Binary Message Protocol and Proof of Concept Prototype Update

Presented by

Microcom Design, Inc.

May 2022

Binary Protocol – A Brief History

- ASCII and Pseudo-Binary (a subset of ASCII characters) have been the bread and butter of DCS messages for decades.
  - Use of ASCII characters and prohibition of certain non-printable ASCII codes dates back to the teletype days, and was a requirement of the NWS.
- Binary is referenced in both the original and second generation High Data Rate Certification Standards (CS1 & CS2), but was never fully defined.
  - CS1 included the statement … "*precise format and error checking for HDR binary transmissions is TBD*".
  - CS2 noted that a Binary Protocol Specification would to be "*published separately*" and would then be appended the to standard.
- While several approaches to come up with a Binary Protocol were attempted even before CS2 was released, the last major discussion on this topic occurred at the May 2012 TWG in Boise, Idaho.
  - Sutron (now Ott), Signal Engineering, and Microcom all submitted recommendations in 2005.
  - EDDN's Binary Transmission Project (BTP) in the 2009-2010 timeframe.

Binary Protocol – Moving Forward

- While the last serious discussion on a Binary Protocol Specification was in 2012, it continued to be a somewhat regular, albeit informal, discussion topic.
  - As part of its sustainment contract, Microcom annually prepared a system analysis document that includes recommendations for work assignment projects.
  - The Binary Protocol has always been a low priority task on the recommendation list with a caveat, that if Binary is ever going to happen, *NOAA must take the lead*.
- In September 2021, Microcom once again prepared a system analysis document and again included Binary Protocol on the list.
- This time however, NOAA decided it was time to move forward and in November 2021 approved a work assignment that directed Microcom to …
  - Perform a "fresh" review and analysis of the previous recommendations with a focus on Pros and Cons of some of the specific suggestions.
  - Develop a proof-of-concept demonstration of the Binary protocol suggested in an attempt to revitalize the topic and gauge user interest in a Binary Protocol.

Binary Protocol – The Fresh Look

- Dr. Brian Kopp and Matt Taylor performed the *fresh look*:
  - Both have extensive expertise in RF communications and data processing.
  - They are intimately familiar with the DCS from a manufacturer's and a user's perspective, but neither were familiar with the prior work on a Binary Protocol.
- To aid Dr. Kopp and Matt Taylor in their review and analysis, Microcom and NOAA held a TIM to get NOAA's input on setting priorities for potential advantages of using binary.
  - NOAA/NESDIS' mission with regard to the DCS is to provide users with a reliable satellite telemetry capability to gather the *user's* environmental data so NOAA's top priority was to offer the user's the option to use binary with as few restrictions as possible.
  - The Binary Protocol should be like the existing ASCII and Pseudo-Binary specifications with just enough specified to allow the data to be collected, and should not be overly concerned with any specific advantage (e.g. efficiency, reliability, compatibility, etc.), but instead should allow user's flexibility in the data portion of the message.
- Kopp's and Taylor's initial recommendations were presented at the December virtual STIWG and the final recommendations were presented to NOAA in early January 2022.

Binary Protocol – Key Recommendations

- Message Length
  - Use message length in place of EOT.
  - Need 14-bit message length field.
  - BCH protect message length and original Flag byte.
- CRC-16
  - Append 16-bit CRC to data field to replace Odd Parity bits in each byte as was used in the Pseudo-Binary and ASCII formats.
  - Use the code polynomial 0xd175 = $x^{16}+ x^{15}+ x^{13}+ x^{9}+ x^{7}+ x^{6}+ x^{5}+ x^{3}+x+1$
- Reduced Flush
  - Originally Microcom believed it was possible to eliminate the flush entirely due to the fact that the message length is known.
  - However, implementation and testing showed that a 16-bit flush is needed.
  - Current HDR specification requires 32 bits of flush.

Binary Protocol – Comparison to ASCII/Pseudo-Binary

Current 300 & 1200 bps ASCII and Pseudo-Binary DCPRS Message Format							
Carrier 0.5s/ 0.25s	Clock States 3 ‘0-1’	FSS 15 bits	GOES ID 32 bits	Flag Word 8 bits	DCP DATA Max: 32,000 bits @ 300 bps 128,000 bits @ 1200 bps	EOT 8 bits	Encoder Flush 32 bits

Proposed 300 bps Binary DCPRS Message Format									
Carrier 0.5s	Clock States 3 ‘0-1’	FSS 15 bits	GOES ID 32 bits	Flag Word 8 bits	Msg Length 14 bits	BCH 10 bits	DCP DATA Max: 32,000 bits	CRC 16 Bits	Encoder Flush 16 bits

Proposed 1200 bps Binary DCPRS Message Format												
Carrier 0.25s	Clock States 3 ‘0-1’	FSS 15 bits	GOES ID 32 bits	Flag Word 8 bits	Msg Length 14 bits	BCH 10 bits	DCP DATA 32,000 bits	CRC 16 Bits	…	DCP DATA Max: 32,000 bits	CRC 16 Bits	Encoder Flush 16 bits

BCH Encoded 31 bits

Maximum of 128,000 bits which will be split into four sections of 32,000 bits each with a 16 bit CRC

Binary Protocol – BCH Protected Length

- BCH Details:
  - DCP address currently utilizes Bose-Chaudhuri-Hocquenghem (31,21) encoding scheme.
  - This error correction/detection code can correct all 2-bit errors and detect most 3-bit errors.
  - Since the length is only 14-bits, the Flag Word and Message Length are combined to produce the 21-bit information portion, which is then used to generate the 10-bit check field.
  - The normal Odd Parity of the Flag Word is still included in the overall 32-bit (4-byte) field bit as the most significant bit (MSB).
- Flag Word:
  - The 7 information bits in the Flag Word will not be changed.
  - Two of the bits (FW6 & FW5) identify the message as ASCII, Pseudo-Binary, or Binary.
  - One of the reserved bit (FW2) was utilized in the prototype demo to identify Compact PB.

Flag Word								Message Length														Parity									
7	6	5	4	3	2	1	0	13	12	11	10	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0

Odd Parity Bit

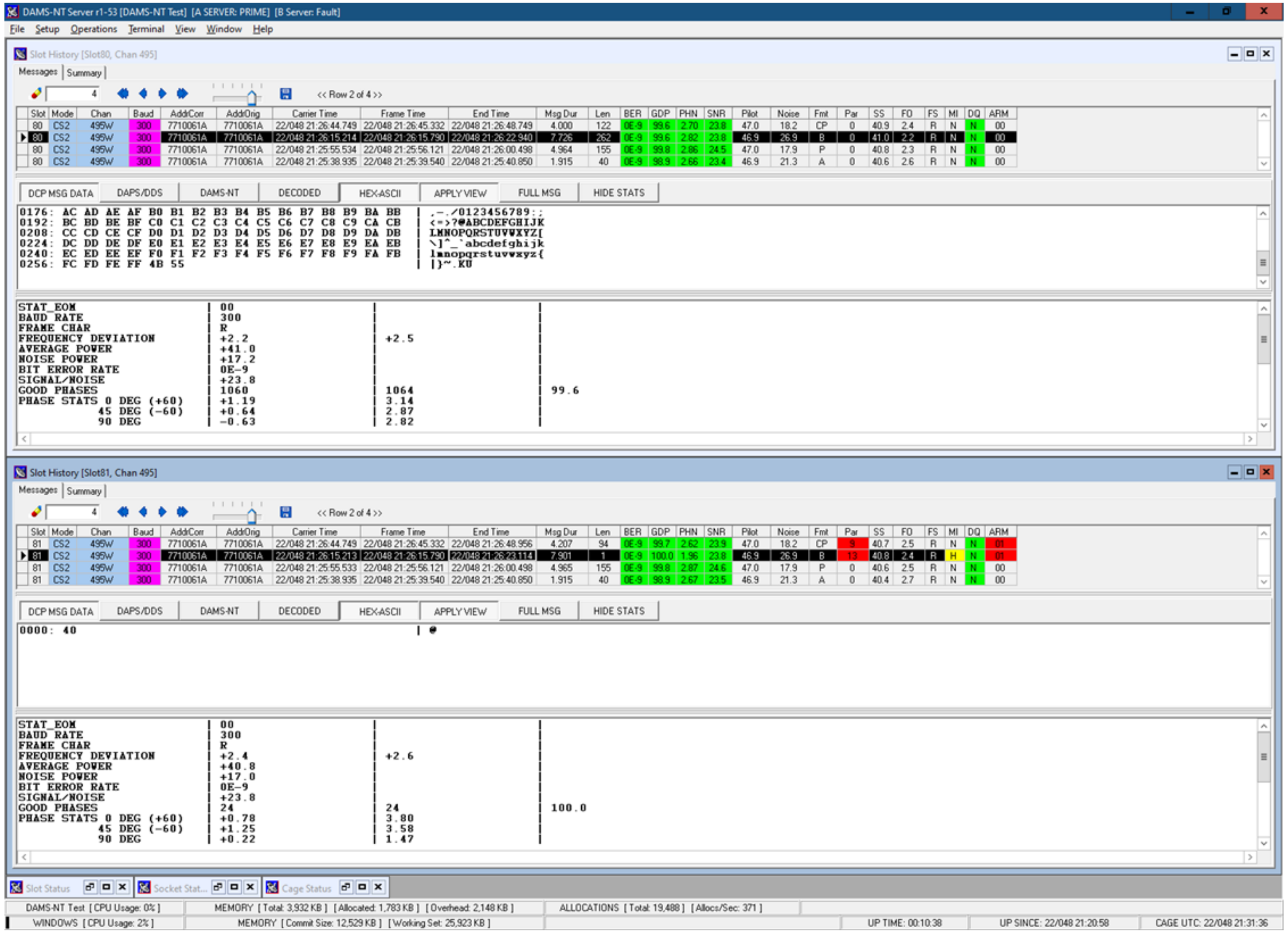
21 Bit BCH Encoded Block

10 Bit BCH Parity

Binary Protocol – Proof-of-Concept Demo

- Once NOAA accepted the Binary Protocol recommendations, Microcom began implementing the protocol in the DCS Pilot/Test Transmitter (P/T Tx) and the DAMS-NT Demodulator and Receiver systems.
  - Prototype work was completed in January and early February 2022; virtual demonstration was provided to NOAA in late February.
  - Microcom commanded the P/T Tx at its office in Hunt Valley and received the test messages on the Sustainment DAMS-NT/DADDS rail also at Microcom's facility in Hunt Valley.
  - Binary messages were also detected by the NOAA sites, but were not properly received since the operational DAMS-NT/DADDS systems had not been updated.
- In addition to sending some test legacy ASCII/Pseudo-Binary messages, the Demo consisted of two main binary examples:
  - A binary fill message that consisted of all 256 8-bit binary values to demonstrate that the implementation would not be confused by key values (e.g. the ASCII EOT character – 0x04).
  - A Compacted Pseudo-Binary example using a typical Florida Department of Transportation message to demonstrate how Compact PB can reduce the message size by ~25% while still conveying the same amount of environmental and system information.

Binary Protocol – Binary Fill Example – Demod Comparison

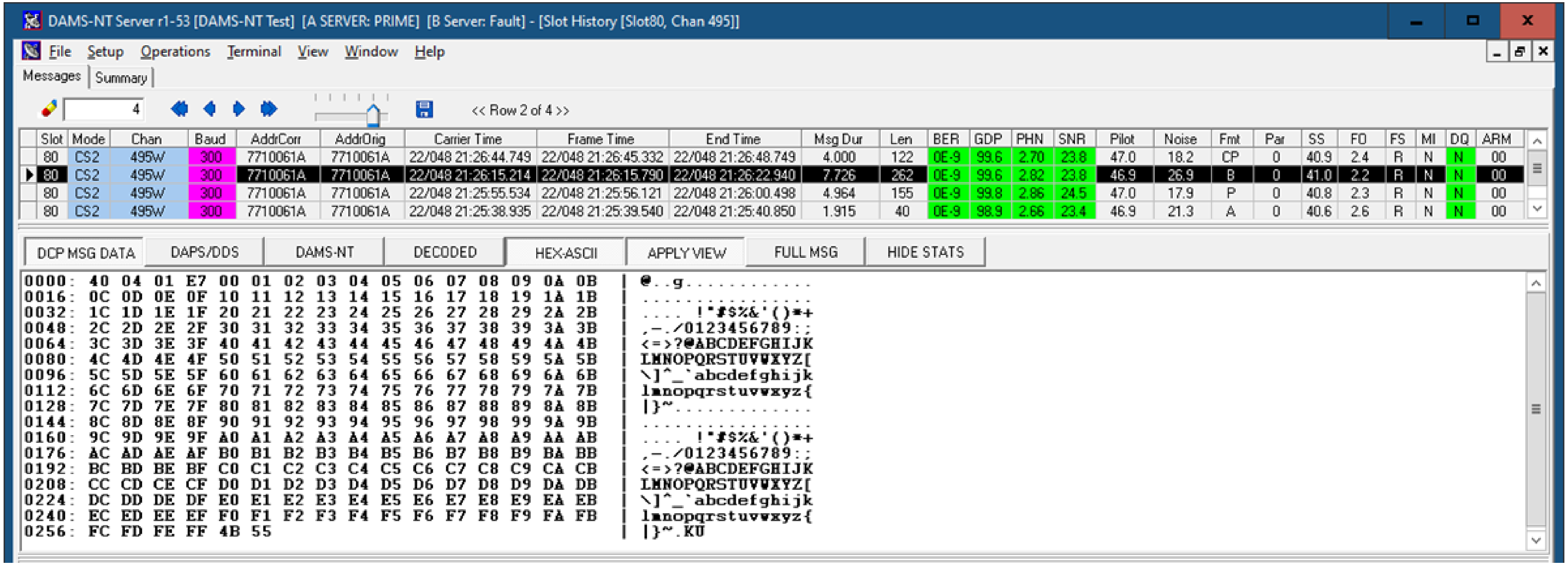


- Received properly by top demod (Slot 80), which was updated to support new binary protocol.
- Not received by bottom demod (Slot 81) that had not been updated.
  - Summary grid shows legacy demod identified message as binary (‘B’).

  - Albeit with parity errors.
  - Even though message lasted nearly 8 seconds, only 1 data byte was reported.

Binary Protocol – Binary Fill Example – Updated Demod

- Updated demod reported 262 bytes received: The 256 binary values (00-FF), the 4-byte Flag & Length field (40 04 01 E7), and the 2-byte CRC (4B 55).
- Note that the second data byte being 0x04 (ASCII EOT) is why the legacy demod truncated the message data to single byte as was shown in the previous slide.



Binary Protocol – Binary Fill Example – Error Detection

STAT_EOM	00		
BAUD RATE	300		
FRAME CHAR	R		
FREQUENCY DEVIATION	+2.2	+2.5	
AVERAGE POWER	+41.0		
NOISE POWER	+17.2		
BIT ERROR RATE	0E-9		
SIGNAL/NOISE	+23.8		
GOOD PHASES	1060	1064	99.6
PHASE STATS 0 DEG (+60)	+1.19	3.14	
45 DEG (-60)	+0.64	2.87	
90 DEG	-0.63	2.82	
135 DEG	-0.41	2.80	
180 DEG	+0.58	2.81	
225 DEG	+0.76	2.52	
270 DEG	-1.42	2.77	
315 DEG	-0.59	2.91	
MOD INDX/PHAS NOISE	+0.02	2.82	+1.42
MESSAGE CARRIER TIME	22/048 21:26:15.214		
MESSAGE SYNC TIME	22/048 21:26:15.674		
MESSAGE FRAME TIME	22/048 21:26:15.790		
MESSAGE END TIME	22/048 21:26:22.940		
BINARY BCH (R|C|L)	400401E7	400401E7	256
BINARY CRC (R|C)	4B55	4B55	

DAMS-NT Test [CPU Usage: 0%] | MEMORY [Total: 3,932 KB] [Allocated: 1,783 KB] [Overhead: 2,148 KB] | ALLOCATIONS [Total: 19,427] [Allocs/Sec: 371]

WINDOWS [CPU Usage: 1%] | MEMORY [Commit Size: 12,439 KB] [Working Set: 25,956 KB] | UP TIME: 00:17:11 | UP SINCE: 22/048 21:20:58 | CAGE UTC: 22/048 21:38:09

- Check of BCH and CRC fields is shown in DAMS-NT Stats section (highlighted lines).
  - For both fields, the received value (R) matches the checked (C) value (i.e. no errors).
- The decoded BCH field shows a data length (L) of 256 bytes, which is the actual message data length and does not include the BCH and CRC overhead bytes.

Binary Protocol – Compacted Pseudo-Binary Example

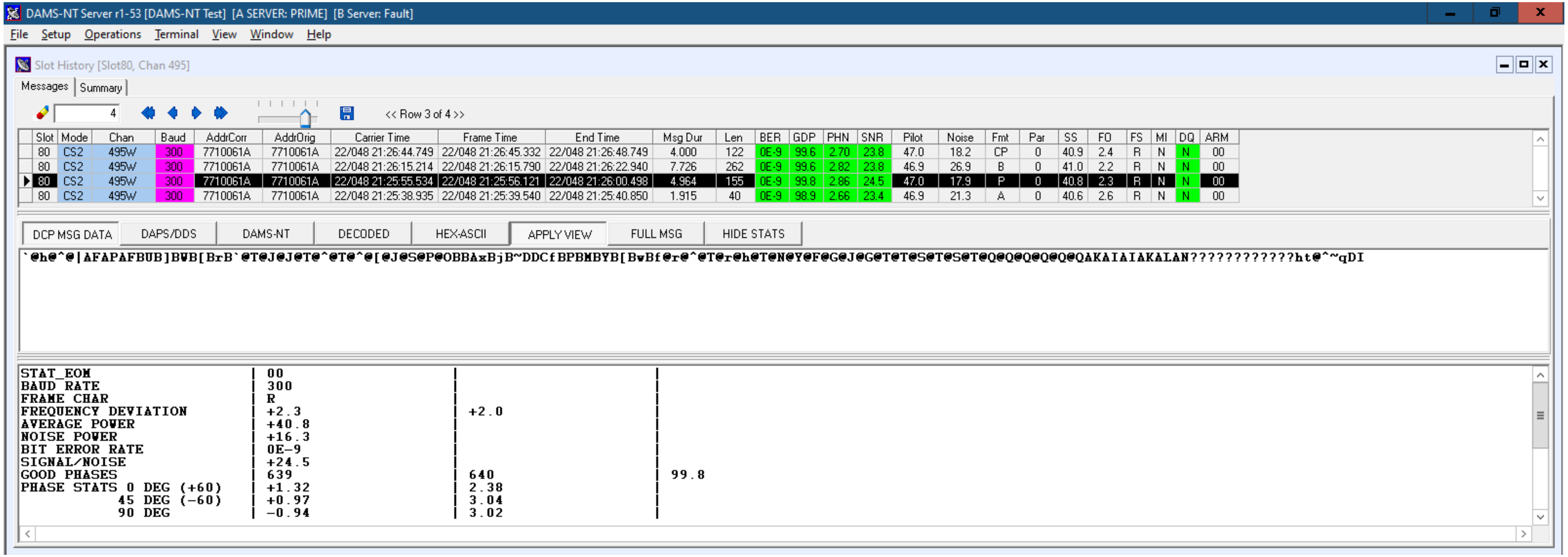
- Microcom's Compacted Pseudo-Binary recommendation was a subset of its overall Binary Protocol recommendation, and was offered as a way to allow users to quickly transition to Binary.
  - It is based on the fact that there are only 6-bits of information in each Pseudo-Binary byte.
  - The other two are the odd parity bit and a forced bit to make the byte a printable character.

$$P_O\ F\ B_5\ B_4\ B_3\ B_2\ B_1\ B_0$$

  - The Forced is typically a 1, but can be a 0 if all the information bits are 1 allowing the question mark character (? = 0x3F) to be sent instead of the ASCII DEL code (0x7F).
- Discarding the 2 non-informational bits and then concatenating the six information bits from four bytes results in a total of 24-bits that can then chopped up into three bytes.
  - This will yield an immediate 25% reduction in raw data size.
  - The compaction is done at the transmitter and the de-compaction occurs at the receiver.
  - Transmitters and data loggers can continue to use the same configuration setups and processing software can use existing decoding scripts – dubbed "IT Transparent".

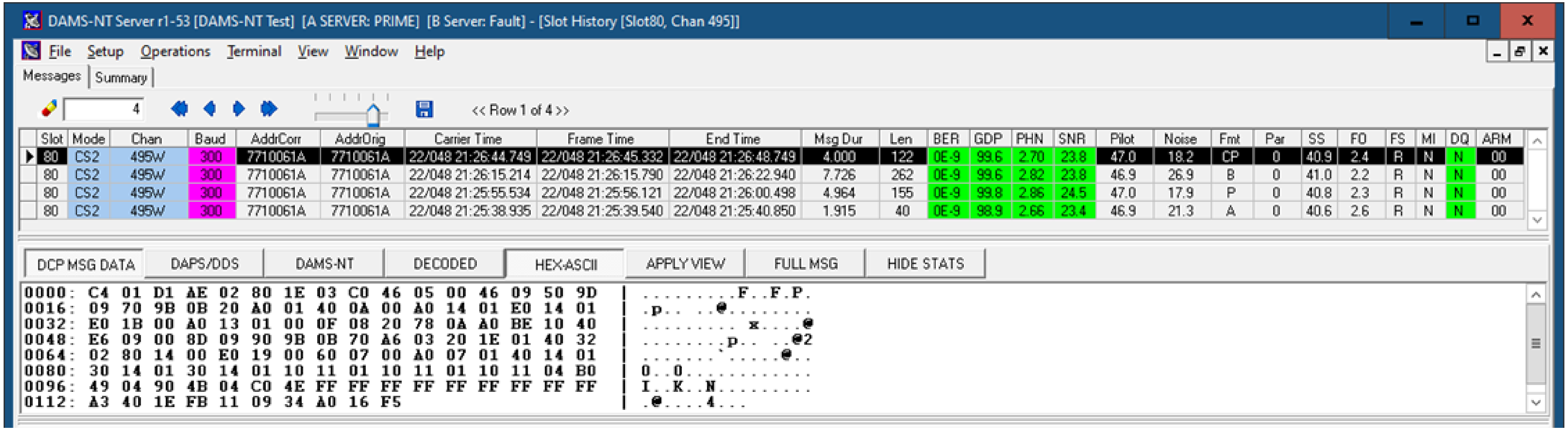
Binary Protocol – Compact PB Example – Legacy PB Message

- Example messaged consisted of 155 Pseudo-Binary characters.
- Total message duration ~5 seconds.



Binary Protocol – Compact PB Example – Raw Binary

- Compact PB message is 122 bytes: 116 bytes of compacted data; 4 Flag & Length bytes (C4 01 D1 AE) and 2 CRC bytes (16 F5).
  - 25% raw data reduction (116/154); 21% total byte count reduction (122/155).
- Total message duration ~4 seconds (20% overall reduction).



Binary Protocol – Compact PB Example – De-Compacted

DAMS-NT Server r1-53 [DAMS-NT Test] [A SERVER: PRIME] [B Server: Fault] - [Slot History [Slot80, Chan 495]]

File Setup Operations Terminal View Window Help

Messages | Summary

4 << Row 1 of 4 >>

Slot	Mode	Chan	Baud	AddrCorr	AddrOrig	Carrier Time	Frame Time	End Time	Msg Dur	Len	BER	GDP	PHN	SNR	Pilot	Noise	Fmt	Par	SS	FO	FS	MI	DQ	ARM
80	CS2	495W	300	7710061A	7710061A	22/048 21:26:44.749	22/048 21:26:45.332	22/048 21:26:48.749	4.000	122	0E-9	99.6	2.70	23.8	47.0	18.2	CP	0	40.9	2.4	R	N	N	00
80	CS2	495W	300	7710061A	7710061A	22/048 21:26:15.214	22/048 21:26:15.790	22/048 21:26:22.940	7.726	262	0E-9	99.6	2.82	23.8	46.9	26.9	B	0	41.0	2.2	R	N	N	00
80	CS2	495W	300	7710061A	7710061A	22/048 21:25:55.534	22/048 21:25:56.121	22/048 21:26:00.498	4.964	155	0E-9	99.8	2.86	24.5	47.0	17.9	P	0	40.8	2.3	R	N	N	00
80	CS2	495W	300	7710061A	7710061A	22/048 21:25:38.935	22/048 21:25:39.540	22/048 21:25:40.850	1.915	40	0E-9	98.9	2.66	23.4	46.9	21.3	A	0	40.6	2.6	R	N	N	00

DCP MSG DATA | DAPS/DDS | DAMS-NT | DECODED | HEX-ASCII | APPLY VIEW | FULL MSG | HIDE STATS

```
d@h@^@|AFAPAFBUB]BWB[BrB`@T@J@J@T@^@T@^@[@J@S@P@OBBAxBjB~DDCfBPBMBYB
[BwBf@r@^@T@r@h@T@N@Y@F@G@J@G@T@T@S@T@S@T@Q@Q@Q@Q@Q@QAKAIAIAKALAN????????????ht@^~qDIMJ
```

De-Compacted message restores PB characters and Flag byte.

```
STAT_EOM                  | 00                 |          |
BAUD RATE                 | 300                |          |
FRAME CHAR                | R                  |          |
FREQUENCY DEVIATION       | +2.4               | +3.1     |
AVERAGE POWER             | +40.9              |          |
NOISE POWER               | +17.1              |          |
BIT ERROR RATE            | 0E-9               |          |
SIGNAL/NOISE              | +23.8              |          |
GOOD PHASES               | 502                | 504      | 99.6
PHASE STATS 0 DEG (+60)   | +0.79              | 2.38     |
           45 DEG (-60)   | -0.12              | 2.72     |
           90 DEG         | -0.79              | 3.27     |
          135 DEG         | -0.76              | 2.90     |
          180 DEG         | +1.65              | 2.63     |
          225 DEG         | -0.04              | 2.45     |
          270 DEG         | -1.20              | 2.46     |
          315 DEG         | -0.01              | 2.82     |
MOD INDX/PHAS NOISE       | -0.06              | 2.70     | +1.65
MESSAGE CARRIER TIME      | 22/048 21:26:44.749 |         |
MESSAGE SYNC TIME         | 22/048 21:26:45.216 |         |
MESSAGE FRAME TIME        | 22/048 21:26:45.332 |         |
MESSAGE END TIME          | 22/048 21:26:48.749 |         |
BINARY BCH (R|C|L)        | C401D1AE           | C401D1AE | 116
BINARY CRC (R|C)          | 16F5               | 16F5     |
```

Binary fields still reported in Stats
Length in BCH is Compacted byte count.

DAMS-NT Test [CPU Usage: 0%] | MEMORY [Total: 3,932 KB] [Allocated: 1,783 KB] [Overhead: 2,148 KB] | ALLOCATIONS [Total: 19,328] [Allocs/Sec: 374]

WINDOWS [CPU Usage: 3%] | MEMORY [Commit Size: 12,304 KB] [Working Set: 25,755 KB] | UP TIME: 00:34:04 | UP SINCE: 22/048 21:20:58 | CAGE UTC: 22/048 21:55:03

Binary Protocol – Key Decisions

- *Move forward with Binary Protocol as recommended?*

  Pro: Will allow users to begin to use Binary DCS messages, and provide system flexibility and security going forward

  Con: Requires updates to Certification Specification, DCS transmitters, and DCS receive systems (initial implementation already done).

- *Include Compacted Pseudo-Binary?*

  Pro: Will allow users to transition to binary quickly using existing message coding schemes and decoding scripts (IT Transparent).

  Con: Requires additional work updating Certification Specification, DCS transmitters, and DCS receive systems (initial implementation already done).

Binary Protocol – ASCII Compaction

- Microcom's original Binary Protocol proposal also included a similar suggestion for compacted ASCII.
  - Subset of all ASCII characters would be encoded in 4 or 5 bits and compacted.
  - If the DCS community is going to make updates, now would be the time to include this approach if it desirable to the DCS community.

- *Include Compacted ASCII?*

Pros: Will allow ASCII message users to transition to binary quickly using existing message coding and decoding schemes (IT Transparent).

Will allow human readable data on DADDS Field Test even when using binary messages.

Con: Requires additional work updating Certification Specification, DCS transmitters, and DCS receive systems (was not prototyped).

Binary Protocol – Summary and Next Steps

- Summary
  - The idea of binary message data has been around for quite some time.
  - It's use has been held back since a complete Binary Protocol specification has never been fully defined and adopted by the DCS community.
  - NOAA funded a new study and proof-of-concept prototype to push forward on a proposed Binary Protocol.
    - Preliminary recommendation has been made, but needs to be accepted and ratified.
    - Successful Proof-of-Concept has been developed, demonstrated and reported on herein.
- Next Steps
  - Decision to move forward.
  - User and manufacturer feedback on protocol recommendations and key decisions.
  - NOAA to fund initial Certification Specification updates that will then be shared with DCS community for questions and comments (may be tied in with Lat/Long Tx ID).