DCB Data Compression

The DCB is equipped with an 8-bit RadHard 8085 processor operating at 4MHz (2 MHz cycle). The software performance numbers are calculated based upon this assumption.

The following concept is that raw uncompressed packets are in memory and the flight software uses its spare time to compress packets “in situ”. As it compresses packets, it changes the APID to mark each packet appropriately.Software Options.

1.1 Software Compressors

Multiple software compression algorithms could be in the system and operate on different APIDs. Each algorithm could be tailored for each APID. Uplinked software algorithms could be tailored for the type of data that we encounter ***after launch***.

1.1.1 Huffman4 Algorithm

A rough program segment of the 8085 software to encode an arbitrary input string using Huffman 4-bit encoding yielded the following information:

40 cycles per byte of input string
110 cycles per byte of output string

Assuming 1KB input and .5 KB output string, the 8085 will need 95 Kcycles or .0475 seconds (21 Kbytes per second). A full memory (750 Mbits) compressed by the 8085 using Huffman4 would take 75 minutes.

1.1.2 Delta Modulator

Using an 8 sample adaptive delta modulator (for waveforms), a rough program segment of the 8085 was constructed with the following results:

680 cycles	Calculate the difference array (8 values)
50 cycles	Decide which encoding length
335-640 cyc	Encoding the output string (variable length)
68-87 cycles	per byte of input string

Assuming 1KB input string, the 8085 will need 70 to 89 Kcycles or .035 to .0445 seconds for each string (>22 KB/second). A full memory (750 Mbits) compressed by the 8085 using Delta Modulator would take 71 minutes.

1.2 Hardware Options.

1.2.1 Huffman4 Encoder.

Assuming the DCB hardware can provide a Huffman encoder, one would expect this would have an 8 MHz output clock, producing 1 Mbyte per second of output, reading bytes from the SSR and shifting out Huffman encoded data at 8 Mbps. A typical 1KB block would be compressed in 1/800th of a second.

The software would have some difficulty keeping up with this packet rate. A more likely performance figure would be 250 packets per second or so. At 1KB per packet, the software/hardware mix would require 94000/250 or about 6 minutes.