

INSTRUKSJONSHEFTE
FOR
ORIC PERSONLIG KOMPUTER
NORSK UTGAVE

Dette heftet er en oversettelse av den originale håndbok og er ment som et hjelpemiddel for dem som ikke tidligere har hatt befattning med datamaskiner.
Bakerst er det gitt endel programmerings eksempler som også kan benyttes av den litt mer erfarne bruker.

FORELØBIG UTGAVE
OSLO FEBRUAR 1983.

AD ELEKTRONIKK A/S, POSTBOKS 32 VINDEREN, OSLO 3

INNLEDNING.

GRATULERER MED DIN NYE ORIC-1 FARGEKOMPUTER.

Dette er en av de mest avanserte mikrokomputere på markedet idag.

Før du kobler ORIC-1 til nettspenning og TV-apparat bør du lese den første delen av dette instruksjonshefte.

Dette heftet på norsk er først og fremst skrevet for deg som er nybegynner.

AD Elektronikk A/S, som er importør av ORIC komputere, legger stor vekt på at instruksjonsmanualen skal være på norsk og skrevet på en forståelig måte.

Til deg som allerede har brukt mikrokomputere og kan programmere i BASIC viser vi til den fullstendige håndboken fra ORIC PRODUCTS INTERNATIONAL LTD.

Øvelse gjør mester heter et gammelt ordspråk. Dette gjelder absolutt for bruk av datamaskiner.

Vær aldri redd for å trykke på knappene .

Selvom du skulle trykke på feil knapp kan du ikke ødelegge komputeren og når du først har trykket feil knapp lærer du iallefall komputerens reaksjon på dette.

For å kunne bruke en datamaskin må den programmeres.

A programmere en datamaskin er ikke så vanskelig som mange tror. Hvis du bare tenker logisk og ikke gjør det vanskeligere enn det er, går dette som en lek.

Iallefall starter du kanskje med de ferdige programmene du får til din ORIC komputer.

Det medfølger for eksempel et demonstrasjonsprogram på kassett som du kan starte med.

AD Elektronikk kommer til å bygge opp et bibliotek av norske programmer for ORIC komputeren. Disse vil bli tilbudt via programvarelister som etterhvert blir sendt til alle registrerte brukere av ORIC komputere.

Fordi dette vil ta noe tid, er det mulig at de første brukerne av komputeren først får dette tilsendt senere.

IGANGSETTING

Før vi går igang med det alvorlige, skal vi se litt på datamaskinen og dens tilbehør.

Når du åpnet pakken fant du først og fremst selve komputeren. Denne ser i grunn bare ut som et tastatur og den må derfor kobles til et TV-apparat før du kan bruke denne.

På baksiden av komputeren ser du flere typer kontakter. Alle disse er vanlige standard kontakter for datamaskiner.

ORIC har riktignok flere kontaktutganger enn mange andre komputere.

Når du holder ORIC med tastaturet mot deg vil kontaktutgangen på høyre side være den som brukes til TV-apparatet.
Du kan koble ORIC både til s/hv- og farge TV.

På venstre side finner du kontakten for strømforsyningen.
De andre kontaktutgangene skal vi se på senere.
Disse er for tilkobling av kasett, diskstasjon og printer.

I pakken er det også en separat strømforsyning fra Mascot. Denne er produsert i Norge og godkjent av NEMKO. Det er forresten også selve komputeren.

Videre finner du både TV ledning og kasettkabel, instruksjonsbok, programkasett og garantikort.

Nå kan du koble ledningen til TV-uttaket på ORIC og den andre enden til antenneinngangen på TV-apparatet ditt. (U.H.F.)
Deretter slår du på TV'en og stiller denne inn på kanal 36.

Så setter du strømforsyningen i nettkontakten på veggen og ledningen i ORIC's spenningsinngang.
Hvis du allerede har stilt TV riktig vil du nå få et bilde på skjermen som vist nedenfor:

```
ORIC EXTENDED BASIC V1.0
@ 1983 TANGERINE
X BYTES FREE
```

READY

Har du ikke fått dette bildet finjusterer du inn kanal 36 på TV-apparatet inntil bildet kommer på skjermen.
Det kan være nødvendig å finjustere lys- og kontrast også.
Dette gjelder spesielt når du bruker fargeTV.

Finjusteringen av TV tar erfaringsmessig litt tid, men vanligvis får du dette til allerede ved de første forsøk.
For å finne kanal 36 på U.H.F. (Ultra High Frequency) bør du studere TV-apparatets instruksjonsbok.
Husk å skru ned lyden på TV før du bruker ORIC.
Tilslutt vil vi nevne at hvis du har et TV-apparat med egen R.G.B. inngang betyr dette at du kan bruke TV som monitor.
Fordi ORIC har egen R.G.B. utgang vil du på denne måten oppnå en bedre fargeoppløsning på skjermen.

AD Elektronikk tilbyr en 22" kombinert fargeTV og monitor som er langt rimeligere enn bare TV-apparatdelen i butikk.
Vi kan også levere en RGB monitor.

Legg tilslutt merke til at under ORIC er det et firkantet hull.
Dette er RESET knappen som vi skal ta opp senere.

HVORDAN DU FORTELLER ORIC HVA DEN SKAL GJØRE

Som du kanskje vet forstår datamaskiner ikke vårt norske språk. Derfor må vi bruke et språk som disse forstår når vi skal "snakke" til dem. Dette språket heter BASIC og ORIC bruker en ny og forbedret utgave av dette språket. Etterat du nå har slått på maskinen og stilt inn TV-apparatet vil du også se et blinkende firkantet svart felt på skjermen. Denne kalles markør (eng. Cursor) og markerer hvor du skriver.

La oss nå se litt på ORIC's tastatur før vi går videre. Alle knappene(tastene) er merket med sine funksjoner. Noen har også to forskjellige merkinger. Det vil altså aldri være mere enn to funksjoner pr. knapp på ORIC. Dette gjør tastaturet lett å bruke.

Hvis du nå trykker på den aller lengste knappen nederst, ser du at markøren flytter seg mot høyre. Dette er altså avstandsknappen. Fire andre knapper er merket med piler. Disse flytter markøren i den retning pilen peker. De fleste knappene er merket med bokstaver og tall og brukes til å skrive med. Noen knapper er merket med: ESC. (escape), CTRL(control), DEL(delete), SHIFT og RETURN.

SHIFT knappen brukes for å skifte over til den andre funksjonen. (Merkingen øverst på knappen)

Nå kan du selv skrive:

```
PRINT "HALLO ORIC"
```

og deretter trykke på RETURN knappen.

Ordene HALLO ORIC kommer da på skjermen og du har nå utført en kommando i BASIC språket til datamaskinen. PRINT er altså en BASIC kommando.

Du kan nå få ORIC til å skrive det du selv ønsker, men husk gjentakelsestegnene " " foran og etter det du vil skrive.

Hvis du nå forsøker å skrive:

```
WRITE "HALLO ORIC"
```

vil du få en feilmelding på skjermen.

SYNTAX ERROR betyr at du har gjort en feil og feilen var her altså bruken av WRITE. Det skulle, som du nå vet, ha vært PRINT istedenfor.

Husk også på bruken av RETURN knappen.

Denne benyttes regelmessig og bl.a. når man skal utføre kommandoer som vist ovenfor.

Vi har nå sett at BASIC kommandoen PRINT er lett å bruke, men at det er avgjørende for datamaskinen at du bruker de rette kommando-ordene. Dette fordi datamaskinen bare forstår de ordene som er i BASIC språket.

LITT OM DATAMASKINER

Datamaskiner er egentlig dumme fordi de bare gjør det de får beskjed om og er beskjeden dum blir resultatet også dumt. Men samtidig er datamaskiner også flinke fordi når du først forstår å bruke dem riktig er det nesten ingen grense for hva den kan gjøre for deg.

En datamaskin består av en prosessorenhet (C.P.U.=Central Processing Unit) som bearbeider alle beskjeder (kommandoer) datamaskinen får fra innlesningsenheten (Tastaturet) Prosessorenheten er ORIC's hjerne kan vi si.

Datamaskinen har to typer hukommelse, nemlig et indre lager (ROM eller RAM) og et ytre lager som kan være en TV-skjerm, printer eller andre ting.

I ORIC ligger hele BASIC språket i ROM og når du selv lager dine programmer bruker du RAM hukommelsen. ROM'en fungerer altså som en oversetter mellom deg og ORIC, mens RAM'en er din egen arbeidshukommelse.

Datamaskinen forstår bare det binære tallsystemet, det vi kaller for totallsystemet fordi det bare er to stillinger i dette, nemlig AV og PÅ.

Populært sagt består derfor ORIC av tusenvis av brytere som står i åpen eller lukket stilling og på denne måten sier disse JA eller NEI (0 eller 1) Et slikt null eller ettall kalles altså for binært siffer og betyr noe inne i datamaskinen.

På engelsk heter binært siffer BINARY DIGIT og en forkortelse av dette kalles BIT.

Slike BITS samles i grupper på åtte stykker og da heter gruppen en BYTE. (uttales bait) Vi kan si at BIT er datamaskinens bokstav og BYTE dens ord.

1000 BYTE utgjør en kilobyte eller 1K.

Din ORIC har innebygget en ROM hukommelse på 16K og en RAM hukommelse på 16K eller 64K (65536 bytes). Altså er dette en måte å måle datamaskinens kapasitet på.

Dette var nok teori nå, så la oss gå videre med noen program eksempler.

ORIC SOM KALKULATOR.

Nå skal vi se hvordan du regner med ORIC.

Skriv PRINT"5" og deretter RETURN.

Så kan du skrive PRINT 5 og husk RETURN.

(Dette er siste gang vi minner om RETURN funksjonen)

Når vi skriver informasjonen med gjentakelsestegn kaller vi dette for en strengfunksjon. En slik funksjon kan inneholde både tall, bokstaver og grafiske symboler.

Når du entrer informasjonen uten disse tegnene antar ORIC at det bare er tall som skrives og du kan derfor på denne måten bruke den til å regne med også uten disse tegnene.

Ved addisjon skriver du da PRINT 75+25 og svaret på skjermen blir altså 100.

Forsøk også PRINT "75+25"

(Legg merke til hvordan nulltallet skrives av datamaskinen)

En subtraksjon vil likeledes være PRINT 75-25

Videre vil multiplikasjon være PRINT 2*4

og divisjon PRINT 4/2

(Multiplikasjonstegnet er SHIFT 8 og subtraksjonstegnet til høyre for 0)

Hvis du gjør feil når du skriver kan du trykke på CTRL knappen og samtidig X knappen for å viske bort hele linjen.

Forsøk også å erstatte PRINT med ?

Nå kan du selv lage noen regneeksempler.

La oss videre se på hvordan ORIC oppfatter en kombinasjon av regningsartene.

skriv PRINT 4+2*3

Resultatet blir altså 10 og ikke 14 fordi multiplikasjon er viktigere enn addisjonstegnet i datamaskinen. La oss derfor skrive en prioritetsliste over de forskjellige matematiske symbolene ORIC bruker.

()

^ potens

+,- når det menes positiv og negativ

*,/

+,- når det menes addisjon og subtraksjon

Husk at ORIC oppfatter det som står innenfor gjentakelsestegnet som tall.

Her ser du at tegnene i datamaskinen er svært like de symbolene vi vanligvis bruker. Det er viktig at du lærer de riktige symbolene med en gang.

Skrevet i BASIC ser da de fire regneoperasjonene slik ut.

3+2 3-2 3*2 3/2

Dette kalles aritmetiske uttrykk.

LITT MERE OM PROGRAMMERING

Nå vet vi hvordan vi skal fortelle ORIC hvilke operasjoner den skal utføre, men vi har ennå ikke gitt den ordre om å utføre slike.

Skriv derfor PRINT H og se hva som skjer. Du skal få et null-tall og det har selvfølgelig ingen betydning.

```
Istedenfor kan vi derfor skrive LET H=4  
PRINT H
```

Denne gangen vet ORIC at du har satt H=4 og vi kaller H for en variabel i BASIC språket.

ORIC vil nå huske at H=4 inntil du enten forandrer verdien av H ved å skrive linjen pånytt eller skrive CLEAR eller eventuelt slå av maskinen.

Forsøk selv med andre bokstaver.

```
Du kan også skrive LET S=3+2  
PRINT S
```

Husk at du kan bruke hele ord som variable, men at ORIC bare vil oppfatte de første to bokstavene av et helt ord.

```
Forsøk: LET JAN =14  
LET JARL=20  
PRINT JAN  
PRINT JARL
```

Du får altså 20 begge ganger fordi ORIC registrerer variablene som JA i begge tilfeller og denne JA variabelen ble først satt til 14 og deretter resatt til 20.

Egentlig kan ORIC huske en rekke forskjellige tall som kan lagres i komputeren ved hjelp av slike LET setninger.

La oss nå oppsummere LET setningen ved hjelp av et siste eksempel. Vi antar at et stykke sepe koster 5 kroner og at du ønsker å hjemme denne opplysningen til senere.

Du skriver da: LET SEPE =5

I ORIC er nå et lagerområde brukt til å gjemme tallet 5 og dette området er gitt navnet SEPE. Denne kombinasjonen kalles altså en variabel.

SEPE er i dette tilfellet en numerisk variabel fordi den har ett tall som verdi. Ønsker du senere å vite hva sepe koster skriver du igjen: PRINT SEPE

ORIC kan også huske variable med navn og en kombinasjon av tall og navn (alfanummerisk)

```
Skriv: LET N$="PER"  
PRINT N$
```

På skjermen kommer da "PER" og vi sier at dette er en streng variabel. Forsøk også : LET X=1/3
PRINT X

Svaret blir som sagt 0,33333333.

NUMMERERING AV SETNINGER

Når vi programmerer i BASIC må alle setninger nummereres. Datamaskinen utfører setningene etter setningsnummrene i stigende rekkefølge uansett hvilken rekkefølge setningene ellers er skrevet i.

La oss velge numrene 10 og 20 og sette opp:

```
10 LET S= 3+2
```

```
20 PRINT S
```

Det er vanlig å velge numrene i dekadere for å reservere tall imellom som kan brukes senere dersom vi ønsker å fylle inn flere instruksjoner. Forsøk dette korte programmet:

```
8 CLS
```

```
10 PRINT "SKRIV NAVNET DITT"
```

```
20 INPUT N$
```

```
30 PRINT "HYGGELIG Å TREFFE DEG,";N$
```

ORIC vil nå gå til den første linjen altså 8 og slette det som måtte allerede stå på skjermen. (CLS=CLEAR. norsk=slette)

Deretter eksekveres linje 10 og 20 hvor ORIC venter på en inngående beskjed fra tastaturet; med andre ord navnet ditt som du da skriver på tastaturet. Tilslutt går den til linje 30 og skriver meldingen på skjermen til deg.

Forat du skal kunne kjøre dette programmet må du skrive RUN

RUN kommandoen må alltid brukes når programmer skal kjøres.

I BASIC er det også vanlig å avslutte et program med END. Dette er ikke nødvendig i ORIC, men du må gjerne bruke dette.

Etterat du har kjørt programmet kan du skrive LIST.

Da kommer programmet tilbake på skjermen. HUSK DETTE !!!

Til programmet ovenfor kan du nå føye til de linjene som vist nedenfor

```
40 PRINT "FODSELSAARET DITT"
```

```
50 INPUT AARSTALL
```

```
60 LET ALDER=1983-AARSTALL
```

```
100 PRINT "DU ER ";ALDER; "AAR GAMMEL";N$
```

```
110 GOTO 200
```

```
200 END
```

```
RUN
```

Hvis du får feil i en linje når programmet skrives kan du viske bort hele linjen bare ved å skrive linjenummeret og RETURN eller du kan skrive linjen pånytt med samme linjenummer.

En alminnelig feil når man arbeider med strenger er å glemme anførselstegnet.

Hittil har du altså lært å programmere med tall og tekst pluss at du kan nummerere linjene i programmet ditt.


```
Tilslutt kan du forsøke dette programmet med ORIC'en din:
10 LET S=3+2
20 LET D=3-2
30 LET P=3*2
40 LET Q=3/2
50 PRINT S
60 PRINT D
70 PRINT P
80 PRINT Q
90 END
RUN
```

LITT MERE TEORI

De fleste datamaskiner har innebygget flere funksjoner slike som SINUS, COSINUS, LOG osv som standard.

Disse kalles biblioteksfunksjoner og vi skal nå se litt på hvordan du kan bruke disse.

Matematisk sett er en funksjon en regel som istedenfor et tall (argument eller operand) gir et annet tall (resultat)

I ORIC finner du følgende funksjoner i BASIC:

SIN	Beregner sinus i radianer	A=SIN(N)
COS	Beregner cosinus "	A=COS(N)
TAN	Beregner tangens	A=TAN(N)
ATN	Beregner arcustangens	A=ATN(Y/4)
ABS	Returnerer absoluttverdien	X=ABS(-4)
SQR	Beregner kvadratroten	A=SQR(N)
EXP	Beregner verdien av e^x hvor $e=2,718$	A=EXP(N)
LOG	Beregner naturlig logaritme	A=LOG(X+Y)
SGN	Returnerer fortegnet -1,0,1	Z=SGN(X-Y)
INT	Returnerer tallet avrundet nedad til nærmeste heltall	X=INT(Y+0,5)
RND	Returnerer et tilfeldig tall mellom 0 og 1	A=RND(X)*100
TAB	Flytter cursor på TV'en til en gitt posisjon X	PRINT TAB(N) "HALLO"

De fleste funksjonene forklarer seg selv bortsett fra et par stykkersom trenger nærmere kommentar.

INT(X) funksjonen gir heltallsdelen av X. Dette er meget nyttig ved avrunding av tall.

Her er noen eksempler:

INT(X) finner alltid største heltall mindre enn X

```
X=3,54      INT(X)=3
X=-4,75     INT(X)=-5
```

Ved avrunding til nærmeste heltall benyttes følgende oppsett:

```
INT (X+0,5)
X=2,85      INT(X+0,5)=INT(3,35)=3
```

ATN(X) funksjonen er den vinkel målt i radianer som har tangensverdi lik X. Vinkelen skal ligge mellom $+\text{PHI}/2$ og $-\text{PHI}/2$. Vi har altså at dersom $X=\text{TAN}(Y)$, så er $X=\text{ATN}(X)$

SGN(X) Returnerer 1 dersom $X > 0$
Returnerer 0 dersom $X = 0$
Returnerer -1 dersom $X < 0$

Linjen PRINT SGN(-4) vil derfor skrive ut -1

ABS(X) Returnerer tallverdien av X
PRINT ABS(-23) vil skrive ut 23

SQR(X) beregner kvadratroten kun av positive X verdier.
Dersom vi bruker funksjonen i et program der vi ikke kan forutsi fortegnet til X, kan vi bruke ABS funksjonen for å rette på dette.
Dette var kanskje et vanskelig avsnitt, men vi mente det var riktig å gå gjennom de funksjonene som er i ORIC fordi mange vil ha lært dette på skolen.

Men la oss bruke litt av det vi nå har snakket om
PRINT SQR 9
Svaret er 3 fordi kvadratroten av 9 er altså 3.
PRINT SQR (2*2)

PRINT TAN(45/180*3,14)
gir resultatet 1, nemlig tangens til 45 grader.
Omvendt skal det ved omregning fra radianer til grader divideres med 3,14 og multipliseres med 180.

NOEN FLERE SMAPROGRAMMER PÅ ORIC

Alle BASIC kommandoer og variable må skrives i store bokstaver.

ORIC har til nå vist oss at datamaskinen er i stand til å utføre ordre og selv ta avgjørelser på forhold som er sanne eller usanne.

Vi skal nå se at ORIC også kan repetere kommandoer så ofte DU vil.

For å oppnå dette på en enkel måte skal vi ta i bruk en ny kommando som heter FOR.....TO/NEXT sløyfe.

Denne programsløyfen vil nemlig gjenta instruksjonen inntil det endelige svaret (f.eks. et tall) er nådd.

La oss forsøke dette:

```
10 FOR X=1 TO 1000 STEP 1
20 PRINT X
30 NEXT X
40 PRINT "PHU! ENDELIG FERDIG"
```

Linje 10 setter telleren X i ORIC til 1 og går så til linje 20 hvor den skriver verdien av X. Så går den til linje 30 hvor den får beskjed om å ta neste verdi av X (NEXT X) og går tilbake til linje 10 for å hente denne.

Nå har verdien av X også steget med 1 p.g.a. STEP 1.

Når 1000 er nådd stopper det hele.

Hvis du forandrer STEP tallet til 2 vil du få tallene 1,3,5,7 osv.

Skal du telle baklengs kan du skrive linje 10 på denne måten:

```
10 FOR X= 1000 TO 1 STEP 1
```

Hvis du vil legge inn en forsinkelse enten i dette telleprogrammet eller når som helst har ORIC en spesiell WAIT kommando.

For å gjøre programmet ovenfor saktere kan du legge inn denne linjen:

```
25 WAIT 4
```

Dette er det samme som å be ORIC om å telle til fire før den skriver et nytt tall.

Generelt kan FOR/NEXT setningen uttrykkes slik.

```
(Linjenummer) FOR(variabel)=(aritmetisk uttrykk) TO(aritmetisk uttrykk) STEP(aritmetisk uttrykk)
```

Hvor mange ganger vil programmet nedenfor skrive ut "GODT GJORT"?

```
10 FOR I=1TO10
20 PRINT "GODT GJORT"
30 NEXT I
40 END
RUN
```

En annen måte å gjøre dette med FOR/NEXT setninger på i BASIC er ved hjelp av IF/THEN setninger.

```
10 FOR N=10TO30 STEP 10
200 NEXT N
```

Dette kan nemlig også skrives slik:

```
10 LET N=N+10
200 IF N<30 THEN 10
```

(< tegnet betyr mindre enn, > betyr større enn)

Dette programmet kunne vi på godt norsk ha formulert slik:

```
10 FOR N=10 TIL OG MED 30 ØK MED 10
200 NESTE N
```

La oss samtidig se på hvordan deler av et program gjentaes flere ganger uten at man bruker FOR/NEXT setninger.

Tar vi for oss telleprogrammet igjen kan det jo være at vi ønsker å informere om denne venteinstruksjonen.

Da kan du sende programmet til en subrutine hvor det venter, skriver ut en melding om kort pause for så å gå tilbake til hovedprogrammet igjen.

La oss forsøke dette.

```
10 FOR X= 1TO10
20 GOSUB 1000
30 PRINT X
40 END
1000 PRINT "DETTE ER EN KORT PAUSE"
1010 WAIT 50
1020 RETURN
```

En slik GOSUB/RETURN programdel kaller vi altså for en subrutine eller underprogram.

For å starte utførelsen av underprogrammet må vi benytte en GOSUB setning. Maskinen husker da linjenummeret etter denne setningen og når vi skriver RETURN går programmet tilbake til dette stedet.

GOSUB benyttes i program hvor den samme sekvensen skal utføres flere ganger.

Før vi nå går over til å se på farger og lyd på ORIC skal vi si litt generelt om programskrivning.

Det er alltid fornuftig å skrive ned programmet på papir først. Dette gjelder iallefall ved litt større programmer.

Du bør forsøke å lage en systematisk beskrivelse som viser den logiske rekkefølgen maskinen må følge.

Når programmer skrives kan du legge inn kommentarsetninger for å forklare hva programmet gjør.

Kommentarsetningene begynner alltid med REM og slike setninger blir ikke tatt med når ORIC kjører programmet ditt.

Derimot vil du alltid få med slike kommentarsetninger når programmet listes ut.

```
10 REM HER STARTER PROGRAMMET
20 FOR N=1TO10:REM TELLER SLØYFEN
30 PRINT AD ELEKTRONIKK IMPORTERER ORIC
40 NEXT N
50 END
60 REM DETTE VAR ET DUMT PROGRAM
```

FARGER OG GRAFIKK

Når du slår på Oric går den automatisk inn i TEXT stilling. Med andre ord kan du skrive tekst direkte på skjermen som vi allerede har sett.

Legg merke til at når skjermen blir full av av tekst ruller den alltid opp en ny linje for deg.

TEXT området er også brukt for den lavoppløselige grafikken.

Før vi ser på den lavoppløselige grafikken (LORES) skal vi notere oss hvilke farger som er tilstede på Oric.

Du vet jo allerede at Oric er en fargecomputer.

Det finnes to fargekommandoer i Oric som du husker lett, nemlig INK og PAPER.

Disse setter forgrunns og bakgrunnsfargene slik at INK gjelder for teksten og PAPER for tekstens bakgrunn.

Når du skriver disse kommandoene skal de alltid etterfølges av et tall mellom 0 og 7. Disse tallene representerer de 8 forskjellige fargene i ORIC.

Disse er:

- 0 = sort
- 1 = rød
- 2 = grønn
- 3 = gul
- 4 = blå
- 5 = lilla
- 6 = turkis
- 7 = hvit

Forsøk å bruke disse fargene ved å skrive PAPER0,PAPER1,PAPER2 etc.

Her er et lite program som viser alle disse fargekombinasjonene.

```
5 REM FARGER
10 TEXT
20 FOR N=1 TO 25
30 PRINT "DETTE ER FARGECOMPUTEREN ORIC"
40 NEXT N
50 FOR I=0TO7
60 FOR P=0TO7
70 INK I:PAPER P
80 WAIT 100
90 NEXT P
100 NEXT I
110 INK 7: PAPER 4
```

Når forgrunnsfargen er lik bakgrunnsfargen kan du selvfølgelig ikke se teksten og noen fargekombinasjoner går også mindre bra mot hverandre.

I disse tilfellene får du lett interferensproblemer på TV skjermen din.

Skal du bruke Oric med lavopløselig grafikk kan du bruke skjermen i TEXT stillingen eller du kan skrive LORES 0. Den delen av skjermen som benyttes til plotting er fra 0 til 38 i horisontalplanet og fra 0 til 26 i vertikalplanet. Posisjon 0,0 er oppe til venstre på skjermbildet. Når du bruker LORES 0 blir også hele TV skjermen sort.

Forsøk dette programmet:

```
10 LORES 0
20 PLOT 16,12,"HALLO"
```

Som du ser vil hallo bli skrevet i midten av skjermen og det er selvfølgelig PLOT kommandoen med X og Y koordinatene som bestemmer plasseringen.

Forsøk dette programmet også:

```
10 LORES 0
20 FOR X=1 TO 63
30 LET Y = SINX +10
40 PLOT X,Y
50 NEXT X
60 GO TO 60
```

I linje 60 holder vi programmet i en endeløs sløyfe og du må derfor bruke CTRL knappen og samtidig C for å stoppe programmet.

For å finne ut hvor de forskjellige karakterene befinner seg på skjermen i enten TEXT eller LORES kan du bruke SCRN (X,Y)

Skriv CLS (CLEAR SCREEN) for å klargjøre skjermen.

Markøren skal nå være oppe i det venstre hjørnet.

Skriv PLOT 10,20,"A"

Bokstaven A kommer frem nede på skjermen.

Skriver du videre PRINT SCRN(10,20) vil du få frem koden til bokstaven A som er 65.

Du kan lese mer om ASCII koden i Oric's håndbok senere.

Det er ikke viktig for deg å forstå den nå.

Her kommer likevel et program som demonstrerer hvordan ASCII koden kan brukes.

Programmet gjentar seg selv inntil en rakett når målet.

```
100 REM ***BRUK AV SCRN(X,Y) ***
110 CLS: INK 1: PAPER 7
120 FOR N=20 TO 25
130 PLOT N,26,"+"
140 NEXT N
150 REPEAT
160 A=INT (RND(1)*36+2)
170 FOR P=0TO24
180 PLOT A,P,"V"
190 WAIT 4
200 PLOT A,P," "
210 NEXT P
220 UNTIL SCRN (A,P+1)= 43
230 EXPLODE
```

Det er altså i linje 220 at ASCII koden blir brukt ved at når raketten er en plass borte fra målet (ASCII koden for + er 43) at det hele avsluttes med en eksplosjon i linje 230.

Dette programmet har vist hvordan man benytter ASCII koden og som sagt kan du lese mere om dette i kapittel 12 i Orics egen håndbok.

Forsøk tilslutt å sette inne en ny linje:

```
115 LORES 0
```

For å avslutte beskrivelsen av den lavoppløselige grafikken la oss ta et eksempel til.

```
10 REM ** LORES FARGEPLOTTING**
20 LORES0
30 STP =2*PI/50
40 R=10: X=10: Y=10
50 REPEAT
60 E=18+RND(1)*6
70 PLOTX+R*SIN(C),Y+R*COS(C),E
80 C=C+STP
90 UNTIL C>2*PI
100 REPEAT: UNTIL KEY$ <>" "
110 CLS
```

KONTROLL KARAKTERER

Før vi nå går videre til høyoppløselig grafikk skal vi gjennomgå noen av kontrollkarakterene i Oric.
Kanskje du har savnet dem allerede.

Som du kanskje vet kan du f.eks. både skrive store og små bokstaver på Oric, men det er bare de store bokstavene som kan brukes i et BASIC program.

Når du er i store bokstaver står det alltid CAPS øverst til høyre på skjermen.

CTRL (Control knappen) og T vil slå av og på denne funksjonen.

Trykker du altså CTRL og T vil du få små bokstaver og gjør du dette en gang til vil de store bokstavene komme tilbake.

En annen slik av/på funksjon er CTRL F. Denne slår av lyden på knappene.

CTRL Q vil fjerne markøren.

Hvis du gjør en feil i en programlinje og du ønsker at hele linjen skal forsvinne slår du CTRL X

Kommandoen CLS vil viske bort skjermbildet.

Når du skal bruke den høyoppløselige delen av Oric må du alltid først be om dette ved å skrive HIRES.

Da ser du at skjermen blir sort bortsett fra et felt på 3 linjer nederst.

Dette er nyttig fordi da kan du likevel se de instruksjonene du skriver når du anvender grafikken.

Vil du tilbake til TEXT for å fortsette f.eks. programskrivningen gjøres dette ved å skrive TEXT.

Både TEXT og HIRES kan brukes som kommandoer i et program.

Før du starter med grafikk bør du vite at skjermen din nå er de.t inn i 240 punkter horisontalt (0 til 239) og 200 punkter vertikalt (0 til 199).

Programmessig kaller vi de horisontale for X og de vertikale for Y. I tillegg husker du at punktet øverst på skjermen til venstre er 0,0.

Det finnes mange spesialkommandoer i Oric for grafisk bruk. La oss se på noen av disse.

CURSET X,Y setter et punkt til et bestemt sted på skjermen bestemt av koordinatene X og Y. Legg merke til at markøren nå ikke er der som du er vant til fra TEXT.

CURSET 100,120,1 setter markøren i sentrum av skjermen. Altså 120 horisontalt og 100 vertikalt. Det siste tallet betyr forgrunn eller bakgrunn (FB).

Nå kan du skrive HIRES og eksperimentere med CURSET.

Den neste kommandoen heter CURMOV
Denne er lik CURSET bortsett fra at X og Y posisjonene er relative til markørens forrige posisjon.

Har du med andre ord allerede satt markøren et sted med CURSET X,Y,FB kan du flytte denne til et nytt sted med CURMOVX,Y,FB. Men pass på at du ikke går utenfor skjermbildet.

DRAW X,Y kommandoen vil tegne en rett linje fra markøren og til den nye posisjonen.

La oss tegne en firkant på skjermen:

```
5 REM ** FIRKANT**
10 HIRES
20 CURSET 60,40,3
30 DRAW 120,0,1
40 DRAW 0,120,1
50 DRAW -120,0,1
60 DRAW 0,-120,1
```

Som du ser benyttes negative verdier for å tegne fra høyre mot venstre og fra nede og oppover.

Husk på at når du skriver HIRES pånytt visker du ut skjermbildet.

Forsøk også dette programmet.

```
5 REM** INTERFERENS**
10 HIRES
20 FOR A=0TO1
30 FOR B=0TO 239
40 CURSET 0,199*A,3
50 DRAW B,199-398*A,1
60 CURSET 239,199*A,3
70 DRAW -B,199-398*A,1
80 NEXT B: NEXT A
```

Hvis du senere ønsker å skrive i høyoppløselig grafikk kan du gjøre dette ved hjelp av kommandoen CHAR.
Dette kan du eventuelt selv studere i den engelske håndboken.

La oss heller se på hvordan du også kan tegne sirkler.

For å tegne sirkler er det nødvendig å skrive CIRCLE og deretter 2 nummer. Det første er sirkelens radius og det andre er FB koden.
Sirkelens sentrum vil alltid være ved markørens posisjon.
Forsøk dette ved å skrive.

```
HIRES
CURSET 120,100,3
CIRCLE 50,1
```

Forsøk også dette.

```
100 HIRES
110 CURSET 120,100,3
120 FOR N= 99 TO 1 STEP-1
130 CIRCLE N,1
140 PATTERN 100-N
150 NEXT N
160 EXPLODE
```

Tilslutt tar vi med oss FILL kommandoen. Denne brukes når du vil fylle igjen et område av skjermen. Du kan fylle 200 rader og 40 celler på skjermen.

Dette programmet illustrerer kommandoen:

```
5 HIRES
10 FORN=0 TO 199
20 X=RND(1)*8+16
30 FILL 1,1,X
40 NEXT N
```

Linje 20 velger tilfeldige bakgrunnsfarger fra linje 0 (toppen av skjermen) til linje 199.

Dette programmet demonstrerte også blandingen av farger og grafikk.

EDITERING AV PROGRAMMER

Når du har skrevet et program og vil forandre en linje er det flere måter å gjøre dette på.
Hvis hele linjen skal fjernes gjør du det ved å skrive bare linjenummeret og slå "RETURN"
Dette er vist nedenfor.

```
10 PRINT "HALLO"  
20 PRINT "OOPS"  
30 PRINT "JEG ER ORIC"
```

Skriv så 20 "RETURN" og skriv LIST.

Du vil altså få programmet frem på skjermen igjen uten linje 20 fordi denne er nå visket bort .

Ønsker du å fjerne alle linjer i et program skriver du NEW

Når du skriver en linje og oppdager at du har gjort en feil kan du viske bort siste karakter ved hjelp av DEL knappen.

Trykker du på denne flere ganger ser du at markøren flytter seg mot venstre og visker ut karakterer etterhvert.

Vil du fjerne hele linjen kan du trykke på CTRL og X.

Markøren flytter seg da ned til neste linje.

Skal du liste ut et større program vil du oppdage at det går fort forbi på skjermen.

Dette kan du ordne ved å trykke på CTRL og C.

Da stopper programlistingen inntil du trykker disse pånytt.

CTRL C vil også stoppe eksekveringen av de fleste BASIC programmer og du kan starte disse igjen ved å skrive CONT såfremt du ikke har forandret programmet eller noen variable etterat du stoppet dette. Da må du skrive RUN eller GOTO for å kjøre pånytt.

COPYING

Skal du forandre deler av en programlinje er det ikke nødvendig å skrive hele linjen omigjen.

Legg forresten merke til at du ikke kan ha programlinjer som er lengre enn 78 karakterer.

Lange programlinjer er dessuten vanskelige å lese når du eventuelt skal feilsøke programmet ditt.

Hvis du skriver linjer som er lengre enn 78 karakterer vil Oric gi fra seg et lydsignal (Ping).

Skal du forandre en linje kan du benytte COPY kommandoen.
For å illustrere denne skriver vi:

```
10 REM ** EDITERINGSPRØVE**  
20 A=20:B=30  
30 C=A*B  
40 PRINT C
```

Hvis du nå vil forandre linje 20 til A=25:B=5 og linje 30 til C=A+B gjør du følgende.

LIST programmet ditt først og skriv gjerne CTRL L for å slette skjermbildet.

Nå kan du flytte markøren ved hjelp av knappene nederst på tastaturet (Knappene med piler).

Når du har flyttet markøren til linjenummer 20 trykker du på CTRL A. Markøren flytter seg da mot høyre på linjen. Når den passerer en karakter vil dette bli registrert av Oric som del av den nye linjen.

Når markøren står over det første nulltallet slipper du CTRL A og slår inn 5 istedenfor.

Deretter fortsetter du å kopiere linjen ved hjelp av CTRL A.

Når markøren er over 3 tallet i 30 stopper du igjen og slår inn 5 istedenfor. Siden du ikke skal ha med nulltallet i den nye setningen kan du slå RETURN.

Nå blir den nye linjen automatisk husket av Oric.

Slå nå CLS og LIST programmet slik at du kan kontrollere den nye linjen din.

Husk på at du ikke kan forandre noe i programmet bare ved å flytte markøren på skjermen.

Programlinjen blir altså entret ved å kopiere de eksisterende karakterene med bruk av CTRL og A eller ved hjelp av nye verdier som slås inn fra tastaturet.

CTRL X vil frigjøre markøren fra den linjen du har editert.

Legg tilslutt også merke til at DEL tasten vil viske bort feil du har slått inn.

I tillegg til dette kan du selv studere TRON og TROFF kommandoene i den engelske håndboken.

LYD

Som du vet har Oric innebygget en meget avansert lydgenerator av den typen man finner i profesjonelle spillmaskiner.

Denne lager lyd over 6 oktaver med tre forskjellige toner og lyden kommer fra Oric's innebygde høytaler.

Du kan imidlertid også ta ut lyden til et stereoanlegg.

Du har også selvfølgelig allerede hørt lyden fra de tastene du har trykket på.

Hvis du slår CTRL F vil du slå av nettopp denne lyden.

Siden dette er en av/på funksjon bruker du også denne for å få

lyden tilbake.

Det ligger noen faste lydkommandoer i Oric som vi kan prøve nå.
Skriv ZAP og slå RETURN.

Dette skal gi en lyd som etterligner f.eks. en laserkanon.

Du kan også forsøke kommandoene PING,SHOOT,EXPLODE

Disse kan du bruke direkte i et Basic program som vist nedenfor.

```
10 FOR N=1TO10
  20ZAP
  30WAITS
  40NEXTN
  50EXPLODE
```

Linje 30 gjør at hver enkelte tone blir ferdig før neste kommer, altså en forsinkelse.

For musikk er det tre andre kommandoer som skal brukes.
Disse er SOUND, MUSIC, PLAY

I de fleste programtyper benyttes SOUND og MUSIC for å definere den type tone du vil ha, mens PLAY bestemmer om lyden skal være som f.eks. en gitar eller f.eks. et orgel.

Det tar litt tid å fortså bruken av disse kommandoene men når du får det til er det altså et utrolig antall muligheter med disse.

La oss ta et lite programeksempel.

```
5 REM **MUSIKK**
  10 MUSIC 1, RND(1)*6, RND(1)*12+1,7
  20 WAIT RND(1)*20+5
  30 GO TO 5
```

Du kan stoppe dette endeløse programmet med CTRL C.

Her kommer et til:

```
5 REM**MUSIKK FRA TASTATURET**
  10 GET A$
  20 A=VAL A$
  30 IF A$="-" THEN A=11
  40 IF A$="=" THEN A=12
  50 IF A$="/" THEN PLAY 0,0,0,0: STOP
  60 IF A$=0 THEN A=10
  70 MUSIC 1,3,A,5
  80 GO TO 5
```

I linje 20 venter programmet på et signal fra tastaturet.
Når du slår "/" stopper du programmet (Linje 50)

Her kommer et siste eksempel:

```
10 REM**MELODI**
20 FOR N=1TO11
30 READ A,B
40 MUSIC 2,3,A,0
45 PLAY 3,0,7,2000
50 WAIT B
60 PLAY 0,0,0,0
80 NEXT N
100 DATA 5,30,5,30,7,30,8,75,5,75
110 DATA 8,60,10,30,7,60,5,30,3,30,5,180
```

Nå kan du lese mer om lyd i den engelske håndboken.

Vi håper at denne oversettelsen var til hjelp for deg som er nybegynner.

Hvis du nå vil lære mere om programmering i BASIC språket finnes det allerede mange lærebøker på markedet.

AD Elektronikk vil kunne tilby deg en omfattende BASIC lærebok skrevet spesielt for ORIC.

Denne vil være ferdig i juli 1983.

Du kan også kjøpe ferdige programmer på kassett til din ORIC.

Du får automatisk tilsendt informasjon om dette senere.

ORIC leverer også både printer og diskstasjon til sin computer.

Kontakt enten importøren eller din forhandler og be om nærmere informasjon om dette.



AD Elektronik AS

Aasmund Vinjesvål 24
Postboks 32, Vinderen Oslo 3
Telefon 02 · 14 12 90