer is the rail to rail input buffer. Since the buffers are in parallel, it is complementary in nature which means this buffer design is robust and works over a wide range of operating voltages. This is a self biasing topology and provides a skew-less output.
- Output would just be inverters in series.



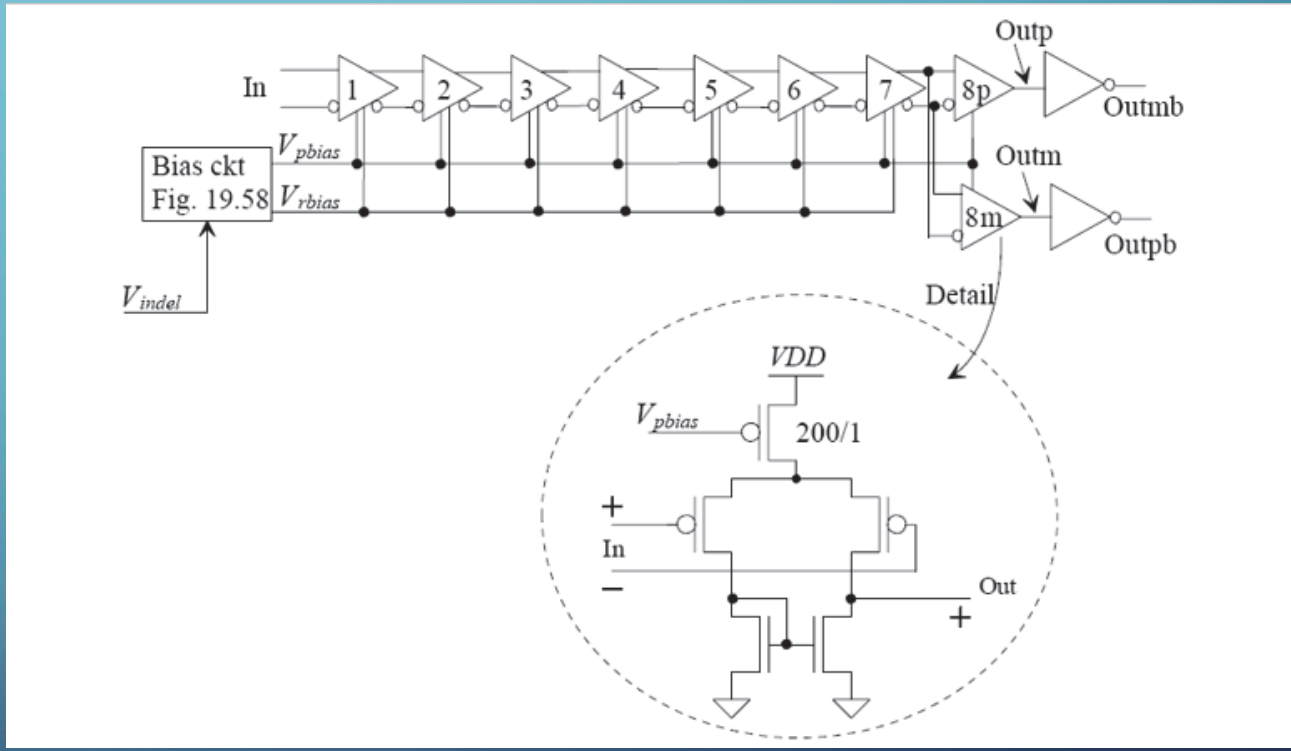
PHASE DETECTOR

To stabilize the movement in the shift register, after making a decision, the phase detector will wait at least two clock cycles before making another decision. A divide by two was included in the phase detector so that every other decision, resulting from comparing the rising edges of the external clock and the feedback clock, was used. This will provide enough time for the shift register to operate and the output waveform to stabilize before another decision by the PD is implemented.[8]



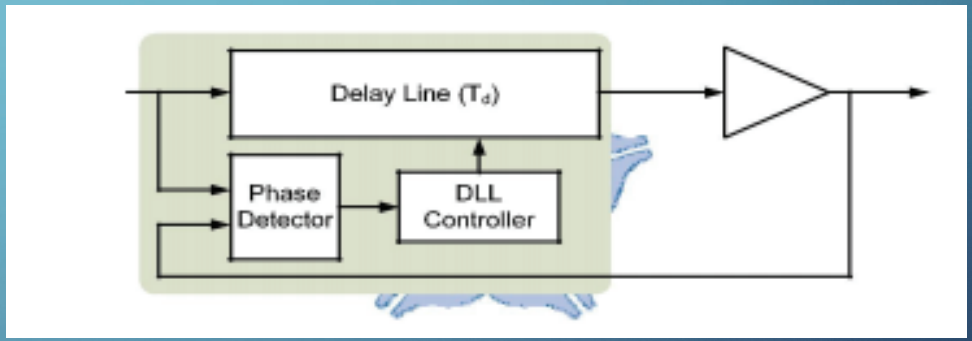
VOLTAGE CONTROLLED DELAY LINE – [7]

In this example the design is modified so that the logic levels will be full, this is done by adding inverters at the end.



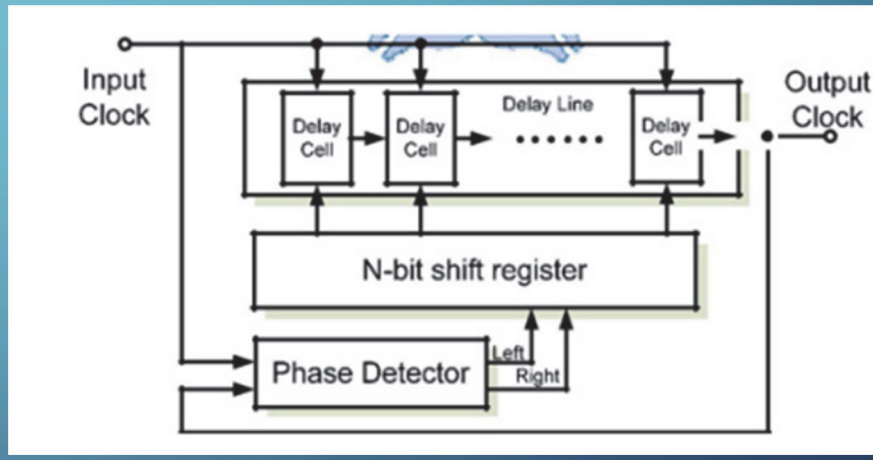
DIFFERENT TYPES OF DIGITAL DLL

A DLL consists of a phase detector, a variable delay line, and a DLL controller to generate the analog or digital control signal for the delay line by PD's output. It selects an optimal delay (Td) to compensate the phase error between reference clock and remote clock. After DLL is locked, the remote clock is synchronized with reference clock. After that, the clock buffer propagation delay (Tc) can be ignored. [9]



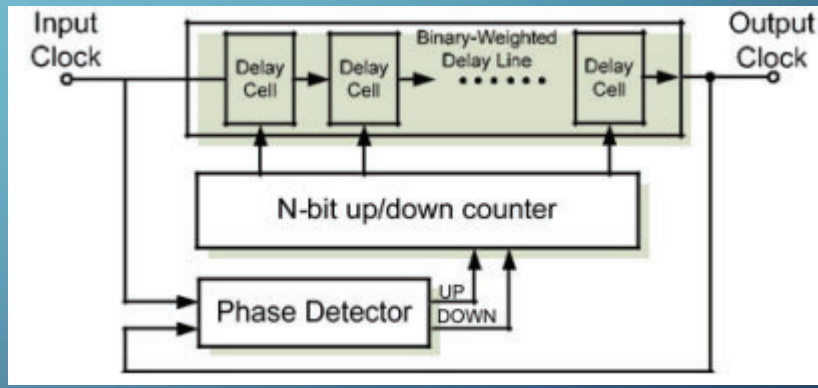
REGISTER CONTROLLED DLL

he n-bit shift registers are controlled by the output of phase detector. The phase detector detects the relationship between input clock and output clock, and generate eft/Right signal for shift register to control the delay time. When the output clock leads input clock, the PD sends “Left” ignal to shift register and the high bits in the shift register vill shift left to increase the delay time and compensate the hase error. Similarly, if PD sends “Right” signal, the high bits n shift register will shift right to decrease the delay time. The LL continuously adjusts the shift register until the phase error s eliminated. [9]



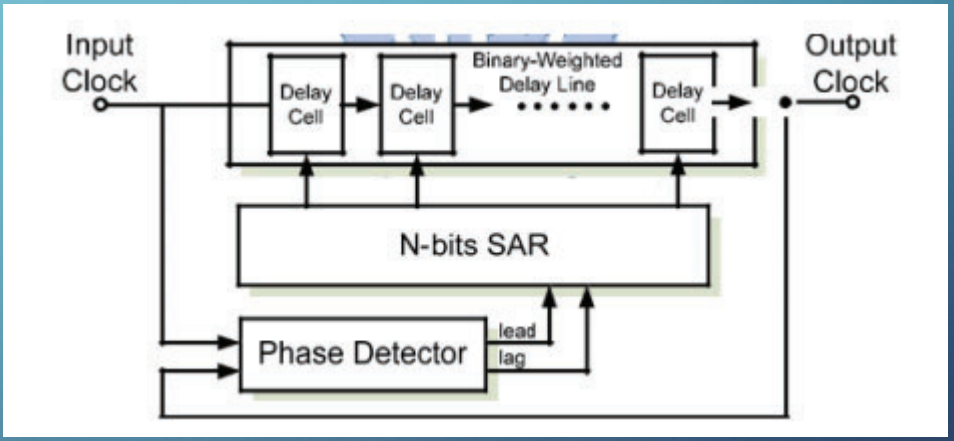
COUNTER CONTROLLED DLL

The phase alignment scheme is similar to register-controlled DLL. The only difference is the structure of the delay cell, the rest of the circuit blocks are similar to register-controlled DLL. The counter-controlled DLL adopts binary weighted delay line, so the delay cells of the delay line have carefully adjusted to maintain a fixed binary-weighted linearity. Meanwhile, the linearity of the delay line will be the key of the resolution of counter-controlled DLL. The n-bit up/down counter updates its value according to the output of phase detector. Phase detector detects lead or lag of the input phase and then outputs UP/DOWN signals to control the Nbit up/down counter. The counter produces binary-weighted control word and adjusts the delay time of delay line until the output clock synchronizes with input clock. The counter-controlled DLL replaces the register-controlled one to reduce the hardware of the controller. [9]



SAR CONTROLLED DLL

The locking time is an important design parameter for Digital DLL. Both register-controlled and counter-controlled DLLs use linear approach manner to search the optimal control code word, thus the locking time will be increased. For example, n-bits counter-based DLL takes 2n clock cycles to lock in worst case. In order to improve the locking time, the binary search algorithm is used to reduce the locking time. [9]



SUCCESSIVE APPROXIMATION REGISTER

- SAR – binary search happens
- The two critical components of a SAR are the comparator and the DAC.

A SAR ADC's speed is limited by:

- The settling time of the DAC, which must settle to within the resolution of the overall converter, for example, ½ LSB
- The comparator, which must resolve small differences in VIN and VDAC within the specified time
- The logic overhead [10]

REFERENCES

- [1] https://www.crucial.com/support/articles-faq-memory/difference-between-ddr4-ddr3-ddr2-ddr-sdram
- [2] https://compas.cs.stonybrook.edu/~nhonarmand/courses/sp15/cse502/res/dramop.pdf
- [3] "Delay-Locked Loop (DLL)." *Delay-Locked Loop (DLL) Definition*, https://www.intel.com/content/www/us/en/programmable/quartushelp/13.0/mergedProjects/reference/glossary/def_dll.htm.
- [4] https://www.micron.com/-/media/client/global/documents/products/technical-note/dram/tn4605.pdf
- [5] Feng Lin, J. Miller, A. Schoenfeld, M. Ma and R. Jacob Baker, "A register-controlled symmetrical DLL for double-data-rate DRAM," in *IEEE Journal of Solid-State Circuits*, vol. 34, no. 4, pp. 565-568, Apr 1999
- [6] https://www.micron.com/-/media/client/global/documents/products/technical-note/dram/tn0456_clock_jitter.pdf
- [7] Baker, R. Jacob. *CMOS: Circuit Design, Layout, and Simulation*. Wiley, 2019.
- [8] Baker, R.. (2000). DRAM Circuit Design: A Tutorial.
- [9] Nagpara, B. H., Karan M. Jani, D.N. Khandhar, "A Review of Different Types of Digital Delay-Locked Loop in 65nm CMOS technology", *International Journal of Modern Trends in Engineering and Research (IJMTER)*, vol. 3, no. 4, ISSN (Online):2349- 9745, Aril 2016
- [10] https://www.maximintegrated.com/en/design/technical-documents/tutorials/1/1080.html