

Format CALL SAMS(address-boundry,page-number[,...])

 CALL SAMS(address-boundry,numeric-variable
 [,...])

 CALL SAMS(command [,...])

Description

The SAMS command will only work with a SAMS memory card.
The address-boundry is a value in Hexadecimal denoted by
2 is >2000 or 3 is >3000 or A is >A000 or D is >D000

EXAMPLE: CALL SAMS(3,page-number[,...])

This 3 stands for >3000 hexidecimal address boundry.

CALL SAMS uses boundry symbols upper case only.

i.e. 2 = >2000, 3 = >3000, A = >A000, B = >B000, C = >C000,
D = >D000, E = >E000 and F = >F000

SAMS turns on the read/write lines of SAMS mapper registers
stores the value into the mapper register chosen. Less wasted
pages results in more memory available. Page numbers can be
from 0 to 16383 so it is hard to explain this easy.

See 16383 would be >FFFF hexidecimal 64 Meg SAMS. Pages 0 to
255 would be a 1 Meg SAMS, Pages 256 to 511 would be a 2 Meg
SAMS, so on up to page 7935 to 8191 would be 32 Meg SAMS.

Pages 8192 to 16383 would be above 32K Meg SAMS so RXB 2020
handles 64 Meg SAMS, but not tested above 32 Meg yet.

(*Note: 16384 to 32767 would be for above 32 Meg to 64 Meg.)

A additional new feature in 2020 RXB SAMS is use of upper 24K
memory can now be switched, but of course care must be taken
or will crash XB by removing the program running SAMS from
upper 24K. Imagine 8 Meg XB program swapping lines.

The order of changing 4K pages does not matter thus a
CALL SAMS(A,55,3,34) example is put 4K page 55 SAMS Memory
at >A000 and 4K page 34 at >3000

Original SAMS commands like ON, OFF, MAP or PASS still work.

"ON" turns on Mapper Registers.

"OFF" turns off Mapper Registers.

"MAP" turns on Map Mode so pages can be changed.

"PASS" default mode making the SAMS just like a normal 32K.