

Ein- / Ausgabe im volksFORTH#

Table of Contents

- [Ein- / Ausgabe im volksFORTH](#)
- [Ein- / Ausgabebefehle im volksFORTH](#)
- [Ein- / Ausgaben über Terminal](#)
- [Ein- / Ausgabe von Zahlen](#)
- [Ein- / Ausgabe über einen Port](#)
- [Eingabe von Zeichen](#)

Ein- / Ausgabebefehle im volksFORTH#

Alle Eingabe- und Ausgabeworte (**KEY EXPECT EMIT TYPE** etc.) sind im volksFORTH vektorisiert, d.h. bei ihrem Aufruf wird die Codefeldadresse des zugehörigen Befehls aus einer Tabelle entnommen und ausgeführt. So ist im System eine Tabelle mit Namen DISPLAY enthalten, die für die Ausgabe auf dem Bildschirmterminal sorgt.

Dieses Verfahren der Vektorisierung bietet entscheidende Vorteile:

- Mit der Input-Vektorisierung kann man z.B. mit einem Schlag die Eingabe von der Tastatur auf ein Modem umschalten.
- Durch die Output-Vektorisierung können mit einer neuen Tabelle alle Ausgaben auf ein anderes Gerät (z.B. einen Drucker) geleitet werden, ohne die Ausgabebefehle selbst ändern zu müssen.
- Mit einem Wort (**DISPLAY**, **PRINT**) kann das gesamte Ausgabeverhalten geändert werden. Gibt man z.B. ein: `print 1 list display` wird Screen 1 auf einen Drucker ausgegeben und anschließend wieder auf den Bildschirm zurückgeschaltet. Man braucht also kein neues Wort, etwa **PRINTERLIST**, zu definieren.

Eine neue Tabelle wird mit dem Wort **OUTPUT**: erzeugt. Die Definition können Sie mit `view output`: nachsehen. **OUTPUT**: erwartet eine Liste von Ausgabeworten, die mit ; abgeschlossen werden muß.

Beispiel:

```
Output: >PRINTER
        pemit pcr ptype pdel ppage pat pat? ;
```

Damit wird eine neue Tabelle mit dem Namen **>PRINTER** angelegt. Beim späteren Aufruf von **>PRINTER** wird die Adresse dieser Tabelle in die Uservariable **OUTPUT** geschrieben. Ab sofort führt **EMIT** ein **PEMIT** aus, **TYPE** ein **PTYPE** usw.

Die Reihenfolge der Worte nach **OUTPUT**: `userEMIT userCR userTYPE userDEL userPAGE userAT userAT?` muß unbedingt eingehalten werden. Entsprechend wird die Input-Vektorisierung gehandhabt.

Ein- / Ausgaben über Terminal#

Das volksFORTH verfügt über eine Reihe von Konstanten, die der besseren Lesbarkeit dienen:

- [c/row](#)
- [c/col](#)
- [c/dis](#)
- [c/l](#)

- [l/s](#)
- [bl](#)
- [#esc](#)
- [#cr](#)
- [#lf](#)
- [#bel](#)
- [#bs](#)
- [standardi/o](#)
- [inputkol](#)
- [outputkol](#)
- [area](#)
- [areakol](#)
- [terminal](#)
- [window](#)
- [full](#)
- [curat?](#)
- [cur!](#)
- [setpage](#)
- [video@](#)
- [savevideo](#)
- [restorevideo](#)
- [catt](#)
- [list](#)
- [\(page](#)
- [page](#)
- [\(del](#)
- [del](#)
- [\(cr](#)
- [cr](#)
- [?cr](#)
- [\(at](#)
- [\(at?](#)
- [at](#)
- [at?](#)
- [col](#)
- [row](#)
- [curoff](#)
- [curon](#)
- [curshape](#)
- [printer](#)
- [print](#)
- [+print](#)
- [lst!](#)

Ein- / Ausgabe von Zahlen#

Die Eingabe von Zahlen erfolgt im interpretativen Modus über die Tastatur, wobei grundlegende Eingabeworte mit **number numbers** und den verwandten Worten definiert werden. Bei der Ausgabe von Zahlen ist wieder die fehlende Typisierung von FORTH zu beachten ? für ein bestimmtes Datenformat (integer, unsigned, double) ist jeweils der geeignete Operator auszuwählen.

- [.](#)
- [u.](#)
- [d.](#)

- [.r](#)
- [u.r](#)
- [d.r](#)

Ein- / Ausgabe über einen Port#

MS-DOS

- [pc@](#)
- [pc!](#)

Eingabe von Zeichen#

In FORTH wird man immer einen Speicherbereich benennen, in dem Zeichen und Zeichenketten verarbeitet werden. Hierfür verwendet man meistens einen kleinen, 80 Zeichen langen Speicherbereich namens **PAD**. Dieser Notizblock ? so die deutsche Übersetzung von pad ? belegt keinen festen Speicherbereich und steht sowohl dem FORTH-System als auch dem Programmierer zur Verfügung.

Dann möchte ich Ihnen mit dem Texteingabe-Puffer **TIB** einen weiteren wichtigen Speicherbereich vorstellen, der den vernünftigen Umgang mit den angeschlossenen Geräten sicherstellt. Weil die Texteingabe über die Tastatur relativ langsam vorsichgeht, werden die Zeichen hier erst in einem freien Speicherbereich, dem Pufferspeicher **TIB**, gesammelt und dann abgearbeitet.

- [tib](#)
- [#tib](#)
- [>tob](#)
- [>in](#)
- [pad](#)
- [input](#)
- [keyboard](#)
- [empty-keys](#)
- [\(key?](#)
- [key?](#)
- [\(key](#)
- [key](#)
- [\(decode](#)
- [\(expect](#)
- [expect](#)
- [span](#)
- [>expect](#)
- [nullstring?](#)
- [stop?](#)
- [source](#)
- [word](#)
- [parse](#)
- [name](#)
- [find](#)
- [execute](#)
- [perform](#)
- [query](#)
- [interpret](#)
- [output](#)
- [display](#)

- [\(emit](#)
- [emit](#)
- [charout](#)
- [tipp](#)
- [\(type](#)
- [type](#)
- [ltype](#)
- [space](#)
- [spaces](#)