amforth 3.1 Reference Card

Arithmetics

```
( n1 -- n2 )
( n1 -- n2 )
( n1 -- n2 )
( n1 -- n2 )
( n1 -- u1 )
( n1 -- n2 )
( n1 -- n2 )
( n1 -- n2 )
( d1 -- d2 )
( d1 -- d2 )
( d -- ud )
( d1 -- d2)
( d1 -- d2)
( d1 d2 -- d3 )
( d1 -- d2 )
( d1 d2 -- d3)
( n1 -- n2)
( n1 -- n2 )
( n1 n2 -- n3)
( n1 n2 -- n3 )
( n1 n2 -- n3)
( n1 n2 -- d)
( n1 n2 -- n3)
( n addr -- )
( n1 n2 -- n3 )
( n1 n2 -- n3)
( n1 n2 -- rem quot)
( n1 n2 -- n3 )
(n1 n2 n3 -- n4)
( n1 n2 n3 -- rem quot)
( d1 n -- rem ud2 )
( ud u2 -- rem quot)
( u1 u2 -- d)
(u1 u2 -- rem quot)
( -- 0 )
```

Character IO

```
( -- 32 )
( -- )
( c -- )
( -- f )
( -- c )
( -- f)
( -- )
( -- )
( n -- )
( addr n -- )
```

Compare

```
( d1 d2 -- flag )
( d1 d2 -- flasg)
( n1 n2 -- flag )
( n -- flag )
( n1 n2 -- flag )
( n1 -- flag )
( n1 n2 -- flasg)
( n1 -- flag)
( n1 n2 -- n1|n2 )
( n1 n2 -- n1|n2 )
( n1 n2 -- flag)
( n -- flag )
( u1 u2 -- flag )
( u1 u2 -- flasg)
```

Compiler

```
( -- )
( -- XT )
( -- )
( -- )
( -- xt )
( n -- )
( -- )
( -- )
( n <name> -- )
( addr1 -- addr2)
( -- )
( -- )
R(xt --)
( -- )
( --  )
( n -- )
( -- )
( --  )
( n <name> -- )
( -- )
( addr len -- )
( -- )
( <cchar> -- )
( -- addr )
( addr -- )
( addr -- )
( n -- )
( n <name> -- )
( -- )
```

Control Structure

```
( addr -- )
( -- addr )
( -- loop-sys )
( -- n )
; R( loop-sys -- loop-sys)
( -- addr )
( -- n )
; R( loop-sys1 loop-sys2 -- loop-sys1 loop-sy
( -- )
R(loop-sys --)
( loop-sys -- )
( addr -- )
( -- addr )
(addr1 -- addr2 )
( -- )
; R(loop-sys -- )
( dest -- orig dest )
```

Conversion

```
( d1 -- n1 )
( n1 -- d1 )
```

Dictionary

```
( n -- )
( -- )
( -- )
( -- XT )
```

Exceptions

```
( n*x -- )
R(n*y --)
( n*x -- )
R(n*y --)
( xt -- )
( n*x -- )
R(n*y --)
( -- addr )
( n -- )
```

Extended VM

```
( -- n2 )
( -- n )
( -- n )
( n -- )
( -- n2 )
( -- n2 )
( n1 -- n2 )
( -- n2 )
( -- n )
( -- n )
( n -- )
( -- n2 )
( -- n2 )
( n1 -- n2 )
( n1 -- n2 )
( n offs -- )
( n1 -- n2 )
( n offs -- )
( n -- )
( n -- )
```

Hardware Access

```
( -- c)
( -- f)
( -- )
(c -- )
( -- f)
( -- )
```

IO

```
( -- f )
```

Interrupt

```
( i -- xt )
( -- sreg )
( --  )
( xt i -- )
( -- n )
```

Logic

```
( n1 n2 -- n3 )
( n1 -- n2 )
( flag -- flag' )
( n1 n2 -- n3 )
( n1 n2 -- n3)
```

MCU

```
( -- )
( -- )
( -- )
( txbyte -- rxbyte)
(  addr len -- )
( -- )
```

Memory

```
( addr - c1 )
(addr-from addr-to n -- )
(addr-from addr-to n -- )
( c addr -- )
( addr - n)
( n addr -- )
( addr -- n )
( c-addr u c -- )
( addr -- n1 )
( n addr -- )
( n addr -- )
```

Multitasking

```
( -- )
```

Numeric IO

```
( -- addr )
( d1 -- )
( d1 n -- )
( -- )
( c base -- number flag )
( n -- )
( n w -- )
( -- )
( -- addr )
( c -- )
( -- )
(addr -- n )
( d1 -- )
( d1 -- addr count )
( d1 -- 0)
( n -- )
( ud1 w -- )
( ud w -- )
( ud1 -- )
( ud w -- )
( ud n -- )
```

Stack

```
( -- n )
( n -- )
( n -- n n )
( n1 n2 -- n1 n2 n1 )
( n1 -- [ n1 n1 ] | 0)
( n1 n2 n3 -- n2 n3 n1)
( -- n)
R(n -- n)
( -- n )
; R( n --)
( n1 n2 -- n2 n1)
( n -- )
; R( -- n)
```

Stackpointer

```
( -- addr)
(  -- n)
( n  -- )
; R( -- xy)
( -- addr)
( -- addr)
(  -- n)
( addr -- i*x)
```

String

```
( addr -- addr+1 n)
( addr1 n1 c -- addr1 n2 )
( addr1 n1 c -- addr2 n2 )
( char "ccc" -- c-addr u )
( addr1 len1 addr2 -- )
( addr1 u1 n-- addr2 u2 )
```

System

```
( addr n1 -- n2 )
( n -- )
( -- )
( xt1 -- xt2 )
( xt1 xt2 -- )
( xt -- )
( -- f_cou )
( -- addr )
( -- )
; R(i*x - j*x )
( xt1 c<char> -- )
( -- addr )
( xt|0 -- )
( -- )
( -- addr n )
( -- addr )
( addr -- )
```

System Value

```
( -- v)
( -- edp)
( -- faddr)
( -- addr)
( -- faddr )
( -- addr )
( -- addr )
( -- n )
( -- n*y )
```

Time

```
( -- )
```

Tools

```
( -- c )
( -- c )
( c-addr len searchstart -- [ 0 ] | [ xt [-1|
( -- )
( addr len -- [ 0 ] | [i*x -1 )
( addr -- [ addr 0 ] | [ xt [-1|1]] )
( r-addr r-len f-addr f-len --  f)
( addr -- addr+1 n )
( addr n --  )
( -- )
( n <name> -- )
( -- n )
( -- )
( c -- addr )
( -- )
```

internal/hidden

```
          (-- )
          (f -- )
          (-- addr )
          (--  )
          (limit counter -- )
          R(-- limit counter )
          (-- )
          (i*x -- j*x )
          (-- n1 )
          (-- )
          R(limit counter -- limit counter+1|)
          (n1 -- )
          R(llimit counter -- limit counter+n1|)
          (limit counter -- )
          R(-- limit counter| )
              .dw XT_FETCH
              .dw XT_EXIT
,)        ( addr len len' -- )
          ( -- addr)
          (spmcsr x addr -- )
          ( n -- )
          R(IP -- IP+1)
          (-- addr )
          (-- addr )
          ( xt1 -- xt2 )
          ( xt1 xt2 -- )
          ( -- addr )
          ( addr -- )
          (w -- )
          ( sreg -- )
          ( -- addr )
          ( addr -- )
          ( xt1 -- xt2 )
          ( xt1 xt2 -- )
          ( -- addr n)
          (x addr -- )
          (addr -- )
          (addr -- )
          (-- )
          (-- )
          (spmcsr x addr -- )
          ( xt1 -- xt2 )
          ( xt1 xt2 -- )
```