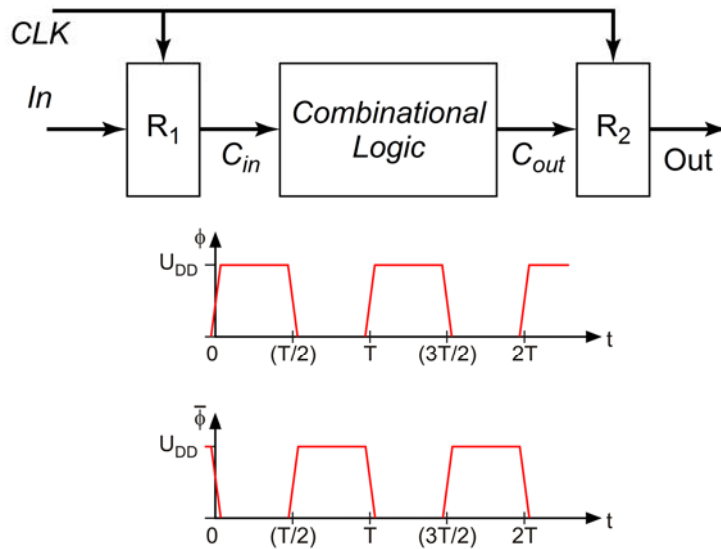


7. Synchrones Timing



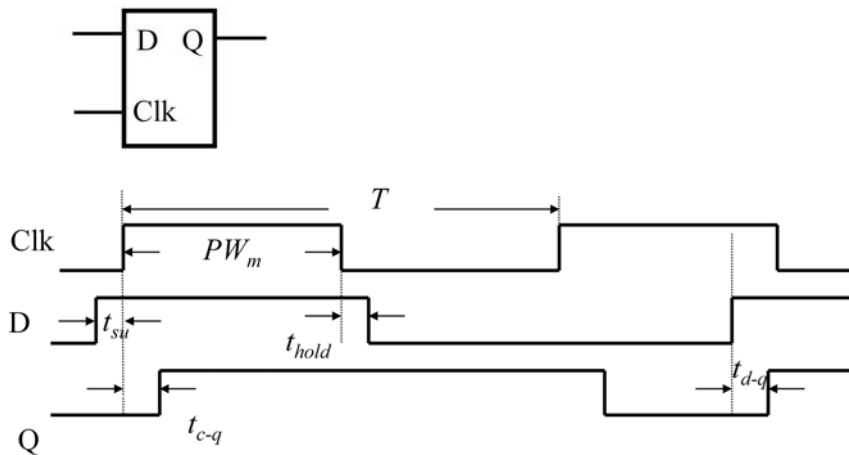
Aufgrund der sehr hohen Taktfrequenzen mit denen logische Schaltungen heute betrieben werden, ist es unumgänglich, sich mit den Herausforderungen an das "Timing" der Schaltungen auseinander zu setzen.

Viele komplexe digitale Schaltkreise werden mit einem so genannten Mehrphasentakt betrieben.

Damir erreicht man höhere Verarbeitungsgeschwindigkeiten, da man immer nur einen Bruchteil der Periodendauer des Taktes als Wartezeit zwischen den Zyklen der Logikblöcke hat.

Hier ist beispielhaft ein 2-Phasen Takt dargestellt.

7.1. Synchrones Timing: Parameter



Delays can be different for rising and falling data transitions

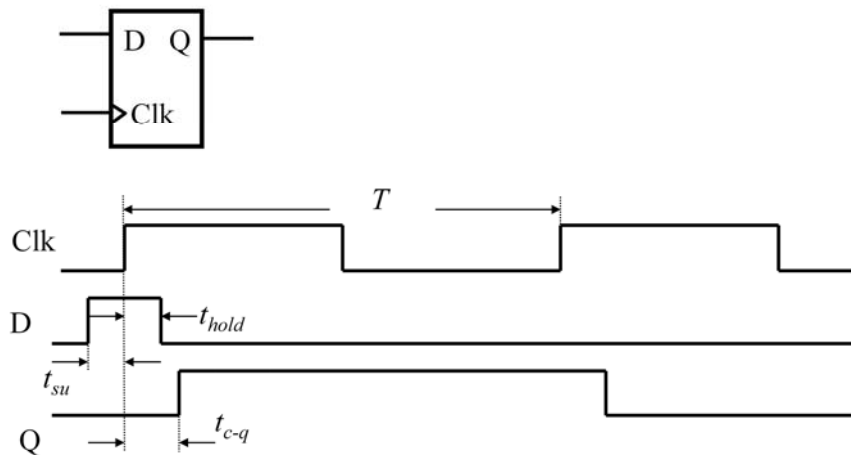
Zunächst wollen wir uns mit den Parametern für synchrones Timing mit transparenten D-FF (Latches) befassen.

Damit die Daten sicher im FF gespeichert werden können, wird eine Setup-Zeit definiert. Dies bedeutet, dass die Daten um diesen Zeitintervall vor dem Wechsel des Taktzustands am Eingang anliegen müssen, damit die korrekte Information am Ausgang anliegt.

Bei Transparenten FF ist diese Größe nicht relevant, da die Daten während des gesamten H-Pegels des Taktes an den Ausgang weiter geleitet werden.

Die Laufzeit durch das FF ist die Zeit t_{c-q} . Wichtiger bei den FFs ist die Zeit t_{hold} , die aussagt, wie lange die Daten am Eingang noch anliegen müssen, wenn das FF in den Speicherzustand übergeht.

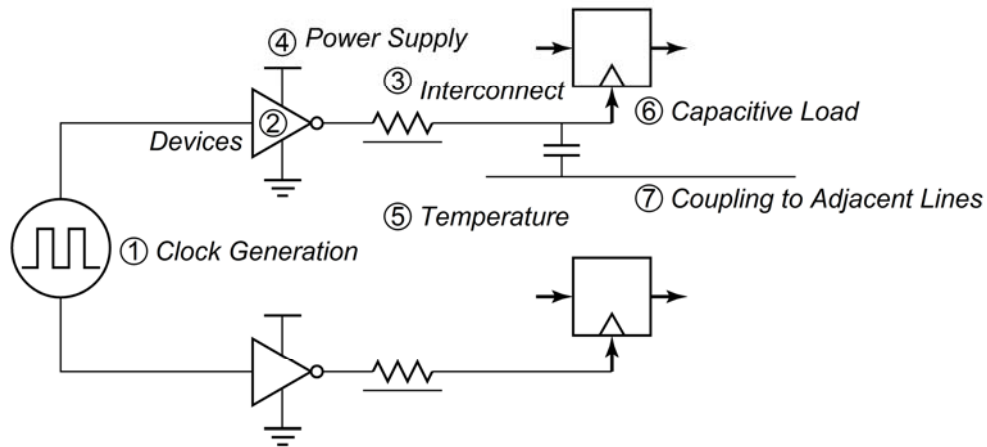
7.1. Synchrones Timing: Parameter



Delays can be different for rising and falling data transitions

Bei flankengesteuerten FF hingegen ist die Zeit t_{su} entscheidend für die korrekte Übernahme der anliegenden Information mit der Flanke der Taktes.

7.1. Synchrones Timing: Takt



Quellen von Taktungenauigkeiten

Da der Takt ein zentrales Element für sehr viele Register auf dem Chip ist, müssen wir auch die Quellen von möglichen Ungenauigkeiten untersuchen.

Schon in der Taktquelle können Ungenauigkeiten entstehen, die zu nicht idealen Periodendauern führen können.

Im weiteren Verlauf der Taktausbreitung können weitere Ungenauigkeiten durch unterschiedliche Laufzeiten in takttreibenden Bauelementen und Verbindungsleitungen zwischen den Bauelementen, durch unterschiedliche kapazitive Lasten in den verschiedenen Pfaden, durch kapazitive und induktive Kopplungen zwischen Takt- und Signalleitungen und unterschiedliche Temperatureinflüsse entstehen.

7.1. Synchrones Timing: Nicht-idealer Takt

■ Takt Skew/Verschiebung

- Verschiebung zeitlich äquivalenter Taktflanken, t_{SK}

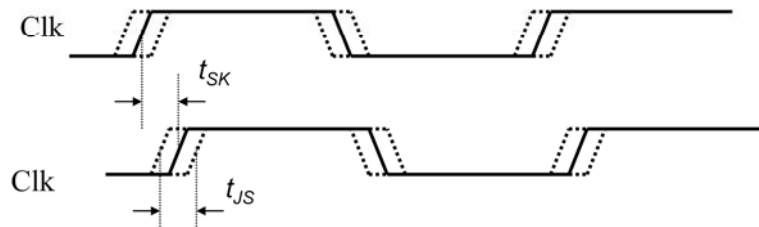
■ Takt Jitter

- Zeitliche Variationen der gleichen Taktflanke
- Zyklus - zu - Zyklus (kurzzeitig), t_{JS}
- Langzeitänderungen, t_{JL}

■ Änderungen der Taktimpulsbreite

Zunächst sollen aber einige Begriffe erläutert werden.

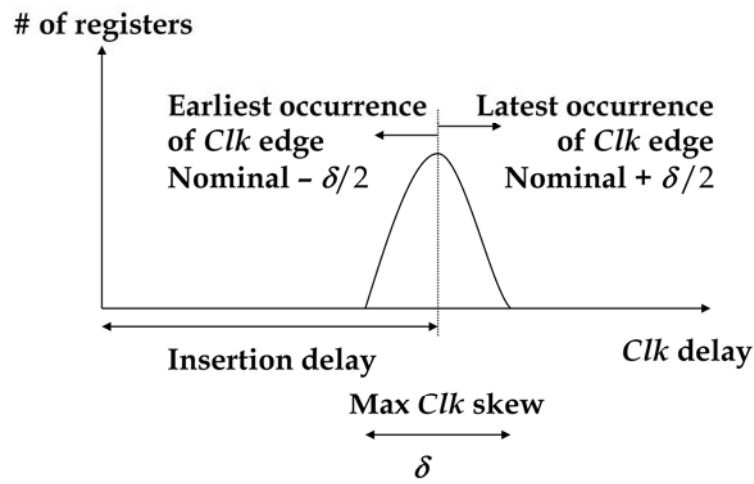
7.1. Synchrones Timing: Skew und Jitter



- Beide Effekte beeinflussen den effektiven Taktzyklus

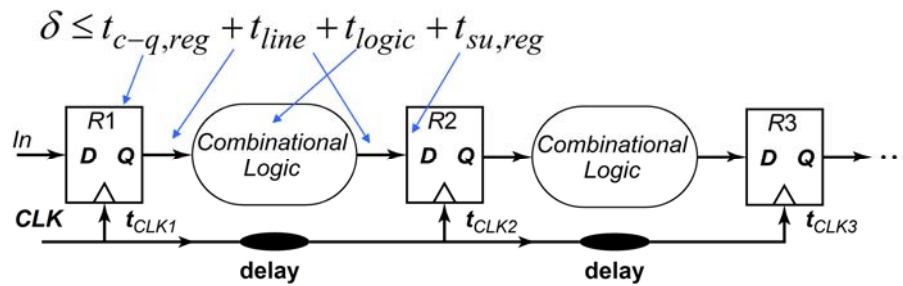
Aus der Darstellung kann entnommen werden, dass sowohl Skew wie auch Jitter den Taktzyklus (Zeitpunkt des Auftretens der Taktflanke und die Periodendauer)

7.1. Synchrones Timing: Skew

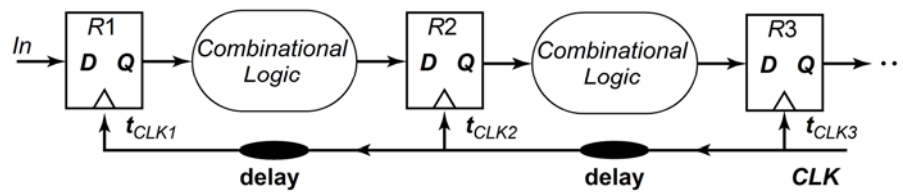


$$\delta = t_{CLK2} - t_{CLK1}$$

7.1. Positive and Negative Skew



(a) Positive skew



(b) Negative skew

Positive Skew:

Daten und Taktausbreitung haben die gleiche Richtung.

In diesem Fall muss der Skew sehr genau kontrolliert werden und der Gleichung genügen.

Wird diese Bedingung nicht erfüllt, kommt es zu Fehlern, unabhängig von der Periodendauer des Taktes.

Andererseits erhöht ein positiver Skew den Datendurchsatz, da die Periodendauer des Taktes um δ reduziert werden kann.

Negative Skew:

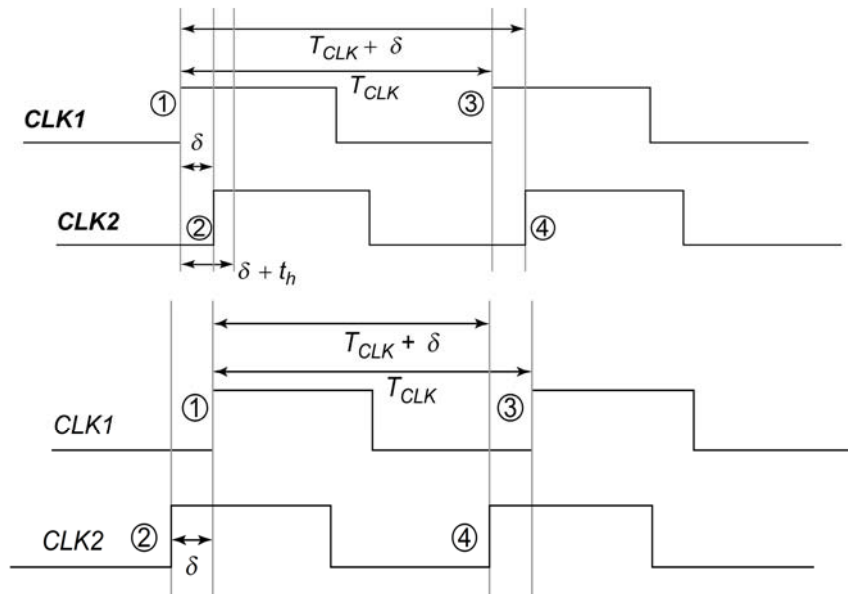
Daten- und Taktausbreitung haben entgegengesetzte Richtungen.

Die Schaltung arbeitet immer korrekt!

Nachteil: die Zeit für aktuelle Berechnungen wird durch den Skew reduziert, so dass die Periodendauer des Taktes um den Wert von $|\delta|$ erhöht werden muss.

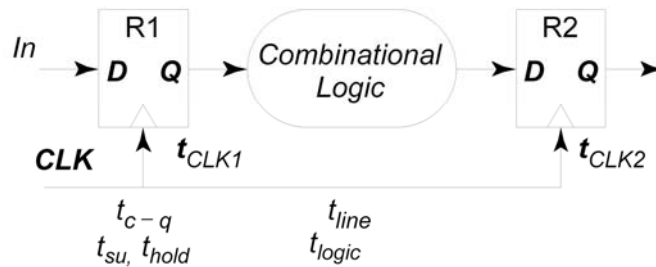
Zusammenfassend lässt sich sagen, dass ein negativer Skew "Katastrophen" verhindert, aber die Performance der Schaltung reduziert.

7.1. Positive Skew, negative Skew



Hier das Impulsbild zu positive und negative Skew

7.1. Timing: Anforderungen



Minimale Zykluszeit:

$$T - \delta = t_{c-q} + t_{su} + t_{line} + t_{logic,max}$$

Hold time constraint:

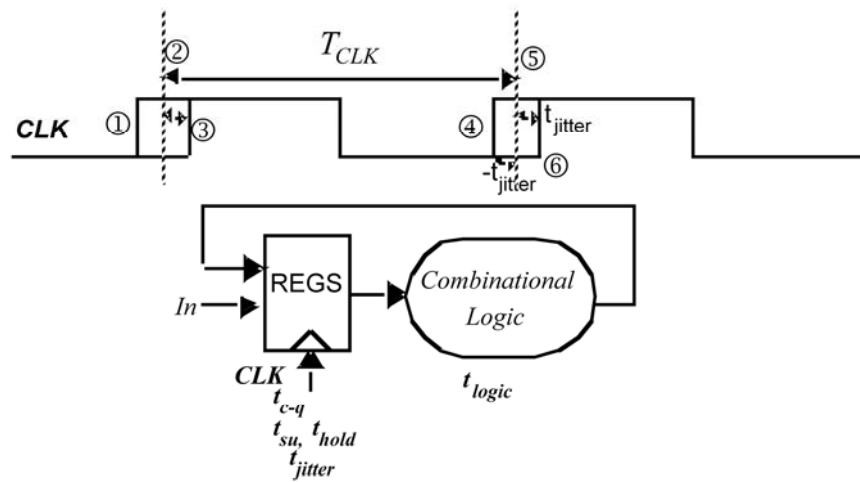
$$t_{c-q} + t_{logic} > t_{hold} + \delta$$

Die Anforderungen an das Timing werden durch Signallaufzeiten in den Logikfunktionen und den Verbindungsleitungen maßgeblich bestimmt.

Die minimale Zykluszeit setzt sich aus den angegebenen Zeiten zusammen.

Die Dauer, die das gespeicherte Signal mindestens anliegen muss ist entsprechend der Formel definiert.

7.1. Timing: Auswirkungen des Jitter

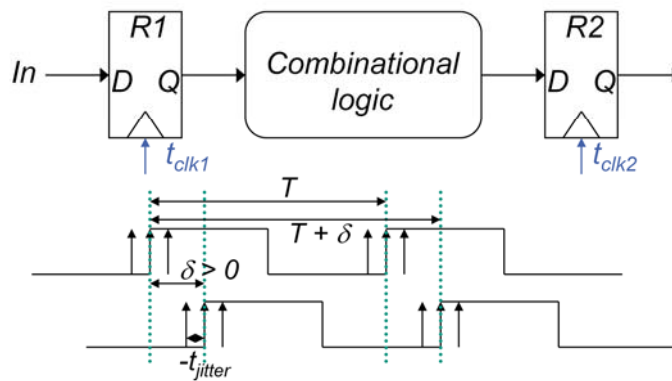


$$T_{CLK} - 2t_{jitter} \geq t_{c-q} + t_{logic} + t_{su} \quad \text{so} \quad T_{CLK} \geq t_{c-q} + t_{logic} + t_{su} + 2t_{jitter}$$

Unter Hinzufügen des Jitters kann man erkennen, dass sich die einzelne Taktperiode entweder vergrößern oder verkleinern kann.

Damit wird durch den Jitter die Leistung des Systems reduziert.

Auswirkungen von Skew und Jitter kombiniert



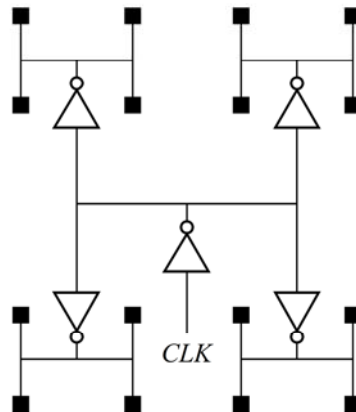
$$T \geq t_{c-q,reg} + t_{logic} + t_{su} - \delta + 2t_{jitter} \quad t_{hold} \leq t_{logic} + t_{c-q,reg} - \delta - 2t_{jitter}$$

Für die minimale Periodendauer des Taktes ergeben sich durch $\delta > 0$ Einschränkungen.

Durch den zusätzlich vorhandenen Jitter wird der akzeptierbare Skew reduziert, was wiederum die Leistungsfähigkeit des Systems reduziert und es erschwert, die Bedingungen für t_{hold} einzuhalten.

7.1. Taktverteilung

H-tree



Techniken der Taktverteilung

Es gibt zwei wesentliche Techniken der Taktverteilung auf hochintegrierten digitalen Chips:

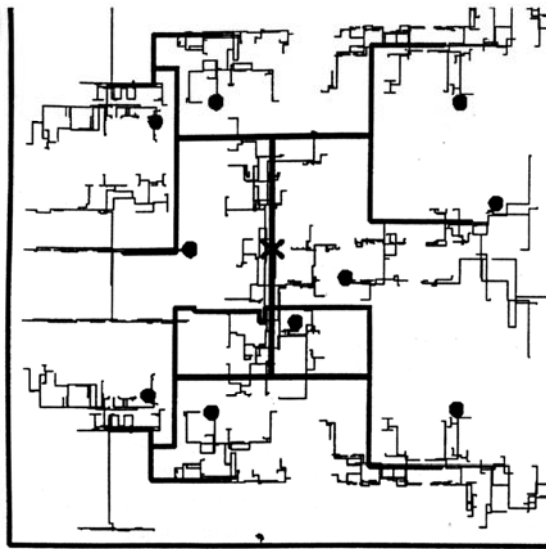
1. Ein so genanntes H-tree Netzwerk mit gematchten RC-trees und
2. ein Grid zur Verteilung der Taktes

Hier zunächst die Struktur eines einfachen H-trees.

Im Idealfall kann damit der Skew fast vollständig eliminiert werden.

Bei hohen Taktfrequenzen kann es aber mehrere Taktzyklen dauern, bis der Takt auch den letzten Ast des Baumes erreicht hat.

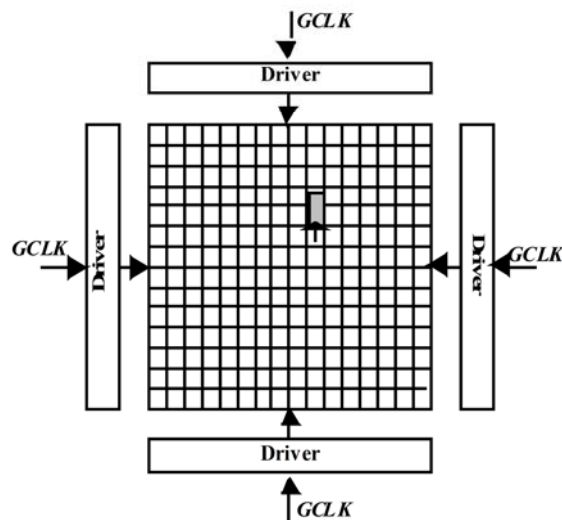
7.1. Realistischer H-tree



[Restle98]

Ein realistischer H-tree auf einem realen Chip ist deutlich komplexer aufgebaut.

7.1. Grid-System



- *Kein RC-matching*
- *große P_V*

Clock grids werden auch gerne für die letzte Stufe des Taktverteilungs-Netzwerks eingesetzt.

Damit kann die absolute Verzögerung minimiert werden, nicht aber die relative Verzögerung.

7.1. Example: DEC Alpha 21164

Taktfrequenz: 300 MHz - 9.3 Mio. Transistoren

Gesamte Taktleitungskapazität: 3.75 nF

Verlustleistung im Netzwerk: 20 W (von gesamt 50 W)

Zwei-PhasenTakt:

- **6-stufige Treiber im Chipzentrum**
- **Sekundäre Buffer treiben links und rechts im Grid in Ebenen Metal 3 and Metal 4**

Gesamte Taktleitungslänge: 58 cm!

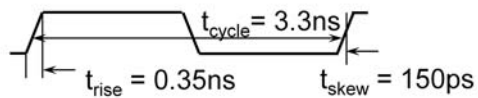
Technologiedaten:

16.5x18.1 mm die size

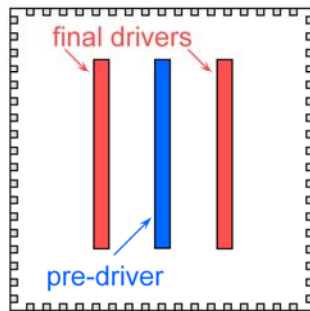
0.55 micron CMOS, 4 layer metal

Clock load accounts for 40% of the total effective capacitance of the chip

7.1. Alpha 21164: Takt



Clock waveform

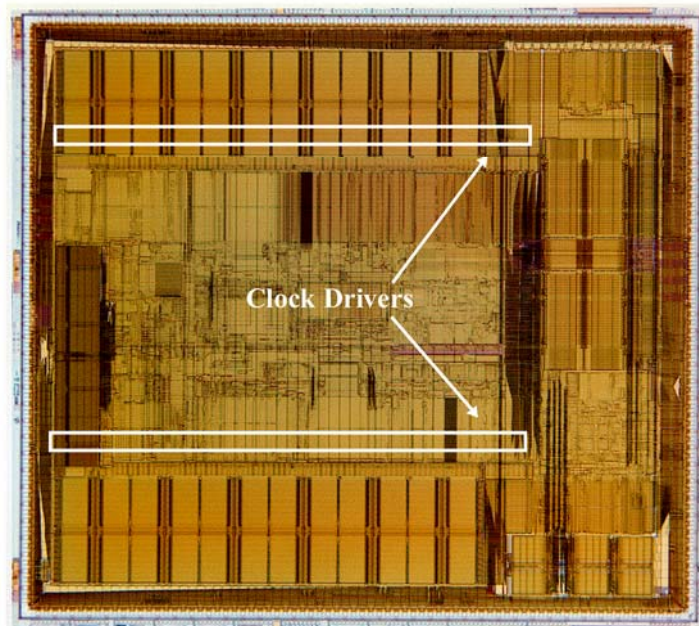


Location of clock driver on die

- 2-Phasen globaler Takt
- 2 verteilte Treiberkanäle
 - Reduzierte RC Skew
 - Bessere Wärmeverteilung
 - 3.75nF Taktleitungskapazität
 - 58 cm Taktleitungslänge
- Lokale Latch-Register
- Abhängige Takte im Cache

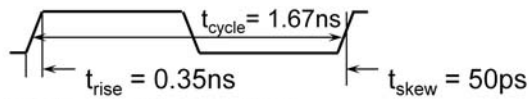
Hier die wesentlichen Daten des Taktsignals und die Treiberplatzierung auf dem Chip.

7.1. DEC Alpha 21164 (EV5)

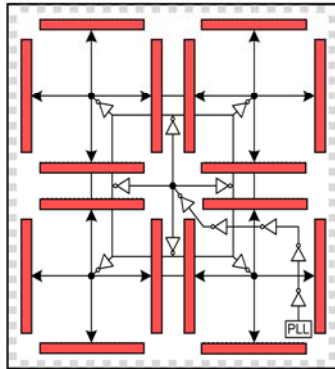


Hier ein Chipfoto des Prozessors

7.1 Alpha 21264: 600 MHz – 0.35 μm CMOS



Global clock waveform



- 2-Phasen, mit abhängigem Takt
 - 2.8 nF Taktleitungskapazität
 - 40 cm Takweg
- Abschaltbarer lokaler Takt
- Reduzierter Skew
- Bessere Wärmeableitung

Bei der nächsten Prozessorgeneration ergaben sich die hier gezeigten strukturellen Änderungen, um die erforderlichen Bedingungen einhalten zu können.

KIT
Karlsruher Institut für Technologie



167

Functions of clock in synchronous design

- 1) Acts as completion signal**
- 2) Ensures the correct ordering of events**

Truly asynchronous design

- 1) Completion is ensured by careful timing analysis**
- 2) Ordering of events is implicit in logic**

Self-timed design

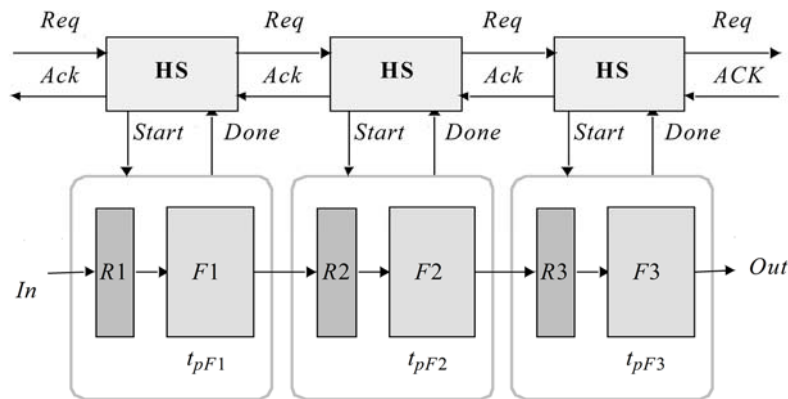
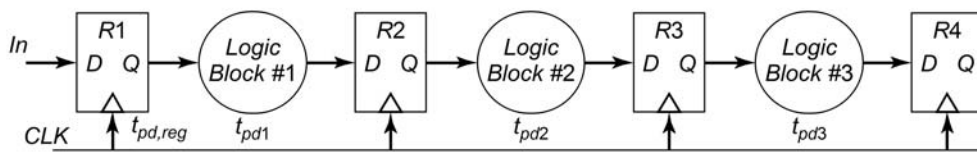
- 1) Completion ensured by completion signal**
- 2) Ordering imposed by handshaking protocol**

Wir wollen nun die unterschiedlichen Varianten von Ablaufsteuerungen betrachten und vergleichen.

Die Eigenschaften und Probleme des synchronen Designs haben wir gerade ausführlich behandelt.

Asynchrones Design bedeutet im allgemeinen, dass der gesamte betrachtete Logikblock ohne jegliche Taktsynchronisation aufgebaut ist. Fehlt also nur noch das self-timed design

7.2. Synchronous Pipelined Datapath

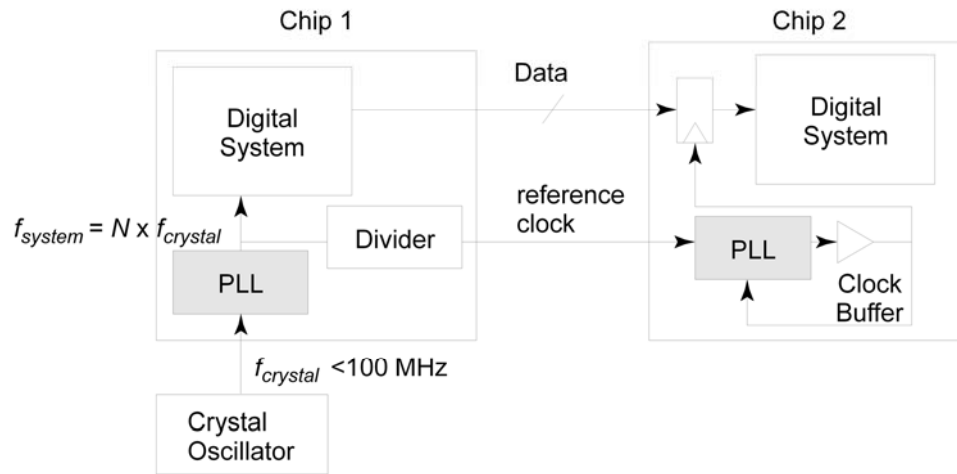


Self-Timed Pipelined Datapath

Anhand von Blockschaltbildern soll der Unterschied zwischen einem synchronen und einem "self timed" Datenweg aufgezeigt werden.

Während bei synchroner Datenweitergabe immer ein Takt eingesetzt wird, dessen Periodendauer auf maximale Verarbeitungszeiten eingestellt werden muss, fällt dieser beim "Self-Timed Pipelined Datapath" weg. die Weitergabe von einer Stufe zur nächsten erfolgt völlig asynchron und wird durch "Anfragen" und "Bestätigungen" der einzelnen Stufen gesteuert.

7.5. PLL-Based Synchronization



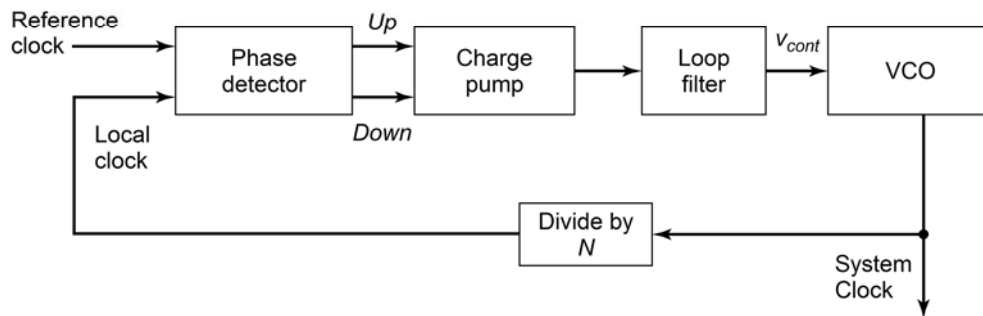
In modernen Digitalschaltkreisen erfolgt die Taktsynchronisation üblicherweise durch PLLs.

Ein einziger Quarzoszillator liefert den Grundtakt, auf den alle weiteren Taktsignale phasengenau synchronisiert werden.

Eine mögliche Arbeitsumgebung ist hier beispielhaft dargestellt.

Der Aufbau einer PLL wird im folgenden näher betrachtet.

7.5. PLL Block Diagram

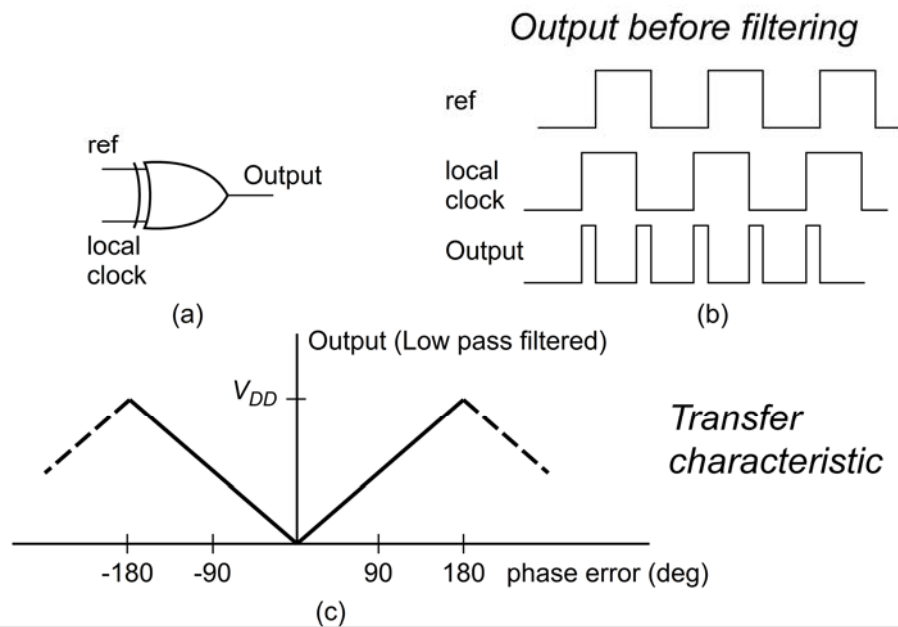


Eine PLL ist heute aus höchstintegrierten Digitalschaltungen nicht mehr wegzudenken.

Das Blockschaltbild zeigt die notwendigen Elemente, um eine PLL aufzubauen.

Im Folgenden sollen diese Blöcke näher betrachtet werden.

7.5.1 Phase Detector



172

Erich Crotti – Design Digitaler Schaltkreise

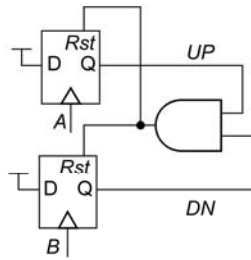
Fakultät für Elektrotechnik und Informationstechnik
Institut für Mikro- und Nanoelektronische Systeme

Das Eingangselement jeder PLL ist ein Phasendetektor.

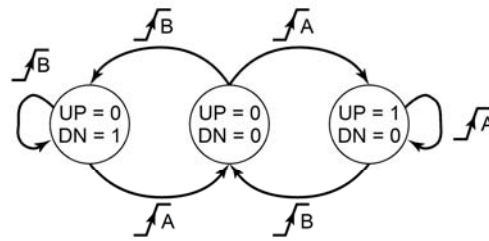
Damit wird die Phase des Referenztakts, der üblicherweise durch einen Quarzoszillator erzeugt wird, mit der Phase des erzeugten Systemtakts verglichen. Grundsätzlich werden Verfahren eingesetzt:

1. Eine Antivalenzschaltung (diese Folie)
2. Eine Schaltung aus einflankengesteuerten D-FFs. (nächste Folie)

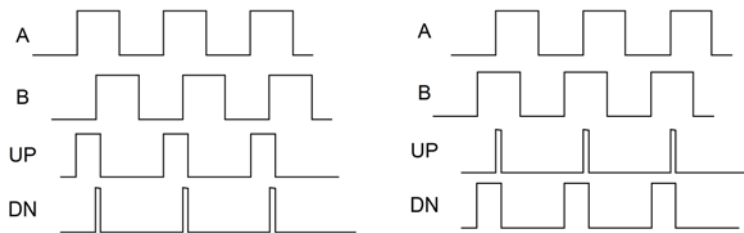
7.5.2 Phase-Frequency Detector



(a) schematic



(b) state transition diagram



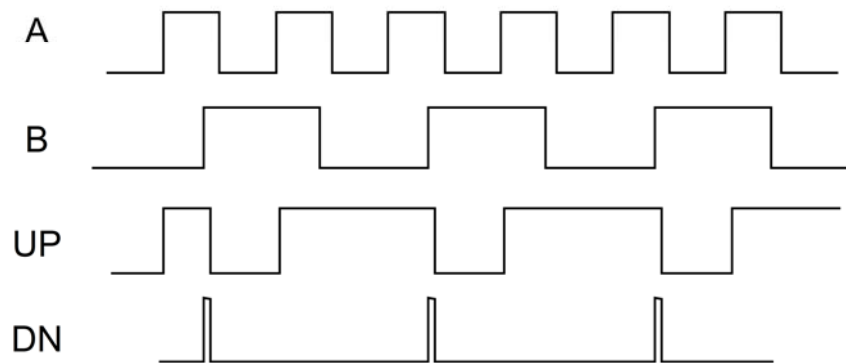
(c) Timing waveforms

Die Schaltung mit D-FFs wird auch als "Phase-Frequency Detector" bezeichnet.

Wenn beide FF-Ausgänge "1" werden, erfolgt ein Reset beider FFs.

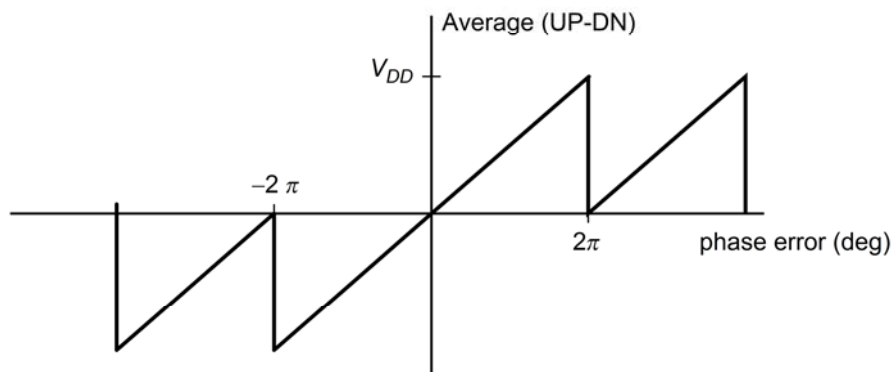
Damit wird im Gegensatz zur Antivalenzschaltung sowohl ein "UP" wie auch ein "DOWN" Signal erzeugt, mit denen dann eine Ladungspumpe angesteuert werden kann.

7.5.2 PFD Response to Frequency



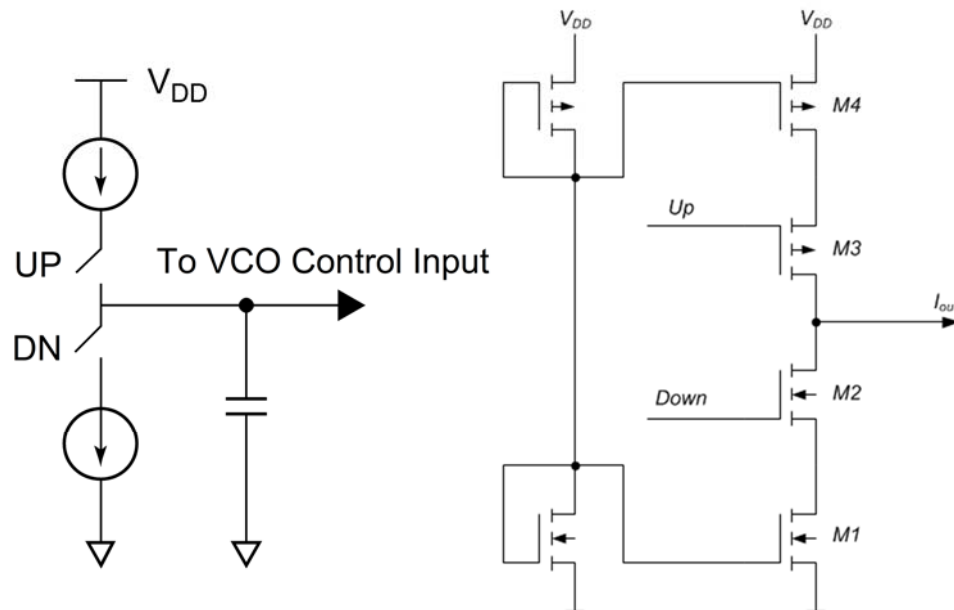
Die Antwort der PFD-Schaltung auf entsprechende Taktsignale zeigt dieses Bild.

7.5.2 PFD Phase Transfer Characteristic



Abhängig vom Fehler in den Phasen der beiden Takte bekommt man die hier gezeigte Transfercharakteristik.

7.5.3 Charge Pump

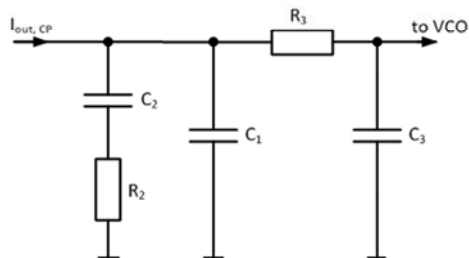


Die Ladungspumpe besteht aus zwei Stromquellen ($M1$, $M4$) und zwei Transfer-Transistoren ($M2$, $M3$).

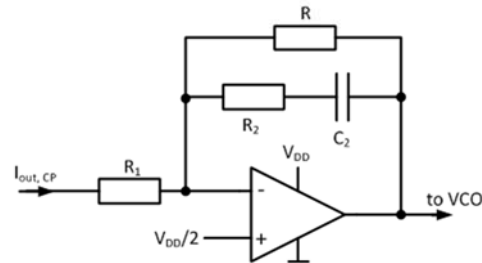
Damit kann der Kondensator am Ausgang mit einem konstanten Strom für definierte Zeiten aufgeladen oder entladen werden.

Ein mögliche schaltungstechnische Realisierung ist rechts gezeigt.

7.5.4 Loop Filter



Passive Loop Filter



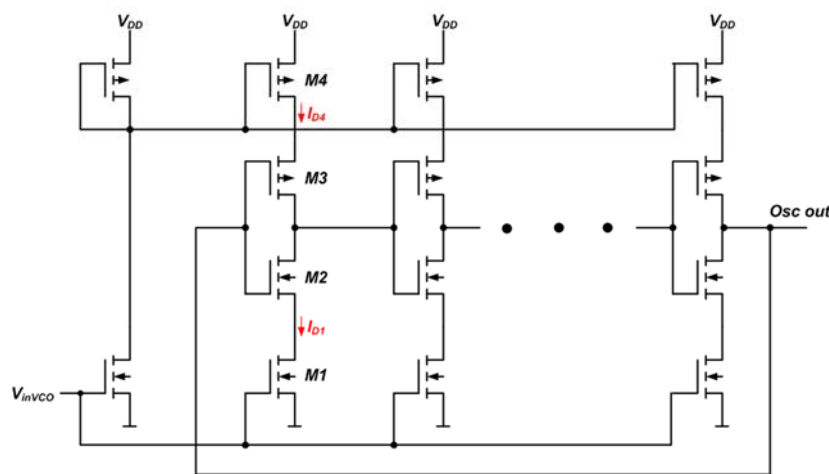
Active Loop Filter

Das Ausgangssignal der Ladungspumpe muss nun noch über eine Tiefpassschaltung gefiltert werden, damit am VCO (spannungsgesteuerter Oszillator) eine "Gleichspannung" angelegt werden kann.

Hier 2 Beispiele solcher Filter:

Links ein passives Filter und rechts eine entsprechende aktive Filterschaltung.

7.5.5 Voltage Controlled Oscillator



$$f_{osc} = \frac{I_D}{N \cdot C_a \cdot V_{DD}} \quad \text{mit } N \geq 5 \text{ und ungerade}$$

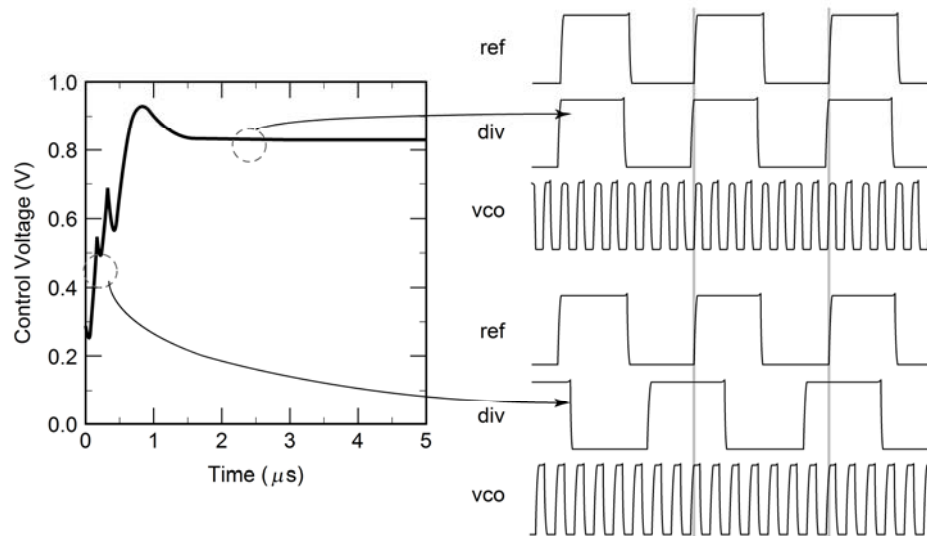
Der Eigentliche spannungsgesteuerte Oszillator kann ebenfalls auf verschiedene Art und Weise aufgebaut werden.

In rein digitalen Schaltkreisen wird gerne die hier gezeigte Schaltung eingesetzt. Sie wird in der Literatur als "Current starved VCO" bezeichnet.

Die Schaltung ist ein stromgesteuerter Ringoszillator bei dem über die "analoge" Stromsteuerung die Zeitkonstanten für die Umladung der jeweiligen Lastkapazitäten an den Ausgängen der einzelnen Stufen gesteuert werden können.

Liegt eine Spannung kleiner U_{thn} oder größer $V_{DD} - |U_{thp}|$ an, wird die Ausgangsfrequenz zu 0, da dann ein Ent- bzw. Aufladen nicht mehr erfolgen kann.

7.5.4 PLL Simulation



Beim Einschalten der Versorgungsspannung muss der VCO erst einmal "einschwingen", bevor die gewünschte Frequenz erreicht wird. Abhängig von den Zeitkonstanten im Filter und den Strömen im VCO ergeben sich die Einschwingzeiten, bis der VCO seine korrekte Ausgangsfrequenz erreicht hat.