

M I N I T O R

Monitor-Programm für QU68000

Benutzer-Handbuch

Version: 2/83

Dieses Handbuch beschreibt alle Kommandos des Minitors auf dem Prozessor 0U68030, sowie die Fehlermeldungen, die der Minitor generiert.

Best.-Nr.: D922.026/2

Version : 2.0

Datum : Mai 1983

Copyright by Peripherie Computer System GmbH, München

Die Vervielfältigung dieser Dokumentation sowie die Verwertung ihres Inhalts ist nur mit ausdrücklicher Genehmigung von PCS gestattet.

Wir sind bestrebt, immer auf dem neuesten Stand der Technologie zu sein. Aus diesem Grund behalten wir uns Änderungen vor.

1. Einführung

Auf der Prozessorbaugruppe QU68030 ist ein PROM-residenter Monitor vorhanden, mit dem auch ohne Betriebssystem MUNIX kleine Maschinenprogramme hexadezimal in den Speicher geschrieben werden und zu Testzwecken ablaufen können. Mit den Monitor-Kommandos kann man Speicher-Inhalte ausgeben und ändern, Breakpoints setzen, Trace ein- und ausschalten, und eigene Programme bzw. das Betriebssystem MUNIX laden.

2. Kommandos

2.1 Format

Es gelten folgende Abkürzungen:

D0..D7	Datenregister D0..D7 (2 Byte)
A0..A7	Adressregister A0..A7 (4 Byte)
PC	Program Counter (4 Byte)
SR	Status-Register
SS	System-Stackpointer (4 Byte)
US	User-Stackpointer(4 Byte)
PCR	Process-Control-Register der Prozessor-Baugruppe (2 Byte)
ESR	Error-Status-Register der Prozessor-Baugruppe (2 Byte)
r	Register
b	Byte (8 bit)
w	Wort (16 bit)
lw	Langwort (32 bit)

Inhalt	Seite
1. EINFÖHRUNG.....	1
2. KOMMANDOS.....	1
2.1 Format.....	1
2.2 Lesen von Registern und Speicherzellen.....	3
2.3 Speicherzellen und Register beschreiben.....	5
2.4 Breakpoint/Tracing.....	8
2.5 Weitere Funktionen.....	10
3. FEHLERMELDUNGEN.....	12

Seite -2 -

Der Monitor meldet sich mit einem Prompt ('.'), nach Betätigen der Reset-Taste zusätzlich mit 'Minitor' am Bildschirm.

In der folgenden Beschreibung werden alle Meldungen, die am Bildschirm vom System ausgegeben werden, unterstrichen dargestellt.

Adressen- und Längenangaben werden hexadezimal angegeben. Die kleinste adressierbare Einheit ist 1 Byte.

Führende Nullen können bei der Eingabe weggelassen werden. Da A.. bzw. D.. Register ansteuern, müssen Speicherzellen, deren Adressen mit a oder d beginnen, allerdings mit führender Null angesprochen werden.

Groß- und Kleinschreibung werden nicht unterschieden.

Der Minitor testet zuerst alle wichtigen Funktionseinheiten auf dem NU68030-Prozessor (RAM, ROM, MMU, SR, PCR, ESR). Nach erfolgreichem Test erscheint am Terminal das Promt-Zeichen ('.') Erscheint das Promt-Zeichen nicht, liegt ein Hardware-Fehler auf der Baugruppe vor. Es gibt keine explizite Fehlermeldung.

2.2 Lesen von Registern und Speicherzellen

Register:

a:	alle Registerwerte ausgeben Die Register werden, wie auch beim Breakpoint/Trace-Modus, umstrukturiert ausgegeben; Register mit Inhalt 0 werden nicht angezeigt
an, n = (0..7):	Adressregister n ausgeben
dn, n = (0..7):	Datenregister n ausgeben n grösser 7: Anzeige ist ungültig, keine Fehlermeldung

Bei obigen Kommandos sind führende Blanks zugelassen.

PC, SR, SS, US werden auch mit diesen Kürzeln angesprochen.
Führende Blanks sind nicht erlaubt.
Achtung: mit a7 wird immer der SS angesteuert.

Die Register PCR und ESR der Prozessorbaugruppe können nicht symbolisch, sondern nur direkt über ihre Speicheradresse gelesen werden.

PCR : ffc043
ESR : ffc045

Beispiel:

<u>d3</u>	Register d3 lesen
<u>pc</u>	Program Counter lesen //
<u>fffc043.2</u>	2 Bytes des PCR ausgeben
<u>fffc045.2</u>	2 Bytes des ESR ausgeben

Speicherzellen:

Anfangsadresse und Länge des auszugebenden Speicherbereichs müssen angegeben werden. Die Längenangabe kann entfallen, der Default-Wert ist 20h Worte, dies entspricht einer Bildschirm-Zeile.

Minitor gibt die Speicherzellen-Inhalte wortweise, durch Blanks getrennt, aus. Ungerade Längenangaben werden zum nächsthöheren geraden Wert gerundet.

Syntax:

$n, m = (0..f)$

nn = Adresse

mm = Länge

Beispiel:

- 3f00 20h Bytes ab 3f00 ausgeben
 - 0a1ff.100 100h Bytes ab a1ff ausgeben
(führende Null !!)

... / 5

2.3 Speicherzellen und Register beschreiben:

Registernamen und Adressen werden wie in 2.2 angegeben, die weitere Eingabe erfolgt analog zur Ausgabe des Minitors:

r = 1w	Registerinhalt setzen
adr = 1w	Speicherzelle beschreiben

Die Blanks können entfallen.

Der Inhalt wird rechtsbündig eingelesen, überflüssige führende Ziffern werden ohne Warnung weggelassen!

Register:

r = 1w , r = (a0..a7, d0..d7, pc,ss,sr,us)

Beispiel:

d3 = eeefffff	D3 mit eeefffff beschreiben
ss = 1a	System-Stackpointer mit 0000001a beschreiben

Speicherzellen:

Die Länge der Informationseinheit, mit der der Speicher besetzt werden soll, kann angegeben werden (in Bytes). Der Default-Wert ist 2, dh. die Standardlänge beträgt ein Speicherwort.

Seite -6 -

Syntax:

adr.einheitslänge = Inhalt

Einheitslänge = (1,2,4):

Der Speicher wird mit Rytés, Worten bzw. Langworten beschrieben.

Angabe eines Datums:

Die Speicherzelle(n) an der angegebenen Adresse wird rechtsbündig mit dem Datum beschrieben.

Beispiel:

3f00 = 123e	ein Speicherwort beschreiben (Default)
3f00.1 = bf	nur ein Ryté beschreiben, neuer Inhalt
3f00.4 = 1	von 3f00: xxbf (rechtsbündig!!) Langwort beschreiben, neuer Inhalt von 3f00 und 3f02: 0000 0001

Angabe mehrerer, durch Blanks getrennter Daten:

.../7

Ab der angegebenen Adresse wird der Speicher mit den angegebenen Daten beschrieben. Die Verteilung auf die einzelnen Speicherzellen ergibt sich durch die Längenangabe. Dabei ist die Anzahl der einzugebenden Informationseinheiten beschränkt und betragen:

30 Bytes oder
38 Worte oder
9 Langworte

Beispiele:

3f00 = 1 2 3 ffff 4 Speicherworte (Wort=Defaultwert) ab
 3f00 beschreiben. Neuer Inhalt:
 3f00: 0001 3f02: 0002
 3f04: 0003 3f06: ffff.

3f00.1 = 1 2 3 3 Bytes ab 3f00 beschreiben. Neuer
 Inhalt:
 3f00: 0102 3f02: 03xx

Es werden 20h Speicherworte am Bildschirm ab der angegebenen Adresse ausgegeben.

Einheitslänge größer 4:

Die Angabe der Einheitlänge bedeutet jetzt die Anzahl der Bytes, die beschrieben werden sollen. Es werden stets Langworte (4 Byte) beschrieben.

Beispiel:

3f00.10 = bb 10h Speicherworte mit 0000 00bb
 aufgefüllt (rechtsbündig!!)

Der Speicherbereich wird nicht ausgegeben.

2.4 Breakpoint/Tracing:

Breakpoints:

Breakpoints dürfen nur auf gerade Speicheradressen gesetzt werden. Andernfalls erfolgt 'odd-address-error'.

Der Minitor baut eine Breakpoint-Tabelle mit max. 7 Einträgen auf. Dort sind die Inhalte der Speicherzellen gerettet, in die Breakpoints gesetzt wurden. Diese Speicherzellen müssen vorher Befehlscodes enthalten haben, keinesfalls Daten (sonst wird der Trap-Befehl als Datum verarbeitet!!).

Syntax:

b adr	Breakpoint in 'adr' setzen
b -	alle Breakpoints löschen
b -adr	Breakpoints in adr löschen

Das Blank darf nicht weggelassen werden.

Am Breakpoint werden die Adresse und alle Registerwerte ausgegeben.

Tracing:

Nach jedem Befehl werden die Registerinhalte wie an einem Breakpoint ausgegeben. Gesetzte Breakpoints werden durch ein zusätzliches R angezeigt.

Vor dem Einschalten des 'Trace'-Modus muß das zu untersuchende Programm mit einem Breakpoint in einen definierten Zustand gebracht werden.

Beim Tracing werden an Breakpoint nicht die Break-Instruktionen (Register ausgabe von) ausgeführt, sondern die in die Breakpoint-tabelle geretteten Befehle. Nach einem Reset ist diese jedoch leer, d.h. Reset mit gesetzten Breakpoints zerstört das untersuchte Programm. Beim Tracing erscheint dann die Meldung : '?? R.. not found'.

Syntax:

T n	: Tracing der nächsten n Befehle
T -	: Tracing ausschalten
TC n	: nur die nächsten n Unterprogramm - und Rücksprünge werden angezeigt.

Das Blank darf nicht weggelassen werden.

Beim Tracen wird auf jeden Fall ein Befehl ausgeführt!

2.5 Weitere Funktionen:

Achtung: Bei den folgenden Kommandos dürfen Blanks nicht weggelassen werden.

Programm im Speicher starten:

G adr adr ist der neue Stand des Benutzer-PC. adr kann auch weggelassen werden, dann wird der alte Stand angenommen.

Kopplung zum Host-Rechner:

H Schnittstelle wird auf Host-Port fffd40 gelegt, bis ctrl-A auftritt
H fff012 Schnittstelle wird auf Port fff012 gelegt

Unterprogramm starten:

C adr P ..P P ..P sind Parameter, die auch weggelassen werden können. Es sind nur Langworte erlaubt.

Speicher durchsuchen:

L inhalt adr Es wird ab der Adresse 'adr' nach einem Speicherwort 'inhalt' gesucht.

Daten transportieren:

M quelladr zieladr länge Es werden die Speicherworte ab 'quelladr' bis 'quelladr+länge' nach 'zieladr' gebracht. Der Inhalt bleibt erhalten.

Seite -11 -

Es werden 5 Vektoren initialisiert:

Adr.:	8	-	bus error
	c	-	odd address error
	10	-	illegal instruction
	24	-	break vector
	80	-	trace vector

Nach dem Laden eines stand-alone-Programms müssen diese Vektoren neu initialisiert werden. Nach einem 8, c, 10 - Trap werden alle Register, "PCR" und "ESR" angezeigt. Oberschreibt der Programmierer einen Vektor, so muß er die Trap-Behandlung selbst durchführen. Break- und Trace-Vektoren dürfen bei Verwendung des Break- und Trace-Modus nicht überschrieben werden.

Programm laden:

Minitor liest das Unix-File-Format.

Zuvor kann jedoch der Device-Typ angegeben werden. Es stehen zu Verfügung:

r0	:	RK06/07 Platte am Port ffff20. Dies ist Default-Wert.
r1	:	RL01/02 Platte am Port fff900
rx	:	Double-Density Floppy-Disc am Port fffe78
rs	:	Cartridge Tape Streamer am Port ffff80
rt	:	Magnetband am Port fff550

Ladekommando:

/programmname. Es kann auch ein ganzer Baum angegeben werden:
/dir1/dir2/.../programmname.

Normalerweise wird mit /unix das Betriebssystem geladen.
Das geladene Programm wird anschließend gestartet.

3. Fehlermeldungen

Der Minitor generiert eine Reihe von Fehlermeldungen, wenn er Kommandos nicht interpretieren kann, oder Fehler entdeckt.

<u>Fehlermeldung</u>	<u>Bedeutung</u>
What?	Ein falsches Kommando wurde eingegeben, das vom Kommandointerpreter nicht verstanden wird.
?cant find file	Die Datei ist auf dem Datenträger nicht vorhanden (siehe 2.5)
??B..not found	Die Breakpoint-Tabelle wurde während des Tracings gelöscht (siehe 2.4)
RL?n	Lesefehler auf der Winchester (rl02). Eventuell wird noch die Blocknummer n angegeben.
rx error	Lesefehler auf der Floppy Disk. Zusätzlich wird vom FD-Controller noch ausgegeben: csr Command Status Register-Inhalt esr Error Status Register-Inhalt eco Error Code
	Die Bedeutung dieser hexadezimalen Zahlen ist dem betreffenden Controller-Handbuch zu entnehmen.

hk error: Lesefehler auf der Winchester (rk06/07).
 Zusätzlich wird noch angegeben:

bn Blocknummer
cs2 error Command Status Register des
Controllers

Die Bedeutung dieser hexadezimalen Zahlen ist
dem Controller-Handbuch zu entnehmen.

tape error Lesefehler auf dem Band (MB100X). Zusätzlich
 wird noch der Inhalt des Fehlerregisters
 ausgegeben.

?streamer error Lesefehler auf dem Cartridge Tape (z.B. Band
 läuft auf Bandmarke beim Lesen).