



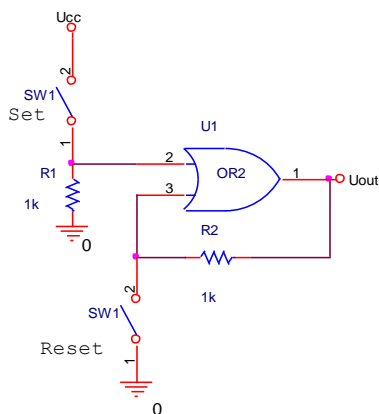
Kompendium

Flip Flops og Tællere

Rettelser og tilføjelser modtages gerne

/ Valle

Flip Flop's



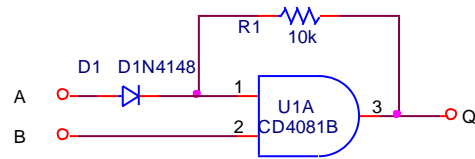
En Flip Flop er et kredsløb, der kan være sat enten i den ene eller den anden tilstand, dvs. enten 0 eller 1 på udgangen. Og den kan huske sin tilstand – i hvert fald indtil strømmen forsvinder.

Dette kredsløb kan bruges til at realisere en 1-bits hukommelse



Her et andet eksempel. Dette kredsløb udgør en R-S Flip Flop.

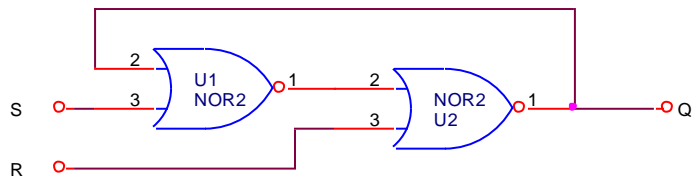
Forklar !! Hvordan skal set / reset-kontakterne monteres ??



SR-NOR

Vha. 2 NOR-gates kan der laves en SR-Flip Flop. (S og R står for Set og Reset). En FF er en 1-bits hukommelse. Den kan bruges i et kredsløb til at huske, at der fx har været trykket på en knap.

Kredsløbet kan tegnes således:



Analysér!

Indgangene må selvfølgelig ikke svæve, for så vil de opsamle støj, dvs. svinge med fx 50 Hz.

En SR-FF kan enten SÆTTES, eller RESETTES. Hermed menes, at en udgang kan tvinges Høj ved SET, eller LAV, ved reset, hvorefter kredsløbet forbliver i denne stilling.

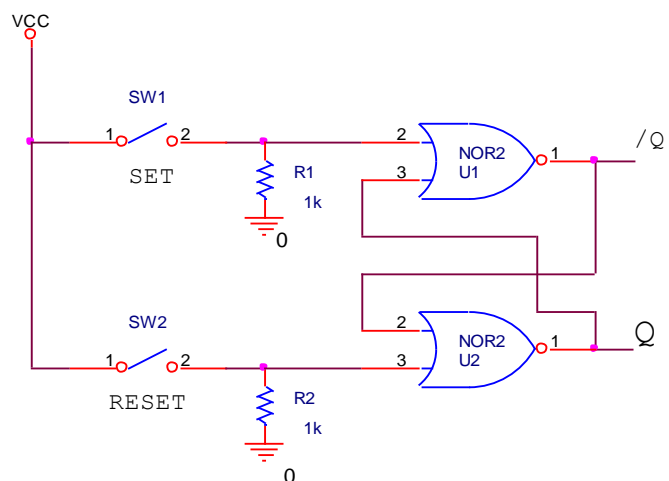
Den udgang, der bliver høj, når SET aktiveres, kaldes Q-udgangen. Den anden er så "Q-ikke".

Normalt tegnes kredsløbet med gatene placeret som følgende:

Nor-SR-FF'en er normalt LOW på indgangene. De skal aktiveres med et højt signal.

Her er der sat pull down modstande på, så omgivelsernes (50 HZ) frekvenser ikke påvirker de høj-impedante indgange.

Pull down-modstandenes værdi kan vælges fra 1 K til 100 K.



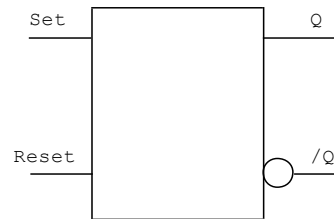


Sandhedsskemaet ser således ud!

S	R	Q-/Q
0	0	LÅST
0	1	0-1
1	0	1-0
1	1	Udef

At udgangene er undefinerede betyder, at man ikke med sikkerhed kan vide, hvad næste "stadie" vil være. Dvs. efter man har sluppet de to knapper. Det afhænger af, hvilken man har sluppet først.

I et diagram kan Flip-Floppen tegnes således:

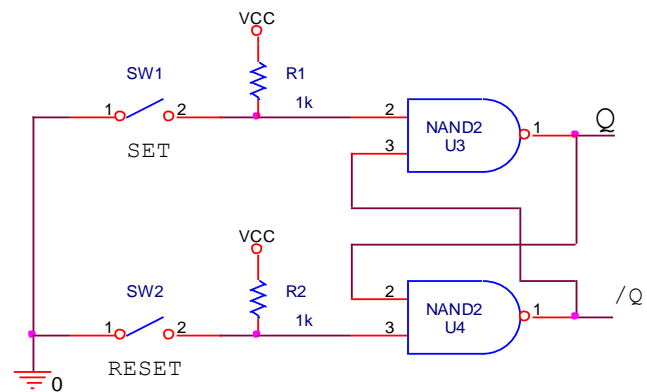


Ved Power-on kan man ikke vide, i hvilken stilling en Flip Flop står!!!

SR-NAND

En FF kan også opbygges af NANDGATES som vist herover. Den er aktiv Lav !!!

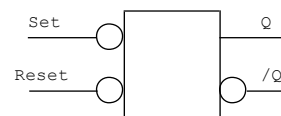
NAND-FF'en er aktiv lav. Dvs. indgangene er NH, (Normally High).



Sandhedsskemaet ser således ud !

S	R	Q-/Q
0	0	Udef
0	1	1-0
1	0	0-1
1	1	Låst

Diagram-symbolet ser således ud:



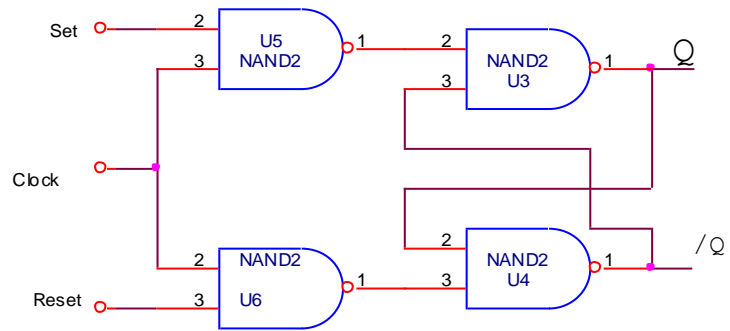


Eksemple på brug af en SR-FF: Styrekredsløb til tandbørste. Med separat Start og Stop.

Clocked SR-FF

Med en clock'ed FF vil Set og Reset-informationerne først overføres når CLOCK bliver høj. Dvs. at man i ro og mag kan tilføje Set og Reset de værdier, der skal overføres ved klock-pulsen !

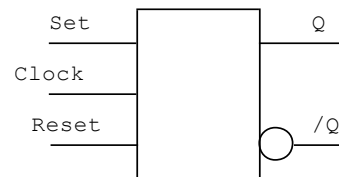
Bemærk, Aktiv Høj !!



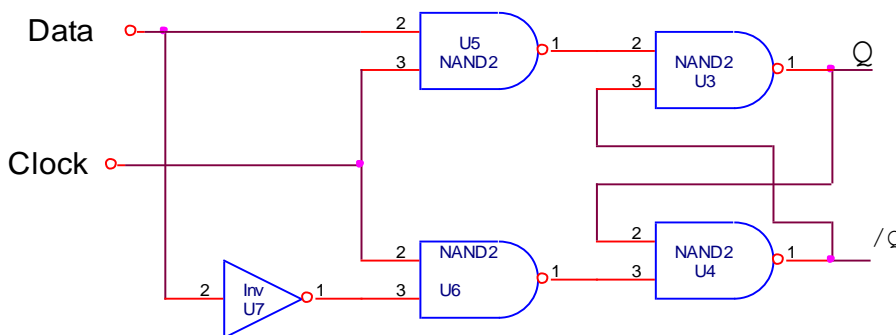
Sandhedsskema for Clocked SR-FF:

S	R	Clk	Q /Q
*	*	0	Låst
0	0	1	Låst
0	1	1	0-1
1	0	1	1-0
1	1	1	Udef

Diagramsymbolet:



Data FF, D-FF



Videreudvikles Clock'ed SR-FF'en kan konstrueres en Data-FF eller blot D-FF.

Normalt er /Q ikke ført ud !!

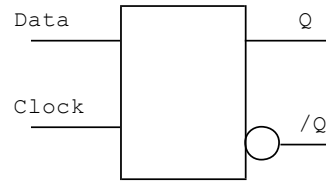
D-FF'en har ingen Udefinerede tilstande !

Værdien på Data-indgangen overføres så snart, clock'en går høj.

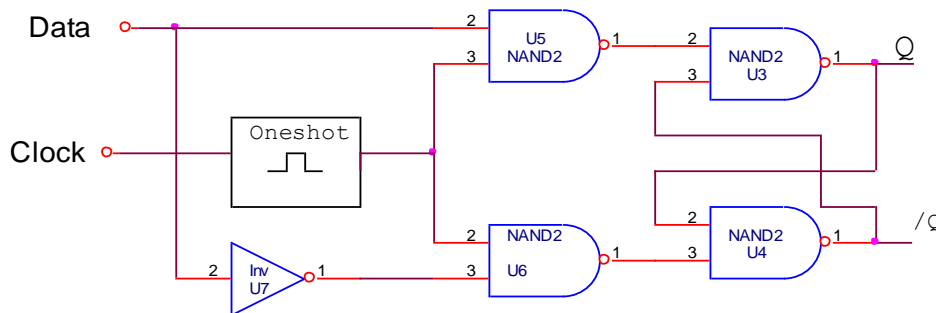
Men også ændringer på data, mens Clock er høj, overføres straks.

Sandhedsskema:

D	Clk	Q /Q
*	0	Låst
0	1	0-1
1	1	1-0



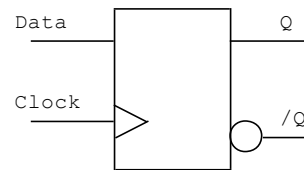
Indføres nu en "ONESHOT" i klock'ens vej i kredsløbet, af en sådan varighed, at FF'en lige netop når at udføre dens funktion, kan man opfatte flip-flop'en som "FLANKE-triggeret". Det er så nu kun i det øjeblik, clock'en skifter til høj, at "Data" overføres til udgangen.



Sandhedsskema:

D	Clk	Q/Q
*	0	Låst
0	↑	0-1
1	↑	1-0

Og Diagramsymbol:

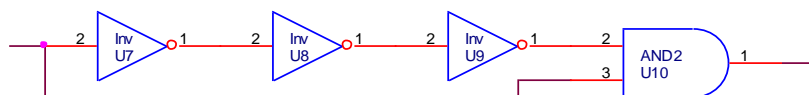


Den lille trekant i symbolet indikerer, at der er flanketrigning, eller kant-trigning. .

Det, der opnås ved at lave flanke-trigning, er, at der kan sættes flere D-FF's efter hinanden, så der kan konstrueres en skifteregister. Se senere.

ONESHOT:

En Oneshot kan fx laves af flg:



De viste gates kan ikke simuleres. De er ideelle, og giver ikke anledning til propagationdelay.

Der kan fx bruges CD4069 og CD4081!

Pga. propagationdelay i gatene vil der komme en lille kort puls på AND-gatens udgang.

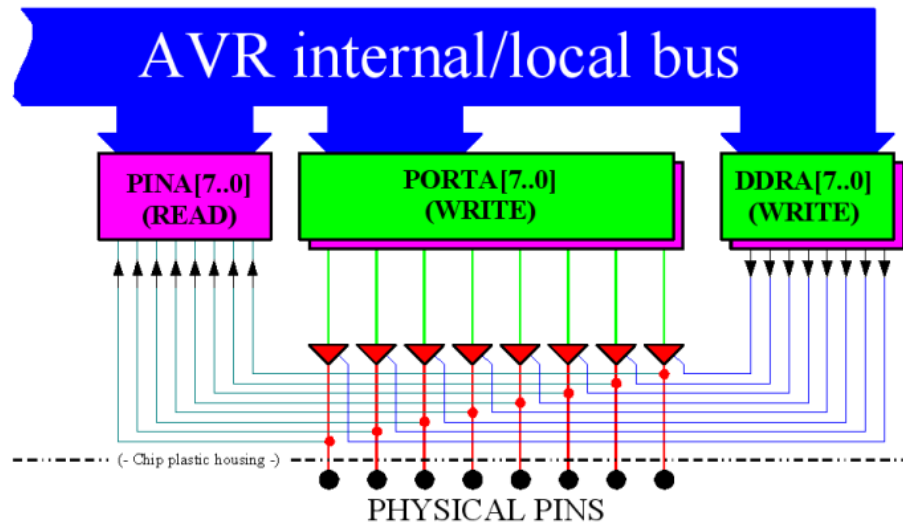


Test ovenstående med WinLogiLab og ORCAD.

Vi skal senere se på Microcontrollere. De har jo inputs og outputs på deres pins.

Her er vist en skitse hvordan de er konfigureret.

Controlleren kan jo fx gøre en pin høj, hvorefter den forbliver høj, indtil den fx resettes til 0 igen.

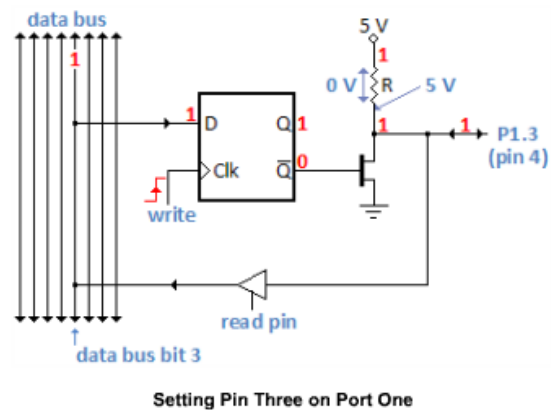


Fra: [http://www.uio.no/studier/emner/matnat/fys/FYS3240/v14/forelesninger/14---uc basic and c basic.pdf](http://www.uio.no/studier/emner/matnat/fys/FYS3240/v14/forelesninger/14---uc%20basic%20and%20c%20basic.pdf)

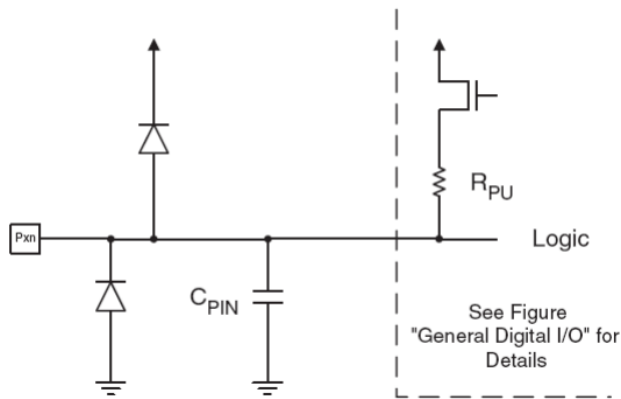
D-FF brugt i en uC-udgang.

Kilde:

<http://www.scribd.com/doc/15924903/EdSim51s-Guide-to-the-8051>



Se på Atmega328P, der bruges i Arduino Uno:



Input kredsløbet.

Bemærk beskyttelsesdioder, der forhindrer at en pin kan trækkes højere end $U_{cc} + 0,7$ dvs. 5,7 Volt.

Eller lavere end $-0,7$ Volt.

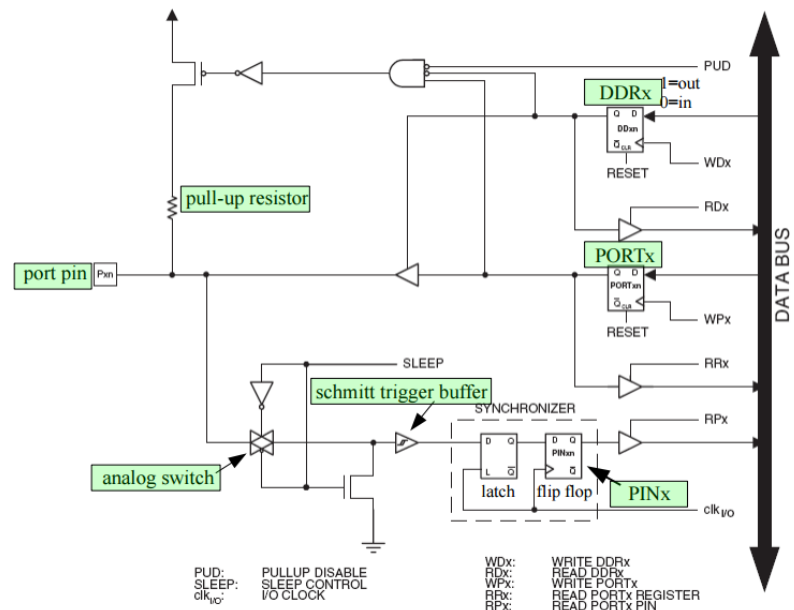
Bemærk også en programmerbar Pull Up-modstand.

Output configuration:

"The pull-up on the port is in the order of 20K-100K"

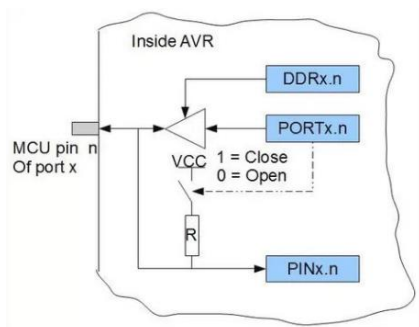
Fra <https://www.arduino.cc/en/Tutorial/DigitalPins>

"The value of this pullup depends on the microcontroller used. On most AVR-based boards, the value is guaranteed to be between 20kΩ and 50kΩ."



Simplet diagram af portpin:

Pin konfigureres vha. SFR, - Special Function Registere, - RAM-adresser.



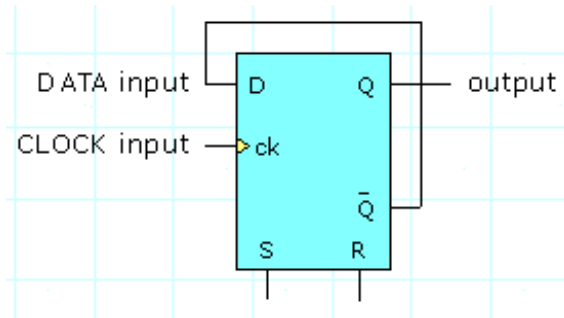
Vha. intern pull up-resistor kan man eksternt nøjes med blot en switch til Gnd. Men man kan med fordel bruge en eksternt modstand alligevel. Hvorfor ??



Toggle-FF lavet med en D-FF

Sådan kan en Toggle FF, eller blot T-FF laves.

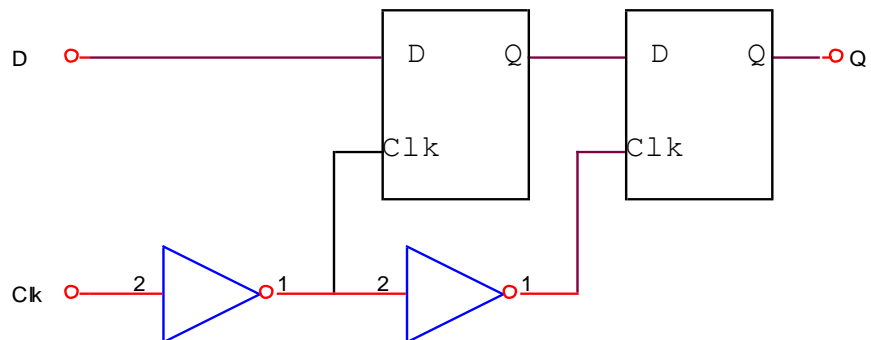
En T-FF, skifter stilling for hver Clockpuls den modtager!



Toggle-FF lavet med en Master-Slave-FF

D-flip flop'ens kanttrigging kan også være konstrueret på følgende vis, vha. en to-sektion flip flop, som også kaldes for master-slave FF.

Positiv kant-trigget D-FF bygget med master og slave D-FF.



Eksempler på brug af FF:

Her følger et par eksempler på elektroniske funktioner, hvor der gøres brug af Flip Flops:



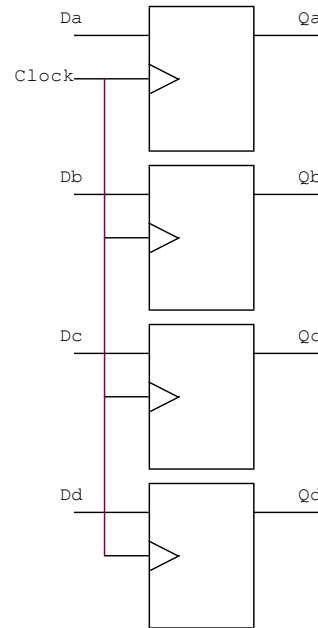
Latch:

Vha D-FF's kan der konstrueres forskellige mere komplekse IC'er. Fx en latch:

Værdierne på indgangene kan overføres og fastholdes på udgangene. Latchen kan enten være udført som flanketrigget eller som niveau-trigget.

En Latch bruges fx til at fastholde værdierne fra en tæller, så man fx kan vise mellemtid på et display! – selvom tælleren kører.

Der skal så ikke anvendes flanketrigning !!



Eksempel på brug af en LATCH.

Latchen består af 4 stk. D-FF. I dette tilfælde skal D-FF'ene ikke være kantriggede.

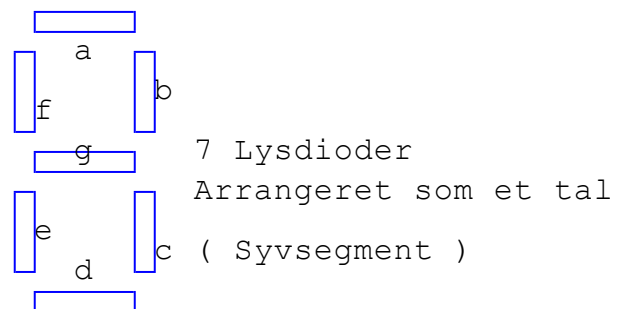
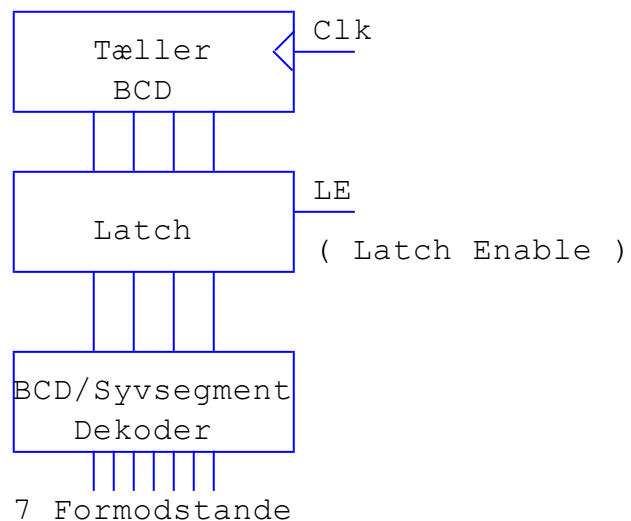
Er Latch Enable (LE) fx lav, går tællerens udgange direkte videre til BCD til 7-segment-dekoderen, der så viser tællerens værdi på et syvsegment.

Er LE høj, fryses visningen på displayet, mens tælleren kan tællere videre!

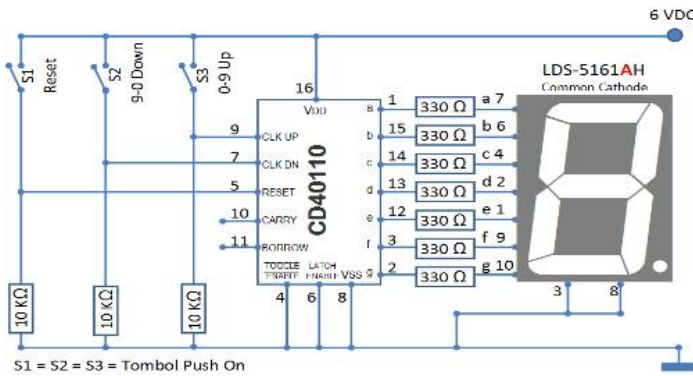
Eksempler:

Tæller: fx 4518,
 Latch: fx 4042
 BCD/7segm: fx 4511

Tælleren 40110 har alle tre blokke indbyggede.



Lav øvelse med 40110. Find opgaven sidst i dokumentet !!

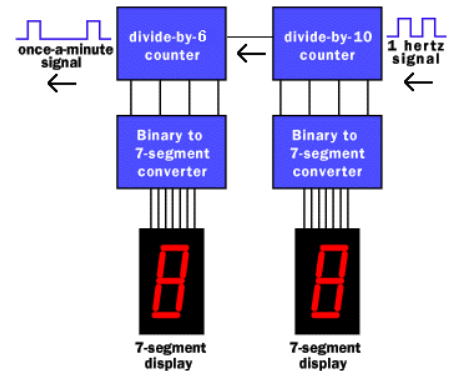


Et eksempel på et 40110-kredsløb !!

Digitale komponenter, fx tællere, der tæller i det binære talsystem.

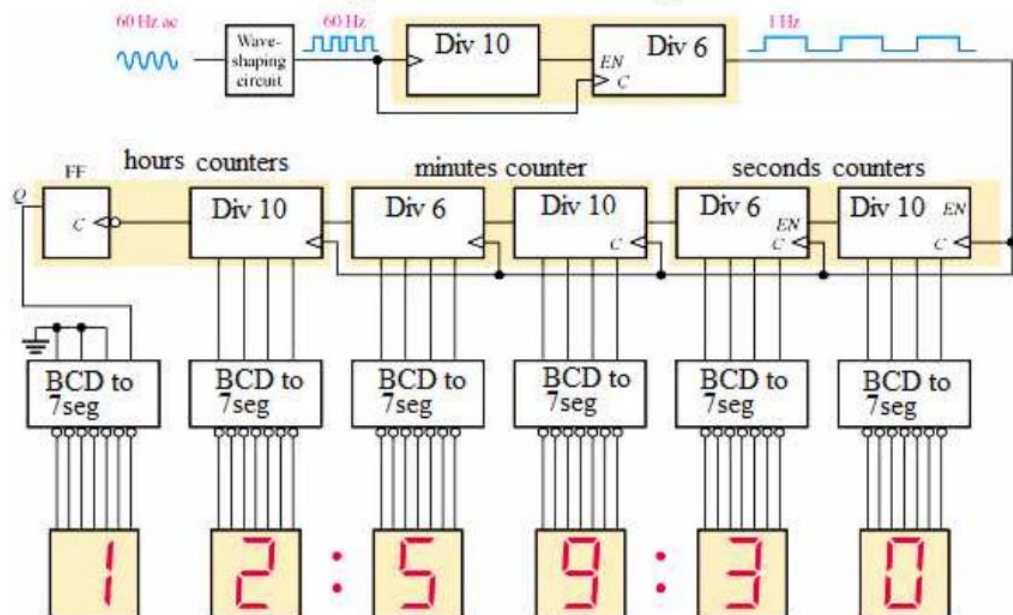
Her vist et tællersystem, der tæller i binære tal. Den øverste højreste tæller tæller fra 0 til 9, og sender så en mente videre til næste tæller. Denne tæller fra 0 til 5.

Ved hjælp af nogle omformere kan man få binære værdier vist som tal – vi kan læse – på nogle displays.

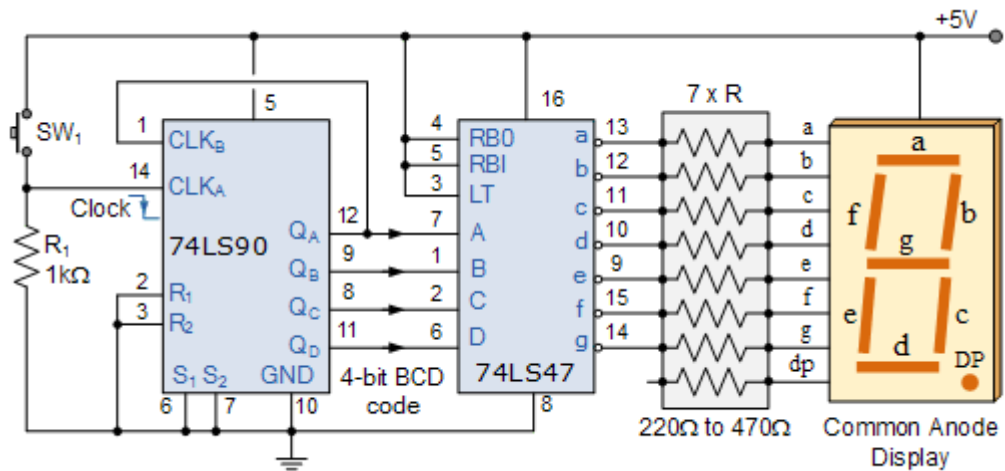


Digital clock block diagram

Eksempel på et ursystem beregnet til 60 Hz net-frekvens.



Fra: <http://www.fatih.edu.tr/~aliadam/EEE122A/EEE122Ch6COUNTERS.pdf>



<http://www.electronics-tutorials.ws/counter/bcd-counter-circuit.html>

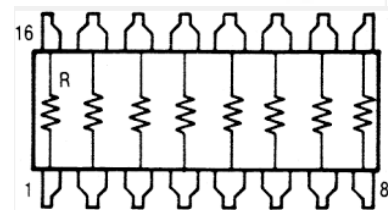
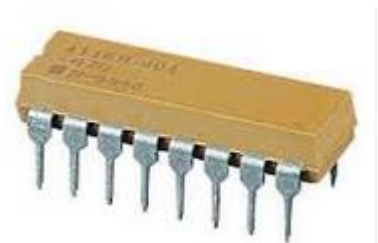
Modstande fås i DIL - IC-huse som vist her.

DIL står for Dual In Line.

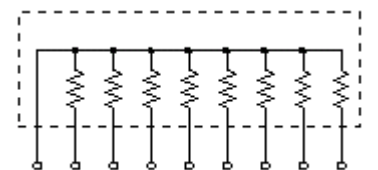
De er typisk størrelses-markeret med nogle tal.

Fx 472

Det betyder 47 og 2 nuller, altså 4700, eller 4,7 Kohm.



Og de fås som SIL, altså Single In Line !!



Kilde: <http://www.talkingelectronics.com/projects/Testing%20Electronic%20Components/TestingComponents.html>

Undersøg tælleren 40110.

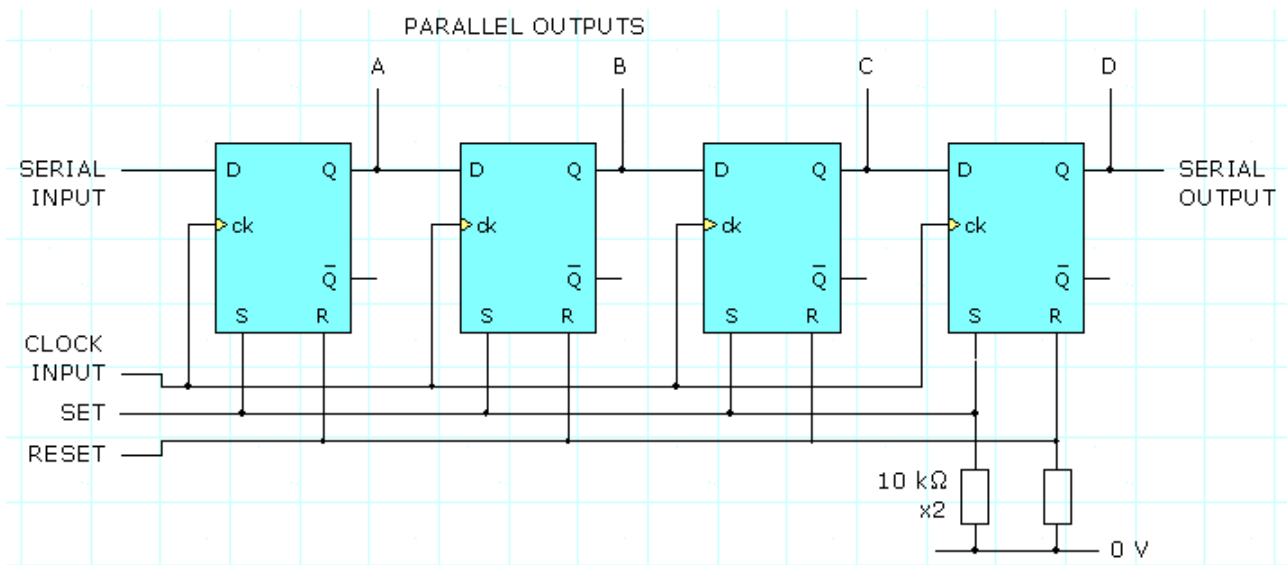
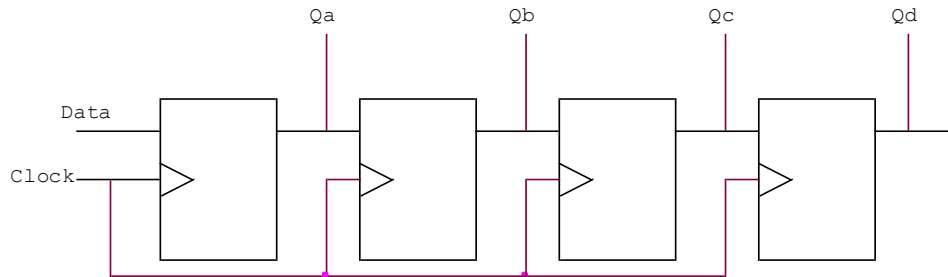
Opbyg fx et kredsløb med 40110 og et 7-segment.

Find flere eksempler, og undersøg deres funktion:

Skifteregister:



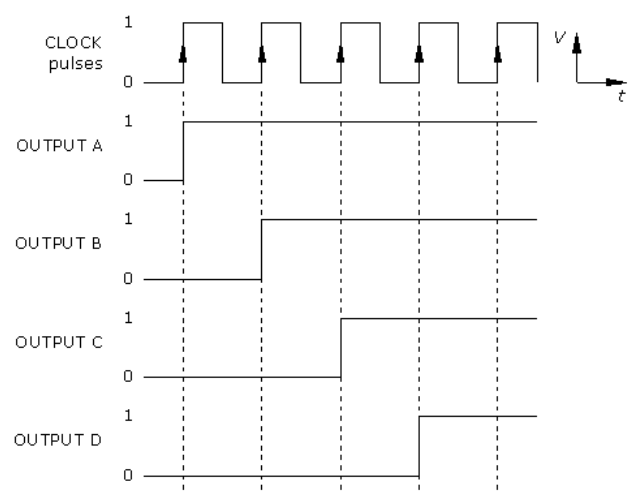
Værdierne på DATA skiftes ind i kredsen hver gang, der kommer en stigende clock-flanke. Er det en række serielle data, der kommer, og det er indrettet således, at der kommer en clock i midten af hver bit, er der konstrueret en serial til Parallel register.



Alle CLOCK-inputs på alle D-FF er koblet sammen. Dvs. at alle FF's smider signalet på "sin" D-indgang over på "sin" udgang. Bitmønstret skifter derfor plads 1 position mod højre. Det sker på Clockens positiv gående flanke, "rising edge" eller "LOW to HIGH transition".

Det tager ganske vist lidt tid at overføre signalet, kendt som propagation delay.

Holdes D- indgangen "SERIAL INPUT" høj, vil situationen kunne illustreres af en pulsplan, eller $U(t)$ graf.

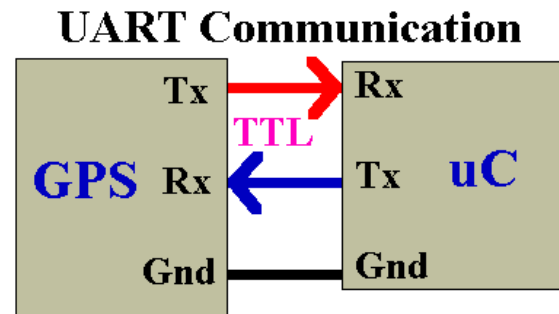




Fra Microcontroller til microcontroller. TxD & RxD

I "vores" microcontroller, fx ATMEGA328, som bruges i Arduino Uno, er der indbygget en **UART**, der står for Universal Asynkron Receiver Transmitter. Denne kan både sende og modtage serielle signaler. Dette betyder, at 2 uC'er kan kobles sammen.

Her et eksempel på kommunikation med en GPS



Data kan sendes både:

Synkron

Der sendes både et signal plus et clock-signal, så modtageren ved, når der skal klockes ind i et skifteregister. Der skal bruges 2 signal-ledninger, - og fælles stel, Gnd, dvs. i alt 3 ledninger.

Asynkron

Data afsendes og kommer til modtageren på et tilfældigt tidspunkt. Derfor skal der være "aftalt" en protokol, dvs. at startbit, stopbit, baudrate osv. skal være kendt.

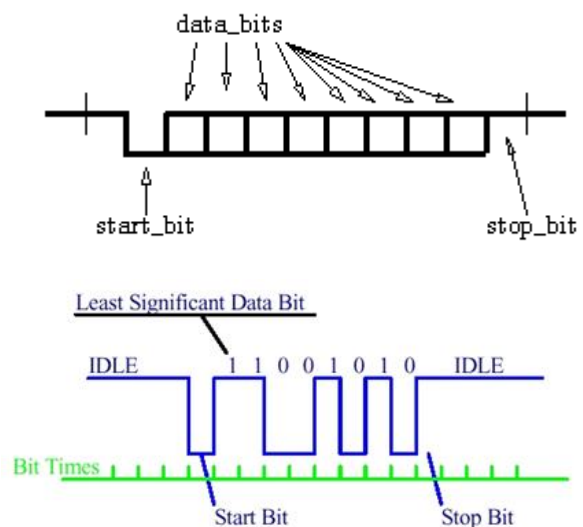
Det er nok med 1 signalledning, plus nul.

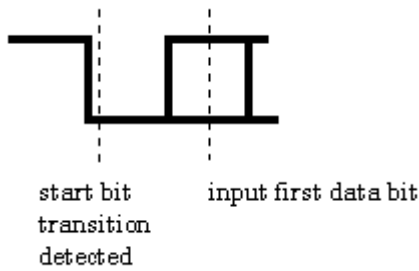
I vores 8-bit uC sendes "pakker" eller bytes asynkront. Der sendes 1 Startbit, 8 databit, og 1 stopbit.

Signaler kan sendes direkte fra uC til uC, dvs. med signalniveauer på 0 og 5 Volt. Tx fra den ene uC skal forbindes til Rx på den anden uC.

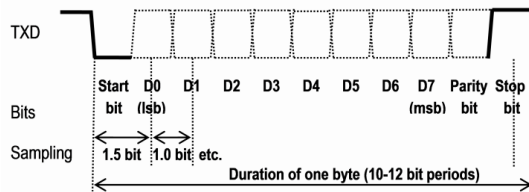
Dette medfører dog ret begrænset rækkevidde pga. evt. støjpåvirkning, der kan medføre at signalet forvanskes så modtageren læser forkert.

Men det kan sagtens bruges til at koble to uC sammen på samme print og over kortere afstande. Fx kan man lade én uC kontrollere et tastatur, og sende de indtastede data til en anden uC via UART'en.

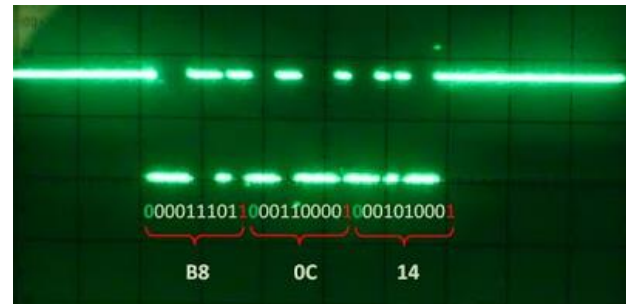




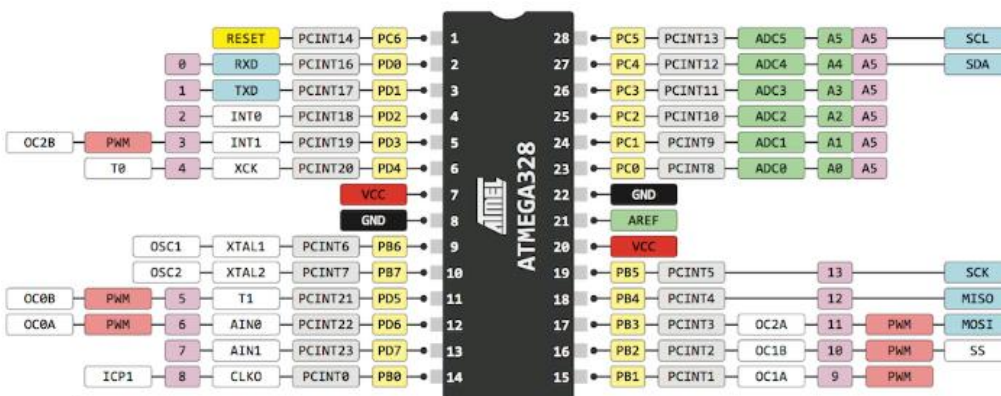
Data skal læses ind i modtagerskifteregisteret midt i den tidsramme, 1 bit varer !!



Til højre ses et eksempel på et scoop-billede af tre byte, sendt direkte fra en uC.



Grønt 0 angiver et startbit, og rødt 1-tal en slutbit.



Atmega328P pinout.

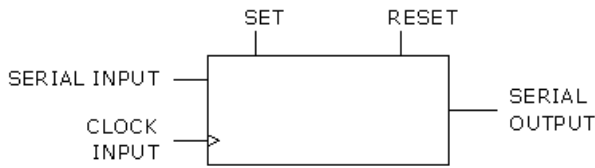
Skifteregister varianter

Følgende skitser viser forskellige typer af skifteregister-konfigurationer.

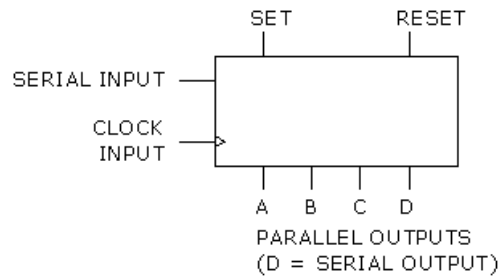
De har navne som: **SISO** (serial in/serial out), **SIPO** (serial in/parallel out), **PISO** (parallel in/serial out) and **PIPO** (parallel in/parallel out). Elektronikken inde i D-FF'ene er ens, men forbindelserne til omverdenen fra kredsens indre er forskellig.



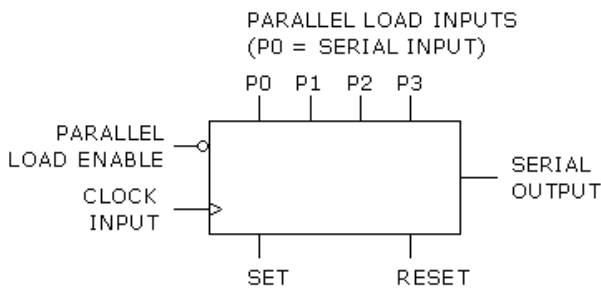
SISO (serial in/serial out) shift register



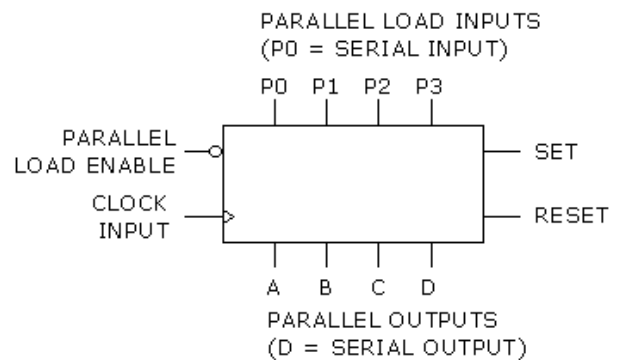
SIPO (serial in/parallel out) shift register



PISO (parallel in/serial out) shift register

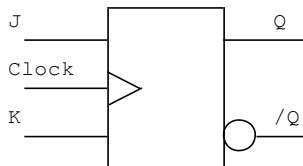


PIPO (parallel in/parallel out) shift register



JK-FF

En JK-FF er en videreudvikling af en SR-FF. Den har yderligere en funktion. Den har ingen udefineret tilstand. Hvis både J og K er "1", vil FF-en Toggle, dvs. skifte tilstand når der kommer et clock-signal.



Sandhedsskemaet:

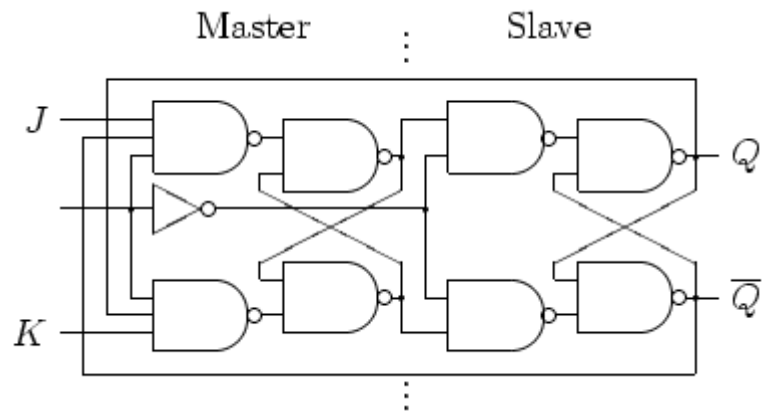
J	K	Clk	Q	/Q
*	*	0	Låst	
0	0	↑	Låst	
0	1	↑	0	1
1	0	↑	1	0
1	1	↑	Toggle	

En JK – FF er fx bygget op som følgende diagram:



Flanketrigning er opnået ved at opbygge FF'en som to FF's efter hinanden. Første halvdel kaldes master, den anden slave.

Opbygning af tæller med JK-FF: Se senere

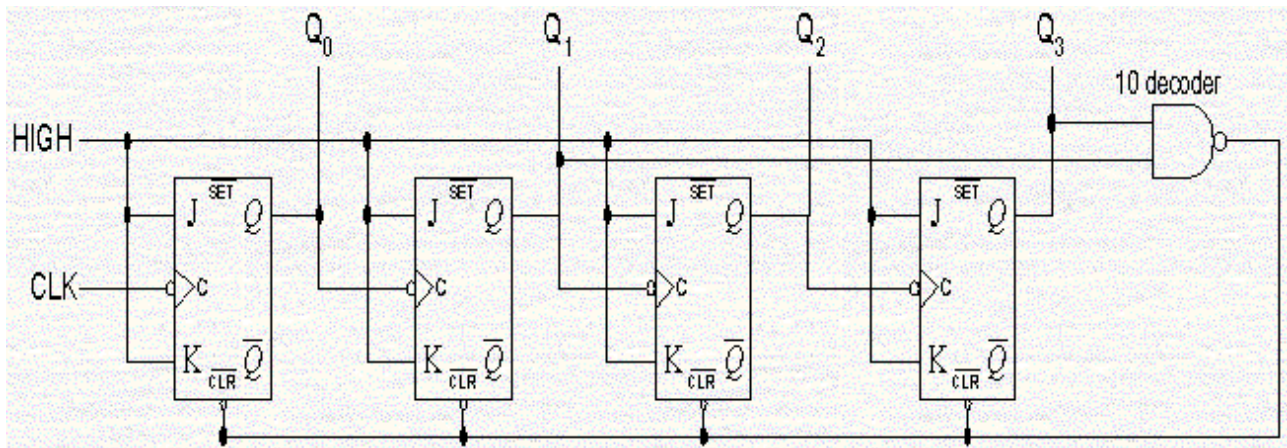


TÆLLERE:

Se link: <http://www.doctronics.co.uk/4060.htm>

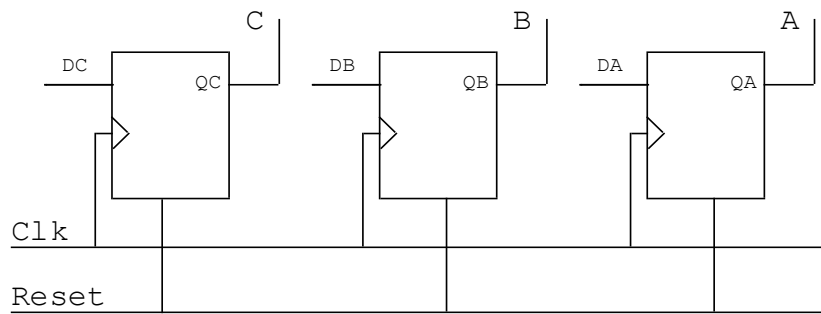
Tællere er opbygget af D-FF eller JK-FF.

Eksempel på opbygning af en tæller, der resettes, når den når 10. Dvs. "10" vil vises et ganske kort øjeblik på udgangen.



Også tællere kan konstrueres af D-FF's. I det følgende beskrives konstruktionen af en 3-bit tæller.

Der ønskes konstrueret en tæller, der tæller i Gray-kode



C	B	A	DC	DB	DA
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Sandhedsskemaet viser de værdier, der skal tilføres „D“-indgangen for de tre forskellige D-FF, for at tælleren skifter korrekt næste gang, der kommer en klock-puls.

Det fører til følgende 3 karnaughkort:

C \ B A	00	01	11	10
0	1	1		
1			1	1

$$D_A = BC + \bar{B}\bar{C}$$

C \ B A	00	01	11	10
0		1	1	1
1				1

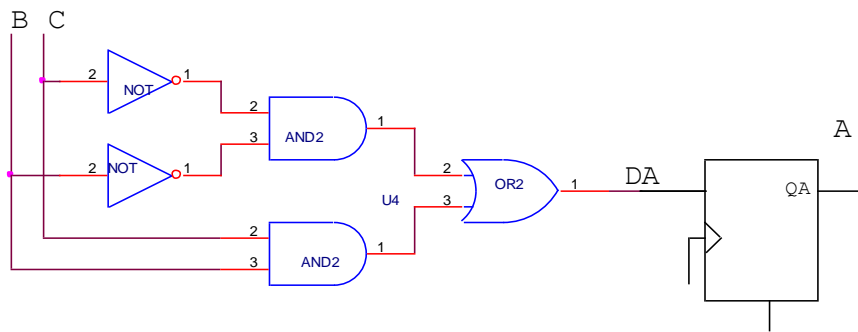
$$D_B = A\bar{C} + \bar{A}B$$

C \ B A	00	01	11	10
0				1
1		1	1	1

$$D_C = \bar{A}B + AC$$

C, B & A er udgangene fra vores tæller, og det er disse variable, der skal føres tilbage og ind i de respektive D-indgange.

Følgende vises kun kredsløbet for D_A



JK-tællere:

Anvendes JK-FF til at opbygge tællere, skal man styre både J og K indgangene for hver udgang. For en 3-bits tæller bliver det: Ja, Ka, Kb, Kc, Kc.

Dvs. hele 6 ligninger skal realiseres på baggrund af de tre udgang, C, B og A.

Men der er dog nogle muligheder for forenkling. Det foregår i karnoughkortene, idet der fx er to muligheder for at tvinge en udgang fx fra nul til én. FF'en kan sættes eller togle.

Analyseres dette nærmere, findes følgende sammenhæng:

Fra		J	K
0 til 1	→	1	*
1 til 0	→	*	1
0 til 0	→	0	*
1 til 1	→	*	0

En stjerne skal opfattes som en "Dont Care". En Dont Care kan medtages i karnoughkortenes sløjfer, eller undlades, efter hvad der er smartest.

Opbygning af JK tæller, fx 3 bit graykode mangler

Synkrone og Asynkrone tællere

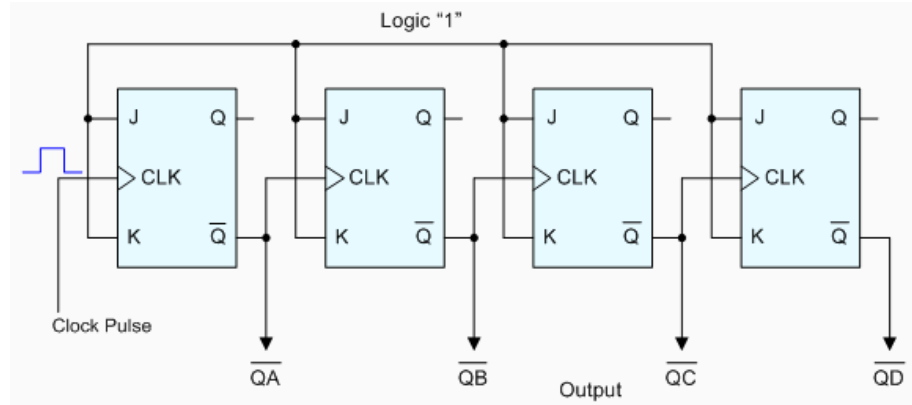
Der findes forskellige typer tællere. Binære, BCD, Johnson- eller Ring-tællere osv. Fælles er, at de er opbygget af gates og FF's.

Asynkron Counter:

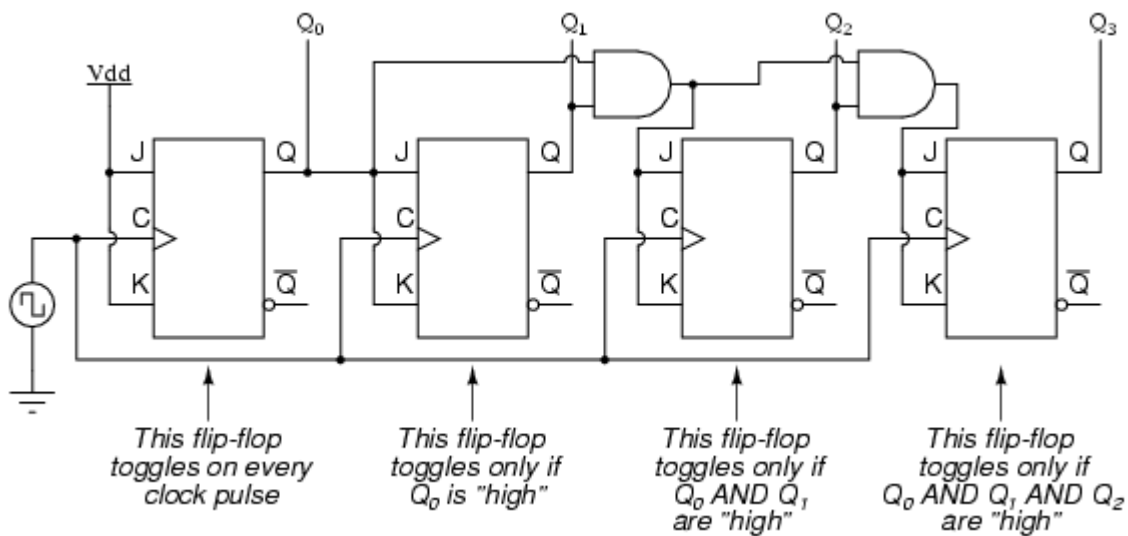


Down-counter eksempel:

Læg mærke til, at clockpulsen ikke kommer til alle FF's samtidig. Derfor en Asynkron tæller.

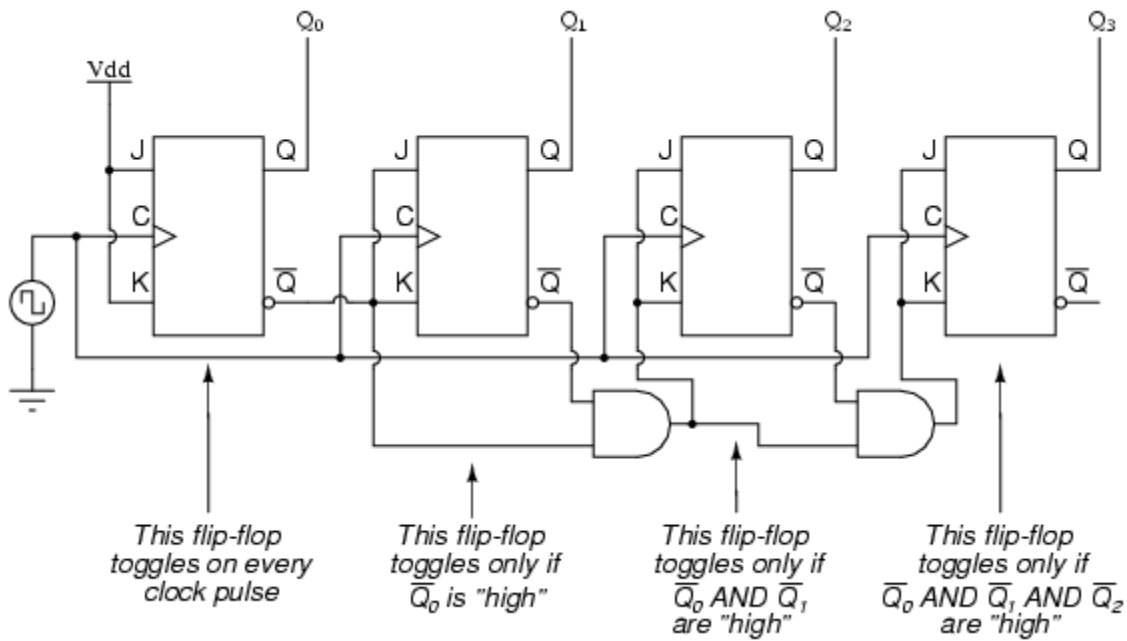


Synkron op-tæller:



Kilde: http://www.allaboutcircuits.com/vol_4/chpt_11/3.html

Synkron Down-counter:



Kilde: http://www.allaboutcircuits.com/vol_4/chpt_11/3.

Undersøg kredsløbet på: <http://www.doctrionics.co.uk/4511.htm>

PEEL-kredse / GAL-kredse

PEEL-kredse er eminente til at implementere sådanne tællere. De indeholder en D-FF i udgangen, og der er mulighed for at programmere de tre ligninger ind.

Mht. til peelkredse, se specielt kompendium.

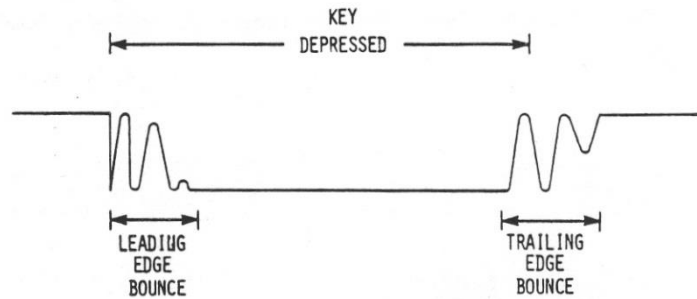
Desværre har PEEL-fabrikanten lukket produktionen. – Erstatningen hedder GAL16V8 !

Kontaktprel-fjerner



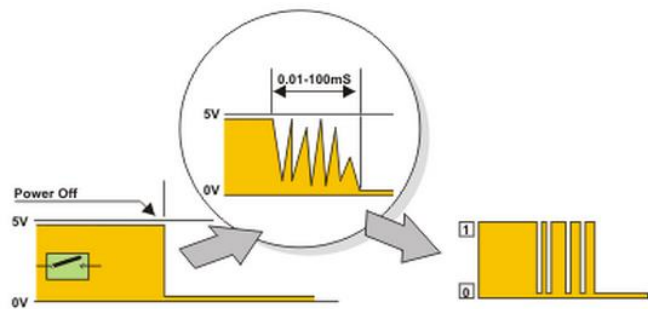
Hver gang, to ledninger forbindes eller en kontakt slutes, vil der altid opstå kontakt-prel. Det kan ikke lade sig gøre at to ledninger, eller kontakter rører hinanden uden at de vil hoppe og danse lidt først inden de lægger sig til hvile i sluttet tilstand. Tilsvarende ved afbrydelse af forbindelsen.

Men vha. lidt ekstra elektronik kan det lade sig gøre at skabe en "ren" puls.

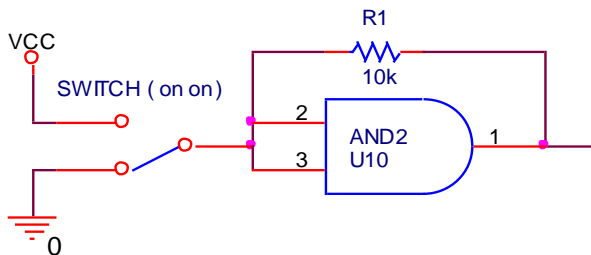


- BOUNCE IS 10-20 MSEC
- HARDWARE SOLUTION: R-C FILTER
- SOFTWARE SOLUTION: VERIFY KEY STATUS FOR 20 MS

Her et andet eksempel, der viser afbrydelse af en kontakt.



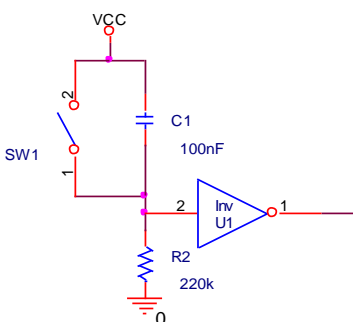
Kilde: http://handsontec.com/8051_Tutor.php



En *virkelig smart* løsning kan realiseres med en AND-gate, men den kræver en skiftekontakt.:

Forklar:

Der findes også andre løsninger:

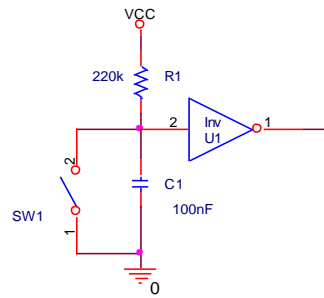


Når kontakten slutes, kortsluttes kondensatoren. Herved bliver spændingen høj på indgangen – første gang, "kontakten danser mod plus".

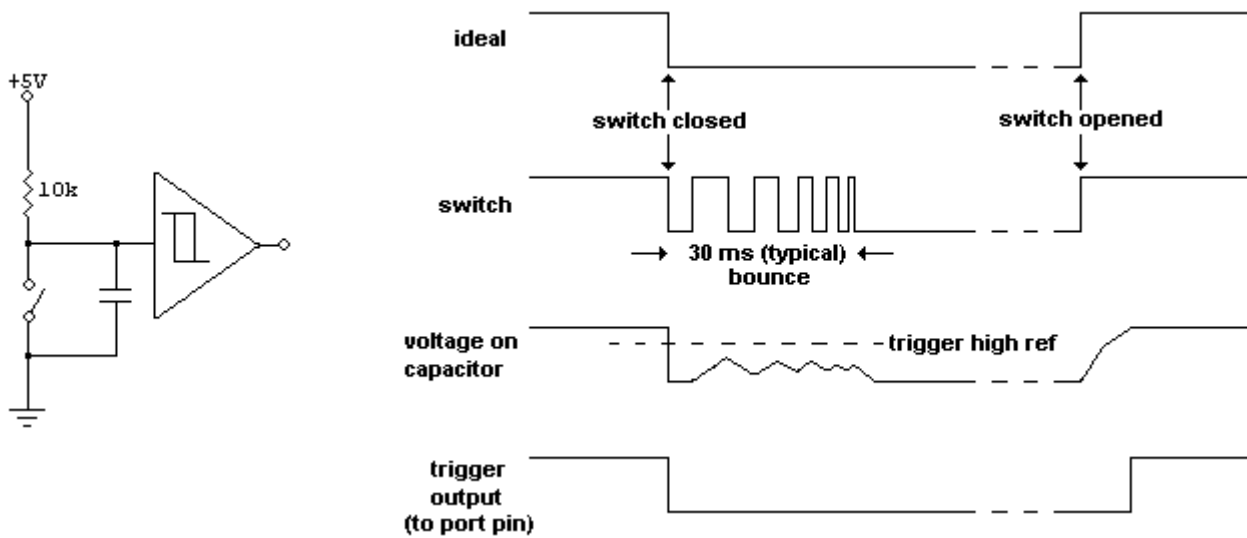
Kontakten når at falde til ro i sluttet tilstand før C1 igen oplades.



Der fås en positiv puls på udgangen, ellers som ovenfor!

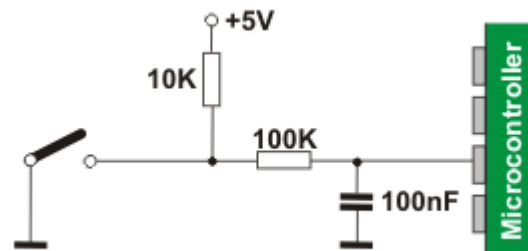


Et eksempel, hvor kondensatorens spænding vises!



<http://www.edsim51.com/8051Notes/interfacing.html#switches>

Her vist en kontaktprel-fjerner der er aktiv lav.



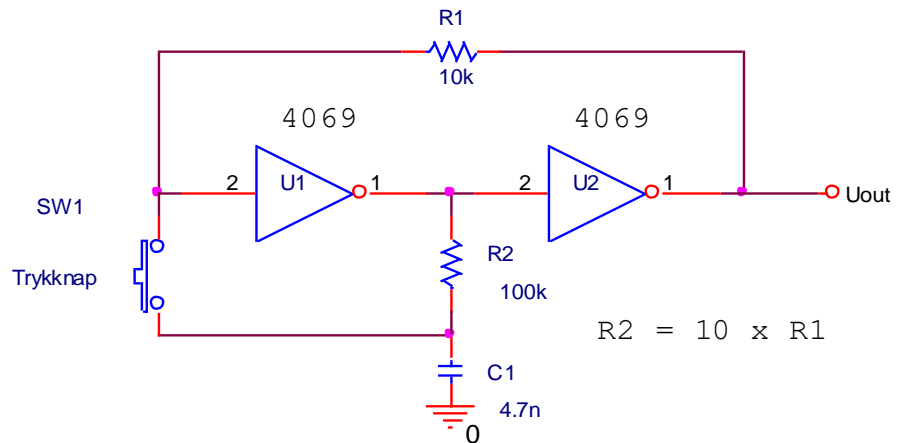
Yderligere findes der Flip Flop-løsninger:

Toggle switch

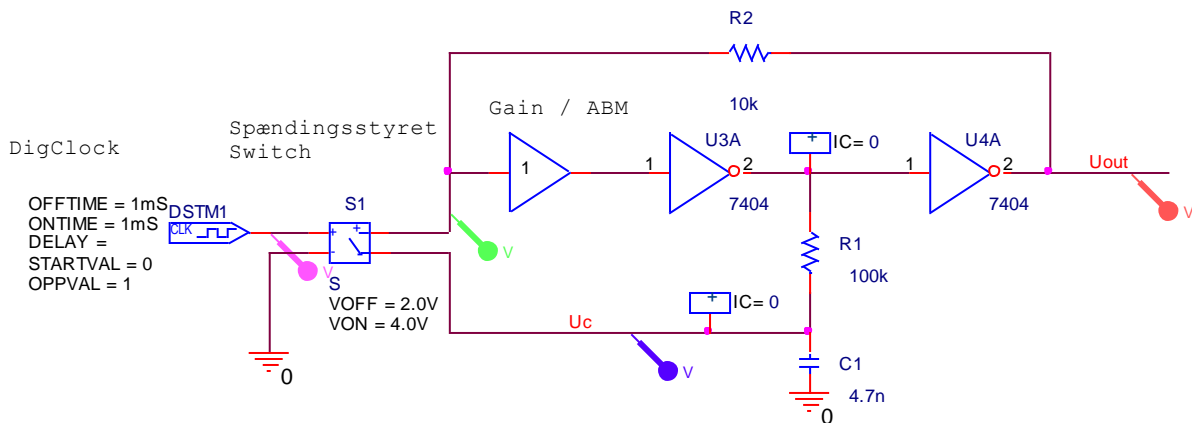


Dette kredsløb er en trykknaps toggle switch!

Den kan bruges som ON/OFF for et kredsløb, idet CMOS kun bruger omkring 20 nA.

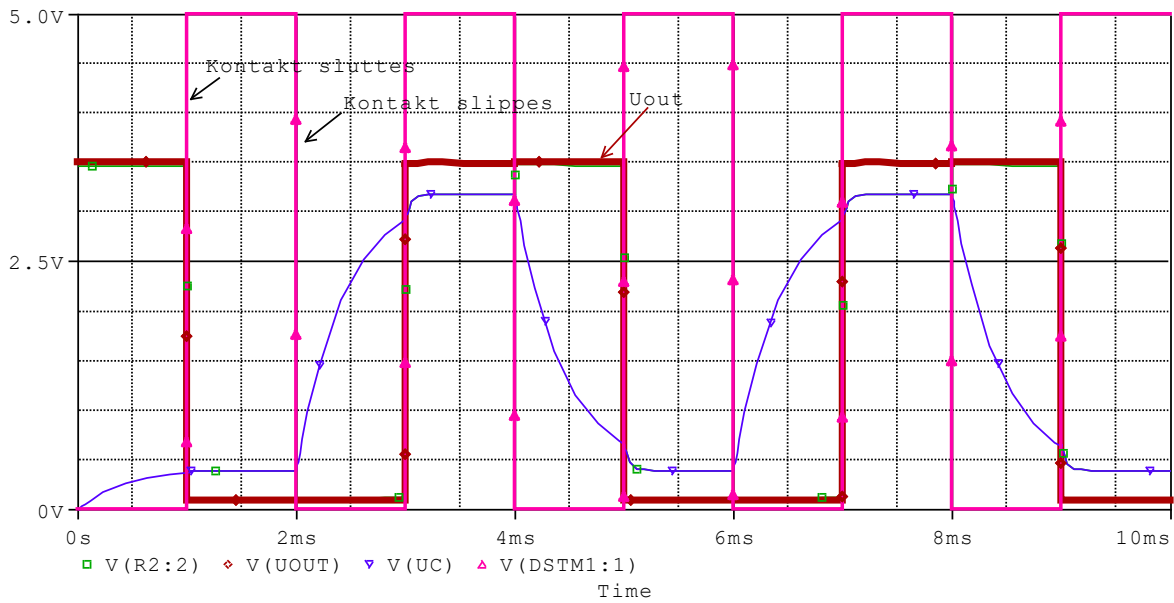


Kredsløbet kan fx simuleres i ORCAD med følgende kredsløb:



Der er anvendt en Digclock til at give pulser til den spændingsstyrede switch. Switchen simulerer en kontakt, der trykkes et antal gange. Forstærkeren Gain, fra /ABM, giver 7404 høj indgangsimpedans. Dvs. ikke belaster kredsløbet. Der skulle egentlig bruges CMOS-kredse !!

Grafen ses herunder:



U_{out} er trukket kraftigt op.

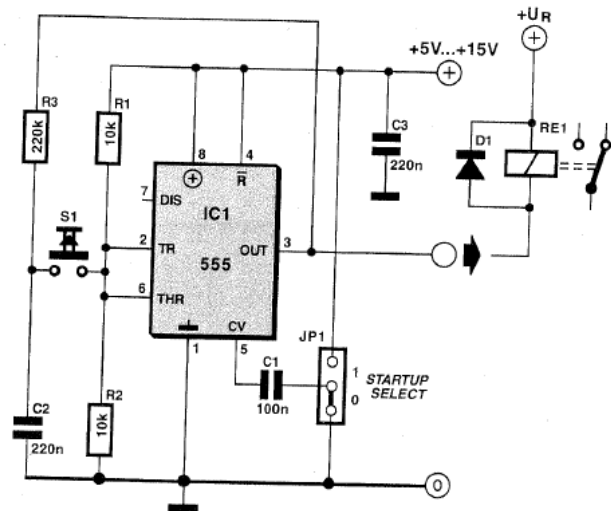
Et andet eksempel, opbygget omkring en 555.

R1 og R2 holder tresholdspændingen på halv forsyningsspænding.

Hvis tresholdspændingen kommer over $2/3$ af U_{cc} , vil udgangen gå Lav.

Kommer $U_{treshold}$ under $1/3 U_{cc}$, går udgangen høj.

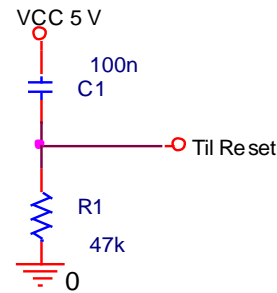
Undersøg IC'en 555.



Power On-Reset:



Til microcontrollere kan fx dette kredsløb bruges: Det giver en høj puls I en periode efter Power On. Perioden kan kort beregnes af $T_{\text{halvliv}} = 2/3 R C$



Reducering af ligninger med analog multiplexer 4051

Hvis følgende ligning er givet,

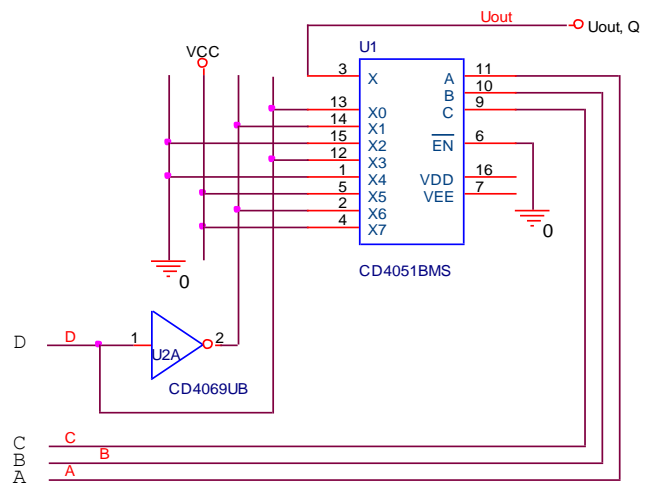
$$F = \overline{A}BCD + \overline{A}BC\overline{D} + ABC + \overline{A}\overline{B}CD + AB\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BC$$

er det måske muligt, at løse eller realisere den med en multiplexer !

Først placeres alle led i et karnaugh-kort, der er lidt anderledes end normalt.

AB CD

	00	01	11	10
00	0	1	0	0
01	0	0	0	1
11	0	1	1	1
10	1	0	1	1



Inden for den første sløjfe (øverste til venstre) ses, at udgangen Q er lige som D. Er D nul, er også udgangen nul. Dvs. at D skal forbindes direkte til indgang 0, eller X0, som den hedder her.

Hvis indgang A, B og C alle er lave, vil X0 være direkte forbundet til udgangen. Q vil altså være rigtig.

Indgang X1 ligger nederst i karnaugh-kortet. Det er indikeret med de små tal forneden i felterne.

Her ses, at udgangen er modsat D. Altså D – inverteret.



Opgave: $F1 = BCD + A\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C + \bar{A}B\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$

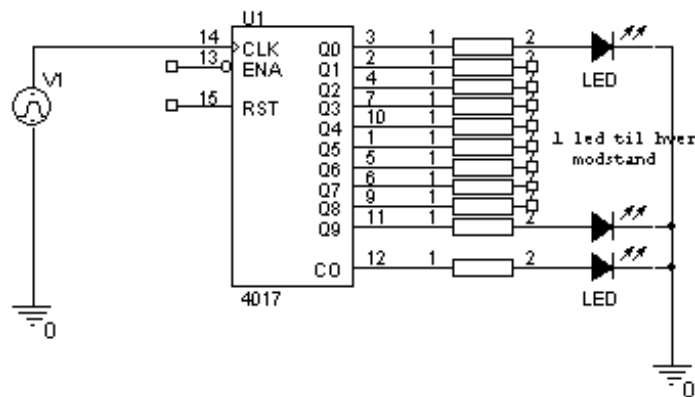
Opgave: $F2 = \overline{\overline{AB} + \overline{BC}} + \bar{D}$



Øvelser:

Se: <http://www.doctronics.co.uk/4017.htm>

TÆLLER 4017



Opbyg viste kredsløb.

Undersøg funktionerne af RST, ENA og CLK. forklar.

Hvad er frekvensen på de forskellige udgange ??? - Og Duty Cyclen ???

I databladet står, at tælleren kan bruges til "DIVIDE BY N" counter, hvor $1 \leq N \leq 10$.

Hvordan forbindes tælleren så der kan måles en syvendedel af clocksignalet på udgangen. Hvilken udgang kan bruges ? Og hvad er Duty Cyclen på udgangene ??

Undersøg hvordan der kan laves et længere løbelys med flere 4017 !!

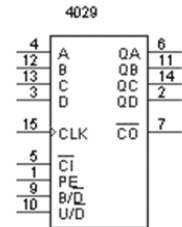


TÆLLER 4029

Undersøg databladet for 4029.

Monter lysdioder på udgangene Q_A, Q_B, Q_C, Q_D og CO. Husk formodstand.

Sæt PE til stel, og undersøg nu og forklar funktionen af de andre styre-indgange. (CLK, B/D, og U/D) Indgangene A til D behøver ikke at være forbundet til noget endnu.



Forbind, så 4029 tæller binært, og opad. Stil om så tælleren tæller BCD. Undersøg !

PE kaldes også Preset Enable. Preset betyder her at sætte nogle udgange til en bestemt værdi. Denne værdi kan sættes på indgangene A til D. Hvis disse alle er sat til 0, hvordan virker så PE ???

Forklar ud fra datablad funktionen af Carry In.

Til højre herfor er vist sandhedsskemaet, hvis tælleren tæller binært. Der ønskes konstrueret et system, der får tælleren og lysdioderne til at tælle op til 12, hvorefter der startes forfra med værdien 5.

Byg og forklar !

Power on-reset. Når en tæller får forsyningsspænding, ved man ikke, hvor den er i sin tællecyklus. Ikke før man fx. resetter den ! Overvej, hvordan der kan laves automatisk "Power on reset". Dvs. her er det Power on Preset til værdien 5.

D	C	B	A	Værdi
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

Forbind Presetindgangene P_A til P_D til 0 Volt. Forbind O_B til Preset. Mål varigheden af "Preset-pulsen". Vælg frekvens, så pulsen kan ses! Forstør evt. pulsen 10 gange. Angiv dens varighed, og forklar.

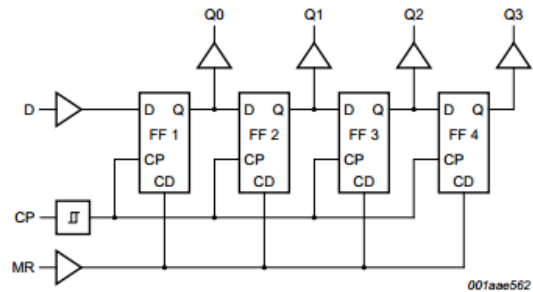
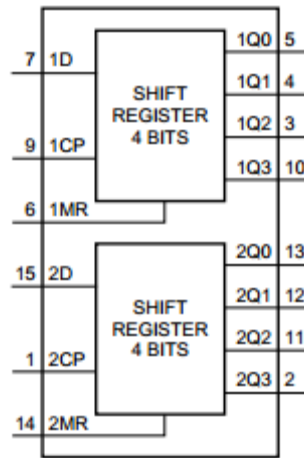


SKIFTEREGISTER 4015

Undersøg databladet for 4015 skifteregister

4015 indeholder 2 stk. 4 bit skifteregistre.

Brug først den ene.



IC'en 4015 monteres på fumlebrættet. Sæt 10 kOhm pulldown modstande på D, Clk, og Reset

Sæt lysdioder med formodstande på de 4 udgange, så deres spænding kan ses.

Med en kontakt, der blot er en ledning, gives nu nogle pulser på clock-indgangen. Ændre spændingen på D-indgangen fra 0 til 1, og giv flere Clock-pulser. Observer, hvad der sker på udgangene.

Observer, at der er kontaktprel på clock-signalet. Forklar hvad det er.

Monter en kontaktprelfjerner på klockindgangen, og se forskellen. Brug fx en andgate-type!

Monter nu den anden 4-bit skifteregister i IC-en, så de to tilsammen udgør en 8-bit-skifteregister, og test igen, nu med 8 lysdioder. Overvej grundigt, hvordan det gøres!

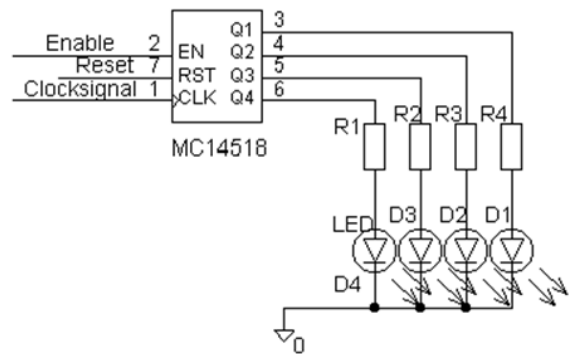
I stedet for manuelle pulser – og kontaktprelfjerner – tages klockpulserne nu fra tonegeneratoren. Indstil den først vha. scoopet til at give korrekte firkant-pulser fra 0 Volt til samme spænding, som er valgt til 4015. Indstil fx til en frekvens på 1 Hz. Hvad sker, hvis D-indgangens spænding varieres mellem 0 og 1?



4518 / 4520

Undersøg først tælleren 4518 og 4520 ud fra datablad. Forklar forskellene. Hvad er Binær kode og BCD-kode?

Byg først op som 4 bit binær tæller, hvor udgangenes status indikeres ved hjælp af lysdioder med formodstande. Forklar hvorfor der skal være formodstande!



Undersøg, hvordan clocksignalet ledes ind til selve IC-en gennem clock og enable-indgangene.

Husk 10 K Pull Down modstande på indgange, så de ikke svæver.

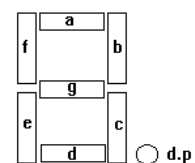
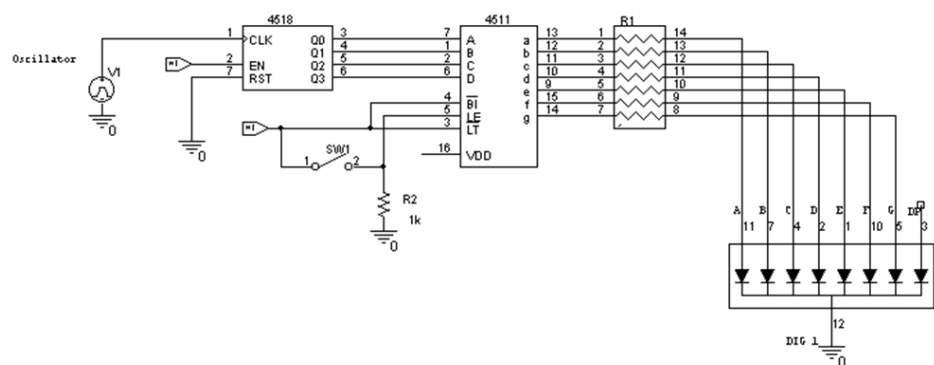
Brug først en løs ledning som clock-kontakt indtil det virker. Dvs. 50 Hz – hvis der ikke er Pull Down modstand på. Observer kontaktprel.

Opbyg evt. en Nandgate-oscillator, eller indstil tonegeneratoren til firkant puls, fra 0 til Ucc. (Ucc er den spænding, der bruges som forsyningspænding, fx 5 eller 12 Volt)

Find et 7-segment med Fælles Katode. Brug en 1K modstand til at teste, og lave et ”datablad” for den. Hvad er forskellen mellem Common Catode og Common Anode?

Undersøg IC'en 4511. Hvad gør indgangen Latch Enable, LE? Undersøg dernæst BI-indput-signalet. – og LT. Se evt: <http://www.doctronics.co.uk/4511.htm>

Monter 4511 mellem tælleren og 7-segmentet. Husk Pull Down modstande på svævende indgange. Husk formodstande før 7-segmetet.



Lav Journal.



TÆLLER 40110

Undersøg databladet for tælleren 40110.

Fastslå hvordan styreindgangene Toggle Enable, Latch-enable og Reset virker. I hvilke situationer skal de være lave eller høje.

Undersøg hvordan et fælles katode 7-segment virker. Lav selv et "datablad" for den. Brug fx. 5 Volt og en 1-Kohm modstand til at begrænse strømmen.

Lav diagram for tællersystemet med terminalnumre og diverse nødvendige data for tælleren 40110 forbundet til et 7-segment. Styreindgangene skal via en "Pull down" eller "Pull up" – modstand være forbundet korrekt. En ledning kan så fungere som kontakt.

Opbyg og test diagrammet på fumlebrædt.

Bemærk kontaktprell. Beskriv begrebet.

Der kan efterfølgende eksperimenteres med en pulsgenerator.

Undersøg om Clock-up, Latch-funktionen, Toggle enable, og Reset – funktionerne virker.

Vis et diagram over hvordan der skal forbindes så der laves en to-cifret tæller "99".

Lav Journal over jeres øvelse!!



Gate og IC-oversigt:

Funktion		HC/ HCT, AC/ACT, VHC/VHCT "74xx"	CMOS 4000B-serien
Gate Buffer	NAND	00, 03, 10, 30, 133	4011B, 4012B, 4023B, 4068B, 4093B, 40107B
	NOR	02, 27, 4002	4000B, 4001B, 4002B, 4025B
	AND	08, 09, 11, 21	4068B, 4073B, 4081B, 4082B
	OR	32, 4072, 4075	4071B, 4072B, 4075B
	EX NOR	266, 7266	4077B
	EX OR	86, 386	4030B, 4070B
	INVERTER	04, 05, 14	4069B, 4502B
	BUFFER	07, 4049, 4050, 7007	4007B, 4009B, 4010B, 4041UB, 4049B, 4050B, 4052B, 40107B
	BUFFER 3-STATE	125, 126, 240, 241, 244, 365, 366, 367, 368, 540, 541	4503B
	BUFFER 3-state, 16-BIT	16240, 16244	
	BIDIREKTIONAL	242, 243, 245, 620, 623, 640, 643, 4245, 3245, C4245, C3245	
	BIDIR, 16 BIT	16245	
	MULTIFUNCTION	51, 4078	4019B, 4030B, 4048B, 4070B, 4077B
SCHMITT TRIGGER	14, 132	4093B, 40106B	
FLIP-FLOP	J_K-FF	73, 76, 107, 109, 112, 113	4027B, 4095B, 4096B
	D-FF	74, 79, 80, 174, 175, 273, 377	4013B, 4076B, 40174B
	3-state	374, 534, 564, 574, 646, 648, 651, 652	
	3-state, 16 BIT	16374	
INTERFACE CIRCUIT			4009UB, 4010UB, 40109B
LATCH		75, 77, 259, 279, 375, 7259	4042B, 4043B, 4044B, 4099B, 4720B
	3-STATE	374, 534, 564, 574, 646, 648, 651, 652	
	16-BIT	16374	
MULTI- VIBRATOR		123, 221, 423, 4538	4047B, 4098B, 4538B
DECODER		42, 131, 137, 138, 139, 154, 155, 237, 238, 4028, 4514, 4515	4028B, 4514B, 4515B, 4555B, 4556B
	7-SEGMENT	4511, 4543	
ENCODER		147, 148	4532B, 40147B
Display Driver			4054B, 4055B, 4056B, 4511B, 4543B
	WITH COUNTER		4026B, 4033B, 40110B
REGISTER		164, 165, 166, 194, 195, 595, 597, 289, 299	4006B, 4014B, 4015B, 4021B, 4031B, 4034B, 4035B, 4017B, 40100B, 4076B, 4099B, 40104B, 40105B, 40108B, 40194B, 40208B
	3-STATE	173, 299, 323, 670, 4094	4094B, 4517B
COUNTER	BINARY	161, 163, 191, 193, 393, 590, 592, 593, 691, 693, 697, 699, 4020, 4024, 4040, 4060, 4520	4017B, 4020B, 4024B, 4039B, 4040B, 4060B, 4516B, 4520B, 4536B, 4541B, 40103B, 40161B, 40163B, 40193B



	DECADE	160, 162, 190, 192, 390, 690, 692, 696, 698, 4518	4029B, 4510B, 40102B, 40160B, 40162B, 40192B
	DIVIDER	292, 4017, 4022, 40102, 40103, 7292, 7294	4017B, 4018B, 4022B, 4024B, 4040B, 4045B, 4060B, 4521B
Multiplexer / Demultiplexer	ANALOG	4016, 4066, 4051, 4052, 4053, 4316, 4351, 4352, 4353	4016B, 4066B, 4051B, 4052B, 4053B, 4067B, 4097B
	DIGITAL	151, 153, 157, 158, 257, 352, 353, 354, 356,	4019B, 4512B, 4519B, 4555B, 4556B, 40257B
SWITCH	ANALOG	66, 67, 384, 719	
Phase Locked LOOP			4046B
OTHERS	ADDER	283	4008B, 4032B, 4038B
	COMPARATOR	85, 688	4030B, 4063B, 4070B, 4077B, 4585B, 40101B,
	ALU / MULTIPLEXER	181, 182	4089B, 4527B, 40181B, 40182B
	Parity Tree	280	