

# Logiske porter

Logiske porter er byggesteinene til nesten alt som finnes av elektronikk. Det er derfor det er så viktig å forstå hvordan de er bygd opp og hvordan de kan brukes til å skape nesten alt vi bruker av elektronikk i dag.

For å forstå oppbygningen av de logiske portene må vi først se på hva de er bygd opp av, nemlig transistoren. Transistoren er komponent med to hovedbruksområder, den kan være en forsterker av signaler eller en bryter uten noen bevegelige deler. Det er den siste funksjonen som gjør den ideell til bruk i logiske porter og boolsk algebra. Transistoren er bygd opp av to dioder som er satt sammen. Så nok en gang må vi forstå dioden før vi kan fortsette videre med transistoren.

## Dioden

Dioden er lagd av et halvledermateriale, som oftest silisium. Et halvledermateriale som for eksempel silikon har 4 elektroner i sitt ytterste skal, dette gjør at den leder strøm, men langt dårligere enn andre materialer med færre elektroner i sitt ytterste skall. Et eksempel på dette er kobber som brukes i stort sett alt av elektriske ledere som ikke krever en spesiell funksjon annet enn å lede, som silikon har. En halvleder leder fortsatt bedre enn andre materialer som for eksempel glass og plast, som ikke har som funksjon å lede.

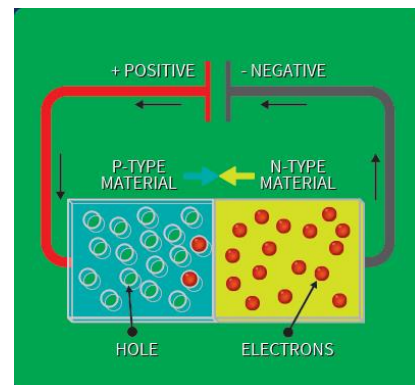
En diode har to sider med silisium (Si, atomnummer 14), hver av de to sidene blir «dopet» av stoffer som endrer sammensetningen av stoffet. Prosessen å dope går ut på å tilsette små mengder av et annet stoff for å endre funksjonen til det originale stoffet. De to sidene av en diode er dopet med henholdsvis bor (B, atomnummer 5) og fosfor (P, atomnummer 15). Fosfor har

kun 3 elektroner i sitt ytterste skal, så når det blir tilsatt i silisiumet, blir det ledige flere ledige elektronhull. Disse hullene blir kontinuerlig fylt av elektroner fra silisium atomene, dette skaper nye hull. Stoffet er nå positivt ladet, det er et overtall av elektronhull, dette er kalt P-doping. Derimot når små doser av fosfor, som har et elektron mer i sitt ytterste skal enn silisium, blir stoffet negativt ladet. Fordi det nå er et flere frie elektroner i stoffet enn det var tidligere.

Dette kalles en N-doping.

Oppbygningen av en diode er som vist på bildet til høyre. Grunnen til at dioden er så spesiell er at den kun leder en vei.

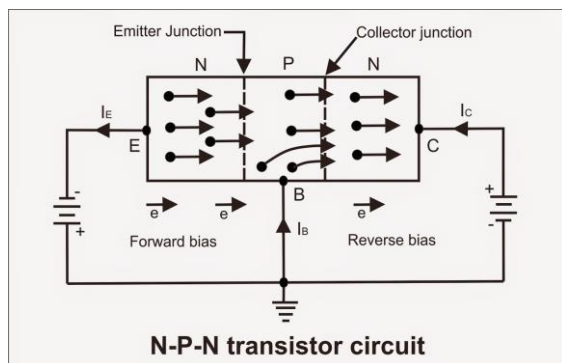
Som vist på bildet vil dioden lede strøm fordi det er en konstant strøm av elektroner som går inn på N-siden av dioden. De elektronene fyller hullene i P-siden og går videre derfra til den positive siden av batteriet. Fra den positive siden av batteriet kommer det konstant nye hull inn på P-siden av dioden, denne prosessen og koblingen gjør at dioden leder strøm. Om man skulle koblet dioden motsatt vei vil den ikke lede strøm, den vil sperre for gjennomgang av noen elektroner. Det som vil skje er at N-siden med overskuddselektronene vil bli tiltrukket den positive siden av batteriet, siden den er positivt ladet. Negativt blir tiltrukket av positivt fordi elektronene kan fylle hullene i det positive stoffet lett, og motsatt med positivt. Samtidig som N-siden blir tiltrukket den positive siden av batteriet vil hullene i P-siden bli trukket mot den negative siden av batteriet. Siden den siden av batteriet har elektroner som lett fyller hullene i det ytterste skallet. Grunnen til at de to sidene av dioden ikke automatisk utfyller hverandre, siden den ene har overskudd og den andre har underskudd, er en nøytralsone mellom de to sidene. Denne oppstår ved at de nærmeste



elektronene fyller de nærmeste hullene mellom hver side. Dette skaper en barriere med materiale som ikke har noen ladning. Denne kalles for ledebarrieren i en diode. Barrieren fører til at elektroner som vil fylle hullene på andre siden må ha et høyere energi nivå. De må på en måte bli dyttet gjennom denne barrieren. Hvis det ikke er koblet til noe batteri eller lignende er denne dytten ikke til stede. Så det ender med de «dopede» sidene av dioden og barrieren i mellom sidene.

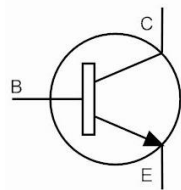
## Transistoren

En transistor som vist på bildet under består



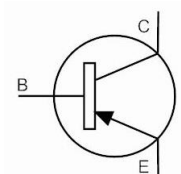
av to slike dioder, også kalt PN dioder.

I denne transistoren er diodene satt sammen slik at det er P-siden som er mellom to N-sider. Dette er en såkalt NPN transistor, den har 3 kontakte, Kollektor, Base og Emmitter. En NPN leder strøm fra Kollektoren og Basen til Emmitteren. Symbolet for en NPN er vist til høyre. En fin huskeregel for å skille mellom NPN og PNP er Nåla Peker Ned, som vist i symbolet så går pilen nedover fra basen.



**n-p-n transistor**

PNP er den andre transistoren som finnes, den fungerer mye på samme måte, men leder strøm fra Emmitteren til Kollektoren. Symbolet for PNP er til høyre. Denne er ikke like mye brukt som NPN transistoren, spesielt



**p-n-p transistor**

ikke når det kommer til logiske porter, derfor blir den bare navnt og glemt nå.

Tilbake til transistorens virkemåte.

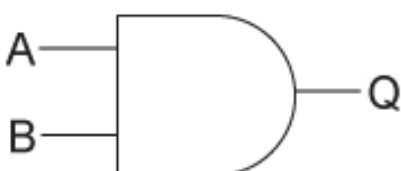
Transistorens funksjon som forsterker og bryter er mye det samme konseptet. På en transistorkobling vil hoved strømmen være koblet på Kollektoren, mens det går en liten strøm inn på basen. Når det er koblet slik vil strømmen fra basen bryte den barrieren mellom de ladde sidene i transistoren, strømmen som er koblet til Kollektoren vil sammen med den fra Basen gå ut mot Emmitteren og videre i kretsen. Det som skjer i transistoren er at det ligger en barriere mellom de to N-sidene av transistoren. Grunnen til at en transistor ikke leder selv om det er koblet strøm på både kollektoren og emitteren, er at barrieren ikke blir brutt selv om det blir press på begge sider. Når det kommer positiv ladning inn fra kollektoren vil det trekke til seg de frie elektronene i N-siden ved kollektoren, det vil bare øke barrieren mellom P- og N-siden. Så selv om det kommer nok elektroner inn fra Emmitteren til å bryte barrieren på den siden, vil det aldri være nok til å bryte barrieren ved Kollektoren. For at elektronene skal kunne flytte seg fra Emmitter til Kollektor må det komme en tilførsel av hull til basen. Disse hullene vil la elektronene flytte fritt fra Emmitter til Kollektor. Transistoren leder strøm fra Kollektor og Base til Emmitter. Måten dette kan brukes til forsterkning er ved å variere strømmen som går inn til basen, den strømmen som kommer inn i basen er proporsjonal med spenningen transistoren leder ut gjennom Emmitteren. Så ved en varierende lav strøm, kommer det en høyere strøm med samme variasjon ut fra Emmitteren.

## Forskjellige porter

Når det kommer til logiske porter og binær matematik, går det kun på om noe er av eller på, ting er enten 1 eller 0. Måten det gjøres er ved å bruke transistoren som en bryter. Ved å sette en lav nok strøm inn på basen, til at transistoren begynner å lede, vil man kunne variere kun mellom høy og lav ut fra Emitteren. En høy er lik 1 og gir ut samme spenning som er koblet til Kollektoren. En lav, er lik 0 og tilsier at transistoren ikke leder noen strøm.

Videre nå til hvordan å bruke transistoren til å bygge logiske porter. Starter med å gå over portenes funksjon og virkemåte. Først er ikke-porten. Den er den enkleste av alle, den gir ut det motsatte av signalet den får inn. Om den får en høy inn, vil den gi ut lavt. Den må kobles til en driftspenning som alle logiske porter. Transistoren brukes som en bryter, inn på basen, så alle krever +5V for å kunne gi et signal ut som enten er høyt eller lavt (1 eller 0). Etter ikke-porten har vi tre brytere som har lignende funksjon. Det er portene som heter OG-porten og NOG-porten. Disse er alle en form for port som krever at begge bryterene er på for å gi verdier. OG-porten er den første, den er bygd opp slik at den kun gir en høy verdi ut om begge verdiene inn er høye. Verdiene inn er ofte referert til etter alfabetet. Den første verdien inn på porten er A, den andre er B også videre. Videre vil vi kun bruke porter med to inganger, A og B, der A er den første inn. Utgangen er ofte markert med Q, eller output (engelsk). Vi kommer til å bruke Q.

Så i en OG-port er ikke utgangen, Q, høy med mindre A OG B er høye. I en NOG-port derimot får vi motsatt verdi ut på Q som vi hadde fått fra OG-porten. En NOG-port gir ut samme verdi som en OG-port, bare negert, motsatt, som vi så på IKKE-porten.



Figur 1: OG-port, Amerikansk symbol

Det er 3 porter til vi skal nevne, igjen går de under samme kategori, der er ELLER-porten, NELLER-porten og XELLER-porten. I en ELLER-port får man høy ut på Q så lenge enten A ELLER B er høye. Så lenge en verdi inn er høy, gir den høy ut. En NELLER-port er nok en gang det samme som NELLER-porten bare negert. Den gir høy ut kun når ingen av ingangene er høye. Den siste porten er XELLER-porten. Siden det er to inngangen på en slik port, vil man ha 4 forskjellige variasjoner på inngangene. Det er Høy og Høy, Lav og Høy, Høy og Lav, og at begge er Lave. Den første muligheten er den som skaper problemer for en port som skal gi høy når A eller B er høy. Sånn den er bygd opp vil den gi høy selv om begge er høye, istedenfor å ikke bruke ELLER-porten slik den var, kom porten som kalles XELLER. Den er eksklusivt høy når enten A eller B er høye. Den gir ut lav, både når begge er høye og når begge er lave.

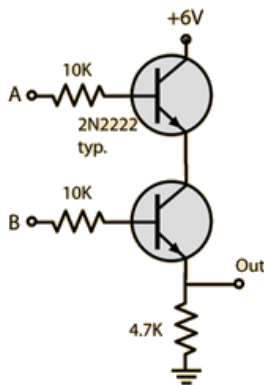


Figur 2: ELLER-port, Amerikansk symbol

## Oppbygning

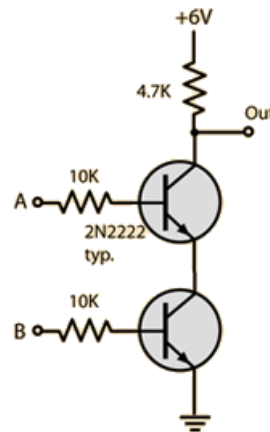
Videre når vi kommer til lagring og forskjellige tellere, kommer vi hovedsakelig til å bruke 4 porter. Derfor vil kun oppbygningen til disse fire portene bli beskrevet. De 4 er OG-, NOG-, ELLER, og NELLER-porter. For å bygge opp disse brukes det en transistor for hver inngang, der verdien inn på basen av transistoren er gitt av inngangene A eller B. Resistansene er for å senke spenningen slik at det kun er nok spenning til å få transistoren til å lede som kommer til basen.

### OG-Porten



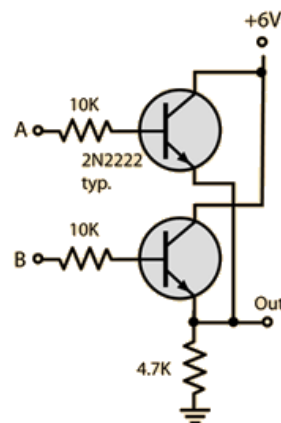
Bildet over viser oppkoblingen med transistorer for at den skal gi de karakteristiske ut-verdiene som en OG-port. Det er kun når A og B er høy, at begge transistorene leder og man får en høy verdi ut på utgangen. Dette er en typisk seriekobling, hvis kabelen er brutt et sted, leder ikke kretsen.

### NOG-porten



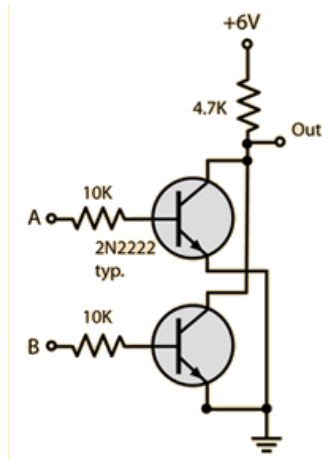
Over er oppkoblingen for en NOG-port vist, den er mye den samme som en OG-port. Den ene forskjellen er det at utgangen for signalet er flyttet opp over transistorene. Denne gir høy ut så lenge ikke både A og B er høy. Eneste gangen utgangen er lav, er når både A og B er høye, da går strømmen ned og ut gjennom jord.

### ELLER-porten



Over er koblingen for en ELLER-port. Så lenge enten A eller B er høy så gir den høy ut. Den er også som sagt høy om både A og B er høy. Den eneste gangen denne gir lav ut er når verken A eller B er høye. Denne kan minne veldig om en parallellkobling, hvis kun en av kablene svikter, vil den fortsatt lede.

## NELLER-porten

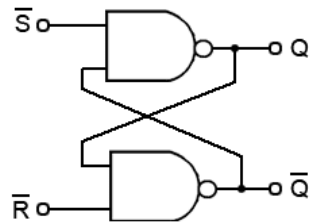


Over er den siste porten, en NELLER-port. Den er som NOG-porten, utgangen er flyttet over transistorene. Den leder kun ut høy om ingen av inngangene er høy, men om en av ingen av utgangene skulle bli høye, går strømmen ned til jord og porten gir ut lav.

## Lagring

Nå som vi er kjent med funksjonen og virkemåten til logiske porter, er det på tide å gjøre et lite dykk inn i hvilken bruk de har når det kommer til lagring av verdier, og hvordan den funksjonen har tilknytning til stort sett alt av teknologi i verden rundt oss. Lagring av binære verdier i sin enkleste form skjer i noe som kalles en «flip-flop».

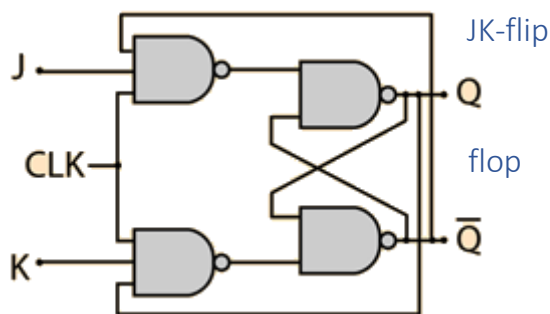
## SR-flip flop



$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$	Comments
0	1	1	0	Q is set to 1 by 0 on $\bar{S}$
1	1	1	0	No change, (1 on Q is remembered)
1	0	0	1	Q is reset to 0 by 0 on $\bar{R}$
1	1	0	1	No change, (0 on Q is remembered)
0	0	1	1	Both inputs at 0 – both outputs are at 1 (Non-allowed state)
1	1	?	?	Inputs change from 0,0 to 1,1 together - outputs will be INDETERMINATE

Som bildet over viser, har denne enkle kretsen evnen til å lagre verdier. Den kalles en SR flip flop, bakgrunnen for navnet er at den har en Set, S, funksjon, den øverste av inngangene og en Reset, R, funksjon, den nederste av inngangene. Måten NOG portene er koblet med utgangene tilbake til inngangen av den motsatte porten er det som gjør den spesiell. Om en av inngangene er lave vil, som vist i tabellen under, den korresponderende verdien for utgangen gå høy. Q er hoved utgangen, den blir satt av Set verdien. Mens den negerte verdien for Q, blir endret ettersom R blir endret.

Problemet med denne koblingen er at den har to inngangskombinasjoner som ikke er mulige. Ved begge inngangene lave, vil begge utgangene være høye, noe som ikke er mulig, det er en ikke-tillat tilstand. Både Q og negert Q kan ikke har samme verdi. Den andre er om begge inngangene er høye, det skaper ett problem for koblingen og timing i systemet. Siden begge inngangene er høye samtidig vil de begge gjøre at NOG-portene begge går lave, men da vil igjen koblingen føre til at NOG-portene går lave. Det kan være vanskelig å se første gangen, men dette er en umulighet, for det skaper to sløyfer som jobber motsatt av hverandre samtidig.



Løsningen på problemene med en SR-flip flop blir ganske enkelt løst i noe som kalles en JK-flip flop. Koblingen er to NOG-porter som er koblet til en SR-flip flop. Denne kvitter seg med de umulige verdiene fra den forrige kretsen. Klokkepulsene fungerer som en av/på bryter. Ved høy på klokkepuls vil den fungere på samme måte som SR-flip floppen, om J er høy alene vil utgangen Q bli satt som høy. Om K er høy alene, vil utgangen Q bli satt som lav. Om begge inngangene er lave, vil det ikke bli noen endring i utgangene, eller verdien som er lagret i systemet.

Om klokkepulsene er lav, vil endringene i verdier på J og K ikke har noen endringer på kretsen eller verdien til utgangen. Den er lagret i forrige posisjon den hadde før klokkepulsene ble lav.

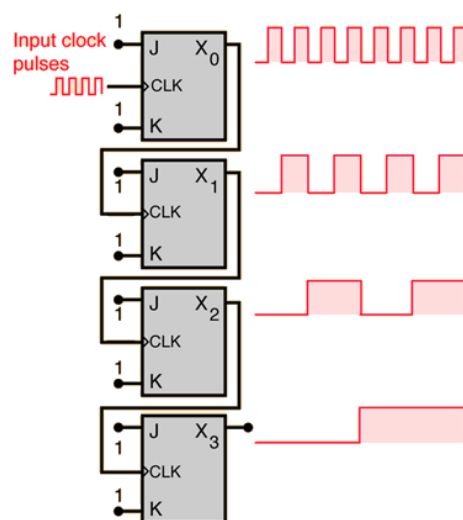
Den siste funksjonen med en JK-flip flop, den viktigste som vi kommer til å bruke, er «toggle» funksjonen. Dette er løsningen på den ikke-tillatte verdien som skjer når begge alle inngangene er høye. Om alle inngangene nå skulle være høye vil kretsen bytte verdi på utgangen. Om den var på lav, vil den gi ut høy. Mens om den var høy før, vil den når alt går til høy, endre utgangen til lav. Dette kommer godt med når vi skal bygge en asynkron klokketeller også kalt en frekvensdeler.

Truth Table

J	K	CLK	Q
0	0	↑	$Q_0$ (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	$\bar{Q}_0$ (toggles)

Figur 3: Sannhetstabellen for JK-flip flop

## Asynkron teller



Over er en asynkron teller, den benytter seg av de 4 første binære sifferplassene. Det betyr at den kan telle 16 siffer i 10-tallssystemet. Altså  $2^4$ , så denne teller fra 0-15. Grunnen til at det kalles en asynkron teller er fordi klokkepulsene ikke er felles for alle flip floppene. Ved hjelp av toggle funksjonen på flip floppene, endres verdiene på utgangen ut ifra klokkepulsene som kommer inn. Pulsene ut fra forrige flip flop er koblet som klokkepuls til neste flip flop. Det vil si at for hver flip flop klokke pulsen går igjennom halveres frekvensen. Ved dette eksempelet vil en frekvens på 16Hz som klokkepuls inn, komme ut av den siste flip-floppen som 1Hz. Denne formen for teller er mye brukt til det, for som ren teller blir den unøyaktig ved flere flip flopper enn 4 stykker på rad. Forsinkelsen skyldes at pulsen må gjennom hver eneste flip flop, alle ledningene den må gjennom fører til en liten forsinkelse. Den er ikke stor, men ikke liten nok heller til å forsvare bruk i større kretser.

Dette var en innføring i logiske porter, hvordan de er bygd opp og bruken av disse.