

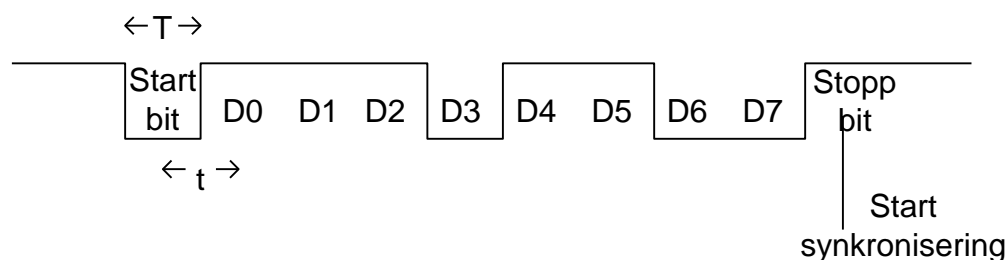
## Seriell kommunikasjon

### Asynkron og synkron dataoverføring.

Andre navn: Com-port, RS232,

#### Asynkron

Her foregår synkroniseringen for hver byte. Det sendes en startbit før databitene sendes. En forbindelse hvor det ikke går data, er på logisk 1. Startbiten er logisk 0, og det er den som mottageren synkroniserer seg på. Det gjøres ved at mottageren punktprøver det mottatte signal med en høyere (16x eller 64x) frekvens enn dataene. Med en gang det oppdages en logisk 0, starter mottageren. En ny punktprøve blir tatt en halv bitlengde senere, slik at midten i startbiten blir prøvet. Hvis det (fortsatt) er logisk 0, er mottageren sikker på at det er startbiten. De kommende 7 eller 8 bit blir punktprøvet mitt i biten, og plassert i riktig bitposisjon i mottageren (USART). Når siste bit er lest, forventer mottageren en (eller 2) stoppbit, som er logisk 1. Når stoppbiten er mottatt, begynner mottageren å lete etter neste startbit, slik at den kan synkronisere seg på den neste byten.



#### Synkron

Ved synkron dataoverføring foregår synkroniseringen i starten av hver blokk. Når synkronisering først er oppnådd, er det ingen synkronisering igjen før neste blokk. Det er da viktig at senderklokka og mottagerklokka er eksakt like, derfor overføres senderklokka i sammen med dataene. Det gjøres ved å kode inn klokkesignalet i datastrømmen. På mottagersiden blir dette klokkesignalet dekodet ut av datastrømmen, og brukes til å klokke inn dataene. På den måten blir mottagerklokka og senderklokka eksakt like.

Synkroniseringen kan foregå på to måter. Det ene er bytesynkronisering. Da er det en bestemt byte, synkroniseringsbyte, som er lagret i mottageren. I starten av blokka er det en slik synkroniseringsbyte. Mottageren sjekker det mottatte bitmønsteret med denne byten for hver bit mottatt. Da bitmønsteret er likt, er synkronisering oppnådd. Hvis denne synkbyten er en del av dataene, setter senderen inn en ekstra byte, som betyr at etterfølgende byte ikke er synkroniseringsbyte, men en databyte. Denne ekstra byten blir tatt ut på mottagersiden.

– Den andre måte en bitsynkronisering. Der ser mottageren etter et bestemt bitmønster. Når det har kommet 6 enere, etter en nuller, og det deretter kommer en nuller, er synkronisering oppnådd. Hvis det samme synkroniseringsmønsteret kommer som en del av dataene, vil senderen sette inn en ekstra 0 i datastrømmen, slik at det ikke kommer 6 1'ere etter hverandre i dataene. Denne ekstra 0'er blir tatt ut igjen ved mottageren