

# 16 CMOS-Schaltungstechnik

Das folgende Kapitel behandelt CMOS-Schaltungstechnik für die Realisierung von logischen Schaltungen. CMOS-Schaltungstechnik verwendet gemeinsam nMOS- und pMOS-Transistoren. Die Bezeichnung CMOS (Complementary MOS) ist daraus abgeleitet, daß diese Transistoren zueinander komplementäre Schalter sind: positive Schalter und negative Schalter (vgl. Tabelle in Bild 16.2a). Da pMOS- und nMOS-Transistoren ein unterschiedliches "Substrat" benötigen, werden als "Unterlage" für "Cluster" von Transistoren eines oder beider Typen Wannen hergestellt. Wir können demnach 3 Arten von CMOS-Technologien unterscheiden (vgl. Schnitte in Bild 16.3):

- n-Wannen-Prozesse mit n-Wannen (*n well*) für pMOS-Transistoren (während sich die nMOS-Transistoren auf dem p-Substrat selbst befinden)
- p-Wannen-Prozesse mit p-Wannen (*p well*) für nMOS-Transistoren (während sich die pMOS-Transistoren auf dem n-Substrat selbst befinden)
- 2-Wannen (*twin well, twin tub*) - Technologie

## 16.1 Gliederung der CMOS-Schaltungstechniken

Es gibt eine große Anzahl von CMOS-Schaltungstechniken (vgl. Übersicht in Bild 16.4). Die bekannteste davon ist die statische CMOS-Schaltungstechnik, deren Entwurfsverfahren in einem späteren Abschnitt behandelt werden.

16.1 Gliederung der CMOS-Schaltungstechniken .....	321
16.1.1 Statische Logiken .....	322
16.1.2 Dynamische Logiken.....	325
16.1.3 P-E-Logiken .....	325
16.1.4 Exotische CMOS-Schaltungstechniken .....	326
16.2 Statische CMOS-Schaltungstechnik.....	326
16.2.1 Entwurfsverfahren für statische CMOS-Schaltungen .....	327
16.2.1.4 Ermittlung des komplementären Transmissionsnetzes .....	334
16.3 Dynamische CMOS-Schaltungen (Clocked-CMOS).....	334
16.4 Precharged CMOS-Schaltungstechnik (PE-Logiken) .....	335
16.4.1 Precharged with gated input .....	335
16.4.2 P-E-Logik .....	337
16.4.3 Das Charge-Sharing-Problem.....	339
16.4.4 DOMINO-CMOS-Schaltungstechnik .....	340
16.5 USP-CMOS Schaltungstechnik.....	342
16.6 Literatur .....	342

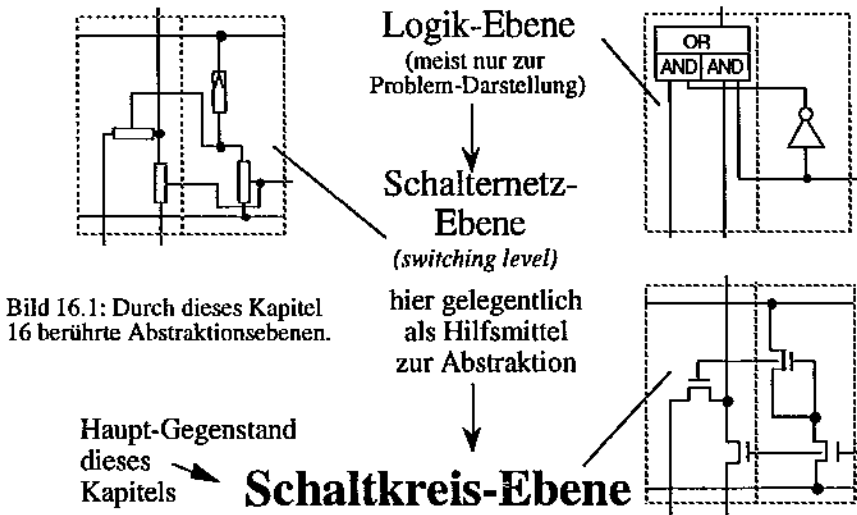


Bild 16.1: Durch dieses Kapitel 16 berührte Abstraktionsebenen.

Die Gliederung von CMOS-Schaltungstechniken erscheint systematischer und abgerundeter, wenn nMOS- und pMOS-Schaltungen als deren Spezialfälle miteingefasst werden. Wir können vier Klassen von MOS-Schaltungstechniken unterscheiden (vgl. Bild 16.4): statische Logik, dynamische Logik, P-E-Logiken und sonstige Logiken, deren Schaltungstechnik wir als "exotisch" bezeichnen, da diese nicht naheliegend sind und keine bei uns gängige Verbreitung haben. P-E steht für "Precharge und Evaluate", was später erklärt wird. Wir machen zur Veranschaulichung der jeweiligen Schaltungstechnik von einem vereinfachten Schema Gebrauch, welches das jeweilige logische Netz als p-Netz oder n-Netz durch ein p-Kästchen bzw. n-Kästchen abstrahiert (vgl. Bild 16.4). Ein n-Netz ist ein Transmissionsnetz mit nMOS-Transistoren (als positive Schalter), während ein p-Netz mit pMOS-Transistoren aufgebaut ist (als negative Schalter, vgl. auch Bild 16.5).

### 16.1.1 Statische Logiken

Statische Logiken werden deshalb so bezeichnet, weil nach dem Abklingen von Einschwingvorgängen ein logischer Wert und sein Träger (eine elektrische Spannung) beliebig lange stabil bleibt und nicht von "vergeblichen Speicher-Effekten", wie etwa durch Kapazitäten, abhängig

	Transistortyp	
	nMOS	pMOS
Schaltsymbol	in —	in —
Modell in der Switching-Ebene	in —  positiver Schalter	in —  negativer Schalter
Kommentar	leitend bei in = 1	leitend bei in = 0

Bild 16.2: CMOS-Schalter-Typen.

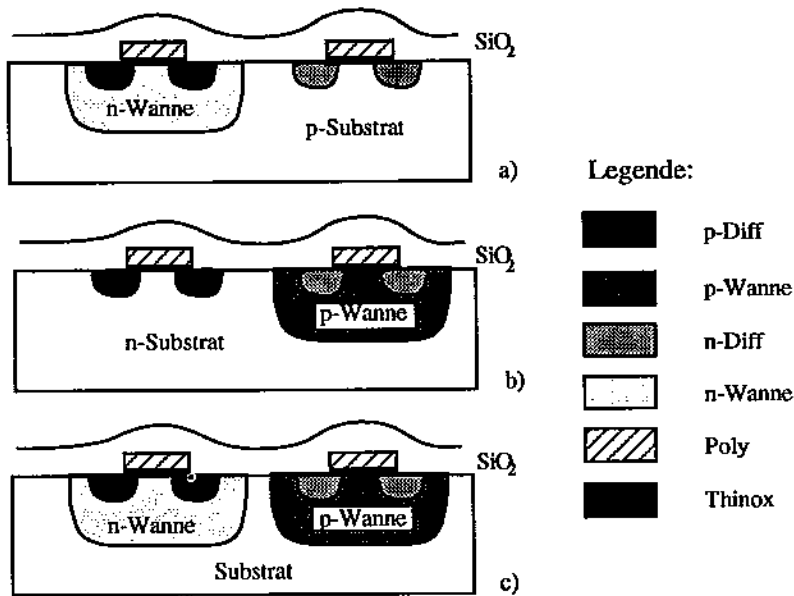


Bild 16.3: Drei Varianten von CMOS-Technologie: a) n-Wannen- (*n well* oder *n tub*) Technologie, b) p-Wannen- (*p well*) Technologie, c) Doppelwannen- (*twin tub*) Technologie.

ist. Statische Logiken benötigen daher keinerlei Wiederauffrischungsprozeduren. Alle in Bild 16.4 gezeigten statischen Logiken, mit Ausnahme komplementärer CMOS-Logik, sind insofern "Verhältnis-Logiken", als hier ein minimales Pull-up/Pull-down-Verhältnis eingehalten werden muß (dies wurde unter "electrical rules" behandelt). Bei komplementären CMOS-Schaltungen hingegen entfällt diese Regel, da diese im Ruhezustand stromlos sind.

**Statische MOS-Logik,** insbesondere nMOS-Logik wurde bereits behandelt. Ein separater "Last-Transistor" dient als flächensparender Ersatz für einen Last-Widerstand. Bei nMOS-Logik ist dies bei heute üblicher Technologie ein selbstleitender Transistor ("depletion load"). Bei heute nicht mehr üblicher reiner pMOS-Technologie wurde für diesen Transistor meist eine separate negative Betriebsspannung benötigt (vgl. "bias" in Bild 16.4), da pMOS-Transistoren gegenüber nMOS-Transistoren einen um ca. den Faktor 2,5 bis 3 höheren Schichtwiderstand haben. pMOS-Schaltungen tendieren also dazu, erheblich langsamer zu sein als nMOS-Schaltungen.

**Statische Pseudo-MOS-Logik.** Stehen in einer Technologie auf dem gleichen Chip beide Sorten von Transistoren zur Verfügung (pMOS und nMOS), so kann der zum logischen Netz jeweils duale Transistor als Last verwendet werden (Bild 16.4). Interessant ist, daß hiermit Verhältnislogik in einer CMOS-Technologie realisiert werden kann (ohne daß ein selbstleitender

Schaltungstechnik		Technologie		
		nMOS	pMOS	CMOS
statische Logiken	statische MOS-Logik (static pMOS, static nMOS)			
	statische Pseudo-MOS-Logik			
	komplementäre CMOS-Logik ('static CMOS')			
dynamische Logiken	getaktete CMOS-Logik (clocked CMOS)			
	dynamische Logik			
P-E-Logiken	nMOS-artige P-E-Logik (‘P-E’ steht für: precharge - evaluate)			
	CMOS-P-E-Logik			
	Domino-Logik			
	NORA-Logik			
exotisch	Transferrgate-intensive Logik			

Bild 16.4: Übersicht über MOS-Schaltungstechniken: NMOS und CMOS.



der nMOS-Transistor verfügbar ist) mit erheblicher Flächensparnis, da bei der Realisierung eines Gatters zu einem Transmissionsnetz kein zweites duales T-Netz notwendig ist.

**Komplementäre CMOS-Logik.** Diese Bezeichnung ist keine Technologie insofern, als das Attribut "komplementär" die spezielle Schaltungstechnik kennzeichnet (das direkte Zusammenspiel zweier zueinander dualer Logik-Netze als verhältnissfreie statische Schaltung: Last-Transistoren entfallen), wohingegen das "C" in CMOS lediglich auf die Zugehörigkeit zur CMOS-Technologie-Gruppe im allgemeinen hinweist.

MOS-Technologien sind derzeit noch die wichtigsten Technologien für komplexe digitale Subsysteme. Eine hohe Integrationsdichte wird erreicht bei geringer Leistungsaufnahme. Höhere Schaltgeschwindigkeiten werden bisher und derzeit noch durch bipolare Technologien erreicht, auf Kosten relativ niedriger Integrationsdichte und hoher Leistungsaufnahme.

### 16.1.2 Dynamische Logiken

Zur dynamischen CMOS-Logik rechnen wir solche Schaltungs-Techniken, die einen Transfer-Transistor oder ein Transfer-Gatter (vgl. Tabelle in Bild 16.2) zur Taktung von einer Stufe in die nächste Stufe haben, oder diesen äquivalente Schaltungstechniken. Diese Logiken tragen die Bezeichnung "dynamisch" deshalb, weil bei gesperrtem Transfer-Transistor bzw. Transfer-Gatter nur eine am Eingang der Zielstufe liegende parasitäre Kapazität den Logik-Wert kurzfristig zwischenspeichert. Es sei an die früher in diesem Text eingeführte dynamische nMOS-Logik erinnert. Die getaktete CMOS-Logik ist (vgl. Bild 16.4) der dynamischen CMOS-Logik äquivalent. Es läßt sich leicht zeigen, daß das zwischen p-Netz und n-Netz dazwischengeschaltete duale Transistorpaar bezüglich der Gesamtfunktion der Anwendung eines Transfer-Gate äquivalent ist.

### 16.1.3 P-E-Logiken

Bei P-E-Schaltungstechniken wird die Gatterfunktion zeitlich unterteilt in eine Precharge-Phase (P-Phase), bei welcher die parasitäre Ausgangs-Kapazität vorsorglich schon einmal aufgeladen wird, und in eine *evaluation phase* (E-Phase), in welcher das logische Netz den tatsächlichen Ausgangswert "berechnet". Sollte die Evaluate-Phase einen vom Precharge-Wert abweichenden Wert zeigen, so wird dieser durch das logische Netz korrigiert.

Die Grundidee der P-E-Techniken dient einer Verbesserung der Schaltungsgeschwindigkeit durch Vermeidung der dynamischen Unsymmetrie der Verhältnis-Logiken (wie etwa aufgrund des Mindest-pull-up/pull-down-Verhältnis bei statischer nMOS-Logik). Da (beim nMOS-Beispiel) im Falle des Umschaltens auf "1" am Ausgang die hochohmige "pull-up"-Last nur sehr langsam reagiert, wird während einer taktungsbedingten Wartezeit der Ausgang im Voraus auf "1" gebracht, da ein evtl. nötiges korrigierendes Herunterziehen in der E-Phase wegen der Niederohmigkeit des Logik-Netzes sehr schnell geht. Zu diesem Zweck wird der Last-Transistor  $T_e$  durch einen Takt während der P-Phase eingeschaltet (vgl. Fp-Signal in Bild 16.4). Gleichzeitig wird zur Vermeidung eines Widerspruches das Logik-Netz über einen durch  $F_e$  gesteuerten Schalter  $T_e$  unterbrochen. Während der E-Phase hingegen ist  $T_l$  ausgeschaltet und  $T_e$  eingeschaltet, sodaß die Pull-down-Funktion des logischen Netzes und nur diese aktiv ist.

MOS-artige P-E-Logik läßt sich durch Einführung des P-E-Verfahrens unmittelbar als statische MOS-Logik ableiten. Wird jedoch für einen der Transistoren  $T_e$  und  $T_l$  ein vom Netztyp ab-

weichender Transistortyp verwendet, so spricht man von "Precharge-Pseudo"-nMOS bzw. -pMOS-Logik. Eine weitere Vertiefung der PE-Logiken erfolgt in späteren Abschnitten an Hand von Beispielen. Es sei hier nur noch die Übersicht über Strukturmerkmale vervollständigt:

- CMOS-P-E-Logik verwendet alternierende Ebenen aus precharged Pseudo-nMOS-Stufen und precharged Pseudo-pMOS-Stufen (vgl. Bild 16.4).
- Domino-CMOS-Logik ist eine Erweiterung von CMOS-PE-Logik derart, daß nach einem Stufenpaar ("CMOS-n-type-PE in Bild 16.4) jeweils ein Inverter angehängt wird.
- NORA-CMOS-Logik ("No Race") ist eine komplizierte Kombination zweier zueinander komplementärer Domino-Techniken, die später behandelt wird.

### 16.1.4 Exotische CMOS-Schaltungstechniken

Zu den sonstigen CMOS-Schaltungstechniken gehören die Transfer-Gate-intensive Logik, die intensiven Gebrauch macht von Multiplexer-ähnlich zusammengeschalteten Transfer-Gattern. Auf diese Schaltungstechnik wird in einem späteren Kapitel ausgiebig eingegangen.

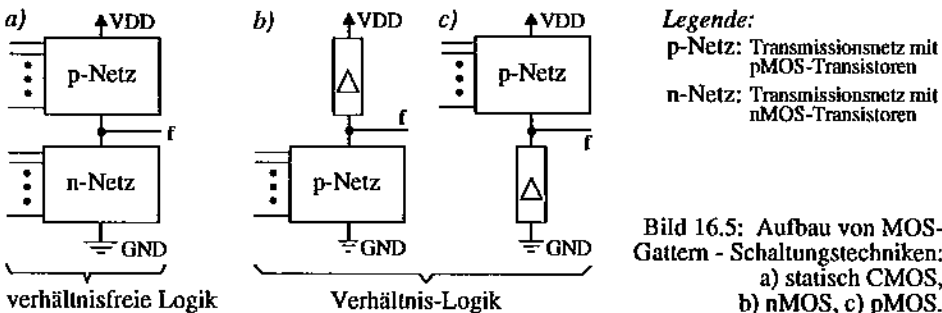


Bild 16.5: Aufbau von MOS-Gattern - Schaltungstechniken: a) statisch CMOS, b) nMOS, c) pMOS.

Letztere Schaltungstechniken versuchen die Realisierung sehr schneller CMOS-Schaltungen in Anlehnung an Schaltungstechniken für sehr schnelle Bipolarschaltungen.

### 16.2 Statische CMOS-Schaltungstechnik

Dieser Abschnitt behandelt die statische CMOS-Schaltungstechnik. Ihr wesentlicher Vorteil gegenüber anderen MOS-Schaltungstechniken liegt im geringen Stromverbrauch beim statischen Zustand. Uhren-Schaltungen, die Jahre mit einer winzigen Knopf-Batterie auskommen, sind derzeit ohne statische CMOS-Schaltungstechnik undenkbar. Ein statisches CMOS-Gatter besteht im Gegensatz zu nMOS oder pMOS aus zwei Durchlaßnetzen, einem p-Netz und einem n-Netz (Bild 16.5).

Das p-Netz ist dafür zuständig, daß für eine logische "1" als Funktionswert der Ausgang des Gatters nach oben gezogen wird ("pull-up" nach VDD). Es ersetzt somit den pull-up-Transistor aus der nMOS-Technologie. Das n-Netz hingegen ist dafür maßgebend, daß für eine logische

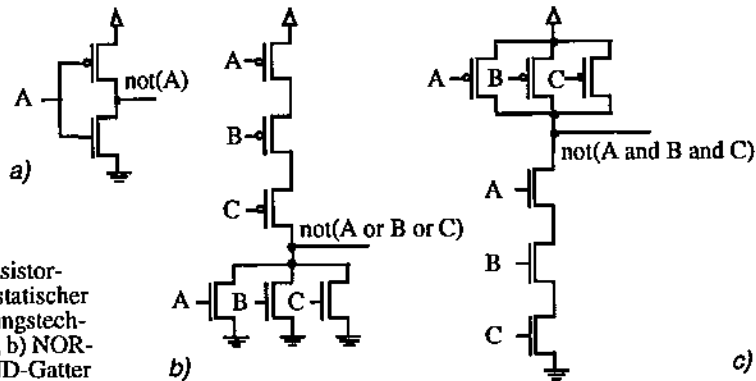


Bild 16.6: Transistor-schaltungen in statischer CMOS-Schaltungstechnik: a) Inverter, b) NOR-Gatter, c) NAND-Gatter

“0” als Funktionswert der Ausgang auf Null heruntergezogen wird (“pull-down” nach GND). Deshalb muß beim Entwurf beachtet werden, daß beide Netze streng komplementär (engl. *complementary*, daher auch die Bezeichnung CMOS) zueinander entworfen werden. Es muß nämlich zum einen sichergestellt werden, daß niemals beide Netze gleichzeitig durchlässig sind, da sonst am Ausgang des Gatters ein Wertekonflikt entstehen würde. Zum anderen darf es niemals vorkommen, daß beide Netze sperren. Dies würde bedeuten, daß der Ausgang hochohmig (*floating*) wird.

Aus der Tatsache, daß zu einem bestimmten Zeitpunkt stets nur ein Netz durchgeschaltet ist, folgt, daß ein Widerstandsverhältnis nicht beachtet werden muß. Deshalb wird die CMOS-Schaltungstechnik auch als verhältnisfrei (*ratio-less*) bezeichnet, im Gegensatz zur statischen nMOS- und pMOS-Schaltungstechnik (Verhältnislogik), für die zur Erzielung akzeptabler Spannungswerte für “0” und “1” ein bestimmtes pull-up/pull-down Verhältnis  $k$  eingehalten werden muß (vgl. Kapitel “electrical rules”). In CMOS findet ein Stromfluß immer nur kurzzeitig beim Umladen der Kapazität statt, d.h. dann wenn durch Änderung der Eingänge der Funktionswert wechselt, was erst bei einer “hochaktiven” Schaltung von Bedeutung ist. Ein Vorteil der statischen CMOS-Technik ist dadurch der “volle Spannungsschub” (*full swing*) zwischen  $V_{SS}$  (Masse) und  $V_{DD}$  beim Wechsel zwischen “0” und “1”. Im Gegensatz zur Verhältnislogik ist hier keiner der Spannungswerte durch Spannungsteilung beeinträchtigt. Wegen des *full swing* ist die Kompatibilität mit anderen Schaltungstechniken gegeben. Außerdem wird dadurch die Schaltung unempfindlich gegen starke Schwankungen der Versorgungsspannung.

### 16.2.1 Entwurfsverfahren für statische CMOS-Schaltungen

Der wichtigste Gesichtspunkt beim Entwurf statischer CMOS-Schaltungen besteht in der Gewährleistung der Komplementarität bei den beiden Durchlaßnetzen (p-Netz und n-Netz). Im folgenden werden nun drei Entwurfsverfahren für CMOS-Schaltungen (Bild 16.6) anhand von Beispielen vorgestellt:

1.a). Ableitung aus der kanonischen Form (durch Serien/Parallel-Synthese)

1.b). Variante zur Ableitung aus der kanonischen Form

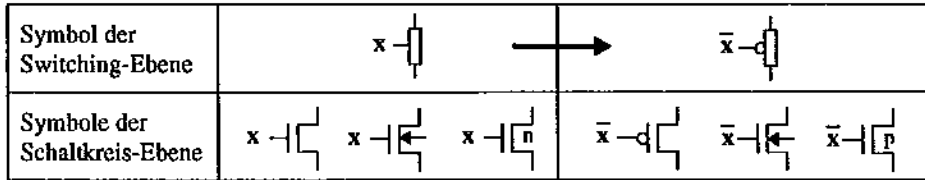


Bild 16.7: Äquivalente Symbole für abstrakte Schalter und pMOS- und nMOS-Transistoren

2). Sutton's Verfahren

3). Ableitung aus dem KV-Diagramm

Dabei muß stets beachtet werden, daß ein nMOS-enhancement-Transistor bei einer logischen "1" durchschaltet, d.h. er verhält sich wie ein positiver Schalter. Ein pMOS-enhancement Transistor verhält sich genau umgekehrt. Er wird bei einer Gatespannung um 0 V herum leitend, was der Funktion eines negativen Schalters entspricht (Bild 16.7).

### 16.2.1.1 Ableitung aus der kanonischen Form

Dieser Abschnitt behandelt die Ableitung statischer CMOS-Gatter aus der kanonischen Form: durch Serien/Parallel-Synthese, sowie eine Variante dieses Verfahrens.

#### 16.2.1.1.1 Serien/Parallel-Synthese aus zwei Gleichungen

Das Verfahren nach der kanonischen Form der Schaltfunktion sei erst einmal in geschlossener Form dargestellt. Daran schließt sich eine Veranschaulichung mit Hilfe eines Beispiels an.

Der Entwurf eines statischen CMOS-Gatters durch Ableitung der kanonischen Funktion erfolgt nach folgender Prozedur:

#### 1. Ermittlung des Pullup-Netzes (p-Netz):

abc	M(a, b, c)
000	0
001	0
010	0
011	1
110	0
101	1
110	1
111	1

a) man wandle die als Problemformulierung gegebene Schaltfunktion in die kanonische Form um (disjunktive Form: Disjunktion von Konjunktionen)

b) durch Serien/Parallel-Synthese erstelle man danach ein Netz aus positiven Schaltern

c) man ersetze alle positiven Schalter durch pMOS-Transistoren (negative Schalter)

d) negiere alle Eingangsvariablen des p-Netzes (Kompensation des Fehlers in Schritt c)

Nach einiger Übung kann Schritt b und c zusammengefaßt werden: man zeichnet bei Schritt b sofort die Transistorsymbole aus Schritt c ein.

Bild 16.8: Majoritätsfunktion M(a,b,c)





## 2. Ermittlung des Pull-down-Netztes (n-Netz):

- man leitet aus der Schaltfunktion  $f$  die negierte Funktion  $\text{not}(f)$  ab.
- überführe diese Schaltfunktion  $\text{not}(f)$  durch de-Morgan'sche Umformungen in die kanonische Form (vgl. 1a), bis nur noch Einzelvariablen negiert vorkommen.
- durch Serien/Parallel-Synthese erstelle man aus  $\text{not}(f)$  ein n-Netzwerk. Da n-Transistoren positive Schalter sind, erhält man sofort das richtige Netz (im Gegensatz zu 1c und 1d).

Wir haben dann zwei zueinander duale Boole'sche Funktionen, eine für jedes der beiden Netze. Meist lassen sich die Netze nachträglich noch minimieren, sodaß die Zahl der Transistoren vermindert wird. Als Beispiel sei für die Veranschaulichung des Verfahrens die Majoritätsfunktion gegeben (Bild 16.8).

Man erstellt zuerst die disjunktive Form der Funktion (vgl. 1a), d.h. die Konjunktionen der "Eins"-Stellen der Funktion werden disjunktiv verknüpft. Die Majoritäts-Funktion  $M$  ist wie folgt definiert:

$$M(a,b,c) = ab \vee ac \vee bc \vee abc = ab \vee ac \vee bc$$

Für das zu erstellende p-Netz setzt man nun die Funktion  $M$  direkt in ein Schalternetz um (oberer Teil von Bild Bild 16.9 a). Da in pMOS negative Schalter realisiert werden, müssen noch die Eingänge invertiert werden (oberer Teil von Bild 16.9 b).

Für das n-Netz ermittelt man die negierte Form  $\text{not}(M)$  der Funktion  $M$  und verwendet diese zum Erstellen des n-Schalternetzes:

$$\overline{M(a,b,c)} = \overline{ab \vee ac \vee bc} = \overline{ab} \overline{ac} \overline{bc} = (\overline{a} \vee \overline{b}) (\overline{a} \vee \overline{c}) (\overline{b} \vee \overline{c})$$

Bild 16.9 (unterer Teil) zeigt das Ergebnis der Parallel/Serien-Synthese des n-Netztes. Durch anschließenden Austausch der Schalter gegen die entsprechenden MOS-Transistoren gewinnt man das Transistornetz (Bild 16.10). Nachteil bei Beispiel in Bild 16.10: Alle Eingänge sind negiert, d.h. es werden noch 6 zusätzliche Transistoren gebraucht. (Abhilfe siehe Beispiel bei Verfahren 2).

### 16.2.1.1.2 Serien/Parallel-Synthese aus einer einzigen Gleichung

Eine Variante des Verfahrens kommt mit einer einzigen Boole'schen Gleichung aus (der für das pull-up-Netz). Diese wird wie gewohnt für eine Serien/Parallel-Synthese angewandt durch etwas tiefergehendes Mitdenken als bei Variante 1a).

### 16.2.1.2 Das Verfahren nach Sutton

Das Verfahren macht von der verallgemeinerten (wiederholt angewandten) de Morgan'schen Regel Gebrauch. Diese Regel beschreibt eine Äquivalenz-Transformation wie bei n-stufigen disjunktiv/konjunktiven Schaltfunktionen eine Negation des Ausgangs durch Negation aller Eingänge eliminiert werden kann. Dieser Satz ist durch Bild 16.11 veranschaulicht.

- Man entferne den Negationsknopf am Ausgang (Bild Bild 16.11 a) und negiere gleichzeitig alle Eingangswerte (Ergebnis: Bild 16.11 b). Gleichzeitig müssen alle Konjunktionen durch

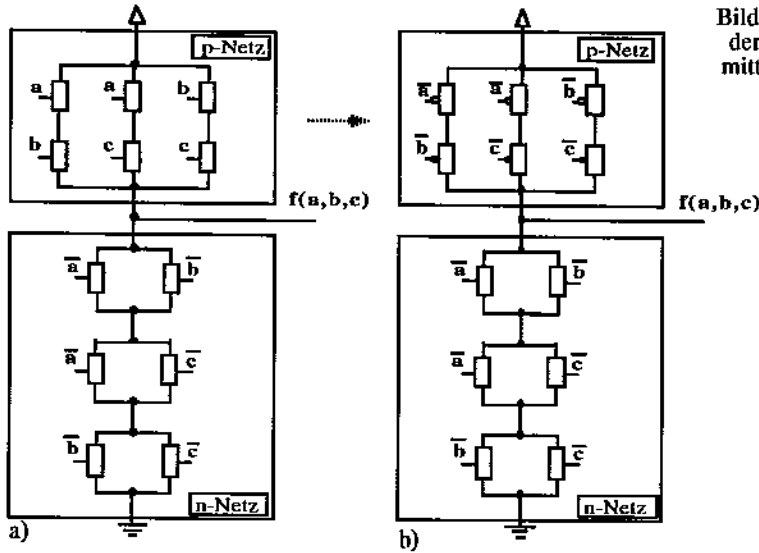


Bild 16.9: Schaltnetz der Funktion  $M$ , ermittelt aus einer disjunktiven Form.

Disjunktionen (AND durch OR), sowie alle Disjunktionen durch Konjunktionen (OR durch AND) ersetzt werden (Ergebnis in Bild 16.11 b).

b) In diesem Verfahren schreibt man die Funktion zunächst als Logik-Diagramm (Bild 16.12 a). Dabei sollte darauf geachtet werden, daß entweder der Ausgang eine Negation besitzt: in diesem Fall kann das pull-down-Netz daraus abgeleitet werden. Ist dies nicht der Fall, so erreicht man dies z.B. am Ausgang durch die Anwendung der Tautologie-Regel (Bild 16.12 b).

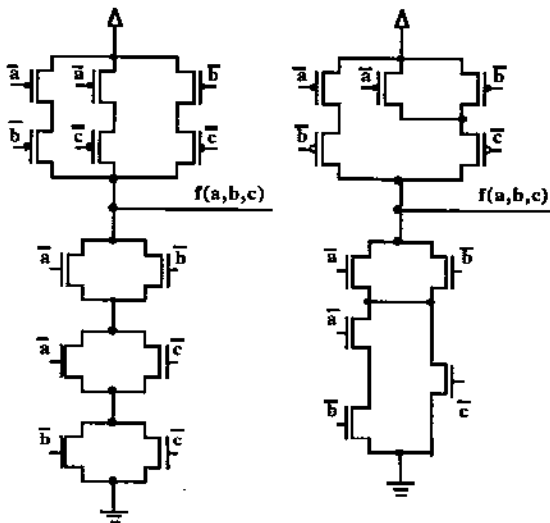


Bild 16.10: Transistorschaltnetz der Funktion  $M$ , ermittelt aus der Disjunktiven Normalform.

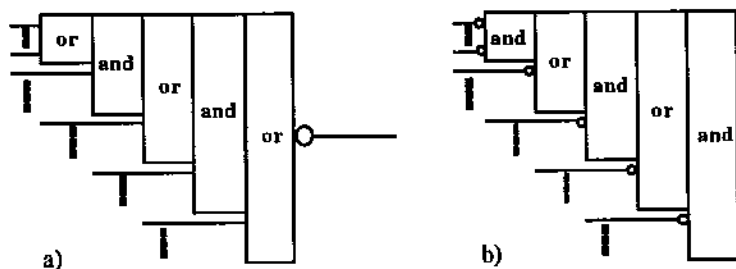


Bild 16.11: Transformation einer Funktion durch mehrfache Anwendung der De-Morgan-Regel. a) Negation am Ausgang b) Negationen am Eingang.

Im zweiten Schritt ermittelt man mit Hilfe der "De-Morgan-Regel" eine zweite Form der Funktion, indem man entweder die Negation am Ausgang durch Umformung an die Eingänge bringt (Bild 16.12 c) oder entsprechend umgekehrt, daß man die Negationen an den Eingängen zum Ausgang bringt. Hieraus kann dann das pull-up-Netz abgeleitet werden.

c) Im dritten Schritt verwendet man nun das Diagramm mit der Negation am Ausgang für die Erstellung des n-Netzes, indem man die Gatter direkt nach der Serien/Parallel Methode umsetzt, ohne jedoch die Negation am Gatterausgang zu beachten.

d) Für das p-Netz wird dann im vierten Schritt das Diagramm mit den Negationen an den Eingängen (ohne sie zu beachten, da negative Schalter!) betrachtet und daraus das p-Netz direkt nach der Serien/Parallel-Methode ermittelt.

In dem vorgegebenen Beispiel muß die Schaltung noch ausgangsseitig invertiert werden, da am Anfang die Tautologie-Regel angewandt wurde (Bild 16.13 a). Danach kann die Schaltung durch entsprechende Transformation noch minimiert werden (Bild 16.13 b).

Wir haben also für die Majoritätsfunktion eine bessere Lösung gefunden. Während die Lösung in Bild 16.10 für jede Eingangsvariable einen Inverter erfordert (die 3 Variablen also 6 Transistoren für 3 Inverter), benötigen wir hier (Bild 16.13) nur einen einzigen Inverter (2 Transistoren), nämlich am Ausgang.

### 16.2.1.3 Die KV-Diagramm-Methode

Die KV-Diagramm-Methode ist der Methode nach der kanonischen Form sehr ähnlich. Jedoch überwiegt hier die graphische Darstellung des KV-Diagramms. Die Schaltfunktion wird jedoch zuerst als KV-Diagramm dargestellt. Zunächst wird die Methode algorithmisch beschrieben.

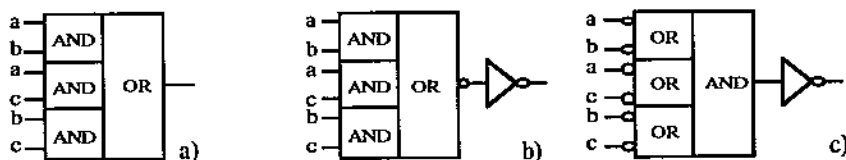


Bild 16.12: Logik-Diagramme der Majoritätsfunktion M: a) direkte Umsetzung, b) nach Anwendung der "Tautologie-Regel" (2 Negationen heben einander auf), c) nach Anwendung der "De-Morgan-Regel" (für M Inverter am Ausgang vorausgesetzt).

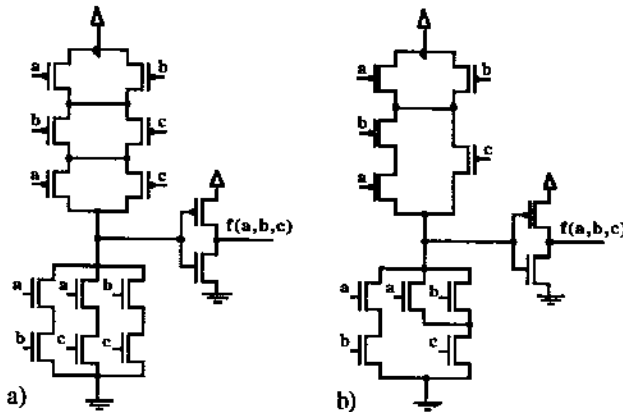


Bild 16.13: Transistorschaltung der Funktion M, ermittelt nach dem Verfahren von Sutton.

- 1.) Erstellen des KV-Diagramms: Erstelle KV-Diagramm (ohne "don't care") für die zu realisierende Boole'sche Funktion.
- 2.) Entwurf des pull-up-Netzes aus dem KV-Diagramm:
  - a.) Verwende alle Einsen der Funktion zum Entwurf des p-Netzes und bilde daraus eine minimale Funktion.
  - b.) Bilde aus der minimalen Funktion das p-Netz durch Parallel/Serien-Synthese, indem eingesetzt wird:
    - für jede nicht negierte Variable ein negativer Schalter mit negiertem Eingang
    - für jede negierte Variable ein negativer Schalter mit nicht negiertem Eingang.
- 3.) Entwurf des pull-down-Netzes aus dem KV-Diagramm:
  - a.) Verwende alle Nullen zum Entwurf des n-Netzes; trage diese in einem Hilfs-KV-Diagramm als Einsstellen ein und bilde daraus eine minimale Funktion h
  - b.) Bilde daraus das n-Netz durch Parallel/Serien-Synthese, indem eingesetzt wird:
    - für jede nicht negierte Variable von h ein positiver Schalter mit nicht negiertem Eingang

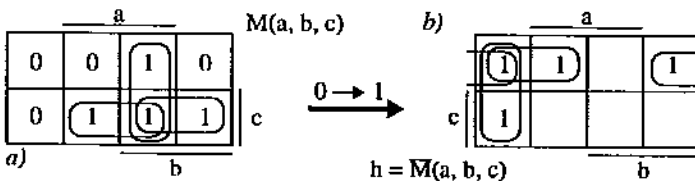


Bild 16.14: KV-Diagramme zur Majoritätsfunktion: a) zur Ermittlung der Funktion M; b) Hilfsdiagramm zur Ermittlung der Funktion  $h = \bar{M}$ .

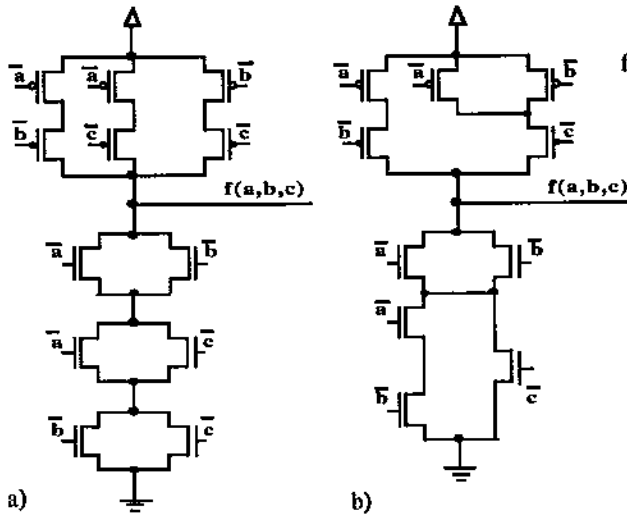


Bild 16.15: Transistor-schaltung der Majoritätsfunktion, ermittelt aus der KV-Diagramm-Methode

● für jede negierte Variable von h ein positiver Schalter mit negiertem Eingang.

Am Beispiel der Majoritätsfunktion soll nun diese Methode noch einmal verdeutlicht werden (vergleiche mit den beiden anderen Methoden). Aus dem KV-Diagramm in Bild 16.14 a ermitteln wir die Funktion M zur Erfassung der Einsstellen, mit  $M = ab + ac + bc$ , und aus dem in Bild 16.14 b die Funktion  $h = \text{not}(M)$  zur Erfassung der Nullstellen, mit

$$h = \text{not}(a)\text{not}(b) \text{ or } \text{not}(a)\text{not}(c) \text{ or } \text{not}(b)\text{not}(c).$$

Die Funktion M verwenden wir für die Erstellung des p-Netzes, indem wir sie mit negativen Schaltern realisieren, d.h. auch die Eingänge invertiert werden. Für das n-Netz verwenden wir die Funktion h, indem wir entsprechend positive Schalter verwenden. Das Ergebnis sehen wir dann in Bild 16.15.

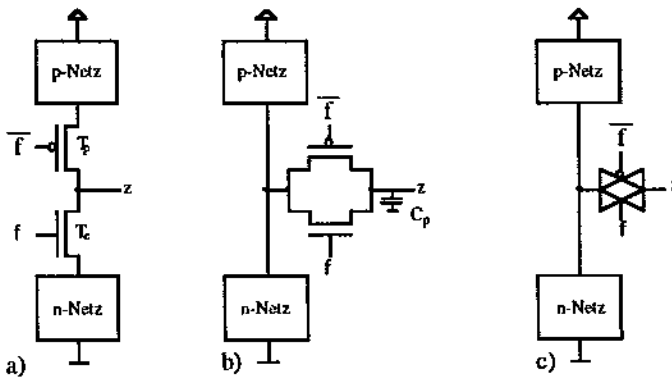


Bild 16.16: Clocked CMOS-Schaltungstechnik a) Prinzip, b) und c) logisches Äquivalent

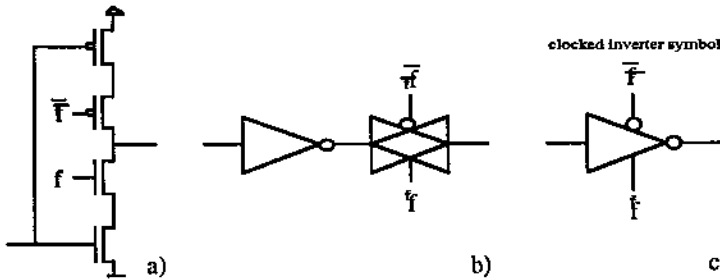


Bild 16.17:  
Clocked  
CMOS-In-  
verter: a)  
Schaltplan,  
b) logisches  
Äquivalent,  
c) Symbol.

### 16.2.1.4 Ermittlung des komplementären Transmissionsnetzes

Dieser Abschnitt gibt einen Algorithmus an zur Ableitung eines komplementären Transmissionsnetzes  $T_{ek}$  aus einem vorgegebenen Transmissions-Ausdruck  $e$ .

Eingabe: Transmissionsausdruck  $e$

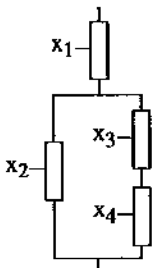


Bild 16.18: T-  
Netz-Beispiel

Ausgabe: Transmissionsnetz  $T_{ek}$

Verfahren: begin

ersetze in  $e$  alle ' $\wedge$ ' durch ' $\vee$ ' und alle ' $\vee$ ' durch ' $\wedge$ ' und  
erhalte so den zu  $e$  komplementären Ausdruck  $e_k$ ;

generate ( $e_k, A, B, T_{ck}$ );

end.

Erläuterung: Zunächst wird der Komplementäre Transmissionsausdruck gebildet und dann aus diesem, durch den Algorithmus von Abschnitt 7.5.3, das gesuchte Transmissionsnetz erzeugt.

Beispiel: Für den Transmissionsausdruck  $(x1 \vee (x2 \wedge (x3 \vee x4)))$  gemäß Bild 7.18 liefert der Algorithmus als Ausgabe das Transmissionsnetz nach Bild 16.18.

### 16.3 Dynamische CMOS-Schaltungen (Clocked-CMOS)

Bei clocked-CMOS-Technik sind zwei komplementäre Netze getrennt durch zwei Transistoren  $T_p$  und  $T_n$ , wie in Bild 16.16 a gezeigt wird. Bei durchlässigem p-Netz fließt Strom über  $T_p$ , bei durchlässigem n-Netz über  $T_n$  nach z. Dies ist äquivalent zur dynamischen CMOS-Schaltung (Bild 16.16 b). Während  $\text{not}(f)$  wird der Ausgabewert z ausschließlich per Kapazität gespeichert ( $C_p$ , siehe Bild 16.16 b). Dies entspricht der dynamischen CMOS-Schaltungstechnik (vgl. Übersicht über MOS-Schaltungstechniken, Bild 16.4). Für einen clocked-CMOS-Inverter (Bild 16.17 a) äquivalent der Schaltung in Bild b) verwendet man das Symbol nach Bild c).

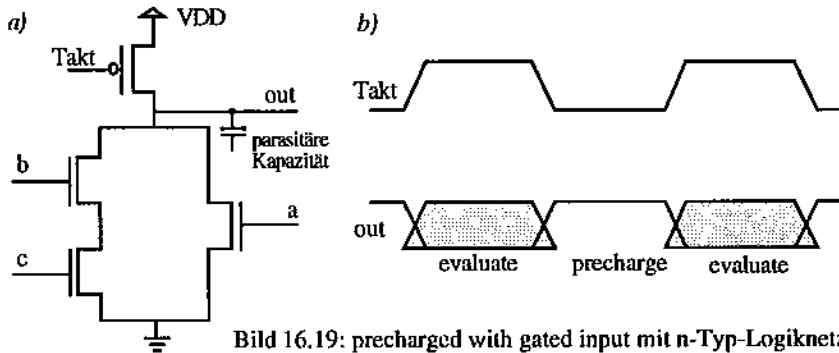


Bild 16.19: precharged with gated input mit n-Typ-Logiknetz.

### 16.4 Precharged CMOS-Schaltungstechnik (PE-Logiken)

Im Gegensatz zu statischen und dynamischen CMOS-Schaltungen verfügen Precharge-CMOS-Schaltungen über **nur ein einziges logisches Transmissionsnetz**. Diese sind eigentlich Varianten einfacher nMOS- oder pMOS Schaltungstechnik. Die Zugehörigkeit zu CMOS ergibt sich daraus, daß noch ein einzelner Transistor der dazu komplementären Technologie hinzukommt. Dieser ersetzt den bei Verhältnislogik üblichen Last-Transistor. Das Ziel gegenüber einfacher nMOS oder pMOS-Schaltungstechnik ist die **Aufhebung der Unsymmetrie des Schaltverhaltens**. Der 0-zu-1-Übergang am Ausgang wird beschleunigt. Einzelheiten hierzu werden in den folgenden Abschnitten behandelt.

#### 16.4.1 Precharged with gated input

Dies ist die einfachste Form einer dynamischen Logik. In Bild 16.19 ist die Realisierung der Funktion  $out = \text{not}((a \text{ and } b) \text{ or } c)$  unter Verwendung von n-Kanal Transistoren dargestellt. Lediglich für den Takteingang wurde ein p-Kanal Transistor verwendet. Die Arbeitsweise der Schaltung gliedert sich in 2 Phasen:

1) **Precharge-Phase (Takt "0")**: Der durch den Takt gesteuerte p-Kanal-Pull-up-Transistor ist geöffnet. Bedingung: da aber ein direkter Pfad von  $V_{SS}$  und  $V_{DD}$  verhindert werden muß, sei vorausgesetzt, daß in dieser Phase  $(a \text{ and } b) \text{ or } c = 0$  gilt. D.h es gibt keine Verbindung des Ausgangs zu  $V_{SS}$ . (Diese Schaltungstechnik heißt deshalb *precharged with gated input*.) Der Ausgang wird daher "1" und die parasitären Kapazitäten des Ausgangs werden geladen. Diese Kapazitäten werden gebildet durch die Gate-Kapazitäten aller an den Ausgang angeschlossenen Eingänge, den Drain-Source-Kapazitäten des p-Kanal Transistors und den Drain-Source Kapazitäten der durch die Signale a und b angesteuerten n-Kanal Transistoren. Diese Phase wird auch *precharge phase* genannt.

2) **Evaluate-Phase (Takt "1")**: Jetzt sperrt der p-Kanal Transistor und der Wert des Ausgangs wird von den Eingangssignalen a, b und c abhängig. Gilt immer noch  $(a \text{ and } b) \text{ or } c = 0$ , so bleibt die Ladung der parasitären Kapazitäten erhalten und der Ausgang ist weiterhin "1". Gilt dagegen  $(a \text{ and } b) = 1$  oder  $c = 1$ , so kann die Ladung abfließen und der Ausgang wird "0". Diese Phase wird *evaluation phase* genannt.

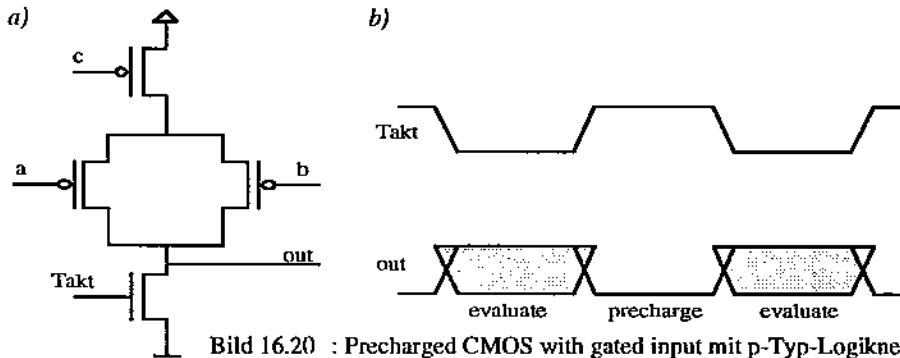


Bild 16.20 : Precharged CMOS with gated input mit p-Typ-Logiknetz

Bild 16.20 zeigt die Realisierung der gleichen Funktion unter Verwendung von p-Kanal-Transistoren und eines n-Kanal-Transistors für den Takteingang. Die Schaltung funktioniert analog zur Schaltung aus Bild 16.19, wobei hier aber in der precharge-Phase (Takt = "1") der Ausgang "0" wird, und in der evaluation-Phase (Takt = "0") der Ausgang in Abhängigkeit von den Eingängen auf "1" gezogen werden kann. Die Grundidee besteht darin, daß im Gegensatz zur Verhältnis-Logik hier der Pull-up-Transistor keinem Inverterverhältnis genügen muß und daher niederohmig sein darf.

Ein Nachteil dieser Schaltungstechnik ist, daß man keine zwei Gatter des gleichen Typs hintereinander schalten kann, um komplexe Funktionen zu realisieren wie dies in Bild 16.21 illustriert ist. Die beiden Gatter haben einen gemeinsamen Takt, und in der gemeinsamen precharge-Phase werden ihre Ausgänge "1". Da der Ausgang x der ersten Stufe aber Eingang der zweiten Stufe ist, erhält man einen Kurzschluß zwischen  $V_{DD}$  und  $V_{SS}$ .

Um die gewünschte Funktion zu realisieren kann man aber zwei Gatter unterschiedlichen Typs kaskadieren, wie dies in Bild 16.22 gezeigt wird. Um eine gemeinsame precharge-Phase zu erhalten, braucht man hier aber ein negiertes Taktsignal. Durch Verschiebungen der beiden Taktsignale gegenüber den "idealen" komplementären Taktsignalen, kann es hier zu Situationen

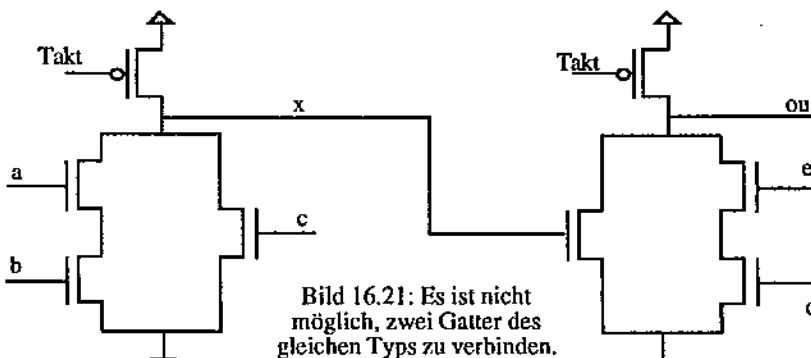


Bild 16.21: Es ist nicht möglich, zwei Gatter des gleichen Typs zu verbinden.



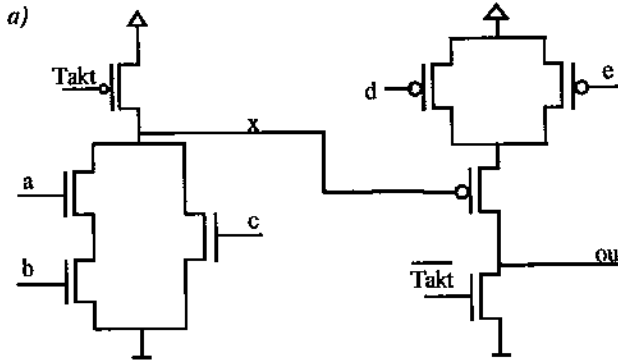
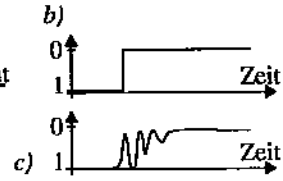


Bild 16.22: Gatter verschiedenen Typs können hintereinandergeschaltet werden (a), c) durch glitches verfälschtes b) Soll-Signal.



kommen bei denen "glitches" auftreten oder bei denen das Ausgangssignal einen falschen Wert annimmt. Der Begriff *glitches* ist in Bild 16.22 b-c veranschaulicht.

**16.4.2 P-E-Logik**

Bei der P-E-Logik (Precharge-Evaluation-Logic) werden die beiden Phasen durch ein einziges Taktsignal und zwei Transistoren gesteuert. Die Schaltung aus Bild 16.23 a arbeitet in zwei Phasen (dazu analog Bild 16.23 b mit p-Kanal Transistoren):

- 1) Takt "0": Der p-Kanal-Transistor leitet und der n-Kanal-Transistor sperrt. Es kann hier also keinen Pfad von  $V_{DD}$  zu  $V_{SS}$  geben. Der Ausgang wird daher unabhängig von den Eingängen auf "1" gezogen und die parasitären Kapazitäten geladen.
- 2) Takt "1": Jetzt sperrt der p-Kanal-Transistor und der n-Kanal-Transistor leitet. In Abhängigkeit von den Eingängen bleibt der Ausgang weiterhin "1" oder er wird auf "0" gezogen.

Der Vorteil der P-E-Logik liegt in der Tatsache, daß hier **Gatter gut kaskadierbar** sind, wie dies in Bild 16.24 gezeigt ist. Um ein korrektes precharging zu gewährleisten, müssen aber alle p-Kanal Eingänge mit n-Kanal Ausgängen verbunden werden. Da ein invertiertes Taktsignal benötigt wird, kann es auch hier Probleme mit Verschiebungen gegenüber den "idealen" Taktsignalen geben, wobei die P-E-Logik mit ihren "precharge"- und "evaluation"-Transistoren aber gegenüber der Technik mit gated input eine zuverlässigere Funktion bietet.

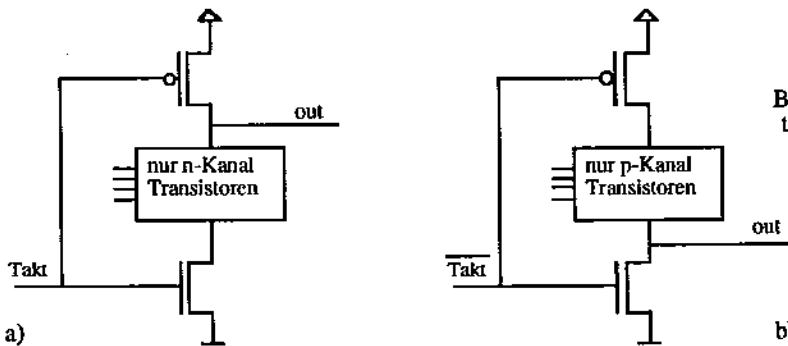


Bild 16.23: Gatter in PE-Logik

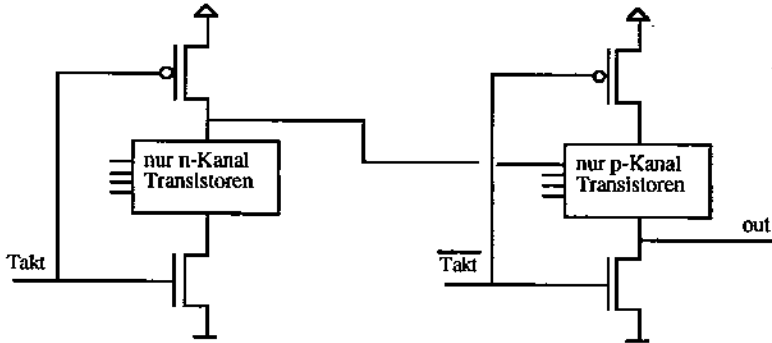


Bild 16.24:  
Kaskadierung  
von PE-Gattern  
ist möglich

Im Gegensatz zu precharged nMOS-Technik wird bei precharged CMOS nur ein einziges Taktsignal  $\phi$  (für evaluate) benötigt, da der precharge-pullup nunmehr ein p-MOS-Transistor ist (negativer Schalter, Bild 16.25). Vorteile dieser Technik sind:

- kein Pullup-Netzwerk wird benötigt (erhebliche Flächeneinsparung möglich)
- da kein direkter Strompfad  $V_{DD}$  und  $V_{SS}$  verbindet: keine statische Verlustleistung.

Precharge-CMOS-Technik nach Bild 16.25 a hat folgende Nachteile:

- eine "1" am Ausgang wird (während not(f)) nur kapazitiv gespeichert: Empfindlichkeit gegen kapazitiv eingekoppelte Störung (Bild 16.26)
- bei Kaskadierung (z.B. die zwei Stufen in Bild 16.25 a) Empfindlichkeit gegen "glitches" und "hazards"

Wenn beispielsweise y während der E-Phase einen 1-zu-0-Wechsel hat, kann dieser sich wegen Laufzeiten innerhalb  $f_2$  eventuell erst gegen Ende der E-Phase auswirken. Je nach Schaltung des Netzes kann es dann vorkommen, daß  $f_2$  die "0" am Eingang nicht mehr rechtzeitig wahrnimmt, womit am Ausgang z fälschlich eine "0" angezeigt wird.

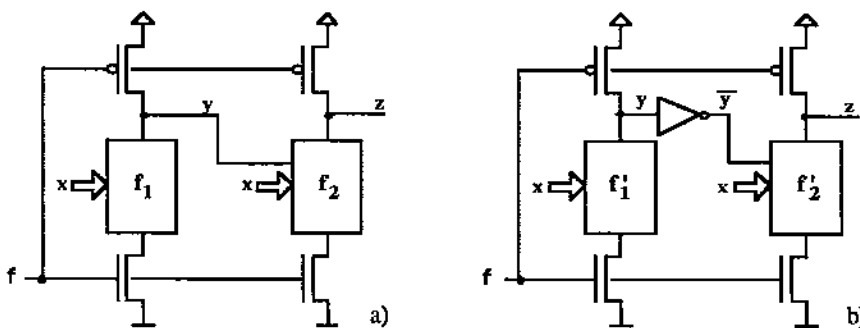


Bild 16.25: precharged CMOS-Technik: a) falsches Precharging, b) DOMINO-Technik.

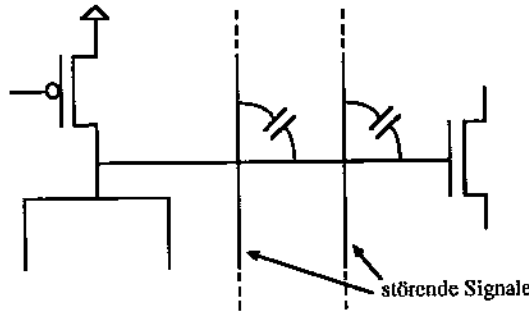


Bild 16.26: kapazitiv eingekoppelte Störungen

Abhilfe kann die Schaltung nach Bild 16.25 b schaffen. Ein eingefügter Inverter (der ein Redesign von  $f_1$  und  $f_2$  erfordert) sorgt dafür, daß während der P-Phase alle Eingänge von  $f_2$  den Wert "0" haben. Ein 1-zu-0-Wechsel dieser Eingänge während der E-Phase ist somit ausgeschlossen. Diese Schaltung wird DOMINO-Technik (siehe Abschnitt 16.4.4 ) genannt. Die Schaltung hat einen Nachteil: DOMINO-CMOS-Technik ist etwas anfällig gegen Ladungsausgleich-Probleme (*charge-sharing*, siehe Abschnitt 16.4.3 ). Ladungsausgleich-Probleme in CMOS sind von Noise untersucht worden [6]. Außerdem tendieren diese Schaltungen zu unsymmetrischer Dynamik (slow-off .... fast-on).

### 16.4.3 Das Charge-Sharing-Problem

Bei PE-Schaltungstechniken können "Charge-Sharing"-Phänomene (Ladungs-Ausgleich) zu falschen Ausgabewerten führen. Das Phänomen sei zunächst an einem Beispiel erläutert (Bild 16.27, eine PE-*and*-Schaltung).

Die Eingangskombination (x, y) sei zunächst (0,1) (siehe Bild Bild 16.27 a). Während der P-Phase wird der Ausgang z auf "1" angehoben (Bild 16.27 a). Während der darauffolgenden E-

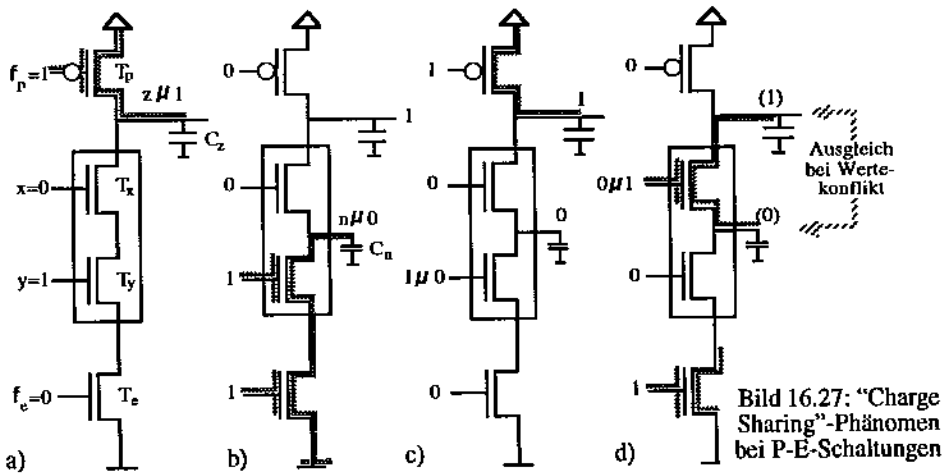
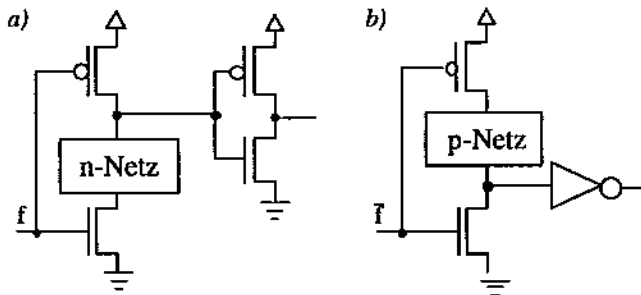


Bild 16.27: "Charge Sharing"-Phänomen bei P-E-Schaltungen


 Bild 16.28: Nur ein einziges Taktsignal  $f$  ist nötig.

Phase (Bild 16.27 b) speichert  $C_z$  den Wert  $z=1$ , da Transistor  $T_x$  gesperrt ist. Der interne Knoten  $n$  hingegen wird auf "0" herunter gezogen, da  $T_y$  und  $T_e$  leitend sind. Während der nächsten Precharge-Phase bleiben die Ladungen  $C_z$  und  $C_n$  erhalten, da  $T_x$  weiterhin nichtleitend ist. Knoten  $n$  bleibt auch dann auf "0", wenn während  $f_p$  der Eingang  $y$  von "1" auf "0" wechselt (Bild 16.27 c). Während der nächsten E-Phase (Bild 16.27 d) erfolgt ein Ladungsausgleich zwischen  $C_n$  und  $C_z$ , wenn  $x$  mit etwas Verspätung von "0" auf "1" wechselt. Am Ausgang erscheint statt der erwarteten "1" ein Spannungswert zwischen "0" und "1" oder sogar "0" (je nach Schwellwertspannung der Transistoren). Es entsteht eine Fehlfunktion durch Ladungsausgleich (Charge-Sharing).

Das Problem ist dadurch zu beheben, daß alle Eingabewechsel rechtzeitig vor Beendigung der P-Phase erfolgen, sodaß alle Eingänge während der E-Phase stabil bleiben. Dies sei am Beispiel nach Bild 16.27 veranschaulicht. Wir legen den 0-1-Wechsel von  $x$  in Bild 16.27 d nach Bild c, also von der E-Phase in die P-Phase. Damit stellt sich vor Beendigung der P-Phase die Konfiguration  $(x, y)$  mit (1,0) ein. Also ist der Transistor  $T_x$  während  $f_p$  leitend, womit Knoten  $n$  und Kapazität  $C_n$  auf "1" gebracht werden ( $T_y$  ist undurchlässig). Zu Beginn der E-Phase (Bild 16.27 d) sind beide Kapazitäten ( $C_z$  und  $C_n$ ) auf "1". Auch ein eventueller leichter Ausgleichstrom führt deshalb nicht zu einem Werte-Konflikt. Als Maßnahmen zur Vermeidung von Fehlern durch Ladungsausgleich sind geeignet:

- Alle Eingänge des pull-down-Netzes müssen rechtzeitig vor Beendigung der P-Phase ihre gültigen Werte erreichen.
- Während der E-Phase müssen diese dann stabil bleiben.

Man kann zusammenfassen, daß die Flächensparnis und Beschleunigung teuer erkauft wird.

#### 16.4.4 DOMINO-CMOS-Schaltungstechnik

DOMINO-CMOS-Technik [5] entsteht aus "precharge-CMOS-Schaltungstechnik" durch Nachschaltung eines statischen CMOS-Inverters (Bild 16.28 a). Neben einem n-Typ (Bild 16.28 a) ist auch ein p-Typ möglich, bei welchem der Kern aus pMOS-Transistoren zusammengesetzt wird (Bild 16.28 b). Wo statische CMOS-Technik  $2k$  Transistoren benötigt, kommt DOMINO-Technik mit  $k+4$  Transistoren aus. Der Kern hat  $k$  Transistoren.

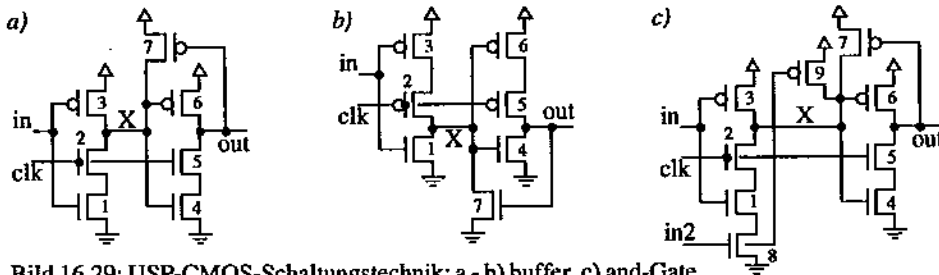


Bild 16.29: USP-CMOS-Schaltungstechnik: a - b) buffer, c) and-Gate.

**DOMINO-Entwurfsregeln.** Zur Vermeidung von Hazards wird die Einhaltung folgender Entwurfsregeln dringend empfohlen. Hierbei wird Unterschieden zwischen *externen* Signalen und *internen* Signalen. Letztere kommen von anderen DOMINO-Gattern, während erstere von statischen Quellen kommen, wie Flipflops oder Latches. Alle internen Signale müssen *stabil* und "0" sein während der P-Phase und zu Beginn der E-Phase. Externe Signale müssen sich in der P-Phase *stabilisieren* und während der E-Phase *stabil* bleiben, gleich ob "0" oder "1". Während der E-Phase können interne Signale "0" bleiben oder einen "0-zu-1"-Übergang machen. Da externe Signale früher stabilisieren müssen, werden deren Transistoren am Besten in die Nähe des Ausgangs platziert. Dadurch wird auch erreicht, daß bei externer "0" ein Zustandsübergang verhindert wird, unabhängig davon, welchen Wert innere Signale haben.

**Fehler durch Ladungsausgleich.** Zur Vermeidung von Fehlern durch Ladungsausgleich (CR: charge redistribution, charge sharing) wird angestrebt, die Kapazität des Ausgangsknotens groß zu machen gegen die anderen parasitären Kapazitäten innerhalb des Kerns. Eine Methode benützt einen großen Inverter (zusätzlicher Vorteil: großes Fan-out; Nachteil: Schaltung ist langsamer). Eine zweite Methode verwendet kleinstmögliche Transistoren im Kern (Vorteil: weniger Fläche; Nachteil: Schaltung ist langsam). DOMINO hat also mit NORA-Technik gemein, daß CR-Vermeidung zu langsameren Schaltungen führt. Das Vorliegen von CR-Problemen in einem Gatter kann durch Analyse-Programme analysiert werden, wie z.B. das MOSMESH-Problem [4].

**Logik-Probleme.** DOMINO-Schaltungen sind logisch unvollständig, da ein Gatter keine Negation realisieren kann. Negation kann nur über die der letzten Stufe nachfolgenden Schaltungen anderer Art erreicht werden. Die passende Funktion kann aus der ursprünglich als Entwurfsaufgabe vorliegenden Form durch "bubble pushing" (verschiebung der Negationen, vgl. hierzu den Abschnitt "Entwurfsverfahren für statische CMOS-Schaltungen", Kapitel 16.2.1) abgeleitet werden. Dies kann z.B. auch durch "expression-tree transformation" erreicht werden, wie im MGMG-Programm [4].

**Bevorzugung von OR-Schaltungen.** Vermeidung von CR ist möglich durch die Vermeidung von Serienschaltungen im Kern: es entsteht ein OR-Gatter. Die Schaltung ist auch besonders schnell durch einen niedrigeren pull-down-Widerstand. Durch parallele Transistoren in einem p-Kern bekommen wir zwei p-Typ-Gatter (Bild 16.28 b) welche ein AND-Gatter realisieren. Diese werden in DOMINO-Technik jedoch nicht verwendet, sondern nur in NORA-CMOS-Schaltungstechnik.

## 16.5 USP-CMOS Schaltungstechnik

Hierbei handelt es sich um eine neuere CMOS-Schaltungstechnik [1] [7] [8] [9]. Durch Optimierung der Transistor-Größen und ein sehr einfaches Taktungsschema (unified single phase clocking scheme: USP clocking scheme) wurden sehr hohe Schaltgeschwindigkeiten erreicht. Der neue Alpha-Prozessor von Digital Equipment wurde nach einer Variante dieser Schaltungstechnik entwickelt [3] in einer 0,75- $\mu$ -Technologie mit 3 Metallisierungsebenen. Der Chip umfaßt 1,68 Millionen Transistoren und erreicht bei einer Betriebsspannung von 3,3 Volt eine Taktrate von 200 MHz, wobei die Transistoren Schwellwertspannungen um 0,5 Volt (bzw. - 0,5 Volt) haben.

Bild 16.29 zeigt einige Schaltungsbeispiele. Bild 16.29 a zeigt eine nicht invertierende Buffer-Stufe für positive Taktung. Die Transistoren 1, 2, und 3 bilden einen Inverter, die Transistoren 4, 5, und 6 einen diesem nachgeschalteten zweiten Inverter, weshalb an *out* der Eingangswert *in* unverändert wieder ausgegeben wird. Für unterschiedliche Einbettungen mit unterschiedlich großen zu treibenden kapazitiven Lasten steht ein Repertoire von Stufen unterschiedlicher Dimensionierungen und elektrischer Eigenschaften zur Verfügung. Die Stufe ist "transparent", wenn *clk* auf "high" ist, was sorgfältig zu beachten ist wegen *race-through*-Gefahr. Der Knoten X ist empfindlich gegen Einstreuungen (noise), weshalb Transistor 7 hinzugefügt wurde (vgl. Bild 16.29 a). Dieser schwache Rückkopplungsweig ist so dimensioniert, daß als Gegenmaßnahme genügend Strom geliefert wird und Transistor 6 sicher gesperrt gehalten wird (wenn die Stufe transparent ist: dann kann X hochohmig *high* sein und *out* hochohmig *low*).

Bild 16.29 a zeigt eine nicht invertierende Buffer-Stufe für positive Taktung und Bild 16.29 b eine Stufe für negative Taktung, die ähnlich zu erklären ist wie die Stufe in Bild 16.29 a. Bild c zeigt eine and-Schaltung. Die Schaltungstechnik ist schwierig zu beherrschen. Eine gegenläufige Ausbreitung von Taktsignal und Datenfluß hat sich gegen die Gefahr von *races* (Wettläufen) als günstig erwiesen. Insbesondere ist auf dem Chip auf sehr sorgfältige Verlegung des Verteilungsnetzes für die Taktsignale zu achten [3].

## 16.6 Literatur

- [1] M. Afghahi, C. Svensson: A Unified Single-Phase Clocking Scheme for VLSI Systems; IEEE JSSC Vol. 25, No. 1, 1990
- [2] M. Anaratone: Digital CMOS Circuit Design; Kluwer Acad. Publishers; Boston, 1986
- [3] D. Dobberpuhl et al.: A 200-MHz 64-b Dual-Issue CMOS Microprocessor; IEEE J. SSC-27,11 (Nov. 1992)
- [4] M. E. Hofmann: Automated Synthesis of Multi-Level Combinational Logic in CMOS Technology; Menu # VCB/ERL M85/53, 1. July 1985, Univ. of Calif, Berkeley, USA
- [5] R. Krambeck, C. Lee, H. Law: High Speed Compact Circuits with CMOS; IEEE J. SC-17, no. 3 (June 1982) pp. 614-619
- [6] R. C. Noice, R. Mathews, J. Newkirk: A clocking discipline for 2-phase digital systems; Proc. IEEE Int'l Conference on Circuits and Computers, Sept/Oct 1982
- [7] J. Yuan, C. Svensson: CMOS Circuits Speed Optimization Based on Switch Level Simulation; Proceedings of IEEE, ISCAS '88
- [8] J. Yuan, C. Svensson: High-speed CMOS circuit techniques; IEEE J.SSC-24,1 (Feb 1989)
- [9] J. Yuan, C. Svensson: Pushing the limits of standard CMOS; IEEE Spectr., Febr. 1991