Interrupt priorities within a process are determined by the binary magnitude of the high order 45 bits of their names.

1) Allows more than one subprocess to have same priority (i.e. can't interrupt each other)

2) Protect system by using early class codes

3) User can control priorities of his own subprocess by remembering the order in which he changes the class codes or by assigning appropriate temp parts (high order 15 bits of 30 bit temp part)

4) Use top of stick interrupt inhibit to disable interrupts at same priority

5) Use real time global inhibit to lock out all interrupts
   Its value compared to real-time
   in sec clock on all swapin
   if expired - cause error
   get to subsidiary interrupt
   can only be set once - another attempt to set before reset is error