

Bedienungsanleitung für Hardware-Debugger

G. Schmitt
Robert-Bosch-Berufskolleg

1.12.1999

Auswahl Menüpunkt <Terminal>



COM-Anschluss <1> oder <2> wählen



Das Monitorprogramm meldet sich.

Syntax der Kommandos

Erklärung

Beispiel

F1 Hardware-Debugger verlassen

F2 Meldung: Input file:
Das ablauffähige Programm wird in den Speicher des Mikrocontrollers übertragen

< LAUF.HEX> eingeben

G (Startadresse (, Stopadresse))

Go: Starten des Programms. Ist kein Breakpoint definiert, kann die Programmausführung nur durch einen Hardware-Reset unterbrochen werden.

#G8100 ; ab der Adresse 8100H
; wird der Programmcode
; ausgeführt

X PC Meldung **PC= 8000**

8100 ; Programm Counter
; springt auf gewählte
; Adresse

X **Examine:** Registerinhalte anzeigen

#X ; Registerinhalte werden
; ab oben gewählter Adresse
; angezeigt.

T (Anzahl der Schritte) **TraceStep:** Anzahl der angegebenen Programmschritte wird ausgeführt.

#T10 ; 10 Programmschritte
; werden ab oben gewählter
; Adresse ausgeführt.

P (Anzahl der Schritte) **TraceStep:** Anzahl der angegebenen Programmschritte wird ausgeführt (Unterprogramme werden als ein Programmschritt aufgefasst).

#P10

DC (Startadresse(,Endadresse)) **DisplayCode:** Der angegebene Bereich des Programmspeichers wird angezeigt.

#DC 8100,8200

DX (Startadresse(,Endadresse)) **DisplayExternalData:** Der angegebene Bereich des externen Datenspeichers wird angezeigt.

#DX C000,C100

DI (Startadresse(,Endadresse)) **DisplayInternalData:** Der angegebene Bereich des internen indirekt adressierbaren Datenspeichers wird angezeigt.

#DI 0,1F

DD (Startadresse(,Endadresse)) **DisplayInternalData:** Der angegebene Bereich des internen direkt adressierbaren Datenspeichers wird angezeigt.

#DD 20,FF

X **Registername** **Examine:** Registerinhalt kann mittels interaktiver Eingabe verändert werden.

#X RO

| | | | |
|--|--|----------------------------|--|
| EC Startadresse | EnterCode: Programmspeicher kann mittels interaktiver Eingabe beschrieben werden. | #EC 8100 | |
| EX Startadresse | EnterExternalData: Der externe Datenspeicher kann mittels interaktiver Eingabe beschrieben werden. | #EX C000 | |
| EI | EnterInternalData: Der interne indirekt adressierbare Datenspeicher kann mittels interaktiver Eingabe beschrieben werden. | #EI 0 | |
| ED | EnterInternalData: Der interne direkt adressierbare Datenspeicher kann mittels interaktiver Eingabe beschrieben werden. | #ED 20 | |
| FILLC Startadresse, Endadresse, Konstante | FillCode: Der angegebene Bereich des Programmspeichers wird mit der angegebenen Konstanten beschrieben. | #FILLC 8100,8200,00 | ;der Programmspeicher ;wird im Bereich von 8100H ;bis 8200H mit der Konstan- ;ten 00 beschrieben |
| FILLX Startadresse, Endadresse, Konstante | FillExternalData: Der angegebene Bereich des externen Datenspeichers wird mit der angegebenen Konstanten beschrieben. | #FILLX C000,C100,FF | |
| A Startadresse | Assemble: Der Programmspeicher kann mittels interaktiver Eingabe mit „Assembler-Code“ beschrieben werden. | #A 8100 | |
| U Startadresse (,Endadresse) | Unassemble: Der angegebene Bereich des Programmspeichers wird disassembliert. | #U 8100 | |
| BS Adresse | BreakpointSet: Definieren eines Breakpoints | #BS 8200 | ;beim Starten des Programms ;mit Kommando Go wird die ;Programmausführung beim ;Erreichen der Adresse 8200H ;unterbrochen. |
| BL | BreakpointList: Anzeigen der definierten Breakpoints. | #BL | |
| BK Nummer | BreakpointKill: Löschen eines definierten Breakpoints | #BK 0 | ;der zuerst definierte Break- ;point wird gelöscht. |
| BK all | BreakpointKill: Alle definierten Breakpoints werden gelöscht. | #BK all | |