

Format

CALL EXE(cpu-address[,...])

CALL EXE(numeric-variable[,...])

Description

The EXE subprogram directly goes to the cpu-address >8300 using the GPL XML >F0 and expects to work like an assembly BL @address so EXE(address) put that address at >8300 thus to return you need an assembly RT to end. Programmers can see this is a BL at a cpu-address. The programmer is responsible for keeping track of the workspace and program space he is using. Also for any registers while doing a BL or another context switch. A RT will end the BL as long as registers R11, R13, R14, R15 are not changed. By using CALL LOAD or CALL MOVES the programmer can set up a BL routine in the lower 8K by filling the registers with values first, then using CALL EXE to directly complete these commands. This is faster than CALL LINK as no interpretation of the access or values are checked.

Here is the example of using EXE doing VDP garbage collect the VDP memory not needing Memory Expansion but using the XB ROM 1 COMPCT routine.

CALL EXE(29656) ! does VDP COMPCT garbage collection
or execute the SCROLL routing in XB ROM 1
CALL EXE(31450) ! does SCROLL screen routine

FOR L=1 TO 4 :: CALL EXE(31450) :: NEXT L

This would scroll the screen 4 times like PRINT does.