```
01F0   AC   PLO  RC   ;Initialize RC to Key scan ROM routine @ 8195
  F1   DC   SEP  RC   ;Do Key scan
  F2   AF   PLO  RF   ;Store instruction in RF.0
  F3   D4   SEP  R4
  F4   00             ;Call Tape Read/Write sub
  F5   90
  F6   F8   LDI       ;Load RD.0 with 00 to assure
  F7   00
  F8   AD   PLO  RD   ;Start (pseudo address) from page beginning
  F9   D5   SEP  R5   ;Return

                  PRINT ADDRESS AND INSTRUCTION

0200   9D   GHI  RD   ;Get first half pseudo address in RD.1
  01   BF   PHI  RF   ;Put in RF.1 to pass to sub
  02   D4   SEP  R4
  03   02             ;Call Convert/Store ASCII in text
  04   1F
  05   8D   GLO  RD   ;Get second half pseudo address in RD.0
  06   BF   PHI  RF   ;Put in RF.1 to pass to sub
  07   D4   SEP  R4
  08   02             ;Call Convert/Store ASCII in text
  09   1F             ;(Address is now printed)
  0A   19   INC  R9   ;Buffer pointer + 01 to create space
 *0B   0A   LDN  RA   ;Get instruction for disassembly
  0C   BF   PHI  RF   ;Put in RF.1 to pass to sub
  0D   D4   SEP  R4
  0E   02             ;Call Convert/Store ASCII in text
  0F   1F
0210   D5   SEP  R5   ;Return (R9 points just after last instruction printed)

                  ASCII CONVERSION (HEX DIGITS)

0211   8E   GLO  RE   ;Get value passed by caller
  12   FD   SDI       ;Value - 09 (If result negative, value > 9)
  13   09
  14   33   BPZ       ;Branch if value ≤ 9 (not a letter)
  15   1A
  16   8E   GLO  RE   ;Get the same value
  17   FC   ADI       ;Add 07 if value is a letter
  18   07
  19   AE   PLO  RE   ;Store in RE.0
  1A   8E   GLO  RE   ;Get value in RE.0 (Either plus 07 or not depending)
  1B   FC   ADI       ;Add 30 hex always to complete conversion
  1C   30
  1D   AE   PLO  RE   ;Store converted value in RE.0
  1E   D5   SEP  R5   ;Return
```

*Note- Entry point for printing arguments to instructions