

Neues vom STRIP Forth-Prozessor

Tagung der Forth-Gesellschaft

März 2014

Bad Vöslau/Österreich

Willi Stricker

STRIP Forth-Prozessor

Inhalt

1. STRIP-Befehlssatz

Bisher: minimaler Befehlssatz

neu: erweiterter Befehlssatz

2. Aufbau eines STRIP-Forth Systems

STRIP Forth-Prozessor Software-Ausstattung Befehlssatz

Minimal-Befehlssatz

Er enthält alle diejenigen Befehle, die für den Aufbau eines Forth-Systems zwingend erforderlich sind.
Außerdem enthält er zusätzliche Befehle, die für die Hardware-Steuerung benötigt werden.

STRIP Forth-Prozessor Befehlssatz

Erweiterter STRIP-Befehlssatz

Zweck der Erweiterung: Höhere Ablaufgeschwindigkeit.

Er enthält weitere in Hardware ausgeführte Funktionen:
Die Auswahl nach folgenden Kriterien:

1. Hardware-Aufwand
2. Benutzungs-Häufigkeit

STRIP Forth-Prozessor

Befehlssatz

Basis: Forth Standard 200X

Er besteht aus:

- Core Word Set
- Zusatzfunktionen

STRIP Forth-Prozessor

Befehlssatz

Core Word Set

er besteht aus 3 Gruppen

Nucleus

Interpreter

Compiler

Der STRIP Befehlssatz enthält nur Words aus dem
Nucleus

STRIP Forth-Prozessor

Befehlssatz

STRIP Nucleus Words

Minimal-Befehlssatz (schwarz)

nur Hardware

Forth Standard Words

zusätzliche Words (*kursiv*)

Erweiterter Befehlssatz

entweder Hardware

als Primitives (**rot/rot**)

oder Software

als High level functions (**blau/blau**)

STRIP Forth

Forth Nucleus-Befehlssatz

System-Befehle

Hilfsfunktionen für den Compiler (runtime code of compiling words)

<i>;S</i>	(--)	<i>return ";"</i>
<i>LIT</i>	(-- data)	<i>immediate Wert „LITERAL“</i>
<i>BRANCH</i>	(--)	<i>unbedingte Verzweigung</i>
<i>?BRANCH</i>	(data --)	<i>bedingte Verzweigung für Structures IF, ELSE THEN BEGIN UNTIL WHILE REPEAT</i>

STRIP Forth

Forth Nucleus-Befehlssatz

Processor Steuerungs-Befehle (Hardware-Steuerung)

<i>DISINT</i>	(--)	Disable Interrupts
<i>ENINT</i>	(--)	Enable Interrupts

STRIP Forth

Forth Nucleus-Befehlssatz

Indirekter Befehls- und Unterprogramm-Aufruf

EXECUTE (address --)

STRIP Forth

Forth Nucleus-Befehlssatz

Return stack manipulation

<i>RP@</i>	(-- RP)
<i>RP!</i>	(RP --)
<i>>R</i>	(data --)
<i>R></i>	(-- data)
<i>R@</i>	(-- data)

STRIP Forth

Forth Nucleus-Befehlssatz

Parameter stack manipulation

SP@	(-- SP)
SP!	(SP --)
DEPTH	(-- SP) <i>identisch mit SP@</i>
DROP	(data --)
PICK	(position -- data)
-PICK	(data position --)
DUP	(data -- data data)
?DUP	(data -- data <i>data if data <> 0</i>)
OVER	(data0 data1 -- data0 data1 data0)
ROT	(data0 data1 data2 -- data1 data2 data0)
SWAP	(data0 data1 -- data1 data0)
ROLL	(data - nth data)
NIP	(data0 data1 -- data1)
TUCK	(data0 data1 -- data1 data0 data1)
NDROP	(data --)

STRIP Forth

Forth Nucleus-Befehlssatz

Memory access

@	(address -- data)	
!	(data address --)	
W@	(address -- 0 word)	{32 bits only}
W!	(word address --)	{32 bits only}
C@	(address -- 0 byte)	
C!	(byte address --)	
+!	(data address --)	

STRIP Forth

Forth Nucleus-Befehlssatz

Logic functions

INVERT (data -- result)
AND (data1 data2 -- result)
OR (data1 data2 -- result)
XOR (data1 data2 -- result)
NOT (flag -- flag) *identisch mit 0=*

STRIP Forth

Forth Nucleus-Befehlssatz

Compare functions

```
0=      ( data1 data2 -- flag )
0<>     ( data1 data2 -- flag )
0<      ( data1 data2 -- flag )
0>      ( data1 data2 -- flag )
=       ( data1 data2 -- flag )
<>      ( data1 data2 -- flag )
<       ( data1 data2 -- flag )
>       ( data1 data2 -- flag )
>=     ( data1 data2 -- flag )
<=     ( data1 data2 -- flag )
U>     ( data1 data2 -- flag )
U<     ( data1 data2 -- flag )
U>=   ( data1 data2 -- flag )
U<=   ( data1 data2 -- flag )
FALSE ( -- false-flag )
TRUE  ( -- true-flag )
```

STRIP Forth

Forth Nucleus-Befehlssatz

Select fuctions

MIN (data0 data1-- data-min)
MAX (data0 data1-- data-max)
ABS (data -- abs-data)

STRIP Forth

Forth Nucleus-Befehlssatz

Shift functions

U2/C	(data -> carry result)
U2*	(data - data*2)
U2/	(data - data/2)
LSHIFT	(data0 data1 -- data0*(2**data1))
RSHIFT	(data0 data1 -- data0/(2**data1))

STRIP Forth

Forth Nucleus-Befehlssatz

Arithmetic functions: plus, minus

+C	(data0 data1 -- sum carry)
+	(data0 data1 -- sum)
NEGATE	(data -- -data)
-	(data1 data2 -- difference)
1+	(data -- data+1)
1-	(data -- data-1)

STRIP Forth

Forth Nucleus-Befehlssatz

Arithmetic functions: multiply

UM* (data0 data1 - prod-l prod-h)
M* (data0 data1 - prod-l prod-h)
***** (data0 data1 - prod)

STRIP Forth

Forth Nucleus-Befehlssatz

Arithmetic functions: Divide

UM/MOD	(data-l data-h div - quot mod)
FM/MOD	(data-l data-h div - quot mod)
SM/REM	(data-l data-h div - quot mod)
/MOD	(data div - quot mod)
/	(data div - quot)
MOD	(data div - mod)

STRIP Forth

Forth Nucleus-Befehlssatz

Arithmetic functions: multiply, divide

```
*/      ( data0 data1 div - data0*data1/div )  
*/MOD  ( data0 data1 div - data0*data1/div mod )
```

STRIP Forth

Forth Nucleus-Befehlssatz

Constants

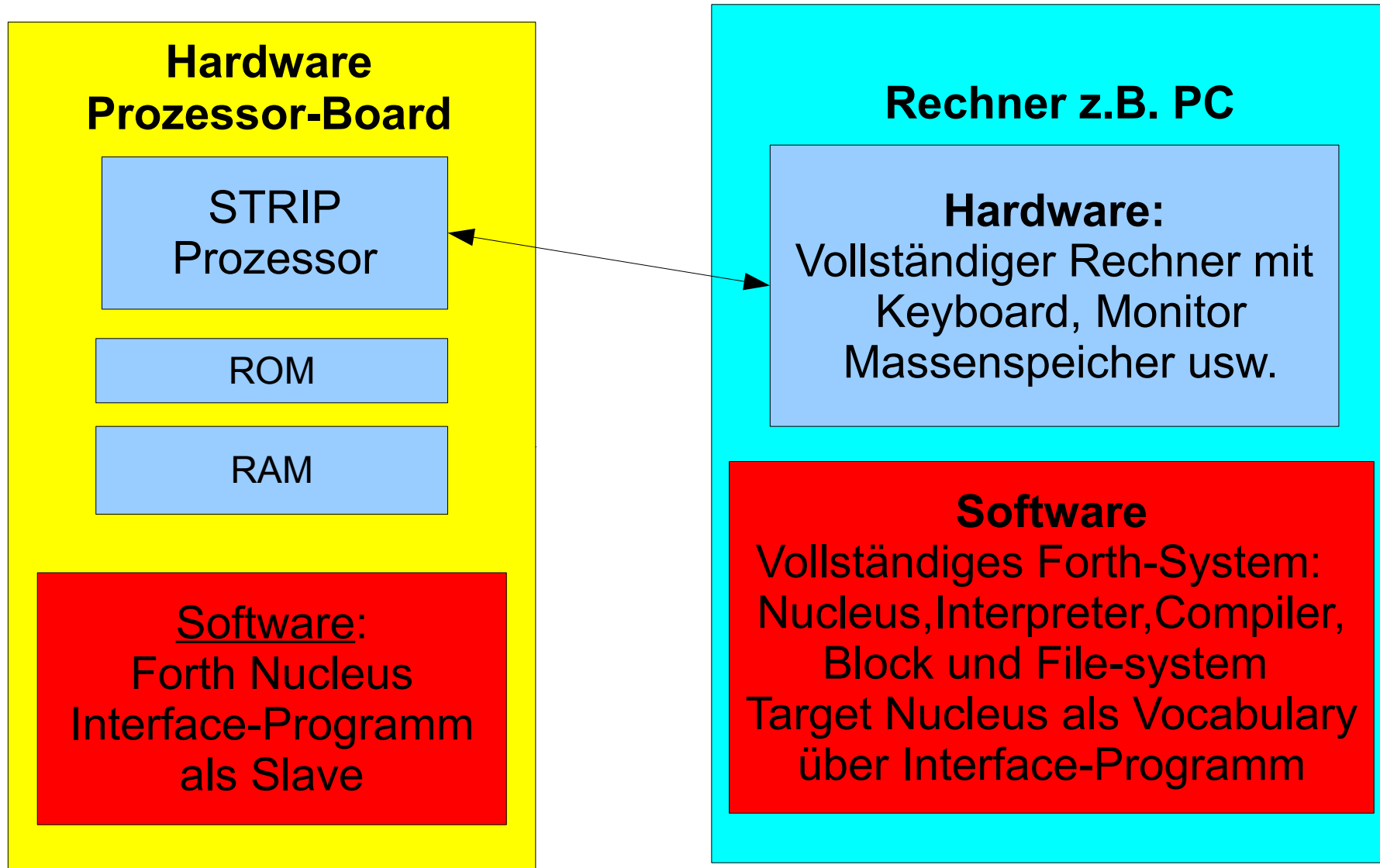
0	(-- 0)	<i>identisch mit FALSE</i>
1	(-- 1)	
-1	(-- -1)	<i>identisch mit TRUE</i>

Gegenwart und Zukunft

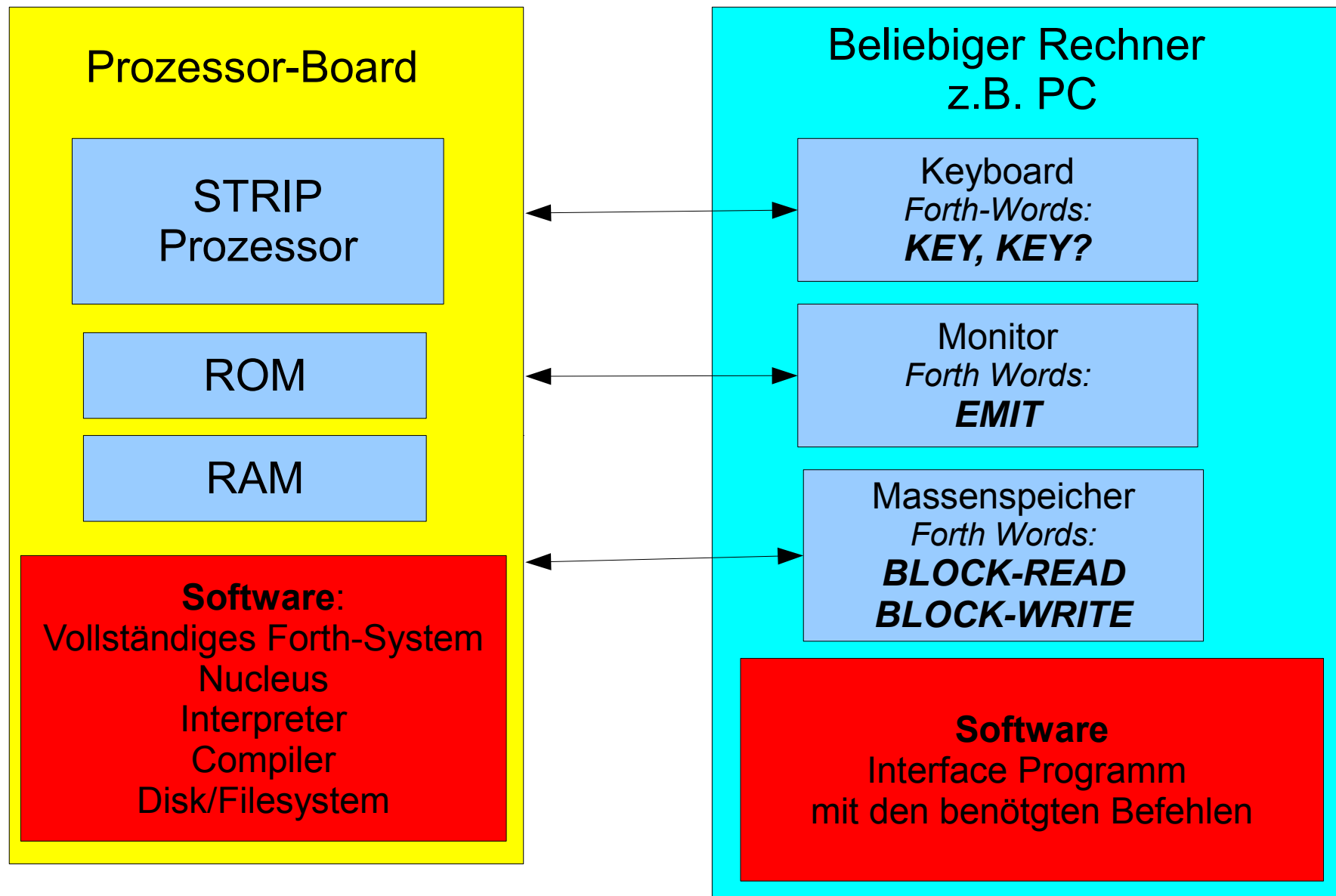
Forth-System mit Strip-Processor

- Strip-System als Forth Target am Host-Rechner
- Strip-System mit angeschlossenem Hilfs-Rechner als „dummes“ Terminal
- Strip-System als eigenständiger Forth Rechner

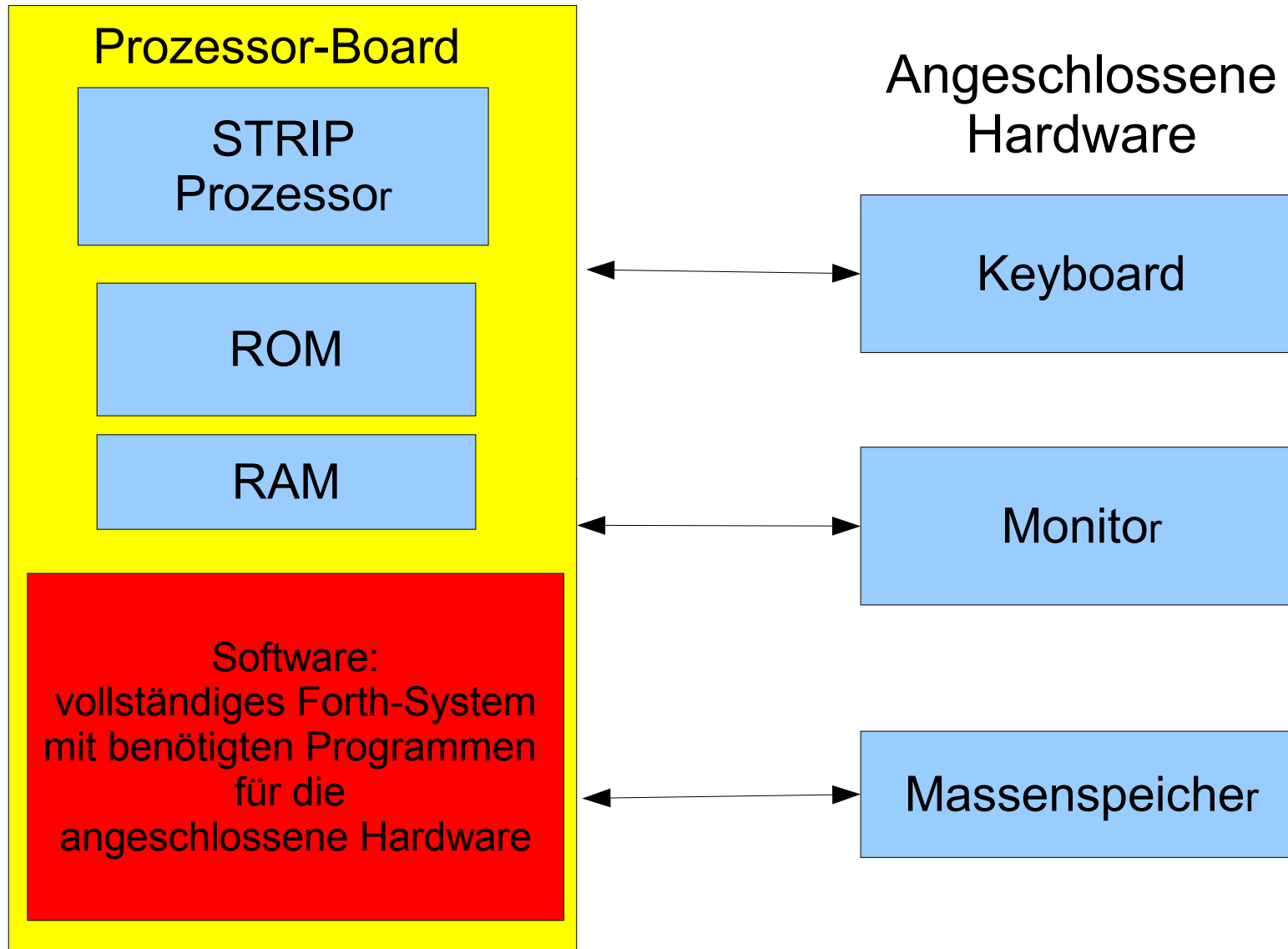
Strip-System als Forth Target am Host-Rechner



Strip-System mit angeschlossenem Hilfs-Rechner als „dummes“ Terminal



Strip-System als eigenständiger Forth Rechner



Ende

Noch Fragen?