

# Forth-Quellcode-Bibliotheken

Wie kommt der Code in den Mikrokontroller ?



Manfred Mahlow  
[manfred.mahlow@forth-ev.de](mailto:manfred.mahlow@forth-ev.de)

Jahrestagung 2009 der Forth Gesellschaft e.V.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Fragestellung

Ist es möglich, für die Programmierung eines Mikrokontrollers die Quellcode-Bibliotheken eines Terminal-Servers zu nutzen, ohne dass im Forth-System des Mikrokontrollers eine entsprechende Schnittstelle implementiert ist?

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Ausgangssituation

(1)

Ein Mikrokontroller mit einem Forth-System

- mit serieller Kommunikationsschnittstelle
- ohne Massenspeicheranbindung
- ohne Bibliotheksschnittstelle

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Ausgangssituation

(2)

Ein Linux-PC als Terminal-Server

- mit Tastatur, Monitor und serieller Schnittstelle
- mit einem Forth-System mit
  - Massenspeicheranbindung
  - Quellcode-Bibliothek

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Quellcode-Bibliothek des Terminal-Servers

Der Quellcode wird in der Form von Quellcode-Modulen, die als sequentielle Textdateien realisiert sind, in einer hierarchischen Verzeichnisstruktur gespeichert.

## Unterstützt wird

- unbedingtes Laden von Modulen
- bedingtes Laden von Modulen.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Modul- und Dateinamen

Modulname : [ccc<space>]...[ccc<space>]ccc

Dateiname : /BSV/[ccc/]...[ccc/]ccc.<ext>

Beispiel : String ==> /BSV/String.4th

String Array ==> /BSV/String/Array.4th

BSV = Bibliotheksstammverzeichnis

ext = Dateinamenerweiterung

Jahrestagung 2009 der Forth Gesellschaft e.V.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Bibliotheks-API des Terminal-Servers

Unbedingtes Laden mit

```
take <Modulname>
```

Lädt das Modul <Modulname> aus der Quellcode-Bibliothek.

Bedingtes Laden mit

```
needs <Modulname>
```

Lädt das Modul <Modulname> nur dann, wenn es noch nicht geladen wurde, d.h. Quellcode wird nicht mehrfach geladen.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Kommunikation mit dem Mikrokontroller

- Der Mikrokontroller legt empfangene Zeichen im Terminal Input Buffer ab und sendet ein Echo an den Terminal-Server zurück.
- Der Inhalt des Terminal Input Buffers (TIB) wird interpretiert, sobald ein CR-Zeichen (0x13) eintrifft oder der TIB voll ist.
- Nach der Interpretation oder im Fehlerfall sendet der Mikrokontroller eine Statusmeldung an den Terminal-Server zurück :

<space>**ok**<CR><LF>

im Interpretermodus

<space>**]**<CR><LF>

im Compilermodus

<space>**? MSG# <Fehlernummer>**<CR><LF> im Fehlerfall

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Lösungsansatz

Ein Terminal-Server der drei Betriebsarten unterstützt:

- Terminal Mode
- Download Mode
- Control Mode

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

## Terminal-Server **Terminal Mode**

- Eingegebene Zeichen werden direkt an den Mikrokontroller gesendet.
- Kein lokales Echo.
- Die Eingabe am Terminal erfolgt zeilenorientiert.
- Beim Drücken der Eingabetaste wird ein CR-Zeichen gesendet.
- % als erstes Zeichen einer Zeile aktiviert den Control Mode.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

## Terminal-Server **Control Mode**

- Eingegebene Zeichen werden nicht zum Mikrokontroller gesendet sondern in einen lokalen Textpuffer abgelegt.
- Lokales Echo
- Beim Drücken der Eingabetaste wird der Inhalt des Textpuffers vom Terminal-Server als **Terminaldirektive** interpretiert.
- Nach dem Ausführen einer Terminaldirektive kehrt das Terminal in den Terminal Mode zurück.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

## Terminal-Server **Download Mode** (1)

- Der Download Mode wird durch **Terminaldirektiven** aktiviert, die eine Quellcode-Datei / ein Quellcode-Modul öffnen.
- Der Quellcode wird zeilenweise aus der zuletzt geöffneten Datei gelesen. Er kann weitere Terminaldirektiven enthalten.
- Eine Zeile, die eine Terminaldirektive enthält, wird vom Terminalserver interpretiert.
- Eine Zeile die keine Terminaldirektive enthält, wird ohne lokales Echo an den Mikrokontroller gesendet.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

## Terminal-Server **Download Mode** (2)

- Am Zeilenende wird ein CR-Zeichen gesendet.
- Nach dem Senden des CR-Zeichens wartet der Terminal-Server auf eine Statusmeldung des Mikrokontrollers.
- Im Fall einer Fehlermeldung werden alle geöffneten Quellcode-Dateien geschlossen, der Download Mode beendet und zum Terminal Mode zurückgekehrt.
- Wird beim Lesen das Ende einer Datei erreicht, wird die Datei geschlossen. Wenn alle Dateien geschlossen sind, wird zum Terminal Mode zurückgekehrt.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Bibliotheks-API für den Mikrokontroller (1)

- Der Mikrokontroller hat keine reale Bibliotheksschnittstelle.
- Das Runterladen von Quellcode-Modulen muss vom Terminal-Server angestoßen werden.
- Wir führen deshalb mit Hilfe von Terminaldirektiven eine virtuelle Bibliotheksschnittstelle für den Mikrokontroller ein.
- In Analogie zur Bibliotheksschnittstelle des Terminal-Servers nennen wir die Terminaldirektiven **needs** und **take** .

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Bibliotheks-API für den Mikrokontroller (2)

Unterstützt werden das unbedingte und das bedingte Laden von Quellcode-Modulen.

Unbedingtes Laden mit

```
take <Modulname>
```

Lädt das Modul <Modulname> in den Mikrokontroller herunter. Der Code wird zeilenweise aus der Quellcode-Datei gelesen und zum Mikrokontroller übertragen. Antwortet der Mikrokontroller mit einer Fehlermeldung, wird die Übertragung abgebrochen und alle geöffneten Quellcode-Dateien werden geschlossen.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Bibliotheks-API für den Mikrokontroller (3)

Bedingtes Laden mit

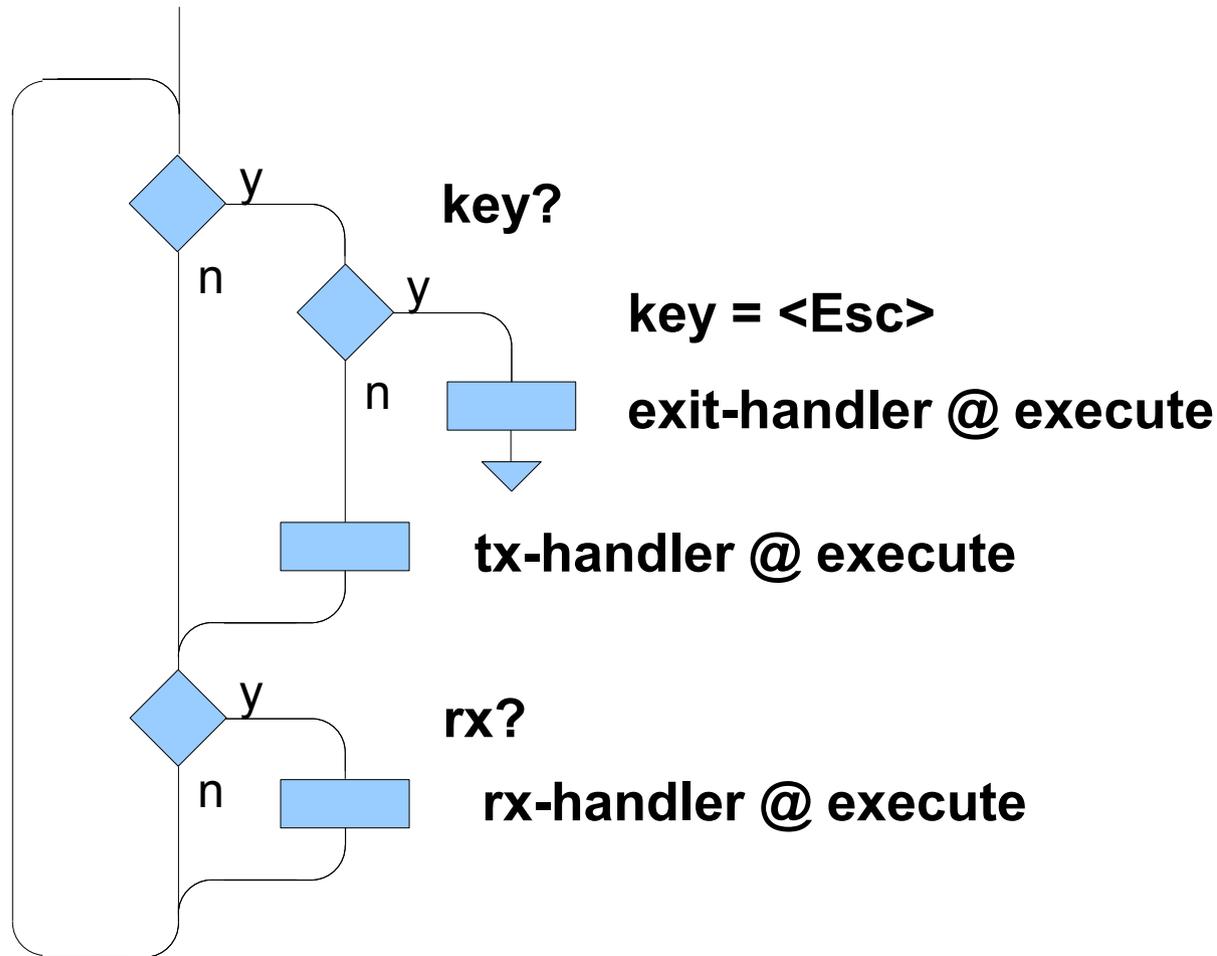
```
needs <Modulname>
```

Lädt das Quellcode-Modul <Modulname> nur dann in den Mikrokontroller herunter, wenn es noch nicht geladen wurde.

- Stellt sicher, dass Code nicht mehrfach geladen wird.
- Geladene Module werden auf dem Terminal-Server registriert.

Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

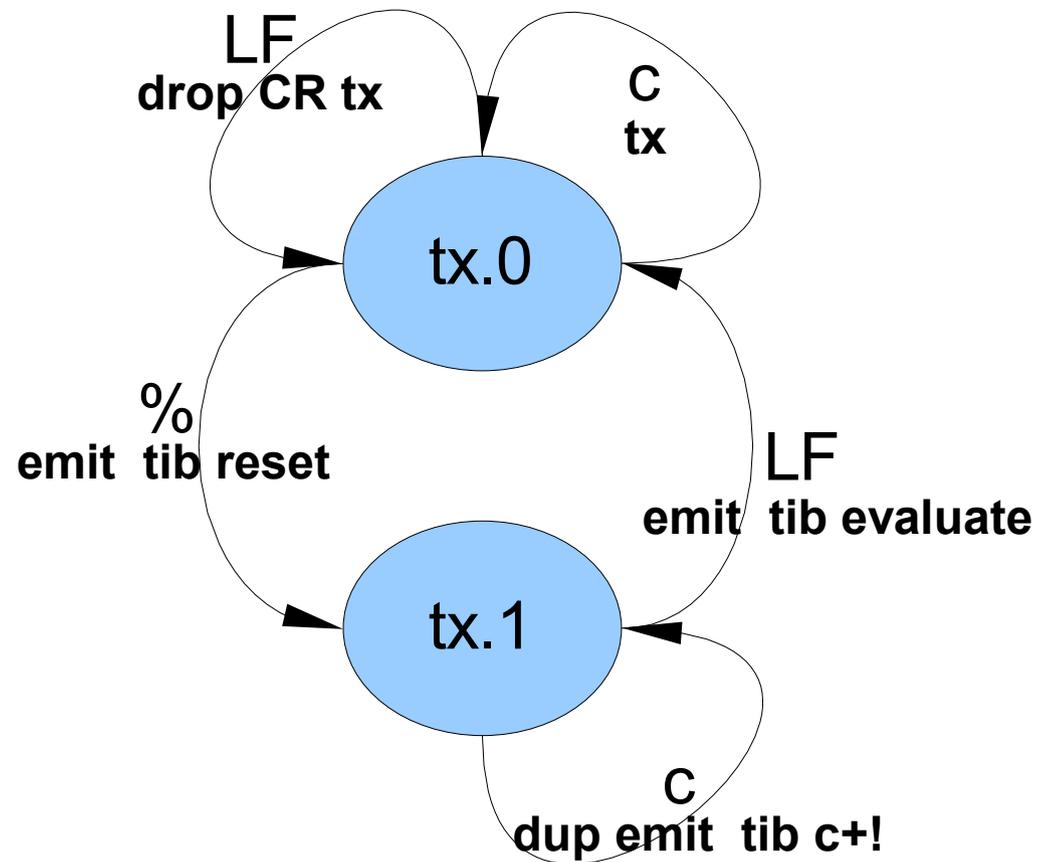
# Implementierung Terminal Flussdiagramm



Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Implementierung

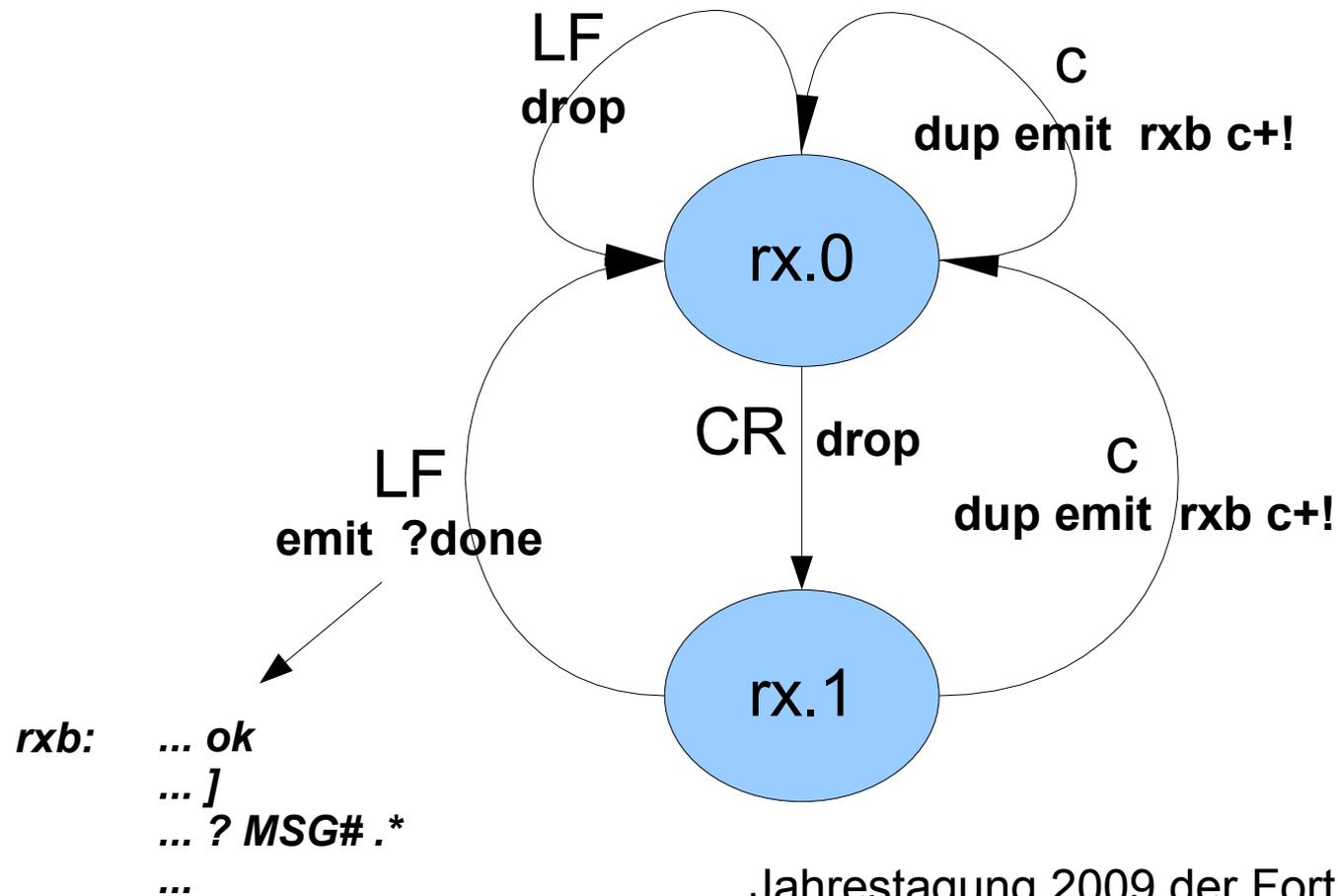
tx-handler



Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Implementierung

rx-handler

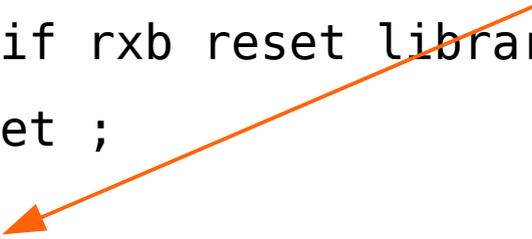


Quellcode-Bibliotheken - Wie kommt der Code in den Mikrokontroller ?

# Implementierung

?done

```
: ?done ( -- )  
  ]? ok? or if rxb reset download exit then  
  error? if rxb reset library close exit then  
  rxb reset ;
```



```
: download ( -- )  
  begin  
    library file read-line if exit then  
  directive while repeat  
  write-line ;
```

Jahrestagung 2009 der Forth Gesellschaft e.V.

```
mahlow@ubuntu: /home/archiv/Programme/csp4th-ans
Datei Bearbeiten Ansicht Terminal Reiter Hilfe
Terminal: device=/dev/ttyS0 fd=4 B9600 8N1 CLOCAL

Press [Esc] to return to the Forth Interpreter.

% NEEDS .S
80C592/.S.4th opened
80C592/DEPTH.4th opened
FORTH DEFINITIONS DECIMAL ok
: DEPTH ( -- n )
  SP@ S0 @ SWAP - 2 / ; ok
80C592/DEPTH.4th closed
FORTH DEFINITIONS DECIMAL ok
: .S ( -- ) ]
  ." [ " DEPTH -DUP ]
  IF ]
    1- -DUP ]
    IF S0 @ 2- SWAP 0 DO 2- DUP @ . LOOP ]
    DROP ENDIF DUP . ]
  ELSE ]
    ." empty " ]
  ENDIF ]
  ." ]TOS " ; ok
80C592/.S.4th closed ok

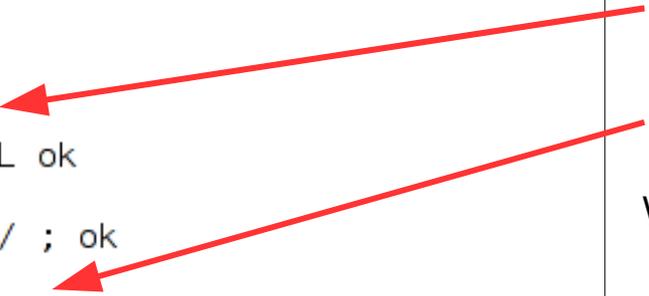
% NEEDS .S
ok
```

```
\ 8051: .S
\ -----
NEEDS DEPTH

FORTH DEFINITIONS DEC

\ Prints the parameter st
: .S ( -- )
  ." [ " DEPTH -DUP
  IF
    1- -DUP
    IF S0 @ 2- SW
    DROP ENDIF
  ELSE
    ." empty "
  ENDIF
  ."
  ."

.../80C592/.S.4th
```



# Einschränkungen des Konzepts

1. Wird das Terminal-Server-Programm beendet, gehen die Informationen über geladene Module verloren.

Abhilfe: Vor dem Beenden des Terminal-Server-Programms den aktuellen Zustand des speichern.

2. Wird ein Warm- oder Kaltstart des Mikrokontrollers durchgeführt, ohne dass die geladenen Module dauerhaft gespeichert wurden, wird die auf dem Terminal-Server geführte Liste der geladenen Module ungültig.

Abhilfe: Terminal-Server-Programm neu starten.