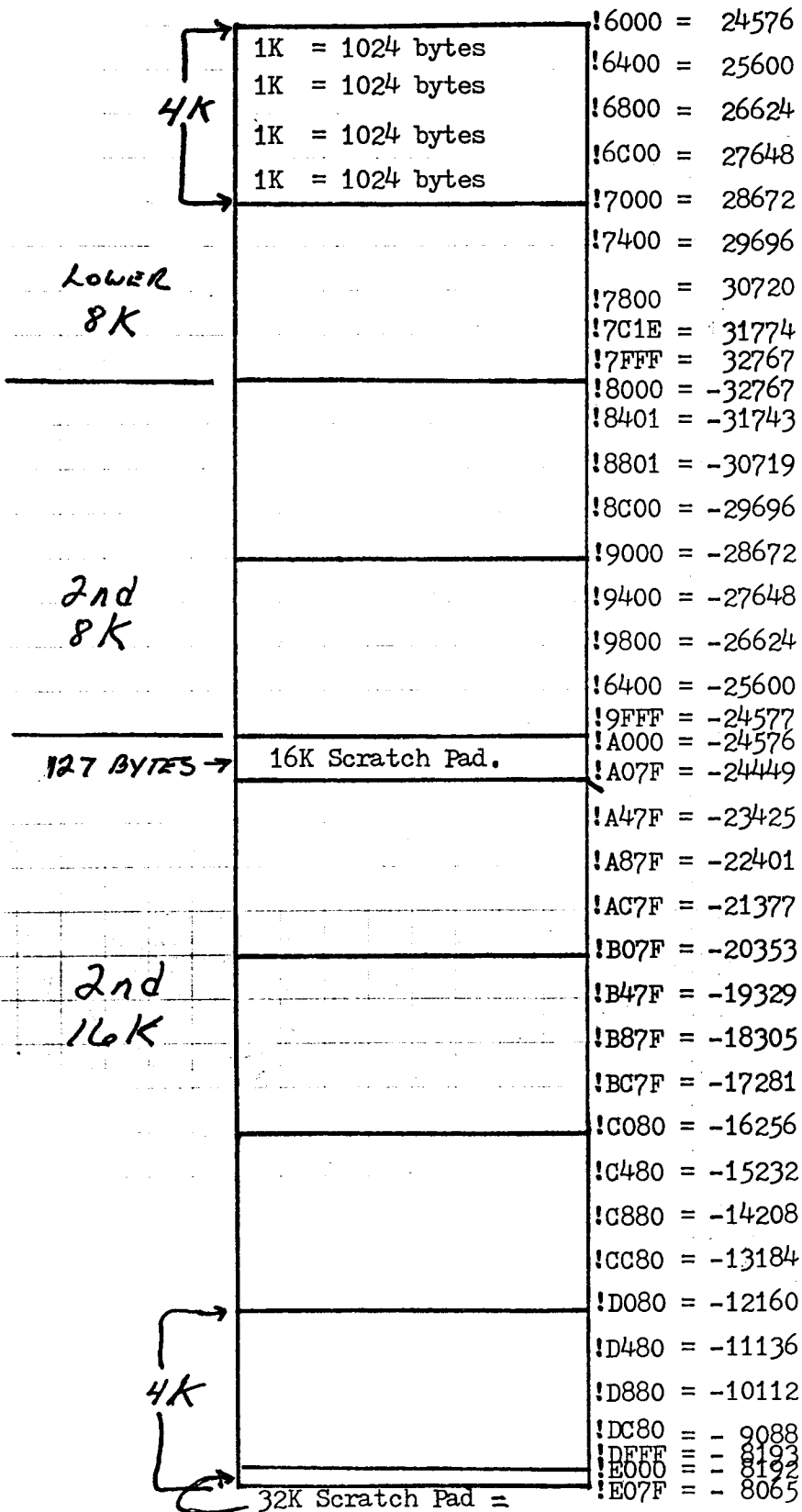


BLUE RAM MEMORY

SNAP & SHOW

12/83



- !6000 = 24576
- !6400 = 25600
- !6800 = 26624
- !6C00 = 27648
- !7000 = 28672
- !7400 = 29696
- !7800 = 30720
- !7C1E = 31774
- !7FFF = 32767
- !8000 = -32767
- !8401 = -31743
- !8801 = -30719
- !8C00 = -29696
- !9000 = -28672
- !9400 = -27648
- !9800 = -26624
- !6400 = -25600
- !9FFF = -24577
- !A000 = -24576
- !A07F = -24449
- !A47F = -23425
- !A87F = -22401
- !AC7F = -21377
- !B07F = -20353
- !B47F = -19329
- !B87F = -18305
- !BC7F = -17281
- !C080 = -16256
- !C480 = -15232
- !C880 = -14208
- !CC80 = -13184
- !D080 = -12160
- !D480 = -11136
- !D880 = -10112
- !DC80 = - 9088
- !DEFF = - 8193
- !E000 = - 8193
- !E07F = - 8065

When program is 3.1K SNAP to !7000 and above. Full screen = 4K bytes.
 From !6000 to !7000 = 4K
 From !7000 to !8000 = 4K

To calculate screen size in bytes let computer calc: $W \div 4 + (RM \# 0) * H + 4$

If I am programming in 3.1K I can SNAP into: !7000, !8000, !9000, !A07F, !B07F, !C080, !D080, !DEFF all with 4K screen images. I can SNAP into memory in between with smaller byte sizes.

If I am programming in 16K I can SNAP into !A07F, !B07F, !C080, !D080 & !DEFF - And in between for smaller sizes.

If my program is a full 32K I can only SNAP into !6000 to !7FFF.

SNAP x,y,w,h,location
 SNAP -10,-10,30,40,%(!8000) example
 SHOW x,y,sm,location
 SHOW -10,-10,1,%(!8000)
 sm=0 overlay / 1 OR / 2 XOR / 3 blank.

SNAPS can be made to @ (n) but additional programming will destroy it unless the SNAP is made at the end of the prog.

SNAPS can also be made to line numbers. In this case a reserve has to be made: For a 20 byte snap pick a LINE # & 10 A1234567890123456789
 SNAP -10,-10,30,40,%(>10) OR 7/0
 SHOW 0,0,sm,%(>10) / SHOW in diff, loc.
 THESE SNAPS CANNOT BE DESTROYED BY ADDITIONAL PROGRAMMING BUT ARE USUALLY ONLY GOOD FOR SHORT SNAPS UNLESS IN 16 or 32K.

IF NB=16 PRINT 32767 = !7FFF
 IF NB=10 PRINT !7FFF = 32767.
 Let computer calc. conversions.

SNAPS STAY IN ↑ MEMORY EVEN AFTER RESET!!
EVEN AFTER Computer Turned off!!

```

10 CLR;NB=16
20 INPUT "ENTER #"

Stanley Kendall


```