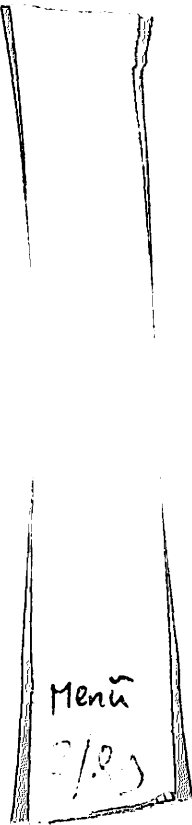


Diskette/Platte in Laufwerk C besitzt den Kennsatz S01
Verzeichnis von C:\CLV\VF\C16

.		<VER>	28.06.89	16.02
..		<VER>	28.06.89	16.02
MENUE		174080	27.02.89	13.03
EH-TERM	387	174081	26.02.89	22.07
UF38-1		174080	26.02.89	22.50
UF38-2		174080	26.02.89	23.40
UF38-3		174080	27.02.89	0.25
UF38-4		174080	27.02.89	2.48
TC38		174080	27.02.89	10.19
EDIT		174080	27.02.89	11.17
DSK-GRF	487	174080	27.02.89	13.55
EXE		<VER>	28.06.89	16.20
	12 Datei(en)		9211904 Bytes	noch frei!



owner='S01:WK:HFI'

SSSSS	00	1			W	W	K	K		H	
S	S	0	0	11		W	W	K	K	H	
S		0	0	1 1	**	W	W	K	K	H	
S		0	0	1	**	W	W	K	K	H	
SSSSS	0	0	1			W	W	W	K	K	HHHHH
	S	0	0	1		W	W	W	KKK		H
	S	0	0	1	**	W	W	W	K	K	H
	S	0	0	1	**	WW	WW	K	K	**	H
S	S	0	0	1		W	W	K	K		H
SSSSS	00	111				W	W	K	K		H

Despooler of 3Share - version 1.3.1

file spooled: 28-jun-89, 16:22:52
file printed: 28-jun-89, 16:40:32
copies=1, priority=50, form type=1

6

91

```

0 \ SU: scr@/! clv06jul88
1
2 &40 Constant cols
3
4 : scr@ ( y x --n1 n2) swap cols * +
5   dup $800 + c@ swap $c00 + c@ ;
6
7 : scr! ( n1 n2 y x-- ) swap cols * +
8   under $c00 + c! $800 + c! ;
9
10 \\ fuer Vorsichtige:
11
12 : ?scr ( adr--adr)
13   dup $800 $1000 uwithin ?exit
14   u. true abort" oh,n!!" ;
15 : scr! ( n1 n2 y x-- )
16   at? &15 0 at .s at
17   swap cols * +
18   under $c00 + ?scr c! $800 + ?scr c! ;
19
20
21
22
23
24

```

7

92

```

0 \ SU: clv06jul88
1
2 \\ will alles nicht
3 &40 Constant cols
4
5 forth forget cpush
6 : cpush 2dup 20 0 at
7   key 3 = abort" break"
8   2dup [ tools ] cpush 5 fill ;
9
10
11
12 : scrpush ( y x h w-- )
13   2swap swap cols * + $c00 + swap
14   ( h adr w )
15   BEGIN rot dup WHILE 1- -rot
16     2dup cpush under + swap
17     REPEAT 2drop drop ;
18
19
20 \\
21 debug scrpush
22
23 5 6 7 8 scrpush
24

```

8

93

```

0 \ MN: sample clv10judst
1
2 &2 &06 &3 &10 Menu: doOne
3   ." 1 is One" ;
4
5
6 &12 &04 &6 &09 Menu: doTwo
7   0 4 at ." 2" 1 3 at ." is"
8   2 2 at ." two" ;
9
10 0 0 &25 &40 ww: wholeScreen
11 : edi wholeScreen wwSave r wwRestore ;
12
13 &12 &04 &10 &23 Menu: doFrosch
14   ." Froesche sind nett" ;
15
16 -->
17
18
19
20
21
22
23
24

```

```

0 \ clv10feb89 @ @ | A ^ A C A U A E A A A A A A A A C A B A E A O A I A T A I A A A A A E
1
2
3 &8 &11 &7 &17 Menue: takeit
4   <s . " Choice One" s> doOne
5   cr <s . " Choice Two" s> doTwo
6   cr <s . " Information" s> info
7   cr <s . " Editor" s> edi
8   cr <s . " Froesche" s> doFrosch
9   3 3 2 2 [ ] words sel! ;
10 &8 &11 &6 &17 Menue: t1
11 cr . " ...lahm, was?" takeit ;
12 &3 &5 &10 &25 Menue: big
13 cr cr . " Dies ist ein fest" cr
14 cr cr . " Na, wie ist das Fenster?"
15 t1 ;
16 : t . " Achtung! Screen &50-60 werden "
17 cr . " veraendert. RET to continue"
18 key #cr - ?exit big ;
19 t
20
21
22
23
24

```

```

0 \ MN: Menue Loadscreen cclv10jul88 clv06jul88
1
2
3
4       2 load \ d- split join
5       3 load \ vArray
6       4 load \ Listen / nil
7       6 load \ scr@ scr!
8       2 &14 +thru \ Windows
9       &20 &24 +thru \ Items
10
11 ' keySel Is select \ mouse not yet
12
13 \      1 +load \ sample
14
15
16 \\ Nachteil:
17 - bei Schachtelung des gleichen Menues
18   wird's immer kleiner
19 - bei falschen Cursorpositionen-> Crash
20 - Schreiben, Zeigen nur auf's aktuelle
21   Fenster moeglich.
22
23
24

```

```

0 \ MN: sample clv10jul88 \ LL: sample clv06jul88
1
2 &2 &06 &3 &10 Menue: doOne      5 Constant listlen
3   . " 1 is One" ;                NewList myList listlen 2* allot
4 &12 &04 &6 &09 Menue: doTwo
5   0 4 at . " 2" 1 3 at . " is"
6   2 2 at . " two" ;
7 0 0 &25 &40 ww: wholeScreen      : linkall nil myList !
8 : edi wholeScreen wwSave r wwRestore ; listlen 1 DO I nth myList link! LOOP ;
9 &8 &11 &6 &17 Menue: takeit
10  <s . " Choice One" s> doOne      : .l myList .link ;
11  cr <s . " Choice Two" s> doTwo
12  cr <s . " Information" s> info
13  cr <s . " Editor" s> edi ;
14 &8 &11 &6 &17 Menue: t1
15  cr . " ...lahm, was?" takeit ;
16 &3 &5 &10 &25 Menue: big
17  cr cr . " Dies ist ein fest" cr
18  cr cr . " Na, wie ist das Fenster?"
19  t1 ;
20 : t . " Achtung! Screen &50-60 werden "
21  cr . " veraendert. RET to continue"
22  key #cr - ?exit big ;
23 t
24

```

12

97

```

0 \ MN: leap (>)select(ed)   clv10jul88
1
2 : leap  rdrop rdrop rdrop rdrop ;
3
4 Defer select   Create: >select select ;
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

13

98

```

0 \ MN: Variables           clv10jul88
1
2 Variable act      act off
3 \ zeigt auf mit Menue: deklarierte
4 \ Window-Parameter des aktuellen WW
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

14

99

```

0 \ MN: room                clv10jul88
1
2 &10 $400 * Constant wwLen
3 &50 $400 * vrange !
4 &60 $400 * vrange 2+ !
5 \ block 50 to 60 reserved for >room
6
7 wwLen VArray >room
8
9 \ pointer into the >room-space:
10
11 Variable wwP  wwP off \ uni-pointer
12 Variable selP selP off
13 \ first selectable Item
14 Variable selected selected off \
15 \ now selected Item
16 ; Variable lastWW lastWW off
17 \ linked list of saved screenparts
18
19
20
21
22
23
24

```


21

106

```

0 \ MN: Format of savings      clv10jul88
1
2
3 \\ The format which wwSave/Restore use
4
5 w:pointer to last saving OR 0
6 w:outputVektor
7 w:selP - selecteable-region-pointer
8 w:selected - last highlighted
9 2 w:CursorX CursorY
10 4 w:wwX wwW wwY wwH - Window-rectangle
11 N w:contents of window / N=wwW*wwH
12
13
14
15
16
17
18
19
20
21
22
23
24

```

22

107

```

0 \ MN: wwopen                clv10jul88
1
2 \ +! wegen .wwbord
3
4 : wwOpen
5   wwSave display .wwBord
6   1 wwX +! 1 wwY +! -2 wwW +! -2 wwH +!
7   wwOut page ;
8
9 : wwClose
10  -1 wwX +! -1 wwY +! 2 wwW +! 2 wwH +!
11  wwRestore ;
12
13
14
15
16
17
18
19
20
21
22
23
24

```

23

108

```

0 \
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```


30

115

```

0 \ IT:empty          clv10jul88
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

31

116

```

0 \ IT:sel! noSel sel  clv10jul88
1
2 \\ used only for MOUSE (later):
3   not tested!!
4
5 &8 &11 &6 &17 Menue: noSel
6   cr cr ." not possible"
7     cr ." press any key" key drop ;
8
9
10 : wrong? ( .y x adr--flag)
11   \ adr points to x y w h cfa-field
12   dup @ 2 pick -
13   over 4 + @ + 0< ?exit
14   dup 2+ @ 2 pick -
15   over 6 + @ + 0< ?exit drop false ;
16 : sel@ ( y x--cfa)
17   wwp @ selP @
18   DO 2dup I wrong? not
19     IF I &10 + @ leap THEN
20       &10 +LOOP ['] noSel ;
21
22
23
24

```

32

117

```

0 \ IT:<s s>          clv10jul88
1
2   \ 10 bytes: x y w h cfa
3
4 : <s absAt? ww, ww, ;
5 : s> absAt? -4 wwp +! \ word follows
6   ww? - 1+ swap ww? - 1+ swap
7   ww, ww, \ w h
8   r> dup 2+ >r @ ww, ; restrict \ cfa
9 : sel! ( h w y x cfa-- )
10  >r ww, ww, ww, ww, r> ww, ;
11
12 \\ Used like:
13
14 5 5 6 6 Menue: name
15 ... <s ..outputWords... s> word
16 ... ;
17
18 declares the field of ..outWords..
19 to be selectable and to execute word
20 if selected. Word is only executed
21 when selected.
22
23 SEL! is another way to do it and not
24 tested yet.

```

33

118

```

0 \ IT:select-primitives      clv10jul88
1
2 | : revers ( char--char' ) $80 xor ;
3 | : highlight
4 | selected @ 0= ?exit
5 | wwp push selected @ wwp !
6 | ww? ww? ww? ww? ( h w y x )
7 | rot swap bounds 2swap bounds
8 | ?DO 2dup ?DO I J scr@ revers I J scr!
9 | LOOP LOOP 2drop ;
10 | : normalight highlight ;
11 | : firstSel ( --adr/0 )
12 | selP @ dup wwHere u< ?exit drop 0 ;
13 | : nextSel ( --adr/0 )
14 | selected @ dup 0= ?exit
15 | &10 + dup wwHere u< ?exit
16 | drop firstsel ;
17 | : action selected @ ?dup
18 | IF 8 + room@ execute THEN ;
19
20
21
22
23
24

```

34

119

```

0 \ IT:dokey keySel          clv10jul88
1
2 $1b Constant #esc      $88 Constant #help
3
4 &18 &04 &6 &17 Menue: info
5   " ESC = exit"
6   cr ." CR = doThis"
7   cr ." HELP = this"
8   cr ." other = select" ;
9
10 : doKey ( key--andflag ) false swap
11   #esc case? IF not exit THEN
12   #cr case? IF action exit THEN
13   #help case? IF info exit THEN
14   drop nextSel selected ! ;
15
16 : keySel firstSel selected !
17   BEGIN highlight key normalight
18   dokey UNTIL wwclose ;
19
20
21
22
23
24

```

35

120

```

0 \ zinseszins              clv18aug88
1
2 .( noch nix laueft ) quit
3
4 \needs menue: 10 load
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```


36

121

```

0 \          clv18aug88
1
2 Create num 6 allot
3
4 &10 &12 &3 &30 Menue: getNumber ( --n)
5   ." Enter a Number"
6   num 6 bl fill
7   num 6 expect
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

37

122

```

0 \          clv18aug88
1
2 &8 &11 &12 &30 Menue: zinsen
3
4   <s ." Zinsberechnung " s> rechne
5 cr
6 cr d @ 6 u.r
7   <s ." DM Darlehen " s> chD
8 cr z @ 6 u.r
9   <s ." 0/00 Jahreszins" s> chZ
10 cr r @ 6 u.r
11   <s ." DM pro Rate " s> chR
12 cr r/z @ 6 u.r
13   <s ." Raten/a " s> chR/a
14 ;
15
16
17
18
19
20
21
22
23
24

```

38

123

```

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```


42

127

```

0 \ MC: clv07jul88
1
2 Class: Item: Super: object
3 method: .It ?:" has no name";
4 method: ?It ?:" no help available";
5 method: xIt ?:" no action";
6 method: pit highlight;
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

43

128

```

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

44

129

```

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```


72

157

```

0 \          clv10aug88
1
2 | : .info ." blk" blk @ dup u. block
3 | " line" >in @ c/l / dup ;
4 | ( blkadr lin ) c/l * + c/l cr type ;
5
6 | : name  bl parse cr type ;
7
8
9
10 \ : pushinput r> blk push >in push >r ;
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

73

158

```

0 \          clv10aug88
1
2 | Variable state state off
3 | 2Variable lastDef
4
5 | : lastDef! >in @ here count - blk @
6 |   lastDef 2! ;
7
8 | Defer ' : : \ forward
9
10 | : defPass \ ungets last name and def
11 |   lastDef 2@ block >in !
12 |   ':: D state on ] ;
13
14
15
16
17
18
19
20
21
22
23
24

```

74

159

```

0 \          clv10aug88
1
2 | : 2^ ( n -- 2^n )
3 |   1 swap 0 ?DO 2* LOOP ;
4
5 | : Ask: F Create D ( n--n+1 ) dup , 1+
6 |   does> ( adr -- adr flag ) @ 2^
7 |   ( adr mask ) over @ and ;
8
9
10
11
12 0 | Ask: creates?      | Ask: setsState?
13 | Ask: parses?      | Ask: parses?
14 | Ask: eatsWord?    | Ask: resetsState?
15 | Ask: eatsChar?
16 | Ask: compiles?
17 drop
18
19
20
21
22
23
24

```

75

160

```

0 \          clv10aug88
1
2 | :: F Create D 0 ,
3 | does> >r clearstack r>
4 |   \ Zahlen werden noch verarbeitet !
5 |   \ F state @ D state @ 0= and
6 |   \ IF defPass exit THEN
7 |   \ eatsWord? IF bl parse THEN
8 |   \ eatsChar? IF 1 >in +! THEN
9 |   \ parses(? IF Ascii ) parse THEN
10 |  \ parses^? IF Ascii ^ parse THEN
11 |  \ creates? IF name .info THEN
12 |  \ setsState? IF ] THEN
13 |  \ resetsState? IF [compile] [ THEN
14 |  \ drop ;
15
16 | :: Is '::
17
18
19
20
21
22
23
24

```

76

161

```

0 \          clv10aug88
1
2 | :: x
3
4 | : ??? ( cfa -- pfa )
5 |   dup @ ['] x @'- abort" not ::"
6 |   >body ;
7
8
9 | : Set: F Create ( n -- n+1 ) dup , 1+
10 | does> @ 2^ ( bitmask )
11 | last @ name> ???
12 | ( mask cfa )
13 | dup @ under or swap ! ;
14
15 | Set: creates      | Set: setsState
16 | Set: parses(     | Set: parses"
17 | Set: eatsWord    | Set: resetsState
18 | Set: eatsChar
19 | Set: compiles
20 | drop
21
22 | : vererbt ; \ nur zur Information
23
24

```

77

162

```

0 \          clv10aug88
1
2 | :: : creates setsState vererbt compiles
3 | :: Variable creates vererbt
4 | :: Constant creates vererbt
5 | :: Create creates vererbt
6 | :: " parses" immediate
7 | :: " parses" immediate
8 | :: " parses" vererbt
9 | :: abort" parses" immediate
10 | :: ( parses( immediate
11 | :: ( parses( immediate
12 | :: Ascii eatsChar immediate
13
14 | \ [compile] ['] [ ]
15
16
17
18
19
20
21
22
23
24

```

78

163

```

0 \
1   clv10aug88
2
3
4 F ' load Alias load
5 F ' +load Alias +load
6 F ' thru Alias thru
7 F ' +thru Alias +thru
8
9 \
10 : [ D state off F [compile] [ ;
11   immediate
12 : ; [compile] [ ; immediate
13
14
15
16
17
18
19
20
21
22
23
24

```

79

164

```

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

80

165

```

0 \ test
1   clv10aug88
2 Variable xxx
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```