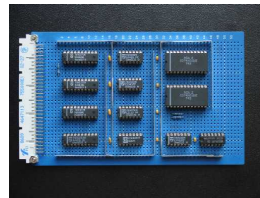


Digital- och datorteknik



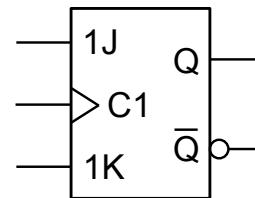
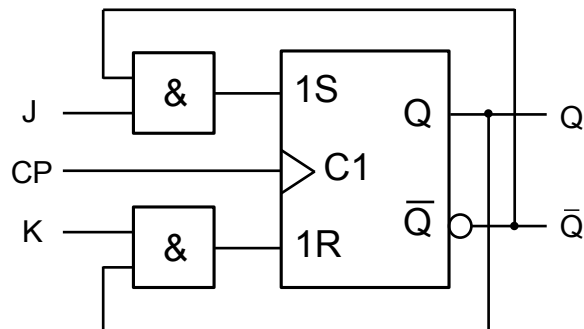
Föreläsning #10

Biträdande professor Jan Jonsson

Institutionen för data- och informationsteknik
Chalmers tekniska högskola

Vippor

Funktions- och excitationstabell för JK-vippa:



Funktionstabell för JK-vippa		
J	K	Q ⁺
0	0	Q
0	1	0
1	0	1
1	1	\bar{Q}

Excitationstabell för JK-vippa			
Q	Q ⁺	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

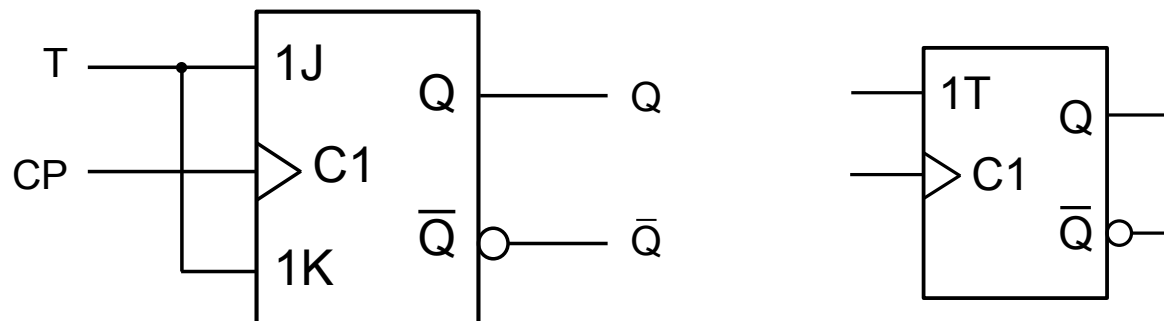
Vippor

Vippa med alternerande tillstånd:

Med JK-vippan fick vi en ny funktion (när $J = 1$ och $K = 1$) som kan användas för att enkelt växla mellan två logiska tillstånd. Om vi kopplar ihop J och K kan vi med en signal styra den alternerande funktionen.

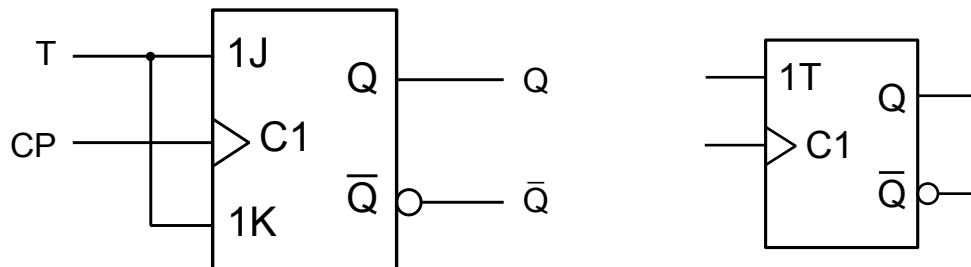
Detta kallas för en T-vippa.

T-vippan är en grundläggande komponent som vi litet senare kommer att använda vid konstruktion av tillståndsmaskiner, t ex räknare och styrenheten i vår dator.



Vippor

Funktions- och excitationstabell för T-vippa:



Funktionstabell för T-vippa	
T	Q^+
0	Q
1	\bar{Q}

Excitationstabell för T-vippa		
Q	Q^+	T
0	0	0
0	1	1
1	0	1
1	1	0

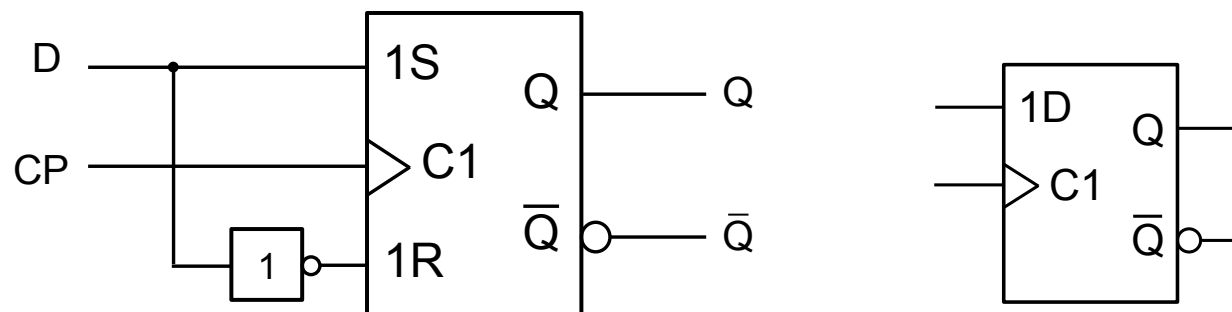
Vippor

Vippa för lagring av en bit:

Vi har sett att JK-vippan kan förhindra att det icke-önskvärda tillståndet i en SR-vippa uppträder. Ett annat sätt att åstadkomma detta är att, med en inverterare, se till att S och R alltid är varandras negationer.

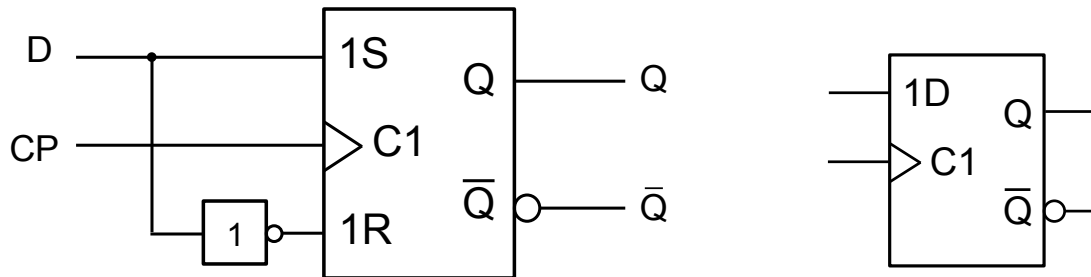
Detta kallas för en D-vippa.

D-vippan är som klippt och skuren för att lagra en bit, och blir därför ett viktigt byggblock vid konstruktion av de register som skall lagra dataord i vår dator.



Vippor

Funktions- och excitationstabell för D-vippa:



Funktionstabell för D-vippa	
D	Q^+
0	0
1	1

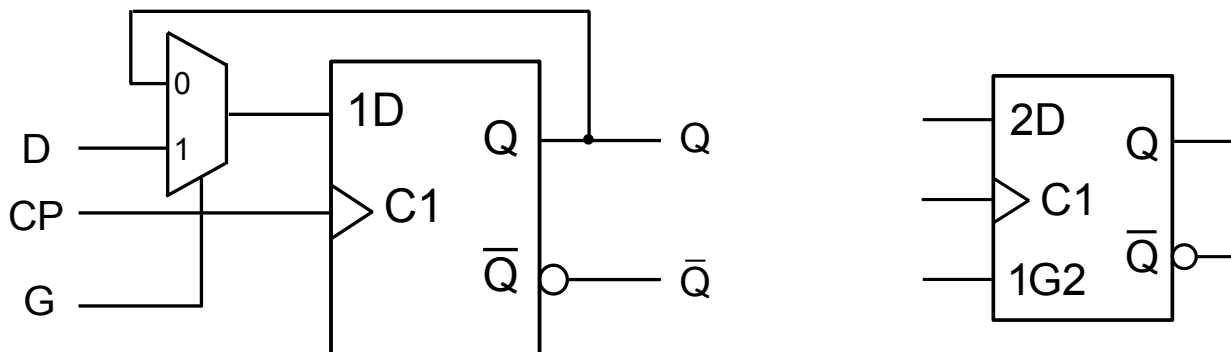
Excitationstabell för T-vippa		
Q	Q^+	D
0	0	0
0	1	1
1	0	0
1	1	1

Vippor

Vippa med villkorlig lagring av en bit:

När vi ansluter register till vår dators dataväg kommer klocksignalen att vara gemensam för alla D-vippor. För att undvika att tillståndet för alla vippor uppdateras vid varje klockpuls behöver vi en extra ingång som anger om vippan skall laddas med nytt värde eller ej. Genom att låta den extra ingången styra en 1-av-2-väljare kan vi välja om vippan skall laddas med ett nytt (D) värde eller med sitt gamla (Q) värde.

Detta kallas för en D-vippa med grind.



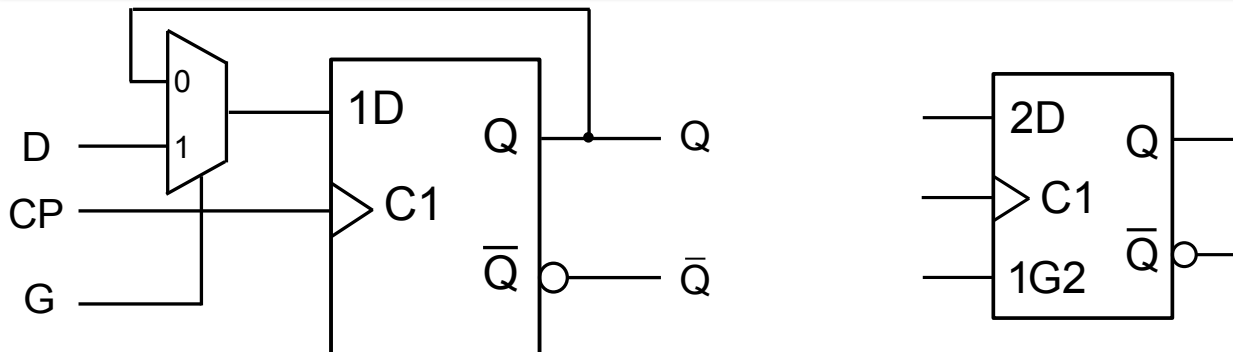
Vippor

Vippa med villkorlig lagring av en bit:

När vi ansluter register till vår dators dataväg kommer klocksignalen att vara gemensam för alla D-vippor. För att undvika att tillståndet för alla vippor uppdateras vid varje klockpuls behöver vi en extra ingång som kan stänga av uppdateringen för vissa vippor. Detta kallas för en enable-ingång. Om vi vill att långa bitar ska lagras i ett register, så kan vi använda en enable-ingång för att stänga av uppdateringen för vissa vippor. Detta kallas för en enable-ingång.

Jämför tidsdiagram för följande vippor:

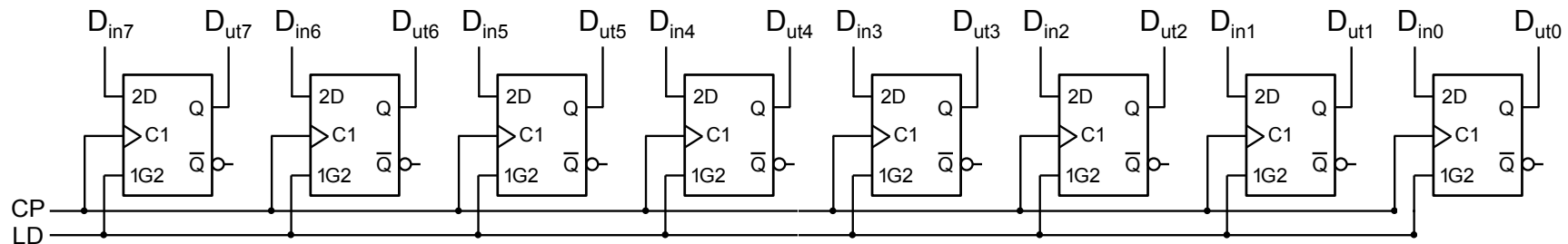
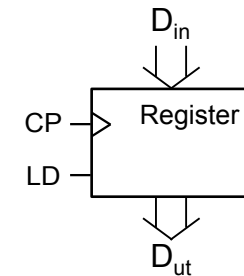
- a) positivt flanktriggad D-vippa
- b) positivt flanktriggad D-vippa med grind



Register

Register med gemensam klock- och grindsignal:

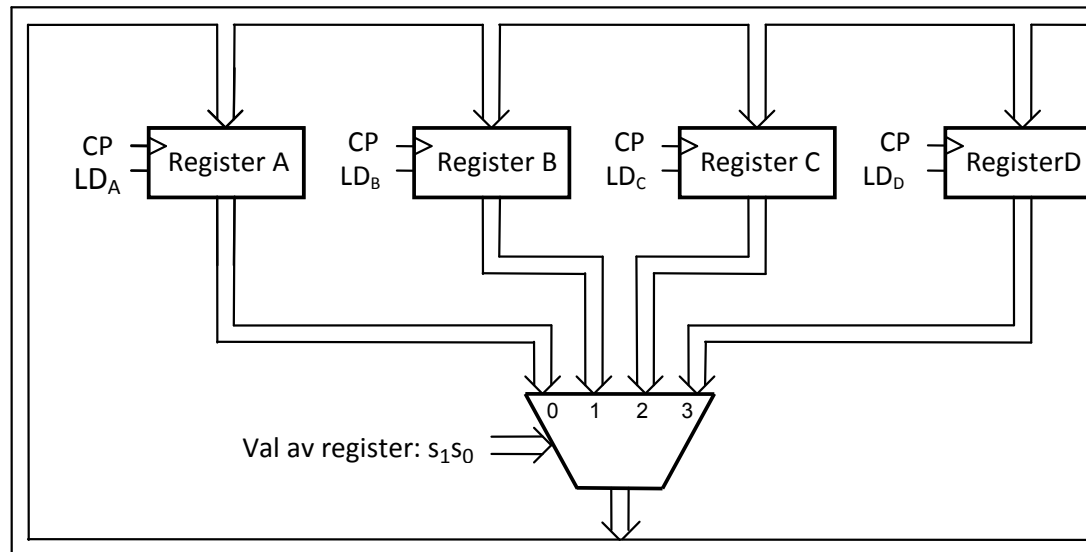
Vi kan nu bygga upp ett register för att lagra flera bitar, anpassat till behovet i vårt digitala system. Vår dator kommer t ex att använda 8-bitars register. För att alla vippor i ett register skall uppdateras samtidigt har de en gemensam klocksignal (CP). Dessutom har de en gemensam grindsignal, kallad load (LD), som gör det möjligt att uppdatera vipporna i just detta register.



Register

Dataöverföring mellan register:

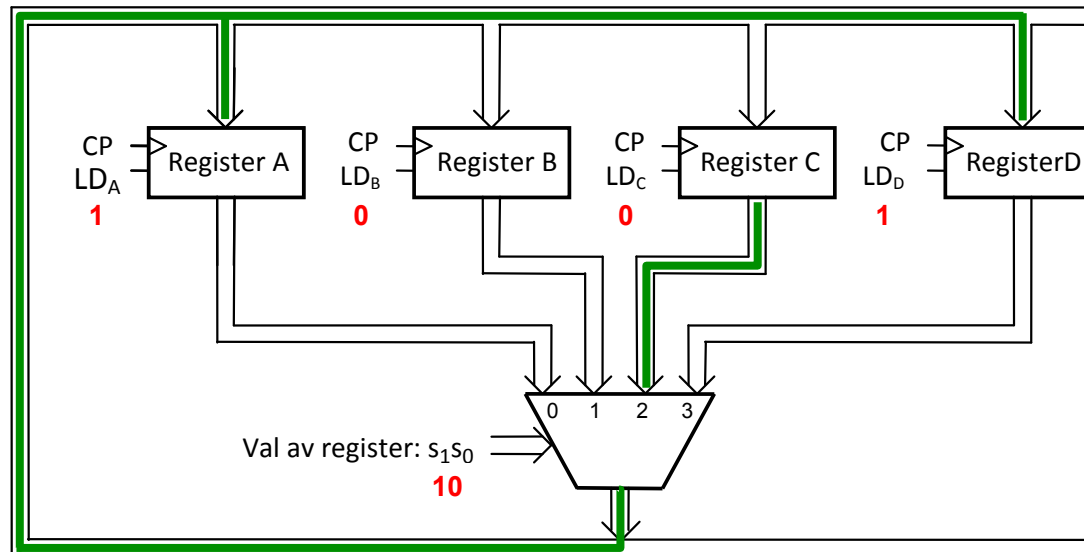
För att möjliggöra överföring av data mellan olika register måste de kopplas ihop på lämpligt sätt. En metod för detta är att överföra data via en väljare. Väljarens selektorsignaler avgör då från vilket register data skall hämtas. Data skrivs till de register vars LD-signal är aktiv.



Register

Dataöverföring mellan register:

För att möjliggöra överföring av data mellan olika register måste de kopplas ihop på lämpligt sätt. En metod för detta är att överföra data via en väljare. Väljarens selektorsignaler avgör då från vilket register data skall hämtas. Data skrivs till de register vars LD-signal är aktiv.

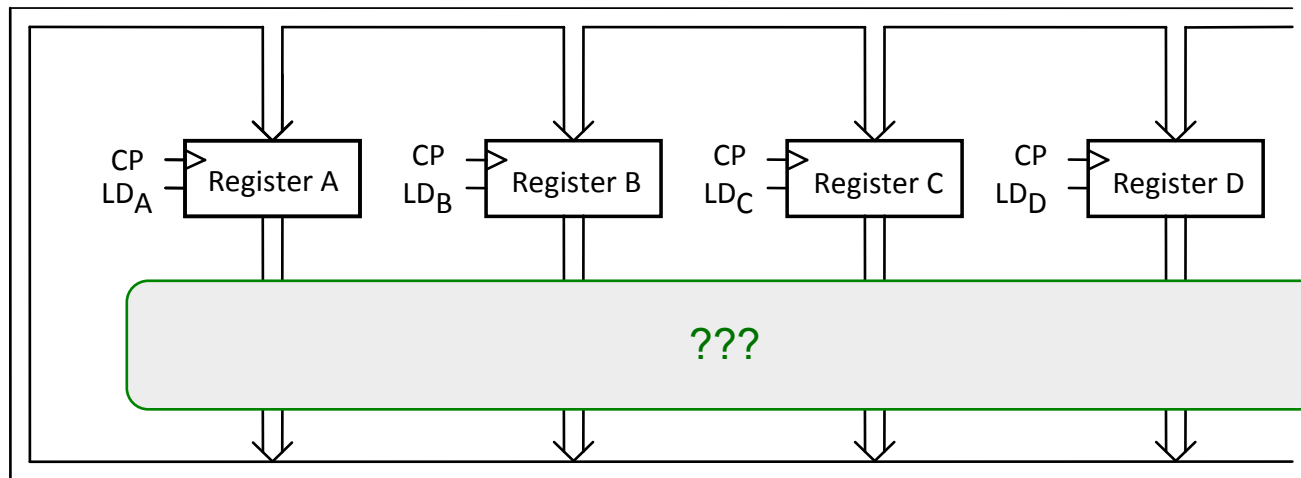


Exempel: Överföring från register C till register A och D.

Register

Dataöverföring mellan register:

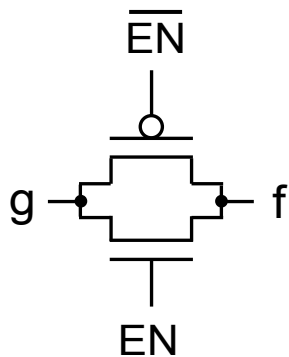
Den vanligaste metoden att överföra data mellan register använder däremot inte en väljare. Denna metod utnyttjar istället ett sätt att ansluta registrens utgångar till en databuss utan att riskera att de får elektrisk kontakt (och därmed skulle kunna förstöras).



Register

Transmissiongrinden:

För att undvika att utgångarna på två eller fler vippor får elektrisk kontakt måste en vippas utgångar kunna isoleras från databussen när vippans data inte skall användas. Detta kan åstadkommas med en transmissionsgrind, ett CMOS-transistorpar som har kanalerna parallellkopplade. När styrsignalen $EN = 1$ är båda kanalerna öppna och grinden släpper igenom signalen på ingång g till utgång f . När $EN = 0$ är kanalerna stängda och utgång f blir isolerad från ingång g .



Funktionstabell för transmissionsgrind		
EN	g	f
0	0	Z
0	1	Z
1	0	0
1	1	1

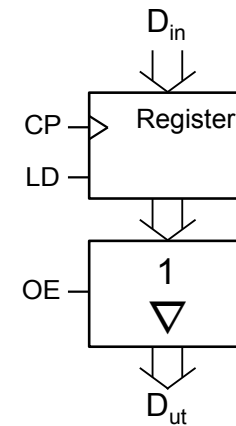
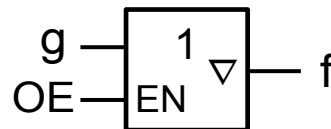
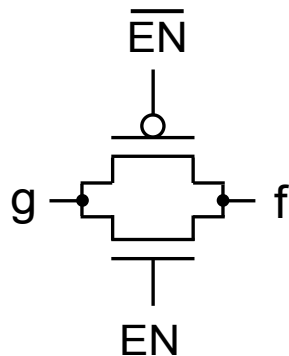
Högimpedanstillstånd med logiknivå 'Z' = varken '1' eller '0'

Grindens utgång kan anta tre olika tillstånd = 'three-state logic'

Register

Register med gemensam klock- och grindsignal:

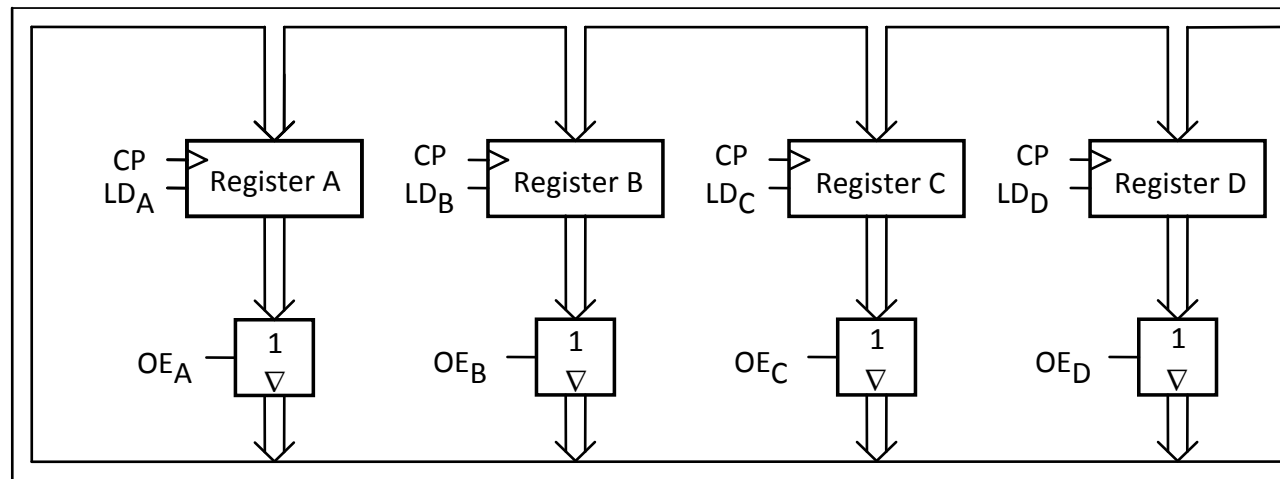
Vi ansluter nu en transmissionsgrind på varje vippas utgång. Alla samhörande vippor i ett register får därefter ytterligare en gemensam grindsignal, kallad output enable (OE), som gör det möjligt att ansluta utgångarna på vipporna till databussen.



Register

Dataöverföring mellan register:

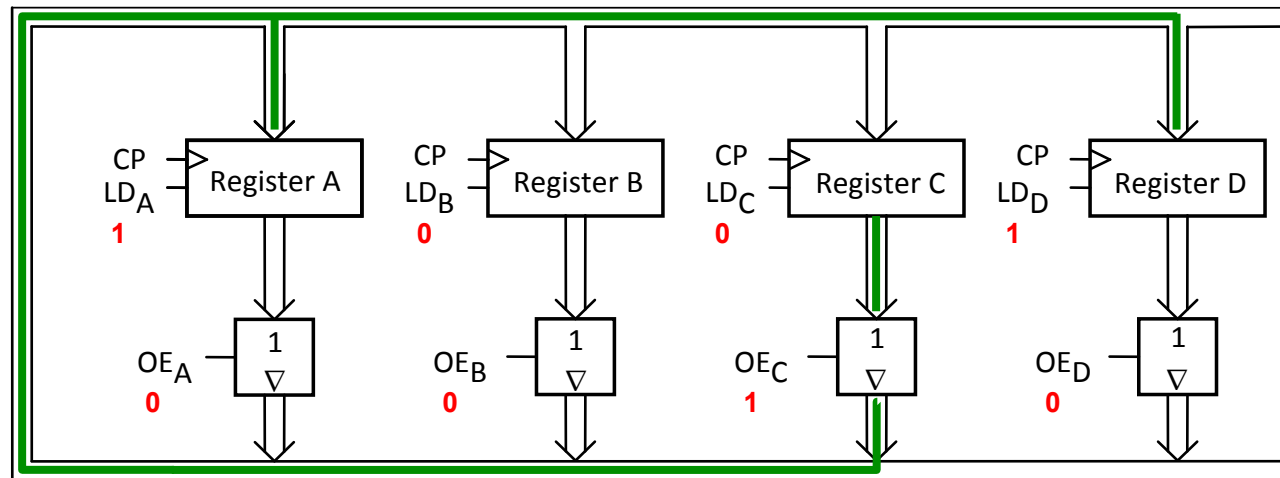
Vi har nu fått ett sätt att överföra data via en transmissionsgrindar. Data läses från till det register vars OE-signal är aktiv, och data skrivs till de register vars LD-signal är aktiv.



Register

Dataöverföring mellan register:

Vi har nu fått ett sätt att överföra data via en transmissionsgrindar. Data läses från till det register vars OE-signal är aktiv, och data skrivs till de register vars LD-signal är aktiv.



Exempel: Överföring från register C till register A och D.

Register Transfer Notation (RTN)

Uppgift 6.1:

Använd figurens dataväg och ange styrsignalssekvenser för följande registeröverföringar beskrivna enligt RTN.

- a) $T \rightarrow A$
- b) $T \rightarrow A ; T \rightarrow R$
- c) $R \rightarrow T ; A \rightarrow R , T \rightarrow A (= A \leftrightarrow R)$

Parallella operationer

Sekventiella operationer

