```
Format         CALL COINC(#sprite-number,#sprite-number,
               tolerance,numeric-variable[,...])

               CALL COINC(#sprite-number,dot-row,dot-column,
               tolerance,numeric-variable[,...])

               CALL COINC(ALL,numeric-variable[,...])
```

Description

See EXTENDED BASIC MANUAL PAGE 64 for more data. The only difference is the use the comma has been added for auto-repeat. Previously a COINC only allowed one sprite comparison per program line. COINC runs from ROM.

Programs
* See EXTENDED BASIC MANUAL page 64

```
Clear screen set and X to 190 | >100 CALL CLEAR :: X=190
Set up 3 sprites to be on     | >110 CALL SPRITE(#1,65,2,9,X,
the same vertical plane.      |  20,0,#2,66,2,9,X,30,0,#3,67,
                              |  2,9,X,-20,0)
COINC scans ALL sprites for a | >120 CALL COINC(ALL,A,#1,#2,1
collision then #1,#2,#3 also. |  2,B,#1,#3,12,C,#2,#3,12,D)
Print results on screen.      | >130 PRINT A;B;C;D
Loop forever to line 120      | >140 GOTO 120
```

The above program in RXB will put a -1 in A,B,C,D variables unlike normal XB that would never detect all 4 collisions.

Options

While characters 144 to 159 are being used, you cannot use sprites. Notice the ALL option MUST ALWAY BE FIRST as it was given highest priority to increase the detection rate. Though the ALL option does not improve much, the normal COINC detections are slightly faster as the interpreter is not looking to find the next COINC command on the next line number. Instead the comma and the next sprite is checked.