

Sie auf diese verzichten, läßt man alle mit Sternchen markierten Zeilen weg und schreiben in Zeile 9 »\$C008« statt »\$C028«.

Das Programm schaltet nach dem Start das Betriebssystem ab, aktiviert eine eigene Tastaturabfrage und wartet, bis die Taste <A> fünfmal gedrückt wird.

```
x: move $dff01c,d0 ;Interrupts aus
ori #$8000,d0
move d0,intena
move #$7fff,$dff09a
move.l 104,vec2 ; Interruptvektoren ändern
move.l 108,vec3 ; *
move.l #inkey,104
move.l #rast,108 ; *
move #$c028,$dff09a ; Interrupts an
moveq #4,d1 ; Tastenzähler
x1: btst #0,keyfl ; auf Taste warten
beq.s x1
move.b keyco,d0 ; Tastencode holen
bclr #0,keyfl ; keyfl löschen
cmpl.b #$20,d0 ; "A"-Code ?
bne.s x1 ; nein => zurück
dbr a1,x1 ; zum fünften Mal gedrückt ?
x2: move #$7fff,$dff09a ; Interrupts aus
move.l vec2,104 ; alte Vektoren zurück
move.l vec3,108 ; *
move intena,$dff09a ;Interrupts an
rts
inkey: move.l d0,-(a7) ; Tastaturabfrage
move #8,$dff09c ; Interruptrequestbit löschen
btst #3,$bfed01
bne inkey2
bclr #0,repfl ; *Hilfsflag aus
move.b $bfec01,d0 ; Tastencode berechnen
ror.b #1,d0
eorl.b #$ff,d0
bmi.s inkey1 ; Taste losgelassen ?
btst #0,keyfl ; bereits ein Code gespeichert?
bne.s inkey1
move.b d0,keyco ; Code speichern
bset #0,keyfl ; keyfl setzen
move.b #20,repco ; *Verzögerungszähler starten
bset #0,repfl ; *Hilfsflag aktivieren
inkey1: move.b #$ff,$bfec01 ; Tastaturleitung auf 0
bset #6,$bfec01
bclr #6,$bfec01
inkey2: move.b $bfed01,d0
move.l (a7)+,d0
rte
rast: move.l d0,-(a7) ; *Rasterinterrupt
move #$0020,$dff09c ; *
btst #0,repfl ; *Hilfsflag gesetzt
beq.s rast2 ; *nein => kein Repeat
move.b repco,d0 ; *Zähler erniedrigen
subl.b #1,d0 ; *
bne.s rast1 ; *gleich 0 ?
move.b #4,d0 ; *neue Verzögerung setzen
bset #0,keyfl ; *keyfl setzen
rast1: move.b d0,repco ; *
rast2: move.l (a7)+,d0 ; *
rte ; *
vec2: dc.l 0 ; alter Interruptvektor (Priorität 2)
vec3: dc.l 0 ; *alter Interruptvektor (Priorität 3)
intena: dc.w 0 ; altes Interruptkontrollregister
keyco: dc.b 0 ; Tastencode
keyfl: dc.b 0 ; Taste gedrückt? ja/nein
repco: dc.b 0 ; *Verzögerungszähler
repfl: dc.b 0 ; *Hilfsflag für Tastenwiederholung ; © 1992 M&T
```

## Tastaturabfrage.asm

Für Programmierer, die ohne Betriebssystem arbeiten

Zur Funktionsweise: Die Tastaturabfrage erfolgt über die Interrupt-Routine »inkey«, die u.a. immer dann aufgerufen wird, sobald ein Signal von der Tastatur gesendet wird. Die Routine überprüft zunächst, ob es sich überhaupt um einen von der Tastatur ausgelösten Interrupt handelt. Falls ja, so wird »repfl« gelöscht, das Stapelregister des zuständigen CIA's gelesen und daraus der Tastencode berechnet. Wurde eine Taste losgelassen (Code > 127), ist die Routine hier zu Ende. Ansonsten wird der Code in »keyco« gespeichert,

»keyfl« und »repfl« gesetzt und schließlich die Tastaturleitung für einen Moment auf Null gesetzt, um der Tastatur mitzuteilen, daß das Signal empfangen wurde.

Die Interrupt-Routine »rast« wird 50mal in der Sekunde aufgerufen und prüft, ob »repfl« gesetzt ist. Das Flag zeigt an, daß die zuletzt gedrückte Taste noch gehalten wird. In diesem Fall erniedrigt der Amiga den Zähler »repco« um 1 und setzt »keyfl« auf Null.

Die Tastaturabfrage erfolgt vom Hauptprogramm aus schließlich dadurch, daß dieses zunächst testet, ob »keyfl« gesetzt ist. Wenn ja, wird der Tastencode aus »keyco« gelesen und »keyfl« gelöscht.

Sie können die Geschwindigkeit der Tastenwiederholung beliebig einstellen, indem Sie die Verzögerungswerte in Zeile 37 und 53 entsprechend verändern.

Frédéric Thiesse/irw

## WIE SAGE ICH ES DER PROZEDUR?

Beim Aufruf eines Unterprogramms in einer Hochsprache müssen die formalen Parameter auf festgelegte Art durch die aktuellen Parameter ersetzt werden. Die Programmiersprache Pascal bietet dazu zwei unterschiedliche Mechanismen an. Bei der Wertübergabe, im Englischen mit »call-by-value« bezeichnet, wird dem Unterprogramm nur der Wert des aktuellen Parameters übergeben. Innerhalb der Prozedur wird selbständig eine lokale Variable mit dem Namen und Typ des formalen Parameters deklariert. Dieser wird anschließend der Wert des aktuellen Parameters (der auch ein Ausdruck sein kann) zugewiesen. Da es sich um eine lokale Variable handelt, kann zwar Prozedur ihren Wert verändern, jedoch wird dieser nicht an den aufrufenden Programmteil zurückgegeben. Dadurch verhindert der »call-by-value«-Mechanismus unerwünschte Seiteneffekte. Die folgende Prozedur könnte, wenn »a« eine INTEGER-Variable ist, z.B. mit p(a+10,20) aufgerufen werden:

```
PROCEDURE p(x,y : INTEGER);
(* ohne tieferen Sinn *)
BEGIN
  x := x+y;
  WRITELN(x);
END;
```

Bei »p(a,10)« würde die Variable »a« nicht um 10 erhöht, da im Beispiel »x« eine lokale Variable innerhalb der Prozedur »p« ist. Den zweiten Mechanismus im Pascal bezeichnet man als Referenzübergabe bzw. »call-by-reference« oder »call-by-address«. In diesem Fall wird beim Prozeduraufruf die Adresse (Referenz, Verweis) des aktuellen Parameters, an welcher die Variable im Speicher steht, übergeben. Aus diesem Grund muß es sich in Pascal um eine Variable handeln, Ausdrücke, wie »a+10« sind nicht erlaubt.

Alle Operationen mit dem formalen Parameter innerhalb der Prozedur führen über den Verweis zu Operationen mit dem aktuellen Parameter. Veränderungen von Referenzparametern haben deshalb Auswirkungen auf die aktuellen Parameter. Durch diese Seiteneffekte können sowohl Werte vom aufrufenden Programmteil zum Unterprogramm als auch in umgekehrte Richtung übergeben werden. In Pascal wird der »call-by-reference«-Mechanismus durch das Schlüsselwort »VAR« vor dem formalen Parameter gekennzeichnet. Beim Aufruf der folgenden Prozedur »P(a,11)« wird der Variablen a des übergeordneten Programmteils der Wert 110 zugewiesen:

```
PROCEDURE P(VAR x:INTEGER;y:INTEGER);
BEGIN
  x := y*10;
END;
TYPE folge = ARRAY[1..10000] OF CHAR;
PROCEDURE Ausgabe(VAR text: folge);
BEGIN
  WRITELN(text);
END;
```

Der »call-by-reference«-Mechanismus kann auch verwendet werden, um dem Compiler anzuweisen, nur einen Zeiger zu übergeben anstatt eine Kopie als lokale Variable für die Prozedur anzulegen. Das trifft speziell auf längere Datentypen zu: