

Buglist

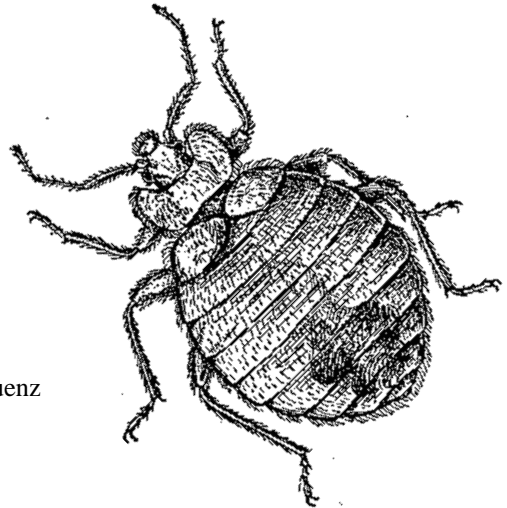
Für Version V1.02 Stand 21.1.06

Änderungen Firmware V1.03 gegenüber V1.02:

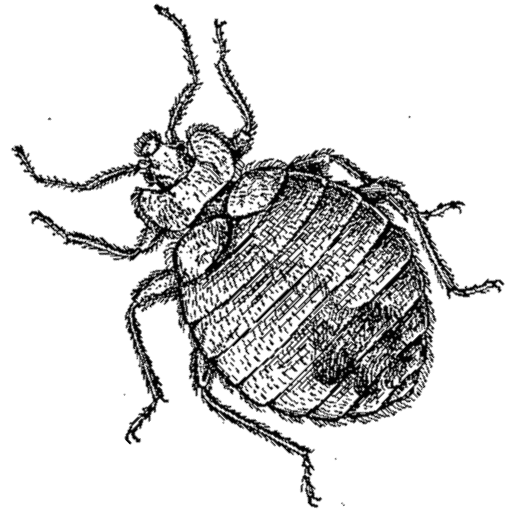
- * Fehler in den Timingroutinen für das Schreiben des FLASHs behoben.
Das Schreiben des FLASH ist nur noch bei 2,45MHz Busfrequenz möglich, damit das Timing sicher eingehalten wird.

Änderungen Dokumentation:

- * GP32 Seite 1.6 Hinweis daß Schreiben des FLASH nur bei 2,45MHz Busfrequenz möglich ist.



Buglist



Für Version V1.01 Stand 29.4.05

Änderungen Firmware gegenüber V1.00:

- * Autostart in COLD geändert
- * Fehler in Vektoren behoben
- * Fehler in Fehlermeldung behoben (?)
- * SPACE in ND. eingebaut
- * -1 eingebaut

Änderungen Dokumentation:

- * GP32 Seite 1.6 : Flußdiagramm, Tabelle 4
- * FORTH Seite 2.11 : Fehlermeldungen
- * IO Seite 3 Mitte : Vektoren
- IO Seite 5 : Neue Beschreibung Autostart

Verbleibene Bugs & Eigentümlichkeiten

Assembler im Interpretermodus

Bei Sprungzieltabellen in Assembler muß der Assembler zugeschalten werden:

```
TABLE MY-JUMPS
[CODE
\ HERE JMP,
\ THERE JMP,
CODE] [
```

Mit CODE] wird auf Compilermodus umgeschaltet, weshalb [nötig ist das auf Interpretermodus wechselt. Man kann natürlich den Opcode auch binär ablegen:

```
TABLE MY-JUMPS
\ HERE , CC C,
\ THERE , CC C,
```

Fehlende Leerzeichen am Anfang von , " und . "

Der Parser eliminiert Leerzeichen die vor druckenden Zeichen kommen:

```
: x1
." my text" ;
\ ^^ druckt diese SPACE nicht
```

```
TABLE x2
," my text"
\ ^^ druckt diese SPACE nicht
```

Einfachste Lösung ist „unauffällige“ Platzhalter einzufügen:

```
: x1  
." ..my text" ;
```

Diese Lösung funktioniert auch hat aber hohen Speicherbedarf:

```
: x1  
SPACE SPACE ." my text" ;
```

Umschalten des Bustaktes

Das Umschalten auf z.B. 8MHz durch das kurze Hilfsprogramm stört durch das langsame Hochlaufen der PLL kurzzeitig die UART. Es können dann fehlerhafte Zeichen empfangen und gesendet werden. Man sollte das Umschalten deshalb manuell vornehmen und nicht mitten in Listings. Oder eventuell mit MSEC eine Verzögerung von ca. 1sec unmittelbar danach ausführen

Illegales Lesen von RAM

Wenn man z.B. mit DUMP über die Obergrenze des RAMs kommt druckt man Speicher aus der nicht existiert. Laut Motorola kann ein Zugriff auf solche Bereiche einen „illegal address reset“ auslösen. Anscheinend wird zwar kein echter Reset ausgelöst aber nominell sind Funktionsstörungen danach nicht ausgeschlossen. Deshalb nun in DUMP Abbruch bei Überlauf auf Adresse 0240.

Relativer Sprungbefehl

Der Sprung anhand absoluter Adresse bei relativem Sprungbefehl compiliert in RAM korrekt aber bricht in FLASH mit Fehlermeldung ab.

```
:CODE XYZ1  
    RTS,  
CODE ;
```

```
:CODE XYZ2  
' XYZ BCC,  
    RTS,  
CODE ;
```

Patch mit JMP , :

```
:CODE XYZ2  
1 $ BCS,  
' XYZ JMP,  
1 $: RTS,  
CODE ;
```