Analysis of Reported
Compact Position Reporting (CPR) Anomalies
in Operational Airspace

Introduction / Abstract:

An appropriate Aviation Regulatory agency close to implementation of Automatic Dependent Surveillance-Broadcast (ADS-B) has produced data that initially indicates that at least one Avionics manufacturer has transponders operating in the airspace that do not encode longitude information correctly in Airborne Position Messages. This paper discusses the analysis performed in regards to the data received and demonstrates results that vindicate the transponders in question but also demonstrate that there are deficiencies in the ADS-B encoding scheme that transmitting and receiving media must be aware of and must compensate appropriately.

Case 1: Airborne Position Message, Odd Encoding ___Aircraft Identification = VOZ858

Part a: Initial Data Received

The reporting agency reported that the following Airborne Position Message was received by the Ground Station and considered to be in error in regards to Encoded Longitude.

Original Odd Encoded Airborne Position Message Received:		
Message Content:	8D7C6D2560C387D7B452F8000000	
Aircraft Address:	7C6D25	**VOZ858**
Encoded Latitude :	**1EBDA** (Hex)	125914 (Decimal)
Encoded Longitude :	**052F8** (Hex)	21240 (Decimal)
Latitude (decoded at Receiver) :	-36.850285934	
Longitude (decoded at Receiver) :	149.963856573	
Receiver Latitude Seed Value :	-36.850285934	
Receiver Longitude Seed Value :	146.77395435	
Expected Decoded Latitude :	-36.850285934	
Expected Decoded Longitude :	146.77314	
Expected Encoded Latitude	**1EBDA** (Hex)	125914 (Decimal)
Expected Encoded Longitude :	**1823A** (Hex)	98874 (Decimal)

At the given receiver latitude and longitude position, the aircraft was expected to provide an Encoded Latitude of **1EBDA** (Hex) and an Encoded Longitude of **1823A** (Hex). Instead, the Encoded Longitude received from the transponder was **052F8** (Hex). This resulted in a Decoded Receiver Longitude that was approximately 3.19072 degrees in error relative to the Expected Decoded Longitude. Effectively, an error in longitude of approximately **153.475** nautical miles appears to have been induced into the system.

The reporting authority indicated that the transponder may not be executing a proper latitude zone look-up in accordance to the applicable specification. If such were the case, then the error would be as demonstrated in the table above.

<u>Part b:</u> **Initial Analysis and Response**

In attempting to verify the condition discussed above, Rockwell Collins implemented updated files of the same Mathcad Simulations that have been used to verify the performance of the Compact Position Reporting Algorithm ever since it was introduced to Avionics vendors many years ago.

Using an updated set of the simulations which implement the exact applicable equations from RTCA DO-260 and DO-260A, it was demonstrated that the Rockwell Collins Air Transport TPR-901 and the Business and Regional TDR-94/94D (-108) transponders should be delivering the proper latitude encoding of **1EBDA** (Hex) and a proper longitude encoding of **1823A** (Hex).

The simulations were then modified to force the **NL** lookup to **48** given the input latitude and longitude positions in the table above. In this case, both the TPR-901 and TDR-94/94D (-108) transponders provided an encoded latitude of **1EBDA** (Hex) and an encoded longitude of **052F8** (Hex). As such, the simulations proved that a transponder could provide the erroneous **0528F8** (Hex) encoding as reported by the reporting Agency if the transponder set the **NL** to **48** (as opposed to **47**) at the given input latitude and longitude position.

Testing was then repeated on both the TPR-901 and TDR-94/94D (-108) transponders in accordance with RTCA DO-260A, Section 2.4.3.2.4.7.1.1, **Table 2-130**, which is intended to demonstrate that **NL** zones are being established correctly. Results indicated that both transponders were transitioning across the latitude zone boundaries appropriately as NO Failures were observed during the testing.

At this point, the results of the simulations and testing were reported to the reporting agency. However, concerns remained that the transponder appeared to be establishing the message encodings based on a latitude zone lookup of **NL=48**, as opposed to **NL=47**.

<u>Part c:</u> **Detailed Testing of Latitude Zone Lookup**

As concerns remained that latitude zone boundaries were not being processed properly, both the TPR-901 and TDR-94/94D (-108) transponders were subjected to detailed testing to establish exactly where the transponders established the zone boundary between **NL=47** and **NL=48**. **Figure 1** illustrates the location of the exact boundary crossing between **NL=47** and **NL=48**, as well as the limits that are tested in RTCA DO-260A, Section 2.4.3.2.4.7.1.1, **Table 2-130**. Likewise, **Figure 1** illustrates exactly where the TPR-901 and TDR-94/94D (-108) transponders are transitioning from **NL=47** to **NL=48**.

Prior to explaining **Figure 1** further, it should be noted that the convention is:

> When moving from the **pole to equator**, one does not transition to the next **higher NL** value until **the boundary is reached**. Alternately, when moving from the equator to the pole, one does not transition to the next **lower NL** value until the boundary has been crossed.
>
> As shown in the figure, **NL=47** for all values less than or equivalent to 38.41241892412246 degrees **UNTIL** approaching the next boundary at 36.85025107593526 degrees. Then, **NL=48** for all values less than or equivalent to 36.85025107593526 degrees **UNTIL** approaching the next boundary at 35.22899597796385 degrees.

Referring to **Figure 1**, it is clear that the input latitude position of -36.850285934 (Negative for South) is 36.850285934 degrees from the equator which is **further from** the equator than the boundary located at 36.85025107593526 degrees. Therefore, the proper zone to be using for encoding or decoding at this latitude is **NL = 47**.

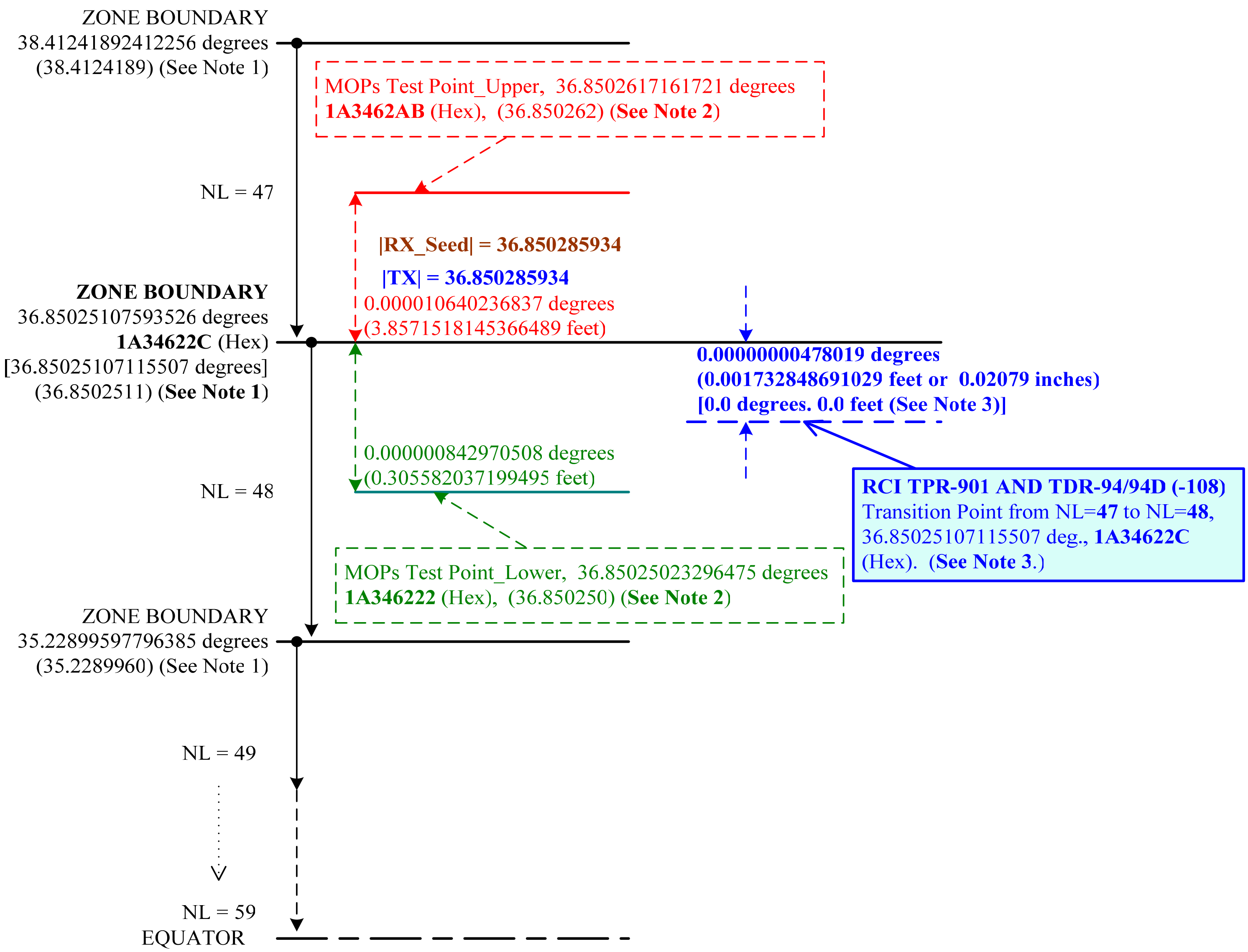
Again, referring to **Figure 1**, note that the best case Angular Weighted Binary (AWB) encoding of the boundary at 36.85025107593526 degrees is **1A34622C** (Hex) which represents a value of 36.85025107115507 degrees. When moving from the equator towards the poles, this value actually occurs 0.00000000478019 degrees or approximately 0.02079 inches **before** the boundary. Next,

Figure 1 illustrates that the TPR-901 and TDR-94/94D (-108) transponders switch from **NL=47** to **NL=48** exactly at **1A34622C** (Hex). As this is the best that the designated boundary can be encoded in Angular Weighted Binary and retain an accuracy of +/- ½ LSB, the TPR-901 and TDR-94/94D (-108) transponders appear to be switching **NL** boundaries **EXACTLY** as required and there is **NO** error.

When referred to the input boundary value of 36.85025107593526 degrees, the error is extremely minute with the transponders switching to **NL=47** 0.00000000478019 degrees **prior to** reaching the boundary. As the switch point is **prior to** the boundary, the case in point for selecting **NL=47** for the given input latitude is **aided** as opposed to being **degraded**, and the transponders will select a value of **NL=47** as demonstrated in simulation and in testing.

Figure 1 also indicates that the Transmitter Latitude is **further from** the equator than the AWB boundary established at 36.85025107115507 degrees and the boundary computed at 36.85025107593526 degrees. Therefore, the Transmitter should use **NL=47** for encoding purposes.

Likewise, **Figure 1** also indicates that the Receiver Seed Latitude is **further from** the equator than the AWB boundary established at 36.85025107115507 degrees and the boundary computed at 36.85025107593526 degrees. Therefore, the Receiver should use **NL=47** for decoding purposes.



Note 1: **Computed Zone Boundary** is computed exactly as per RTCA DO-260A, Appendix A, Section A.1.7.2.2.d for NL(lat). Then the number is converted to the best possible 32 bit Angular Weighted Binary (AWB) number retaining +/- 1/2 LSB accuracy where the LSB is equal to 180*2^-31. The number in [brackets] represents the true boundary value used for the look-up as given by the translated Hexidecimal or **AWB Zone Boundary** value. The less precise number in parenthesis is from ICAO DOC. 9688-AN/952, Table A-1, and is in accordance with the look-up table generation equation provided in RTCA DO-260A, Appendix A, Section A.1.7.2.2.d. Note 5.

Note 2: Limit line is established in accordance with the best case Angular Weighted Binary Hex value provided in RTCA DO-260A, Section 2.4.3.2.4.7.1.1, Table 2-130, Verification of Latitude Transition. The difference in degrees and feet is then computed relative to the Computed Zone Boundary discussed in Note 1.

Note 3: When computing the difference between the **Transition Point** value and the **Computed Zone Boundary** (See Note 1), the slight error indicates that the transition point is being reached slightly before the **Computed Zone Boundary** when moving from the equator towards the pole.
Yet it should be noted that the error indicated in inches is less that that of 1/2 AWB LSB which is 0.000000041909516 degrees or 0.1823095 inches.
However, when computing the difference between the **Transition Point** value and the **AWB Zone Boundary** Value, there is **NO ERROR** and the transition zone is being selected at the **EXACT** point that it should be in the Angular Weighted Binary Number system which is what the transponder must use since data is provided in that format by the Navigation data source.

Figure 1: Boundary Crossing at NL=47 to NL=48

Part d: Summary of Findings in regards to ADS-B Transmitter

Based on the discussions above in regards to the ADS-B Transmitter, the following preliminary conclusions have been demonstrated:

1. The Rockwell Collins, Inc., TPR-901 and TDR-94/94D (-108) Mode-S Transponders are properly selecting the appropriate **NL** value based on the requirements of RTCA DO-260 and DO-260A.
2. The CPR algorithm appears to be working as expected as demonstrated in the simulations and unit testing discussed up to this point.
3. The establishment of the appropriate **NL** value based on the input Latitude cannot be based on computations that have more resolution than that provided by the AWB encoding which has a maximum resolution of 0.5*180*2^-31 or 0.000000041909516 degrees (e.g., 0.1823095 inches) which represents 0.5 LSB of the possible encoding.
4. Extreme care must be exercised by the receiving equipment to recognize and discard the anomalies prior to providing the information to any applications. As it is understood that the original data has come from raw radar files, these anomalies are going to happen very close to boundary regions simply by nature of the CPR algorithm and the method that it uses for even and odd encodings to provide the capability to globally and unambiguously resolve a reporting aircraft's or vehicle's position.

With these conclusions made, the question still stands as to why the transponder appeared to encode a longitude value in error at the presumed altitude. Consider that the transponder is encoding properly as demonstrated in the simulations and testing previously discussed. Next, consider that the difference between the presumed latitude of the target and the boundary at which the transponder changes **NL=47** to **NL=48** is approximately 0.000034860 degrees. This translates into an approximate distance of 12.6370 feet, which is considerably less than the length or wing span of an aircraft. That considered, it is most possible that the navigation source sustained a momentary error.

Part e: Receiver Considerations

Keeping the table provided in **Part a** in mind, consider the case where the receiver has received the suspect encoded longitude value of **052F8** (Hex) under the circumstances defined in the table given in **Part a**. Review of the following table and subsequent discussion reveals an interesting situation.

Receiver Decoder Results for Given Encoding (Latitude = -36.85028593987227 and Longitude = 146.77314)								
	Transmitted Encoding		Receiver Decoder Processing					
	Encoded Latitude (Y17za)	Encoded Longitude (X17za)	Y19za (4 * Y17za)	Computed Latitude (Rlat)	Computed Latitude Zone	NL (lookup)	Computed Latitude Output Rlat_out	Computed Lontitude Output Rlon_out
1	**1EBDA** (Hex)	052F8 (Hex)	07AF68 (Hex)	-36.85028593419	47.999978082241	47	-36.8502393819518	149.9638565726902
2	**1EBDB** (Hex)	052F8 (Hex)	07AF6C (Hex)	-36.850239381952	48.0000073528	48	-36.8502393819518	146.7731362200798

Row 1 represents the encoding received by the reported ground station having the suspect **052F8** (Hex) encoded longitude. Note that the encoded latitude is correct at **1EBDA** (Hex). Now, in accordance with the equations provided in RTCA DO-260A, the CPR algorithm multiples the received encoded latitude by 4 in order to get back to a 19 bit encoding. This results in the Y19za value of **07AF68** (Hex) which is then used to establish the Rlat value of -36.85028593419. This value is then used to establish the proper **NL** zone for the receiver which turns out to be **NL=47**. Moving through the final stages of the CPR Local Unambiguous Odd Decode results in the longitude value of 149.9638…, as shown. Why the error? Well, as discussed at length above, the **NL** was forced to **48** to get the encoding of **052F8** (Hex) for encoded longitude.

Now, consider row 2 where the encoded latitude received has been increased by one LSB to **1EBDB** (Hex). Interesting that moving through the CPR decode process results in an **NL=48** and a final longitude value of **146.77316**… degrees, which is extremely close to the desired value of **146.77314** degrees.

What this all says is that the receiver determination of **NL** values is just as sensitive as the transmitter encoder is. In fact, the transition from **NL=47** to **NL=48** by the receiver happened in less than 1 LSB of the encoded latitude value. This means the change was done within inches, and ultimately resulted in a massive error from the desired 146….degrees to 149….degrees.

<u>Part f:</u> Case 1 Summary / Ending Commentary

The CPR algorithm was designed to meet a desired accuracy need and be globally unambiguous while compressing both latitude and longitude data into 17 bits in the extended squitter message. In order to meet the desired accuracy the minimum number of AWB bits needed is approximately 22 for latitude and 23 for longitude, +/- 1 bit for each. So, the algorithm was developed to compress into 17 bits and save the number of bits needed in the limited space extended squitter message. This makes the algorithm very sensitive to resolution and accuracy when dealing with the boundaries and determining the appropriate **NL** zone. This is particularly true when a ground station receiver may have unlimited precision in its computational capability when the transmitter is limited to a best case accuracy of the input data source which currently has a maximum resolution of l LSB in 32 bit Angular Weighted Binary and an accuracy of +/- ½ LSB. Also, the algorithm uses even and odd encodings specifically to assist in resolving the boundary issues. If one decode of the position is reasonable and the other is not, the decode that is not reasonable must be discarded.

Selecting the wrong **NL** zone either on the transmitter side or the receiver side, will result in the exact type of errors demonstrated in this case. The important part is that the receiver must realize that there has been a boundary difference selection between the transmitter and the receiver which has caused a large unexpected error. Therefore, it is incumbent on the receiver to ensure that such error data is rejected or heavily filtered before providing data to the application. Effectively, it is the transmitter's responsibility to ensure that encoding is performed in accordance with the algorithm while it is the receiver's responsibility to ensure that ambiguities are properly resolved and that the decoded position is reasonable.

Consider that the discussion to this point has been based on the transmitter having access to 32 bits of AWB data from the navigation source. The only sources providing that much data resolution today are GPS/GNSS based. FMS systems only provide 24 bits of AWB resolution with the MSB=180 degrees. Therefore, the demonstrated error situation will get worse at the boundary zones when the transmitter is using FMS data and the Ground Station is using unlimited precision.

In closing this commentary, the **CPR** algorithm **WORKS** provided the implementer recognizes the limitations and why they exist. In short, when data does not appear to be reasonable, it must be rejected. The entire solution cannot be based on one data point solution as there are too many variables in the system. Just to name a few: navigation source performance, aircraft installation performance, transponder performance, airspace ether permutations (minimal effect but it could), receiving system installation performance, receiver performance, and last but not least, receiver data interpretation performance.

Case 2: **Airborne Position Message, Odd Encoding ___Aircraft Identification = VOZ858**

Part a: **Initial Data Received**

The reporting agency reported that the following Airborne Position Message was received by the Ground Station and considered to be in error in regards to Encoded Longitude.

Original Odd Encoded Airborne Position Message Received:		
Message Content:	8D7C6D22686B95AF395A53	
Aircraft Address:	7C6D22	**VOZ324**
Encoded Latitude :	**0D79C** (Hex)	55196 (Decimal)
Encoded Longitude :	**15A53** (Hex)	88659 (Decimal)
Latitude (decoded at Receiver) :	-27.93897725768008	
Longitude (decoded at Receiver) :	150.0674849290114	
Receiver Latitude Seed Value :	-27.93709	
Receiver Longitude Seed Value :	153.01105	
Expected Decoded Latitude :	-27.93897725768008	
Expected Decoded Longitude :	153.0099722055288	
Expected Encoded Latitude	**0D79C** (Hex)	55196 (Decimal)
Expected Encoded Longitude :	**033F0**(Hex)	13296 (Decimal)

At the given receiver latitude and longitude position, the aircraft was expected to provide an Encoded Latitude of **0D79C** (Hex) and an Encoded Longitude of **033F0** (Hex). Instead, the Encoded Longitude received from the transponder was **15A53** (Hex). This resulted in a Decoded Receiver Longitude that was approximately 2.9425 degrees in error relative to the Expected Decoded Longitude. Effectively, an error in longitude of approximately **175.552** nautical miles appears to have been induced into the system.

The reporting authority indicated that the transponder may not be executing a proper latitude zone look-up in accordance to the applicable specification. If such were the case, then the error would be as demonstrated in the table above.

Part b: **Initial Analysis and Response**

As in Case 1, simulations demonstrated that the Rockwell Collins Air Transport TPR-901 and the Business and Regional TDR-94/94D (-108) transponders should be delivering the proper latitude encoding of **0D79C** (Hex) and a proper longitude encoding of **033F0** (Hex) based on an **NL** lookup value **NL=53** as opposed to **52** indicated by the reporting agency.

The simulations were then modified to **force** the **NL** lookup to **52** given the input latitude and longitude positions in the table above. In this case, both the TPR-901 and TDR-94/94D (-108) transponders provided an encoded latitude of **0D79C** (Hex) and an encoded longitude of **15A53** (Hex). As such, the simulations proved that a transponder could provide the erroneous **15A53** (Hex) encoding as reported by the reporting Agency if the transponder set the **NL** to **52** (as opposed to **53**) at the given input latitude and longitude position.

Part c: **Detailed Testing of Latitude Zone Lookup**

As the reporting agency indicated that the proper latitude encoding should be **033F0** (Hex) and the **NL** should be **52**, both the TPR-901 and TDR-94/94D (-108) transponders were subjected to detailed testing to establish exactly where the transponders established the zone boundary between **NL=52** and **NL=53**. **Figure 2** illustrates the location of the exact boundary crossing between **NL=52** and **NL=53**, as well as the limits that are tested in RTCA DO-260A, Section 2.4.3.2.4.7.1.1, **Table 2-130**. Likewise, **Figure 2** illustrates exactly where the TPR-901 and TDR-94/94D (-108) transponders are transitioning from **NL=52** to **NL=53**.

Prior to explaining **Figure 2** further, it should be noted that the convention is:

> When moving from the **pole to equator**, one does not transition to the next **higher NL** value until **the boundary is reached**. Alternately, when moving from the equator to the pole, one does not transition to the next **lower NL** value until the boundary has been crossed.
>
> As shown in the figure, **NL=52** for all values less than or equivalent to 29.91135685731809 degrees **UNTIL** approaching the next boundary at 27.93898710121905 degrees. Then, **NL=53** for all values less than or equivalent to 27.93898710121905 degrees **UNTIL** approaching the next boundary at 25.82924707058776 degrees.

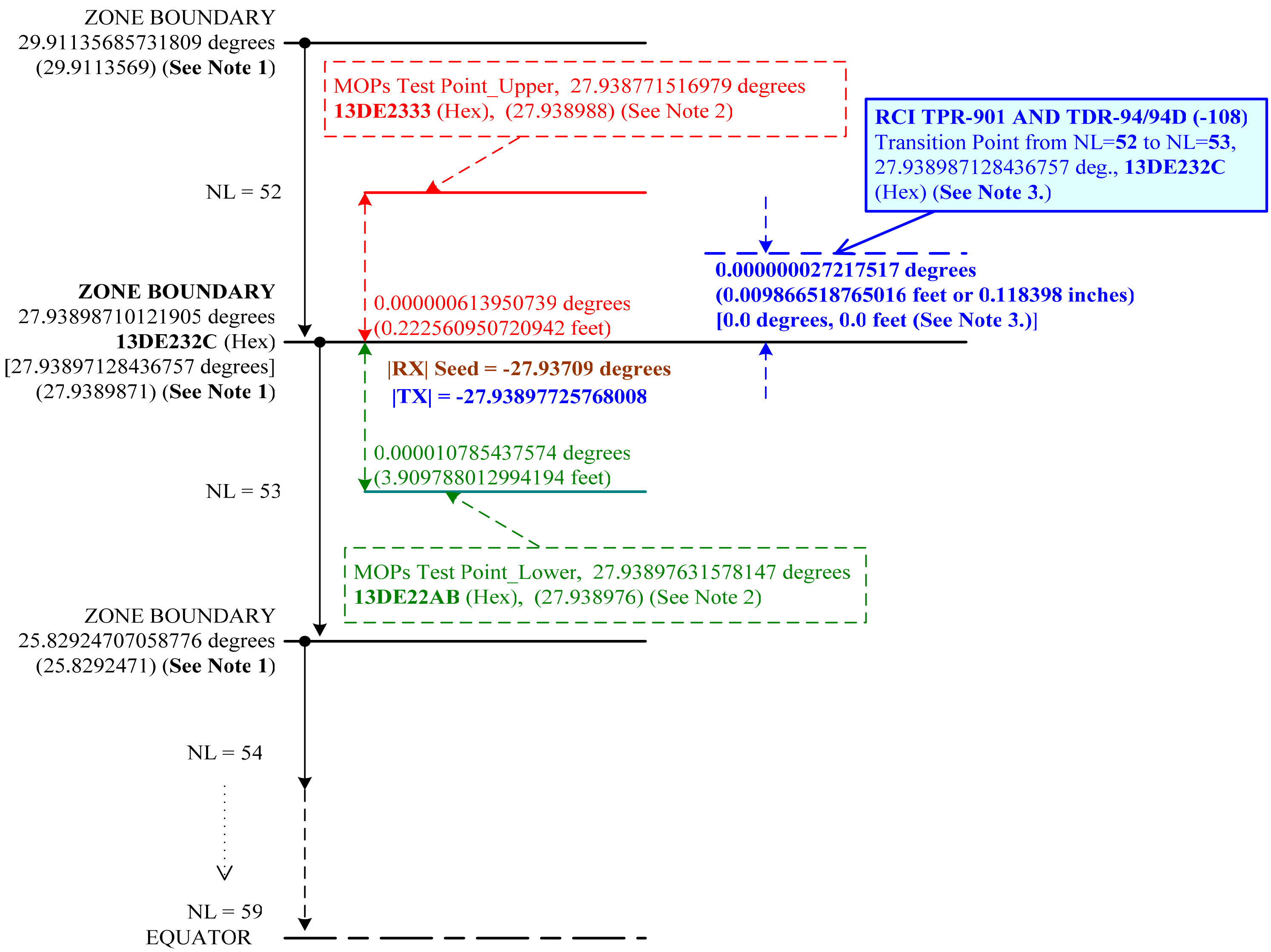
Referring to **Figure 2**, it is clear that the input latitude position of -27.92897725768008 degrees (Negative for South) is 27.92897725768008 degrees from the equator which is **closer to** the equator than the boundary located at 27.93898710121905 degrees. Therefore, the proper zone to be using for encoding or decoding at this latitude is **NL = 53** as illustrated in **Figure 2.**

Again, referring to **Figure 2**, note that the best case Angular Weighted Binary (AWB) encoding of the boundary at 27.93898710121905 degrees is **13DE232C** (Hex) which represents a value of 29.93897128436757 degrees. When moving from the equator towards the poles, this value actually occurs 0.000000027217517 degrees or approximately 0.1184 inches **after** the boundary. Next, **Figure 2** illustrates that the TPR-901 and TDR-94/94D (-108) transponders switch from **NL=52** to **NL=53** exactly at **1DE232C** (Hex). As this is the best that the designated boundary can be encoded in Angular Weighted Binary and retain an accuracy of +/- ½ LSB, the TPR-901 and TDR-94/94D (-108) transponders appear to be switching **NL** boundaries **EXACTLY** as required and there is **NO** error.

When referred to the input boundary value of 27.93898710121905 degrees, the error is extremely minute with the transponders switching to **NL=53** 0.000000027217517 degrees **after** the boundary. As the switch point is **after** the boundary, the case in point for selecting **NL=52** for the given input latitude is **degraded** as opposed to being **aided**, and the transponders will select a value of **NL=53** as demonstrated in simulation and in testing.

Figure 2 indicates that the Transmitter Latitude is **closer to** the equator than the AWB boundary established at 29.93897128436757 degrees and the boundary computed at 27.92897725768008 degrees. Therefore, the Transmitter should use **NL=53** for encoding purposes**.**

Likewise, **Figure 2** indicates that the Receiver Seed Latitude is **closer to** the equator than the AWB boundary established at 29.93897128436757 degrees and the boundary computed at 27.92897725768008 degrees. Therefore, the Receiver should use **NL=53** for decoding purposes**.**



Note 1: **Computed Zone Boundary** is computed exactly as per RTCA DO-260A, Appendix A, Section A.1.7.2.2.d for NL(lat). Then the number is converted to the best possible 32 bit Angular Weighted Binary (AWB) number retaining +/- 1/2 LSB accuracy where the LSB is equal to 180*2^-31. The number in [brackets] represents the true boundary value used for the look-up as given by the translated Hexidecimal or **AWB Zone Boundary** value. The less precise number in parenthesis is from ICAO DOC. 9688-AN/952, Table A-1, and is in accordance with the look-up table generation equation provided in RTCA DO-260A, Appendix A, Section A.1.7.2.2.d. Note 5.

Note 2: Limit line is established in accordance with the best case Angular Weighted Binary Hex value provided in RTCA DO-260A, Section 2.4.3.2.4.7.1.1, Table 2-130, Verification of Latitude Transition. The difference in degrees and feet is then computed relative to the Computed Zone Boundary discussed in Note 1.

Note 3: When computing the difference between the **Transition Point** value and the **Computed Zone Boundary** (See Note 1), the slight error indicates that the transition point is being reached slightly before the **Computed Zone Boundary**.
Yet it should be noted that the error of indicated in inches is less that that of 1/2 AWB LSB which is 0.000000041909516 degrees or 0.015192 inches.
However, when computing the difference between the **Transition Point** value and the **AWB Zone Boundary** Value, there is **NO ERROR** and the transition zone is being selected at the EXACT point that it should be in the Angular Weighted Binary Number system which is what the transponder must use since data is provided in that format by the Navigation data source.

Figure 2: Boundary Crossing at NL=52 to NL=53

Part d: **Summary of Findings in regards to ADS-B Transmitter**

As was the case in **Case 1** earlier, the TPR-901 and TDR-94/94D (-108) transponder appear to be functioning as required and all parts of **Case 1, Part d**, through subparagraph 4, apply equivalently to this **Case 2** discussion.

Part e: **Receiver Considerations**

Information from the reporting agency has previously indicated that the encoded longitude should be **033F0** (Hex), **13296** (Decimal), and that the **NL** zone lookup should be **NL=52** at the given input latitude and longitude. However, the previous discussion in **Case 2, Part c**, as well as **Figure 2**, clearly indicates that the **NL** zone lookup should be **NL=53** at the given input latitude and longitude. For verification purposes simulations were performed at the given input latitude and longitude with the receiver decoder **NL** zone lookup forced to **NL=53**. The resultant decoded received longitude was indeed the desired value of 153.0099722055288 degrees. Thereby proving that the Receiver should be using **NL=53** for decoding purposes at the latitude and longitude encodings and the given receiver seed value latitude and longitude.

Special Concerns during Globally Unambiguous Decoding:

Data received from the reporting agency indicates that the encoded latitude and encoded longitude values given for this **Case 2** were actually used in a globally unambiguous decode which resulted in a longitude of **-50**, e.g. **50 degrees WEST**. This value was then used as a seed position for locally unambiguous decoding which resulted in a track being established along the **50 degree West** value until such time as another globally unambiguous decode was established and moved the longitude location back to approximately **153 degrees East**. Note that this is an error of approximately **203** degrees going **west** (or **157** degrees going **east**).

As the receiving seed position indicates a position somewhere in **Northeastern Australia**, it should be obvious that the **-50 West** is **wrong** as that position is located somewhere in far **South Brazil**.

As such, this example underscores precisely why the receiver has to be more robust in resolving ambiguities and dealing with boundary crossing of the CPR algorithm.

Part f: **Case 1 Summary / Ending Commentary**

All of the summary / commentary material previously provided in **Case 1, Part f**, applies equivalently to this **Case 2** discussion.

CONCLUSION:

This analysis has demonstrated that the transponders tested did properly determine the **NL** zones and properly established the required position encodings in regards to the two cases presented. Likewise testing demonstrated that the transponders tested did properly pass the RTCA DO-260 / 260A tests required to validate Zone selections near boundary transitions. As such, this analysis concludes that the transponders tested are functioning properly in regards to CPR position encoding.

This analysis has also demonstrated the errors that can occur when the ADS-B transmitter and ADS-B receiver do not select the appropriate (e.g., same) **NL** zone. But most of all, this analysis has demonstrated that the ADS-B Receiver must always verify that the decoded position data is reasonable. If the decoded position data is not reasonable, then as a bare minimum, the Receiver should attempt to execute the decode again using the next closest possible NL zone for the decoding. If that does not work, then the erroneous data point must be rejected.

As a commentary, when originally introduced for use with “Extended Squitter”, the CPR algorithm implemented a Boundary Adjustment computation near the end of the encoding that was intended to improve boundary ambiguity resolution by the receiver. At that time, the accuracy of the CPR algorithm was stated to be approximately **15** meters for Airborne Encoding and **5** meters for Surface Encoding. Analysis and Simulations demonstrated that the **5** meter Surface accuracy could not be maintained world wide without adding a small sub-zone search about the decoded latitude and longitude zones for the best fit. Once this was done, the accuracy was well within the desired **5** meter requirement. When RTCA DO-260 was developed many years later, it was determined that the accuracy requirements could be relaxed and the Boundary Adjustment to the Encoding was removed. Although the Boundary Adjustment may improve the situations discussed in this document, it will not remove the need of the Receiver to always verify that the decoded data is reasonable and compensate accordingly.

Respectfully presented by:

Robert H. "Bob" Saffell
Principal Electrical Engineer
(Mode-S / TCAS / ADS-B)
Rockwell Collins, Inc.

Phone: 321-768-7062 Work
321-652-4641 Cellular
Email: rhsaffel@rockwellcollins.com Work
rhssx@msn.com Home

RHS/rhs