

**Velkommen til
introduktions-
kursus til
Interface-sproget
i AMOS Pro**

HUSK:

Interface = Brugerflade

Meningen med brugerflade er
at skabe et enkelt og
overskueligt arbejdsmiljø for
brugeren!

Brugerflade i AMOS Pro er:

- * Bokse, linier og andet grafik - Baggrund
- * Knapper - Til at trykkes på!
- * Lister - Du kan vælge med musen
- * Linealer (Sliders) - Flytte rundt i lister
- * Hypertekst - Tekst, som samarbejder
med brugeren

Notesæt til Interfacesproget i AMOSPro

Indholdsfortegnelse

1. - Interface-programmer	2
2. - Forklaring af postfix-udtryk	2
3. - Variabler	2
4. - Specielle kommandoer	3
5. - Grafik	4
6. - Knapper	6
7. - Funktioner til brug i udtryk	7
8. - Editeringszoner	8
9. - Styring af Interface fra AMOS	9
10. - Programeksempler	12
11. - Oversigt over standard-ressourcerne	15
12. - Oversigt over gennemgåede kommandoer	16

Under forklaring af kommandoer står først med fed, hvordan kommandoen skrives og dens parametres navne. Bagefter står i parentes, hvordan man forkorter kommandoen og, om det er en funktion, der skal indgå i et udtryk. Under forklaringen er parametrenes navne skrevet med fed, så de ikke forveksles. Hvis et parameternavn refererer til en anden kommando, er reference til denne angivet i parentes.

Dette notesæt er lavet af Jan Normann Nielsen 1994 for D.A.B.G. til brug ved Dansk AMOS-møde 1994. Enhver form for distribution/ offentliggørelse af teksten, det være sig i sin helhed eller enkelte dele, må kun gøres ved skriftlig tilladelse fra forfatteren. Enkelte dele af denne tekst er direkte oversættelse fra 'AMOS Professional User Guide' (C) 1992 Europress Software.

Interface-programmer

Interface-programmerne består ligesom AMOS-programmer af kommandoer. Disse skal bare adskilles af semikolon (;)- sådan:

```
kommando;
kommando;
```

...

Programmerne ligger samlet i en streng. For at gøre dem mest overskuelige, kan de skrives sådan:

```
A$="" :Rem Sletter tidligere programmer i A$
A$=A$+"BOx 0,0,1,640,256;"
A$=A$+"LIne 0,8,41,640;"
A$=A$+"EXit;"
```

Kommandoerne består af to STORE bogstaver og alt, der står med småt, bliver ignoreret. Derfor kan man fyde sine programmer med kommentarer skrevet med småt, og for at gøre dem mere letlæselige, kan man fyde kommandoerne ud med bogstaver til hele ord, så det er lettere at se, hvad der sker, ligesom ovenfor. Læg mærke til, at bemærkninger kun kan indeholde bogstaver.

Udtryk

Alle udtryk skal skrives på følgende måde: Først de enkelte operander, så operatoren. For eksempel skrives:

```
2 * 3 skrives som 2 3 *
23 + 48 skrives som 23 48 +
2+3*4-12/3 skrives 2 3 4 *12 3 / - + eller 2 3 4 * + 12 3 / -
```

Når fortolkeren møder en operator tages de to sidste operander og disse regnes på. Disse udtryk kaldes POSTFIX-udtryk.

BEMÆRK: Interfaceprogrammer kan kun benytte HELTAL og STRENGE (dvs. ingen kommatal). Strenge skrives med apostroffer omkring.

Tilladte operatorer (Taget fra 09.01.05):

Operator	Eksempel	Res.	Forklaring
+	1 2 + 3	3	Lægger to sidste tal sammen
-	2 3 -	-1	Træk det sidste tal fra det næstsidste
*	2 3 *	6	Gang de to sidste værdier sammen
/	6 2 /	3	Divider det næstsidste med det sidste
NE	2 NE	-2	Ændrer sidste tal til negativ værdi
!	'H''ej'!	'Hej'	Lægger sidste to strenge sammen
#	42 #	'42'	Gør sidste tal til en streng
MI	6 49 MI	6	Minimum af to sidste tal
MA	6 9 MA	9	Maksimum af to sidste tal

Variabler

Som standard er der 17 variabler i hvert Interface-program. Man får adgang til en variabel ved at skrive nummeret på den efterfulgt af VA. For eksempel returnerer 2VA indholdet af variabel 2, hvad enten det er en streng eller et tal.

Specielle kommandoer

Exit; (EX)

Afslutter et Interface-program. Denne kommando skal der være i ethvert program, ellers bliver der meldt fejl.

SetVariabel nummer,værdi; (SV)

Sætter variabel nummer nummer til at indeholde værdien værdi, som kan være en streng omsluttet af apostroffer eller et heltal. Husk, at der som standard kun er 17 variabler (0-16).

RunUntil forsinkelse,flag; (RU)

Standser programudførslen og venter. forsinkelse er antal 1/50-sekunder, der skal ventes før Interface-progåmmet fortsætter. flag er et binært tal med fire cifre, der betyder:

Bit 0 - Sletter tastaturbufferen før der testes på den. Svarer til CLEAR KEY-kommandoen

Bit 1 - Ignorerer alle de klik på musen, der har været før

Bit 2 - Venter til en tast trykkes

Bit 3 - Venter til en museknap trykkes

De enkelte funktioner slås til med 1 og fra med 0. Fx

RunUntil 500,%0101;

venter 10 sekunder eller til en tast trykkes.

Hvis både forsinkelse og flag er nul, slutter programmet, når der trykkes på en afslutningsknap. Se BQ-kommandoen.

Grafik-kommandoer**BAse x,y; (BA)**

Sætter udgangspunkt for koordinater til alle efterkommende kommandoer. x,y bliver for fremtiden 0,0.

SIze bredde,højde; (SI)

Sætter størrelsen af det område, man vil arbejde med på skærmen. Sammen med BAse-kommandoen har man entydigt defineret dette område på skærmen, og derfor kan man bruge næste kommando:

SAve nummer;

Husker på det område, der er defineret af BAse- og SIze-kommandoerne, så dette bliver genskabt nøjagtigt som det er, når Interface-programmet er færdigafviklet. nummer er nummeret på den blok, man vil definere.

PRint x,y,værdi,farve; (PR)

Skriver værdi, som kan være et heltal, en streng eller et udtryk, ud på skærmen på position x,y med farven farve.

PrintOutline x,y,værdi,kantfarve,farve; (PO)

Ligesom PRint-kommandoen, men tegner en kant rundt om alle bogstaverne med kantfarve.

INK tegne,baggrund,kant; (IN)

Skifter de aktuelle farver, der tegnes med. tegne er den farve, der bruges til at tegne med (fx kasser), baggrund er den farve, baggrund tegnes med, og kant bruges fx til at tegne kanter på udfyldte kasser. Negativt tal angiver, at denne farve ikke skal ændres fra før.

GraphicBox x1,y1,x2,y2; (GB)

Tegner en udfyldt kasse med den aktuelle tegnefarve med øvre venstre hjørne på pos. x1,y1 og nedre højre hjørne x2,y2. Hvis kantvalg er sat til (se SetPattern-komandoen), tegnes der en kant rundt om kassen med kantfarve (se INK-komandoen.)

GraphicSquare x1,y1,x2,y2; (GS)

Tegner en firkant med tegnefarven.

GraphicLine x1,y1,x2,y2; (GL)
Tegner en linie fra (x1,y1) til (x2,y2).

Graphic Ellipse x,y,radius1,radius2; (GE)
Tegner en ellipse med centrum i x,y med radier radius1 og radius2. Hvis radius1=radius2 tegnes en cirkel.

SetPattern mønster,kantvalg; (SP)
Vælger mønster til udfyldning i bl.a. kasser. Hvis kantvalg sættes til 1, tegnes der en kant med kantfarve (se INk-kommandoen) rundt om alle udfyldte kasser. Slå dette fra med 0 som kantvalg.

SetFont nummer,snit; (SF)
Virker som den almindelige Set Font-kommando. Angiv nul som nummer, hvis du kun vil ændre snittet.

SetWriting status; (SW)
Virker som Gr Writing-kommandoen. (Se kap. 6.4)

VertText x,y,strenge,farve; (VT)
Skriver teksten lodret med de enkelte bogstaver oven på hinanden.

UNpack x,y,billedet; (UN)
Placerer billedet nummer billedet fra Ressource-banken på position x,y. x-koordinaten rundes ned til nærmeste 8.

LIne x1,y,billedet,x2; (LI)
Tegner en linie mellem x1,y og x2,y med tre ressourcer, hvoraf den første angives i billedet. x-koordinaten rundes ned til nærmeste 8.

VLine x,y1,billedet,y2; (VL)
Som LIne-kommandoen, bare en lodret linie. Højden af linien (y2-y1+1) bliver automatisk rundet op til nærmeste 3.

BOx x1,y1,billedet,x2,y2; (BO)
Tegner en kasse mellem x1,y1 og x2,y2 med ni ressourcer, hvoraf den første angives i billedet. x-koordinaten rundes ned til nærmeste 8.

nME (ME-Funktion)
Returnerer streng nummer n fra Ressource-banken. Disse strenge indtastes i Ressource-editoren.

Bemærk: Ved LIne, VLine og BOx er det i manualen skrevet, at man skal angive længder og bredder, men det er ikke rigtigt - man skal angive slutkoordinat(er).

Knapper

BUTTON nummer,x,y,brd,højde,status,min,max;[tegn][ændr] (BU)
 Definerer en knap på skærmen, som brugeren kan trykke på. nummer er nummeret på den knap, der skal defineres. x,y er øverste venstre hjørne på knappen (ud fra BAse-koordinaterne). brd,højde er bredden og højden af knappen i pixels. Knapper kan have mere end en status, fx trykket eller ikke-nedtrykket. Med status bestemmer du, hvilken status knappen skal have fra start og min og max sætter den mindste hhv. største status en knap kan have. Når knappen trykkes på, bliver status gjort en større, og hvis denne værdi er større end max bliver den sat til min, men alt dette kan du selv styre ved hjælp af: tegn, som er et Interface-program, der kaldes, hver gang knappen skal tegnes på skærmen (dvs. når den trykkes på den), og ændr er et Interface-program, der kaldes, hver gang, man har trykket på knappen og slippet musen, med mindre man bruger NW-kommandoen (se denne). Det sidste program kan være ingenting. Bemærk: I disse to programmer er altting lokalt, dvs. koordinater beregnes ud fra knappens start-koordinater og SX- og SY-kommandoerne bliver størrelsen af knappen. Bemærk også: Flere knapper kan gives samme nummer uden problemer. I stedet for forskellige knapper bliver de så forskellige objekter af samme knap (Se Rdialog-kommandoen).

ButtonQuit; (BQ)

Bruges i ændr-programmet til at angive, at denne knap afslutter programmet, hvis den bliver trykket. Har kun effekt, hvis der bruges Dialog-bokse (Se Dialog Box- og RunUntil-kommandoerne).

ButtonPosition (BP-Funktion)

Returnerer den aktuelle status af knappen. Bruges i tegn-programmet.

ButtonReturn status; (BR)

Ændrer knappens status til status. Bruges i ændr-programmet.

ButtonChange nummer,status (BC)

Ændrer status af en anden knap, der er defineret. Hvis flere knapper er defineret med samme nummer påvirker det alle knapperne på nær den aktuelle, hvis du er i ændr-programmet.

Keyboard ascii,shift (KY)

Bruges umiddelbart efter definition af knap og den tilknytter en tast til knappen. Hvis denne tast trykkes, er det det samme som at trykke på knappen. ascii er ASCII-værdien for tasten, og hvis du i stedet vil angive en skankode, skal du lægge 128 til. shift er et binært tal, der angiver, hvilke styretaster der skal trykkes ned også.

Bit 0=Venstre shift	Bit 1=Højre shift	Bit 2=Caps Lock
Bit 3=Control	Bit 4=Venstre alt	Bit 5-Højre alt
Bit 6=Venstre Amiga (eller Commodore)		Bit 7=Højre Amiga.

BEMÆRK: Kun en Shift-tast kan testes ad gangen.

Fx:

KY 65,%00100100;
 tilknytter Caps+Højre alt+"A" til den sidste definerede knap
 KY 65,%11001000;
 vil aldrig kunne aktivere den sidst definerede knap (Hvorfor?)

NoWait (NW)

Angiver, at knappen aktiveres ligeså snart brugerens trykker på tasten og ikke til han slipper knappen, dvs. ændr-programmet udføres med det samme. Bruges som første kommando i ændr-programmet. (Se BUtton-kommandoen)

Funktioner til brug i udtryk

nVA (VA)

Returnerer indholdet af variabel nummer n. Der kan som standard bruges 17 kommandoer i hvert Interface-program, men dette kan ændres med Dialog Open-kommandoen (Se denne).

BaseX (BX), BaseY (BY)

Returnerer tallene sat med BAse-kommandoen, hhv. x- og y-værdien.

ScreenWidth (SW), ScreenHeight (SH)

Returnerer bredden hhv. højden af den aktuelle skærm.

SizeX (SX), SizeY (SY)

Returnerer størrelsen af det sidst definerede område med SIZe-kommandoen. Bruges mest til centrering. Hvis du vil have en boks med størrelsen 160x64 centreret på skærmen, skal du gøre sådan:

```
SIZE 160,64;
BAse SW SX -2/,SH SY -2/;
SAve 1; kan evt udelades
```

...
 (Bemærk, at centreringsangivelsen i manualen ikke virker!)

XB, YB, XA ,YA (Funktioner)

Returnerer de aktuelle hhv. forrige koordinater for grafikcursoren.

TextWidth (TW), TextHeight (TH), TextLength (TL), CentreX (CX)

Disse fire funktioner bruges i udtryk efter en streng.

streng TW : Antal pixels teksten vil fylde i bredde på skærmen

streng TH : Antal pixels teksten vil fylde i højden på skærmen

streng TL : Antal tegn i strengen

streng CX : Den X-koordinat, teksten skal starte på, hvis den skal centreres på skærmen (eller i en knap, hvis kommandoen bruges i tegn-programmet (se BUtton))

Editeringszoner

EDitzone zonenummer,x,y,bredde,max,strenge,baggrund,tfarve; (ED)
 Opretter en zone, hvor brugeren kan indtaste tekst. zonenummer er nummeret på den zone, der skal oprettes. x,y er positionen på skærmen i forhold til BAse-koordinaterne. X-koordinaten rundes ned til nærmeste 16 pixels. bredde er bredden af feltet i antal karakterer. Dette tal rundes automatisk af til nærmeste 2. max er det maksimale antal tegn, der kan indtastes i feltet. strenge er en streng eller variabel, der skrives i feltet fra starten. baggrund og tfarve er baggrunds- hhv. tekstfarve for feltet. Hvis der tillades flere tegn i editeringszonen end bredden (**max > bredde**) flytter teksten til højre og giver plads til nye tegn. Hvis man trykker [Tab] eller [Return], flytter man markøren til den næste editeringszone (i erklæringsrækkefølge) og et tryk ved den sidste zone, bliver ingen zoner aktiveret, men i stedet kan tastetrykket aktivere en tast (med KY-kommandoen). Endnu et tryk på [Tab] aktiverer den første editeringszone igen. Indholdet af editeringszonen kan aflæses fra AMOS-programmet med Vdialog\$-funktionen. Hvis man vil have tilgang til editeringsvinduet fra AMOS-programmet har vinduet nummer 1000+z, hvor z er nummeret på den zone, editeringsfeltet har.

DIGit zonenummer,x,y,bredde,værdi,flag,baggrund,tfarve;
 Opretter et felt ligesom EDit, men her kan man kun indtaste tal (heltal). Se denne kommando for forklaring af zonenummer, x, y, bredde, baggrund og tfarve. værdi er, hvad der skal stå i feltet fra starten. flag=1 angiver, at værdi skal skrives fra starten, men med flag=0 skrives tallet ikke i feltet. Værdien, der er indtastet i feltet kan aflæses med Rdialog-funktionen fra AMOS-programmet.

ZoneNumber (ZN-Funktion)

Returnerer nummeret den aktuelle zone. De zoner, der normalt er tale om, er editeringszoner, dvs. ZN returnerer nummeret på den zone, cursoren står i.

ZonePosition (ZP-Funktion)

Aflæser en zones status, fx en knap, et andet sted i programmet end i tegn-programmet, hvor BP bruges (se BUtton).

ZoneChange nummer,ny værdi; (ZC)

Ændrer indholdet af zone nummer nummer til ny værdi. Påvirker:

Knapper : Knappen får status svarende til ny værdi og tegnes

Editeringszoner : Ny værdi kommer til at stå i feltet.

Linealer og lister : Se manualen

Styring af Interface fra AMOS-Programmer

Resource Bank nummer

Angiver, i hvilken bank, der ligger de ressourcer, du vil benytte. Skal sættes i starten af programmet. Hvis banken ikke eksisterer, vil standard-banken blive brugt, på samme måde som hvis kommandoen slet ikke bruges.

Resource Screen Open nummer,bredde,højde,blink

Åbner skærm nummer nummer med størrelsen bredde x højde som den normale Screen Open-kommando. Men antallet af farver og oplossningen findes i ressource-banken (så den svarer til ressourcerne). blink angiver hvilken farve, du vil have til at blinke. Hvis denne sættes til nul blinker ingen farver.

Streng\$=Resource\$(nummer)

Virker som ME-funktionen (se denne), der bruges i Interface-programmet, men denne kommando bruges jo i selve AMOS-programmet. nummer er nummeret på den streng, du vil have.

Resource Unpack nummer,x,y

Virker som UNpack-kommandoen (se denne), men denne kommando bruges i AMOS-programmet.

knap=Dialog Box(program\$)

knap=Dialog Box(program\$,værdi,strengh,y)

Første udgave af kommandoen afvikler Interface-programmer, der ligger i program\$ og venter på, at EXIT-kommandoen nås eller af afslutningsknap (se BQ-kommandoen) er trykket. Dernæst returneres nummeret på den sidst nedtrykkede knap i knap. Hvis et område er husket med SAve-kommandoen, gendannes det efter Interface-programmets afslutning. Den anden udgave af kommandoen gør ligesådan, men værdi, som er et heltal (evt. i en variabel), overføres til variabel 0 og strengh, som er en streng, overføres til variabel 1. Samtidig bliver BAse-værdierne sat til x og y, dvs. virker som om der står en BAse-kommando i starten af programmet. Hvis brugeren ikke skal kunne afbryde dialogboksen med Ctrl-C, skal du bruge Break Off i AMOS-programmet.

Ulempen ved denne kommando er, at AMOS-programmet går i stå. Det kan forhindres med den næste kommando:

Dialog Open kanal,program\$

Dialog Open kanal,program\$,antalvar,buffer

Åbner en kommunikationskanal, hvori Interface-programmet program\$ kan udføres og styres. Der kan også angives to ekstra parametre, og antalvar er antallet af tilgængelige variabler i programmet (normalt 17), og buffer er det antal bytes, der skal afsættes til zoner osv. (Normalt 1k=1024 bytes). Denne kommando gør intet tilsyneladende, men bruges i samarbejde med:

knap=Dialog Run(kanal)
knap=Dialog Run(kanal,label,x,y)

Afvikler Interface-programmet, der er defineret i kanal nummer kanal via Dialog Open-kommandoen. Som variant af kommandoen kan du angive tre ekstra parametre: label, som angiver, hvor programmet skal startes fra (se manualen), og x,y som fungerer på samme måde som Dialog Box-kommandoen. Hvis der trykkes en knap ned, mens Interface-programmet afvikles, returneres det i knap. Når Interface-programmet er færdigt, bliver kontrollen overladt til AMOS-programmet igen.

Dialog Close

Dialog Close kanal

Første udgave lukker alle kanaler, mens anden udgave kun lukker kanal nummer kanal. Hvis der er brugt en SAVe-kommando i Interfaceprogrammet, bliver området gendannet.

knap=Dialog(kanal)

Aflæser, hvad der er sket siden sidste kald. knap vil indeholde nummeret på den sidst nedtrykkede knap, hvis en sådan er trykket på:

knap<0 : Enten er kanalen ikke åbnet, eller programmet er slut

knap=0 : Der er ikke sket noget

knap>0 : Dette er nummeret på den sidst nedtrykkede knap

Bemærk: Hvis brugeren har trykket RETURN i en af editeringszonerne, returneres nummeret på denne i knap!

Dialog Update kanal,zonenummer

Dialog Update kanal,zonenummer,param

Giver dig mulighed for at styre dit Interface-program fra AMOS. Hvis du bruger første udgave, kan du gentegne en knap eller en editeringszone ved at angive det tilsvarende zonenummer. Den anden udgave giver dig mulighed for at ændre:

Knapper : param (tal) bliver den nye status af den valgte knap

ED-zone : param (streng) bliver skrevet i editeringsfeltet

DI-zone : param (tal) bliver skrevet i editeringsfeltet

Du skal selvfølgelig også her angive det korrekte zonenummer.

Dialog Freeze

Dialog Freeze kanal

Fryser et/flere Interface-programmer, så der ikke kan trykkes på knapper eller skrives i editeringszoner. Den første udgave fryser alle kanaler, mens den anden kun fryser kanal. Affrysese igen med Unfreeze-kommandoen (nedenfor).

Dialog Unfreeze

Dialog Unfreeze kanal

Affryser tidligere frossede kanaler ligesom Dialog Freeze-kommandoen ovenfor.

Dialog Clr kanal

Fjerner alt, hvad Interface-programmer i kanalen kanal har lavet, og evt. gentegner området, hvis Save-kommandoen er brugt. Herefter er Interface-programmet inaktivt og skal startes op igen med Dialog Run-kommandoen.

```
knap=Rdialog(kanal,knapnummer)
```

```
knap=Rdialog(kanal,knapnummer,objekt)
```

Læser den aktuelle status af en knap i den angivne kanal eller indholdet af en editoringszone til tal (se Digit-kommandoen). Her er knapnummer nummeret på den zone, der skal aflæses. Hvis det valgte knapnummer har flere knapdefinitioner (se Button-kommandoen), skal du angive objektnummeret, som går fra 1 og opad og svarer til erklæringsrækfølgen af de enkelte objekter.

```
streng$=Rdialog$(kanal,zonenummer)
```

Læser indholdet af en editoringszone til tekst. Virker som Rdialog-kommandoen ovenfor.

```
værdi=Vdialog(kanal,variabelnummer)
```

```
Vdialog(kanal,variabelnummer)=værdi
```

Første udgave af kommandoen læser indholdet af en variabel i et Interface-program, der er under afvikling. Bemærk, at du får nul returneret, hvis variablen var en streng. Anden udgave sætter værdien af en variabel til værdi. Se SetVar-kommandoen.

```
Vdialog$(kanal,variabelnummer)=streng$
```

```
streng$=Vdialog$(kanal,variabelnummer)
```

Læser en streng, der ligger i en variabel i interface-programmet. Se Vdialog-kommandoen ovenfor.

```
position=Edialog
```

Hvis der er opstået en fejl i Interface-programmet under afviklingen, får du her den position i strengen, hvor fejlen er fundet. Du kan så skrive:

```
Print Mid$(A$,Edialog,80)
```

Hvis programmet ligger i A\$ for at se, hvor i programmet, fejlen er. (Bemærk: Denne kommando virker først ordentlig fra v1.12)

Oversigt over de (læs:Jans) hyppigste fejl:

- Husk EXIT-kommandoen til sidst i programmet.
- Intet semikolon efter den sidste klamme i Button-kommandoen.
- Husk semikolon efter SetVar-kommandoen.
- Husk at angive skrifefarve i PRint-kommandoen.

```

' Projekt 1
' -----
' Der skal laves et Interface-program, der kan lave en dialogboks, der
' giver brugeren en besked, hvorefter han trykker på [Ok]-knappen.

' Klargør skærm. Husk Wait Vbl, når der er dialog-bokse med det samme!
Resource Screen Open 0,640,256,0 : Cls 0 : Wait Vbl
A$=""
A$=A$+"SetVar 0,'Intern meddeelse';
A$=A$+"SIZe 1VA TW 96 +160 MA,72;
A$=A$+"BAse SW SX -2/,SH SY -2/;
A$=A$+"SAve 1;";
A$=A$+"BOx 0,0,1,SX,SY;
A$=A$+"POutlin OVA CX,8,OVA,0,7;
A$=A$+"LIne 0,YB 1+,10,SX;
A$=A$+"POutlin 1VA CX,YB 6+,1VA,0,3;
A$=A$+"BUtton 1,SX 64 -2/,SY 20-,64,16,0,0,1; [UNpack 0,0,13BP+;
A$=A$+" POutlin 3ME CX,4,3ME,4,0;
A$=A$+" ][BReturn 0;BQuit];
A$=A$+"RunUnti 0,0;" 
A$=A$+"EXit;"
```

overskrift"
størrelse af boks"
centrer"

tegn boksen"
skriv overskrift"
understreg overskrift"
skriv besked"
ok knap"
tegn knappen"
skriv ok i knap"
knap afslutter prg"

```
D=Dialog Box(A$,0,'Dette er en prøve')
```

```

' Projekt2
' -----
' Der skal laves et Interface-program, der kan lave en dialog-boks, der
' består af en overskrift, en besked og en eller to knapper, der kan
' trykkes på. Alle disse ting skal selv kunne vælges.
```

```
Resource Screen Open 0,640,256,0 : Cls 0
BESKED[1,"Intern besked",7,"Computeren er gået ned!",3,"Ok",4,"Afbryd",4]
```

```

Procedure BESKED[KANAL,OVER$,C1,BESK$,C2,KNAP1$,C3,KNAP2$,C4]
A$=""
A$=A$+"SIZe OVA TW 1VA TW MA 160 MA 64+,72;      vælg størrelse af boks"
A$=A$+"BAse SW SX -2/,SH SY -2/;                centrer boksen"
A$=A$+"SAve 1;";
A$=A$+"BOx 0,0,1,SX,SY;
A$=A$+"PRint OVA CX,8,OVA,"+Str$(C1)+";
A$=A$+"LIne 0,YB 1+,10,SX;
A$=A$+"PRint 1VA CX,YB 8+,1VA,"+Str$(C2)+";
A$=A$+"BUtton 1,16,SY 20-,64,16,0,0,1;           første knap"
A$=A$+" [UNpack 0,0,13 BP+;"                     tekst på første knap"
A$=A$+" PRInt 2VA CX,4,2VA,"+Str$(C3)+";]       tekst på anden knap"
A$=A$+" ][BReturn 0;]"
```

If KNAP2\$<>""

A\$=A\$+"BUtton 2,SX 16- 64-,YA,64,16,0,0,1; anden knap"
A\$=A\$+" [UNpack 0,0,13 BP+;"

A\$=A\$+" PRInt 3VA CX,4,3VA,"+Str\$(C4)+";] tekst på anden knap"
A\$=A\$+"][BReturn 0;]"

End If

A\$=A\$+"EXit;"

afslut"

' Klargør program
Dialog Open KANAL,A\$

' Overfør strenge til program og start
Vdialog\$(KANAL,0)=OVER\$: Vdialog\$(KANAL,1)=BESK\$
Vdialog\$(KANAL,2)=KNAP1\$: Vdialog\$(KANAL,3)=KNAP2\$
D=Dialog Run(KANAL)

' Vent på en knap trykkes

```
Repeat
  D=Dialog(KANAL)
Until D>0
```

' Luk og afslut
Dialog Close KANAL
End Proc[D]

' Projekt 3

' -----
' Der skal laves felter, hvor brugeren kan indtaste sit navn (30 tegn),
' adresse (30 tegn), by (20 tegn) og tlf.nr. (8 tegn). Samtidig skal
' det angives, om vedkommende har en computer eller ej. Til sidst skal
' der trykkes på [Ok] for at godkende indtastningerne (eller Return)
' Programmet vil ikke acceptere navne, der begynder med Kurt.

A\$="" : Rem Sletter tidligere programmer
A\$=A\$+"SSize 480,96;
A\$=A\$+"BAse SW SX -2/,SH SY -2/;
A\$=A\$+"SAve 1;
A\$=A\$+"BOX 0,0,1,SX,SY;
A\$=A\$+"SetVar 0,'Indtast personlige data';
A\$=A\$+"POutlin OVA CX,8,OVA,0,7;
A\$=A\$+"LIne 0,YB 1+,10,SX;
A\$=A\$+"PRint 48,YB 8+,'Navn:',3;"
A\$=A\$+"LIne 128 8-,YA 4-,17,128 32 8*16++;
A\$=A\$+"EDitzon 1,128,YA 4+,32,30,'',0,3;
A\$=A\$+"PRint 48,YB 12+,'Adresse:',3;"
A\$=A\$+"LIne 128 8-,YA 4-,17,128 32 8*16++;
A\$=A\$+"EDitzon 2,128,YA 4+,32,30,'',0,3;
A\$=A\$+"PRint 48,YB 12+,'Tlf.nr.:',3;"
A\$=A\$+"LIne 128 8-,YA 4-,17,128 9 8*16++;
A\$=A\$+"DIgit 3,128,YA 4+,9,0,0,0,3;
A\$=A\$+"PRint XA 112+,YA,'Computer?',3;"
A\$=A\$+"BButton 4,XB 8+,YA 2-,24,12,0,0,1;
A\$=A\$+" [UNpack 0,0,15 BP+;][]"
A\$=A\$+"BButton 5,SX 16- 64-,SY 22-,64,16,0,0,1;
A\$=A\$+" [UNpack 0,0,13 BP+;]"
A\$=A\$+" SetFont 1,%010;
A\$=A\$+" POutlin 3ME CX,4,3ME,0,4;"
A\$=A\$+" SetFont 2,0;
A\$=A\$+"][BReturn 0;]
A\$=A\$+"Keyboar 13,0;
A\$=A\$+"EXit;

størrelse af boks"
centrer boks"
gandan området efter slut"
tegn boksen"
overskrift"
centrer overskrift"
understreg overskrift"

kant"
første editeringszone"

kant"
anden editeringszone"

kant"
tredje editeringszone"

ja nej knap"
ok knap"

stor skrift fed"
normal skrift"
slippes igen"
return aktiverer knap"
afslut"

```
' Klargør skærm, fonte og program
Resource Screen Open 0,640,256,0 : Cls 0
Get Rom Fonts
Dialog Open 1,A$
D=Dialog Run(1)

Repeat
  D=Dialog(1)
  NAVN$=Rdialog$(1,1)
  If Lower$(Left$(NAVN$,4))="kurt"
    Dialog Update 1,1,"XXXX"+Mid$(NAVN$,5)
  End If
Until D=5 : Rem Vent på, at der trykkes på [Ok]-knappen

' Her kan data nu aflæses
NAVN$=Rdialog$(1,1)
ADR$=Rdialog$(1,2)
TELEFON=Rdialog(1,3)
COMPUTER=(Rdialog(1,4)=1) : Rem True eller false

Dialog Close 1
' Her kan programmet fortsætte...
```

Oversigt over standardressourcer

Nummer	Type	Størrel.	Bruges til
1	Boks	16 x 8	Kasser (baggrund/knapper)
10	HLinie	16 x 8	Understregning af tekst
13/14	Elem.	64 x 16	Stor knap *
15/16	Elem.	24 x 12	Ja/Nej-knap (Tændt/Slukket) *
17	HLinie	8 x 16	Kant omkring editeringzoner
22/23	Elem.	24 x 10	Afbryd-knap [X] *
31	HLinie	16 x 4	Smal linie (understregning/kant)
34/35	Elem.	16 x 7	Lille pil op *
36/37	Elem.	16 x 7	Lille pil ned *
38	VLine	16 x 4	Baggrund til linealer (passer til 34-37)
41/44	HLinie	16 x 10	Smalle knapper (forskellig bredde) *
47/50	HLinie	16 x 8	Baggr. til linealer (passer til 53-56) *
53/54	Elem.	16 x 8	Pil til venstre *
55/56	Elem.	16 x 8	Pil til højre *
57/60	VLinie	16 x 4	Baggr. til linealer (passer til 63-66) *
63/64	Elem.	16 x 8	Stor pil op *
65/66	Elem.	16 x 8	Stor pil ned *

Forklaring: Type angiver, hvilken kommando, der skal tegne ressourcen: Boks (BOx), HLinie (LIne), VLinie (VLine) og Elem. (UNpack). Størrelse angiver de tal, der BØR gå op i bredde/højde af kasser/linier for at få det bedste resultat. Sidste kolonne (Bruges til) giver en kort forklaring af, hvad ressourcen bruges til. Hvis der er en * angiver det, at det første nummer er ressourcen normalt, mens det andet nummer er ressourcen nedtrykket (til knapper).

Oversigt over Interface-kommandoer

BA.....BBase (K)	4	PR.....PRint (K)	4
BC.....ButtonChange (K)	6	RU.....RunUntil (K)	3
BO.....BBox (K)	5	SA.....SAve (K)	4
BP.....ButtonPosition (F) ..	6	SI.....SIZe (K)	4
BQ.....ButtonQuit (K)	6	SP.....SetPattern (K)	5
BR.....ButtonReturn (K)	6	SF.....SetFont (K)	5
BU.....BUtton (K)	6	SH.....ScreenHeight (F)	7
BX/BY..BaseX/BaseY (F)	7	SV.....SetVariabel (K)	3
CX.....CentreX (F)	7	SW.....SetWriting (K)	5
DI.....DIGitzone	8	SW.....ScreenWidth (F)	7
ED.....EDitzone	8	SX/SY..SizeX/SizeY (F)	7
EX.....EExit (K)	3	TH.....TextHeight (F)	7
GB.....GraphicBox(K)	4	TL.....TextLength (F)	7
GE.....GraphicEllipse (K) ..	5	TW.....TextWidth (F)	7
GL.....GraphicLine (K)	5	UN.....UNpack (K)	5
GS.....GraphicSquare (K) ...	4	XA/YA..Gamle koord (F)	7
IN.....INK (K)	4	XB/YB..Aktuelle koord (F) ..	7
KY.....KeYboard (K)	6	VA.....VARIABEL (F)	7
LI.....LIne (K)	5	VL.....VertLine (K)	5
ME.....MEssage (F)	5	VT.....VertText (K)	5
NW.....NoWait (K)	7	ZC.....ZoneChange (K)	8
PO.....PrintOutline (K)	4	ZN.....ZoneNumber (F)	8
		ZP.....ZonePosition (F)	8

Oversigt over AMOS Pro-kommandoer

.....Dialog (F)	10Dialog Update (K) ...	10
.....Dialog Box (F)	9Edialog (F)	11
.....Dialog Close (K)	10Rdialog (F)	11
.....Dialog Clr (K)	11Resource\$ (F)	9
.....Dialog Freeze (K) ...	10Resource Bank (K) ...	9
.....Dialog Open (K)	9Res. Screen Open (K) .	9
.....Dialog Run (F)	10Resource Unpack (K) .	9
.....Dialog Unfreeze (K) .	10Vdialog (F)	11

(K) - Det er en kommando.

(F) - Det er en funktion.