

CALLBACK word set:

callback (--)

callback-stack (*u* --)

end-callback (--)

handled (*flag* --)

DEFER word set:

action-of ("name" -- *xt*)
deferred? (*xt* -- 0 | 1 | 2)

defer ("name" --)
is (*xt* "name" --)

defer! (*xt2 xt1* --)

defer@ (*xt1* -- *xt2*)

DICTIONARY word set:

(find) (*c-addr u* -- 0 | *lfa flag*)
inline (--)
xt>abs (*xt* -- *a-addr.*)

(header) (*c-addr u* -- colon-sys)
lfa>xt (*lfa* -- *xt*)
xt>size (*xt* -- *size*)

.name (*xt* --)

allwords (--)

literalxt Compilation: (*xt* --) Run-time: (-- *xt*)

EVENTS word set:

(ekey) (*ticks.* -- event-type)
eventhandler (--)

(handle-event) (--)
get-event (*ticks.* -- event-type)

event (-- *a-addr*)

event-id (-- *u*)

handle-event (-- handled?)

ID word set:

(id) ("cccc" -- *d*)

[id] Compilation: ("cccc" --) Run-time: (-- *d*)

id ("name" --) name Execution: (-- *d*)

INTERPRETER word set:

(binary) ("... --)
(octal) ("... --)
unknown (*c-addr u* --)

(decimal) ("... --)
(radix) (*base* "... --)

(float) ("... --) (*F*: -- *r*)
parse-word (-- *c-addr u*)

(hex) ("... --)
skip (*char* --)

LAUNCHCODE word set:

activate-launchcode (*u* --)

cmdPB (-- *addr.*)

launchcode (-- *u*)

launchflags (-- *flags*)

MEMORY-ACCESS word set:

!a (*x a-addr.* --)
>rel (*addr* -- *addr.*)
cs! (*x a-csaddr* --)
csc, (*char* --)
rp! (*a-addr.* --)

2!a (*x1 x2 a-addr.* --)
@a (*a-addr.* -- *x*)
cs, (*x* --)
csc@ (*a-csaddr* -- *x*)
rp@ (-- *a-addr.*)

2@a (*a-addr.* -- *x1 x2*)
c@a (*char c-addr.* --)
cs@ (*a-csaddr* -- *x*)
cshore (-- *csaddr*)
sp! (*a-addr.* --)

>abs (*addr.* -- *addr*)
c@a (*c-addr.* -- *char*)
csc! (*char c-csaddr*)
csunused (-- *u*)
sp@ (-- *a-addr.*)

MISC word set:

(2>r) (*x1 x2 x3* -- *x3*) (*R*: -- *x2 x1*)
about (--)
dinvert (*x1.* -- *x2.*)
dswapends (*x1.* -- *x2.*)
enough? (*u* --)
fpdissect (*F: r* --) (-- sign exponent umantissa.)
m/mod (*number. denominator* -- *quotient. remainder*)
noop (--)
shash (*c-addr u* -- *shash*)

(2r>) (*x3* -- *x1 x2 x3*) (*R*: *x2 x1* --)
alias (*xt* "name" --)
dlshift (*x1. n* -- *x2.*)
dxor (*x1. x2.* -- *x3.*)
for Compilation: (--) Run-time: (+n --)
registered ("... --)
swapends (*x1.* -- *x2.*)

-1 (-- -1)
cold (--)
dor (*x1. x2.* -- *x3.*)
end-select Compilation: (--) Run-time: (-- *xt*)
hash (*c-addr u* -- *hash*)
mem (-- free. max.)
select Compilation: (--) Run-time: (*u* --)
xt Compilation: ("name" --) Run-time: (--)

0 (-- 0)
dand (*x1. x2.* -- *x3.*)
drshift (*x1. n* -- *x2.*)
fp0 (-- *f-addr*)
next (--)

MODULE word set:

end-module (--)
public: (--)

expose-module ("name" --)

module ("name" --)

private: (--)

OUTPUT word set:

?cr (--)
currenty (-- a-addr)
MainFormID (-- formID)
window-bounds (-- a-addr)

beep (--)
echo (newsetting -- oldsetting)
more (newflag -- oldflag)
wrap (newflag -- oldflag)

BlankFormID (-- formID)
font (fontID --)
ShowForm (formID --)

currentx (-- a-addr)
MainForm (--)
TitledFormID (-- formID)

SOURCE word set:

include ("filename" --)

needs ("filename" --)

set-memodb (creator-id. type --)

warnings (newflag -- oldflag)

STAND-ALONE word set:

CopyRsrc (number type. --)
NewRsrc (number type. -- err)

DelRsrc (number type. -- err)

generate-symbols (flag --)

MakePRC (xt creator. "name" --)

SYSTRAP word set:

>byte (n1 -- n2)
systrap (systrap --)

a0.L (-- x.)

d0.L (-- x.)

d0.L! (x. --)

TERMINATION word set:

(bye) (--)

byeThrow (-- n)

Legend:

prefix 'u' indicates an unsigned value
prefix 'a-' indicates a cell-aligned address
prefix 'c-' indicates a character-aligned address
suffix '!' indicates a double-cell 32-bit value

<i>flag</i>	flag	1 cell	<i>csaddr</i>	codespace address	1 cell
<i>char</i>	character	1 cell	<i>f-addr</i>	float aligned address	1 cell
<i>n</i>	signed number	1 cell	<i>colon-sys</i>	definition compilation	
<i>x</i>	unspecified cell	1 cell	"..."	parsed text	