**RTCA Special Committee 186, Working Group 3**
**EUROCAE WG-51, SG-1**

**ADS-B 1090ES MOPS Maintenance**

**Meeting #30**

**RTCA Headquarters**
**Washington, DC**
**18 – 21 August 2009**

# GPS Time Tag and UTC Time Synchronization Issues
**and**
# Proposed Appendix "S"
# Revision 1

**Prepared by:** **Don Walker, Honeywell Inc.**
**Robert H. "Bob" Saffell, Rockwell Collins, Inc.**
**Presented by:** **Bob Saffell**

**Summary**

During Meeting #29 at RTCA, review of modifications made to Time Tag requirements by Bob Saffell resulted in some considerable discussion and disagreement on what the Time Tag meant and what its relation to actual UTC Time at the leading edge of the Time Mark were. A subsequent telecon on Thursday, July 30, 2009, generated an Action Item for the situation to be addressed and resolved by Don Walker, Honeywell, and Bob Saffell, Rockwell Collins. This document addresses that Action Item in the following manner.

First, minor changes are recommended to be made in section 2.2.5.1.6.2 to implement the need for a process or method to synchronize UTC TIME with the TIME Mark. Then, appropriate sections are added to require input of UTC and UTC Fine data from the GPS/GNSS receiver. Last, Appendix "S" (attached as the end of this document) was developed to design or specify a method such that the Time Mark and UTC time can be synchronized at ANY Time within ANY Epoch on ANY sub-epoch.

Reviewers of this document should ensure that they review Appendix "S" very closely and can relate to the concept.

**Introduction:**

Due to limited time, this introduction is going to be very short. The Working Paper Title Page has already introduced the subject or task at hand, so one may as well get right to it.

The following sections written in "Blue" Font "14" provide instruction or comment on the changes actually being made into the DO-260B MOPs.

Mops changes are then made and highlighted as described before the change is made.
After all the changes are made, Appendix "S" is provided at the end of this document.

**The current DO-260B MOPs provide baseline Time Mark requirements in section 2.2.3.2.3.5 for the Airborne Position Message and in section 2.2.3.2.4.5 for the Surface Position Message. Both of the MOPs sections are repeated below. Note that Section 2.2.5.1.6 is referenced in Note 1 of both section 2.2.3.2.3.5 and section 2.2.3.2.4.5 as indicated by "yellow" highlighting. Therefore, it should not be necessary to make changes in these sections in order to implement the Synchronization process.**

**2.2.3.2.3.5 "TIME" (T) Subfield in ADS-B Airborne Position Messages**

The "TIME" (T) subfield is a 1-bit ("ME" bit 21, Message bit 53) field that **shall** indicate whether or not the epoch of validity for the horizontal position data in an Airborne Position Message is an exact 0.2 second UTC epoch. If the time of applicability of the position data is synchronized to an exact 0.2 second UTC epoch, the "TIME" (T) subfield **shall** be set to "1;" otherwise, the "TIME" (T) subfield **shall** be set to ZERO (0).

***Notes:***

1. *An ADS-B Transmitting Subsystem that sets the "TIME" (T) subfield to ONE (1) must accept a GNSS TIME MARK input from the navigation data source in order to be able to update the position data from the navigation data source to an exact 0.2 second UTC epoch (See §2.2.5.1.6).*
2. *An arithmetic description of the intended synchronization implementation is provided in the **"Commentary"** paragraphs provided in §2.2.3.2.3.7.2 for precision Latitude Position Extrapolation and in §2.2.3.2.3.8.2 for precision Longitude Position Extrapolation.*

**2.2.3.2.4.5 "TIME" (T) Subfield in ADS-B Surface Position Messages**

The "TIME" (T) subfield is a 1-bit ("ME" bit 21, Message bit 53) field that **shall** be used to indicate whether or not the epoch of validity for the horizontal position data in a Surface Position Message is an exact 0.2 second UTC epoch. If the time of applicability of the position data is synchronized to an exact 0.2 second UTC epoch, the "TIME" (T) subfield **shall** be set to ONE (1); otherwise, the "TIME" (T) subfield **shall** be set to ZERO (0).

The "TIME" (T) subfield may be set to ONE (1) only for TYPE Codes 5 and 6 (see §2.2.3.2.3.1 and Table 2-16) when used to indicated synchronization as discussed in the previous paragraph.

***Notes:***

1. *An ADS-B Transmitting Subsystem that sets the "TIME" (T) subfield to ONE (1) must accept a GNSS TIME MARK input from the navigation data source in order to be able to update the position data from the navigation data source to an exact 0.2 second UTC epoch (see §2.2.5.1.6).*

2. *An arithmetic description of the intended synchronization implementation is provided in the "Commentary" paragraphs provided in §2.2.3.2.4.7.2 for precision Latitude Position Extrapolation and in §2.2.3.2.4.8.2 for precision Longitude Position Extrapolation.*

## In order to implement synchronization requirements, section 2.2.5.1.6.2 is modified as shown in "blue" font and highlighted in "blue" in the following section 2.2.5.1.6.2.

**2.2.5.1.6.2 Case, where TIME ("T") = 1**

a. If the ADS-B Transmitting Subsystem is capable of setting the TIME ("T") subfield (see §0) to ONE in the Airborne Position Message (see §2.2.3.2.3) or in the Surface Position Message (see §2.2.3.2.4), then the ADS-B Transmitting Subsystem **shall** accept and process a GPS/GNSS Time Mark pulse or an equivalent time synchronization indication from the source of navigation data.

**Change Subparagraph a. to read as follows:**

a. Establishing the Time Mark Reference (TMR)

The leading edge of the Time Mark indicates the beginning of the USER TIME EPOCH (Figure 8-3, ARINC-743A). The UTC time indicates the UTC Time at the leading edge of the Time Mark and may or may not be at the beginning or end of an UTC EPOCH.

The UTC of Time Mark shall be used to establish synchronization of the position data to the 200 millisecond UTC sub-epochs. Refer to Appendix "S" which provides an acceptable method of establishing the required synchronization to the Time Mark Reference (TMR) based on UTC of Time Mark.

***Notes:***

1. *When synchronization is required, in addition to accepting the Time Tag information, the ADS-B Transmitting Subsystem needs to accept appropriate UTC time information in order to establish the EVEN or ODD Epoch and the appropriate EVEN or ODD 0.2 second sub-epochs.*

   *Typically, this information can be determined from the ARINC-743A label "140" and "150" words. (See §2.2.5.1.53 and §2.2.5.1.54.*

2. *If UTC Time indicates that the beginning of the EPOCH is within +/- 5 milliseconds of the leading edge of the Time Mark, then the GPS/GNSS Time Mark can be used to establish the Time Mark Reference (TMR) that represents the time of applicability as discussed in §2.2.5.1.6.2.b.*

b. The leading edge of the GPS/GNSS Time Mark pulse, or equivalent (e.g., **TMR**, see §2.2.5.1.6.2.a.(2)), **shall** indicate the exact moment (epoch of the UTC time scale) that represents the time of applicability of Position, Velocity, and Time (PVT) information that is received from the navigation source.

***Note:*** *A possible implementation of the GPS/GNSS Time Mark pulse is illustrated in Figure 2-15. The PVT data is expected to be provided by the navigation data source no later than 200 milliseconds after the leading edge of the GPS/GNSS Time Mark pulse, or equivalent.*

c. When the ADS-B Transmitting Subsystem sets the TIME ("T") subfield to ONE in the Airborne Position Message (see §2.2.3.2.3), then the ADS-B Transmitting Subsystem **shall**

use the GPS/GNSS Time Mark pulse or TMR (see §2.2.5.1.6.2.a.(2)) to accomplish the following:

(1). Synchronize the loading of the Airborne Position Message such that the register is properly loaded 100 milliseconds ±50 milliseconds prior to the time of applicability of the data being loaded into the register.

(2). Establish the "Time" subfield in ADS-B Airborne Position Messages (see §2.2.3.2.3) as specified in §2.2.3.2.3.5,

(3). Establish the "CPR Format" subfield in ADS-B Airborne Position Messages (see §2.2.3.2.3) as specified in §2.2.3.2.3.6,

(4). Extrapolate Airborne Latitude Position in ADS-B Airborne Position Messages (see §2.2.3.2.3) as specified in §2.2.3.2.3.7.2.

(5). Extrapolate Airborne Longitude Position in ADS-B Airborne Position Messages (see §2.2.3.2.3) as specified in §2.2.3.2.3.8.2,

d. When the ADS-B Transmitting Subsystem sets the TIME ("T") subfield to ONE in the Surface Position Message (see §2.2.3.2.4), then the ADS-B Transmitting Subsystem **shall** use the GPS/GNSS Time Mark pulse or TMR (see §2.2.5.1.6.2.a.(2)) to accomplish the following:

(1). Synchronize the loading of the Surface Position Message such that the register is properly loaded 100 milliseconds ±50 milliseconds prior to the time of applicability of the data being loaded into the register.

(2). Establish the "Time" subfield in ADS-B Surface Position Messages (see §2.2.3.2.4) as specified in §2.2.3.2.4.5,

(3). Establish the "CPR Format" subfield in ADS-B Surface Position Messages (see §2.2.3.2.4) as specified in §2.2.3.2.4.6,

(4). Extrapolate Surface Latitude Position in ADS-B Surface Position Messages (see §2.2.3.2.4) as specified in §2.2.3.2.4.7.2, and

(5). Extrapolate Surface Longitude Position in ADS-B Surface Position Messages (see §2.2.3.2.4) as specified in §2.2.3.2.4.8.2.

**Add the following sections to the MOPs:**

**2.2.5.1.53 UTC HRs:Min:Sec Data**

The ADS-B Transmitting Subsystem shall accept GNSS UTC Hrs:Min:Sec information via an appropriate data input interface and use such data to establish an appropriate Time Mark Reference (TMR) as specified in §2.2.5.1.6.2.a.(2).

***Note:*** *Typically, the required UTC Hrs:Min:Sec data is provided by the GPS/GNSS Receiver via ARINC Label "150" words (See Appendix S, Table S-2).*

**2.2.5.1.53 UTC Fine Seconds Data**

The ADS-B Transmitting Subsystem shall accept GNSS UTC Fine Seconds information via an appropriate data input interface and use such data to establish an appropriate Time Mark Reference (TMR) as specified in §2.2.5.1.6.2.a.(2).

***Note:*** *Typically, the required UTC Fine Seconds(e.g., fractional seconds) data is provided by the GPS/GNSS Receiver via ARINC Label "140" words (See Appendix S, Table S-3).*

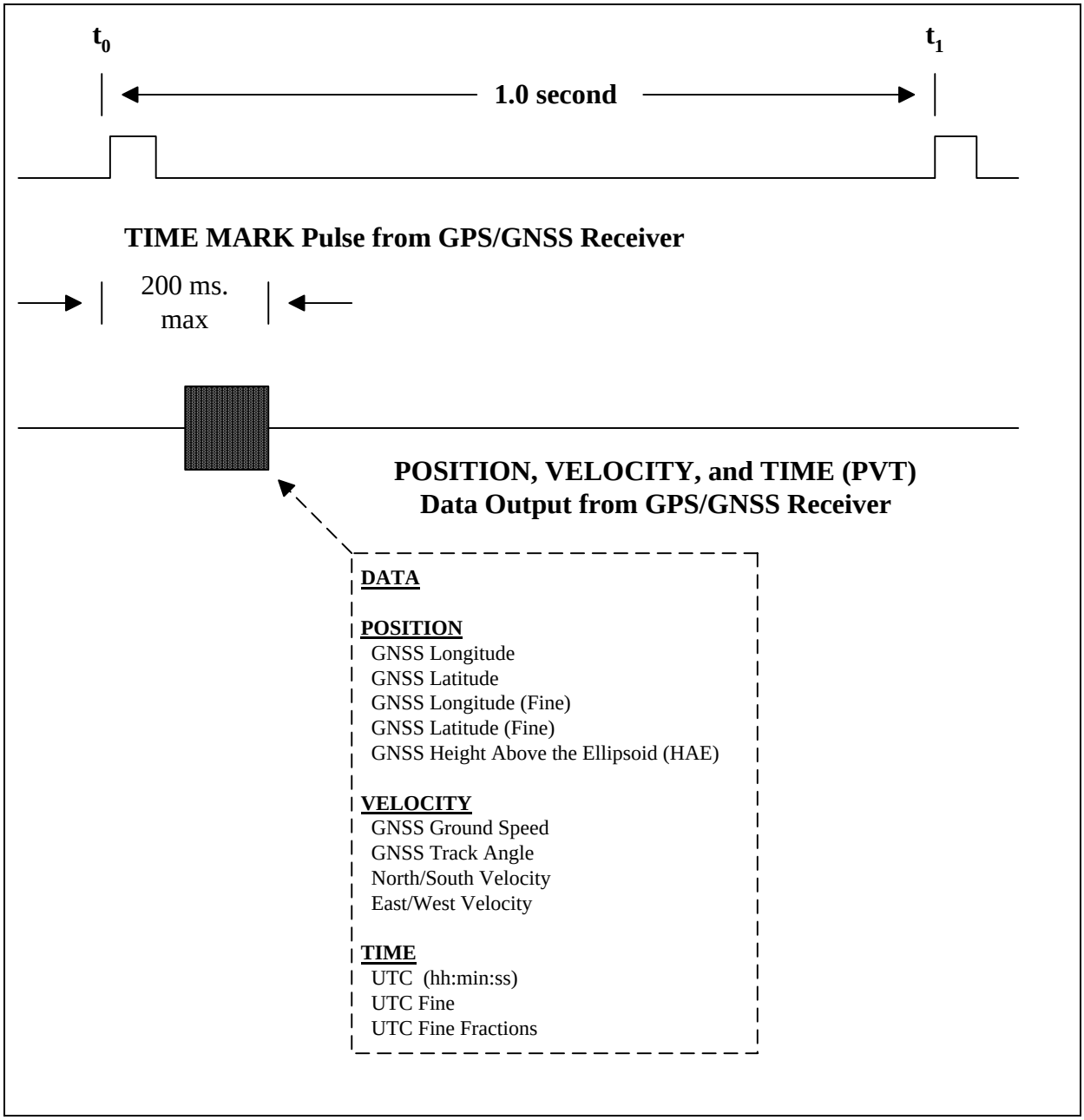



**Figure 2-15:** **GPS/GNSS Time Mark Pulse**

***Note:*** *Navigation and UTC time information should be available within 200 milliseconds following the leading edge of the Time Mark at time $t_0$. The information is considered to be valid at time $t_0$. Specifically, the information is NOT projected to be valid at time $t_1$, which represents the beginning of the next GPS/GNSS Epoch.*

Test procedure modification for the changes made above is expected to be minimal. Upon agreement to the above modifications and acceptance of Appendix "S", test procedures will be addressed and hopefully it will take only a few short paragraphs.

This concludes the changes necessary in the main body of the MOPs at this time. Appendix "S" follows as an attachment.

# Blank

**(This page left Blank for document structure purposes)**

# APPENDIX "S"

# GPS TIME TAG
# and
# UTC TIME SYNCHRONIZATION

# Blank

**(This page left Blank for document structure purposes)**

## S. GPS Time Tag and UTC Time Synchronization

### S.1 Introduction

RTCA DO-260B section 2.2.3.2.3.5 requires that the validity of the Airborne Position Messages be established at exact 0.2 UTC epochs when the "Time" (T) subfield is set to "1". Section 2.2.3.2.3.6 then requires that the CPR encoding be alternated between EVEN and ODD encodings each 0.2 UTC epoch. Section 2.2.3.2.3.6.b Note 2 is more specific in that it requires that the distribution of EVEN and ODD encodings for each 0.2 UTC epoch be started on an EVEN epoch. Figure S.1 illustrates the intended encoding. Inspection of Figure S-1 shows that the intent is to start the even/odd CPR encoding distribution starting at the beginning of the Even Epoch with an EVEN sub-epoch. The result is an equal number of even and odd sub-epochs distributed over two seconds. Further inspection of Figure S-1 shows that synchronization could also be done at the beginning of the ODD Epoch provided that the odd/even CPR encoding distribution starts with an ODD sub-epoch.

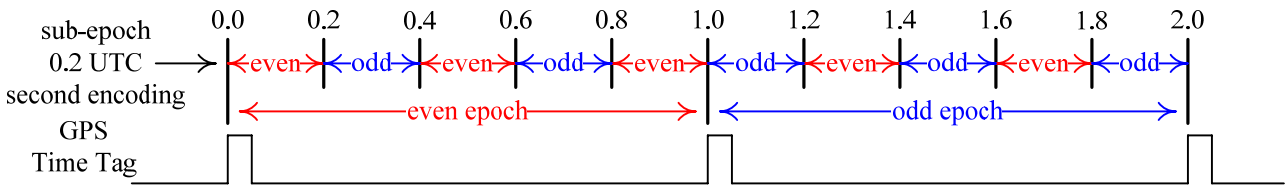



**Figure S-1: Exact 0.2 UTC sub-epoch Encoding**

RTCA DO-260B section 2.2.3.2.4.5 and 2.2.3.2.4.6 then provide similar requirements for the Surface Position Messages.

RTCA DO-260B section 2.2.5.1.6.2 then establishes requirements for the ADS-B Transmit Subsystem to accept the GPS/GNSS Time Mark in order to synchronize the 0.2 second UTC epochs when the "Time" (T) subfield is set to "1". At the time that these requirements were first established during the development of RTCA DO-260 it was established that the leading edge of the GPS Time Mark would be within +/- 5 milliseconds of the beginning of the epoch. Likewise, it was established that position, velocity, and time (PVT) data can be expected from the GPS no later than 200 milliseconds after the leading edge of the Time Mark. Review of most GPS systems has indicated that PVT data is being provided within 200 milliseconds of the Time Mark. However, review of the relationship between the leading edge of the Time Mark and the beginning of the UTC Epoch indicates that the Time Mark and UTC Epoch are not tightly coupled. Table S-1 provides data taken from one major GPS vendor. Note that the data is based on the ARINC **"140"** and **"150"** labels received which are detailed in Table S-2 and S-3.

Review of Table S-1 indicates that the particular GPS gives UTC Time of the Time Mark Leading Edge as 17:08:49.79439544677 which is approximately 0.8 seconds into the ODD epoch or approximately 0.2 seconds before the beginning of the EVEN epoch. Other data reviews indicate that re-cycling of power on the GPS will result in a completely different relationship between the Time Mark and the UTC Time of the Time Mark Leading Edge. Consequently, it appears that the industry cannot count on the GPS Time Mark and the UTC of the Time Mark being close to the actual beginning of the epoch. Therefore, a method is needed to synchronize the Time Mark and the UTC Time of the Time Mark such that the exact 0.2 UTC sub-epochs required by ADS-B can be established based on the time provided by the UTC Time of the Time Mark Leading Edge.

The following sections detail an acceptable method for synchronizing the UTC Time of the Time Mark Leading edge such that exact 0.2 UTC sub-epochs can be established with the proper even/odd CPR encoding sub-epoch distribution.

***Note:*** *The time of **17:08:49.79439544677** or **17:08:49.8** will be used as the **Reference Time** in the following discussions in section S-1 inclusive.*

# Blank

**(This page left Blank for document structure purposes)**

| Sample | Time (sec) | Rx | Label | Hex | Translation | Value | | | | | | Units | "150" Time Hrs:Min:Sec | "140" Binary Count | "140" Time fractional seconds | Total time Hrs:Min:Sec.frac | Time to Next 0.2 sub_epoch | Time to Next Re-Sync |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.702 | 1 | 125 | 05C223 | UTC | 1708.8 | | | | | | Hr:Min | | | | | | |
| | 0.702 | 1 | 150 | 68918F | | 0110 | 1000 | 1001 | 1001 | 1000 | 1111 | | 17:08:49 | | | 17:08:49.794395446777 | | 0.205604553223 |
| | 0.703 | 1 | 140 | 6CB5D8 | | 0110 | 1100 | 1011 | 0101 | 1101 | 1000 | | | 832984 | 0.794395446777 | | 0.005604553223 | |
| | 0.703 | 1 | 141 | 6BEC03 | | 0110 | 1011 | 1110 | 1100 | 0000 | 0011 | | | | | | | |
| 2 | 1.702 | 1 | 125 | 05C223 | UTC | 1708.8 | | | | | | Hr:Min | | | | | | |
| | 1.703 | 1 | 150 | 689197 | | 0110 | 1000 | 1001 | 0001 | 1001 | 0111 | | 17:08:50 | | | 17:08:50.794396400452 | | |
| | 1.703 | 1 | 140 | 6CB5D9 | | 0110 | 1100 | 1011 | 0101 | 1101 | 1001 | | | 832985 | 0.794396400452 | | 0.005603599548 | |
| | 1.703 | 1 | 141 | 635C03 | | 0110 | 0011 | 0101 | 1100 | 0000 | 0011 | | | | | | | |
| 3 | 2.704 | 1 | 125 | 05C223 | UTC | 1708.8 | | | | | | Hr:Min | | | | | | |
| | 2.705 | 1 | 150 | 68919F | | 0110 | 1000 | 1001 | 0001 | 1001 | 1111 | | 17:08:51 | | | 17:08:51.794397354126 | | 0.205602645874 |
| | 2.705 | 1 | 140 | 6CB5DA | | 0110 | 1100 | 1011 | 0101 | 1101 | 1010 | | | 832986 | 0.794397354126 | | 0.005602645874 | |
| | 2.705 | 1 | 141 | 601003 | | 0110 | 0000 | 0001 | 0000 | 0000 | 0011 | | | | | | | |
| 4 | 3.702 | 1 | 125 | 05C223 | UTC | 1708.8 | | | | | | Hr:Min | | | | | | |
| | 3.702 | 1 | 150 | 6891A7 | | 0110 | 1000 | 1001 | 0001 | 1010 | 0111 | | 17:08:52 | | | 17:08:52.794397354126 | | |
| | 3.703 | 1 | 140 | 6CB5DA | | 0110 | 1100 | 1011 | 0101 | 1101 | 1010 | | | 832986 | 0.794397354126 | | 0.005602645874 | |
| | 3.703 | 1 | 141 | 6CC803 | | 0110 | 1100 | 1100 | 1000 | 0000 | 0011 | | | | | | | |
| 5 | 4.702 | 1 | 125 | 05C223 | UTC | 1708.8 | | | | | | Hr:Min | | | | | | |
| | 4.702 | 1 | 150 | 6891AF | | 1000 | 1001 | 0001 | 0001 | 1010 | 1111 | | 17:08:53 | | | 17:08:53.794398307800 | | 0.205601692200 |
| | 4.703 | 1 | 140 | 6CB5DB | | 0110 | 1100 | 1011 | 0101 | 1101 | 1011 | | | 832987 | 0.794398307800 | | 0.005601692200 | |
| | 4.703 | 1 | 141 | 67BC03 | | 0110 | 0111 | 1011 | 1100 | 0000 | 0011 | | | | | | | |
| 6 | 5.702 | 1 | 125 | 05C227 | UTC | 1708.9 | | | | | | Hr:Min | | | | | | |
| | 5.703 | 1 | 150 | 6891B7 | | 1000 | 1001 | 0001 | 0001 | 1011 | 0111 | | 17:08:54 | | | 17:08:54.794398307800 | | |
| | 5.703 | 1 | 140 | 6CB5DB | | 0110 | 1100 | 1011 | 0101 | 1101 | 1011 | | | 832987 | 0.794398307800 | | 0.005601692200 | |
| | 5.703 | 1 | 141 | 6F2C03 | | 0110 | 1111 | 0010 | 1100 | 0000 | 0011 | | | | | | | |

**Table S-1: Observed GPS Output UTC TIME Data**

| ARINC LABEL "150" UTC FORMAT | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Bit | 32 | 31 -- 30 | 28 --- 24 | 23 ---- 18 | 17 ---- 12 | 11 | 10 -- 9 | 8 ------ 1 |
| **Field** | **Parity** | **SSM** | **HOURS**<br>0 - 23 | **MINUTES**<br>0 – 59 | **SECONDS**<br>0 - 59 | ***** | **SDI** | **LABEL**<br>**('150")** |

**Notes:**

1. "*" Bit 11 of label "150" should be encoded with a "1" when the GNSS system clock is being used as the source of time. Otherwise, Bit 11 should be encoded as "0".
2. Maximum Data Range is 23:59:59 in Hours:Minutes:Seconds
3. Reference: ARINC 429 Part 1-16, Attachment 6 Table 6-12 and Attachment 6 Table 6-27 and Attachment 2, Table 2, Label "150".
4. Reference: ARINC 743A-4, Attachment 4-2, label "150", Page 29.

**Table S-2: ARINC Label "150" Data Format**

| ARINC LABEL "140" UTC FINE FORMAT | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Bit | 32 | 31 -- 30 | 29 | 28 --- 24 | 24 -- 21 | 20 -- 17 | 16 -- 13 | 12 -- 9 | 8 ------ 1 |
| **Field** | **Parity** | **SSM** | **PAD**<br>0 | **Binary Count, 0000 –to- FFFFF Hex**<br>**LSB = Resolution = 0.9536743164 microseconds**<br>**Maximum Count = 0.9999990663 microseconds** | | | | | **LABEL**<br>**('140")** |

**Notes:**

1. Reference: ARINC 429 Part 1-16, Attachment 2, Table 2, Label "140".
2. Reference: ARINC 743A-4, Attachment 4-2, label "140", Page 29

**Table S-3: ARINC Label "140" Data Format**

## S-2 Method for Time Mark / UTC Time Synchronization

The following sub-sections detail the method of establishing synchronization to the UTC Time based on receipt of the GPS Time Mark, Label **"150"** and Label **"140"** from the GPS receiver. Note that appropriate formats for the **"150"** and **"140"** labels were previously provided in Table S-2 and S-3 respectively.

Advantages of the method are:

a. Synchronization can be performed at virtually any time during the epoch based on the time given in the **"150"** and **"140"** words.

b. Once synchronization is established, it should not need to be repeated more frequently than approximately every 10 seconds. As a worse case, synchronization would be done every two seconds.

Figures S-2a and S- 2b detail the synchronization method. Review of the figures indicates that each of the flow diagram boxes has a number from "1" –to- "33" shown in bold **"Red"** font inside the box. These "Block" numbers are used as reference numbers in the following sections.

### S-2.1 START of Synchronization (Block #1 through 3)

Synchronization starts with waiting for the Leading Edge of the GPS Time Mark (Block #2). Upon detection of the leading edge of the Time Mark, a timer is started to count the time that expires from the leading edge of the Time Mark. Figure S-2a Block #3 identifies this timer as $\mathbf{timer_{tag_edge}}$. This timer should be an ***UP*** counter with a minimum resolution of approximately 1.0 milliseconds. Those systems that desire better precision should use much finer resolution of perhaps 100 microsecond increments. Range of the counter should be up to 1.0 second at the maximum.

### S-2.2 Wait for GPS Data (Block #4)

Next, wait for GPS data to be received from the GPS receiver. This must include Latitude, Longitude, North/South Velocity, East/West Velocity, Label **"150"** UTC, and Label **"140"** UTC Fine data as a minimum to execute the synchronization process. Ensure that all data is properly stored for later use in the synchronization process.

### S-2.3 Compute Current Time (Block #5)

Current Time represents the time at which the synchronization process computations are completed: Specifically, Current Time is given by the following Equation 1.

**Equation 1:** $\mathbf{time_{current} = time_{in_label_140} + timer_{tag_edge} + time_{est_comp}}$

where: $\mathbf{time_{in_label_140}}$ represents the time in fractional seconds provided in the label **"140"** word received from the GPS receiver.

$\mathbf{timer_{tag_egde}}$ represents the time in fractional seconds that is indicated by the timer started in Block #3.

$\mathbf{time_{est_comp}}$ represents the estimated time of computation that it will take to complete the synchronization computation process. This time may be considered optional since the time of computation is expected to be short.

### S-2.4 Determine Sub-Epoch Base and Remainder (Block #6)

The $\mathbf{sub\text{-}epoch_{base}}$ is used to establish which sub-epoch the current time resides in and should be a number from "0" –to- "4". Specifically, $\mathbf{sub\text{-}epoch_{base}}$ is given by the following Equation 2:

**Equation 2:** $\mathbf{sub\text{-}epoch_{base}} = \mathbf{floor}(\mathbf{time_{current}}/0.200)$

where: **floor** represents the integer "floor" function

$\mathbf{time_{current}}$ represents the current time established in Equation 1 for Block #5

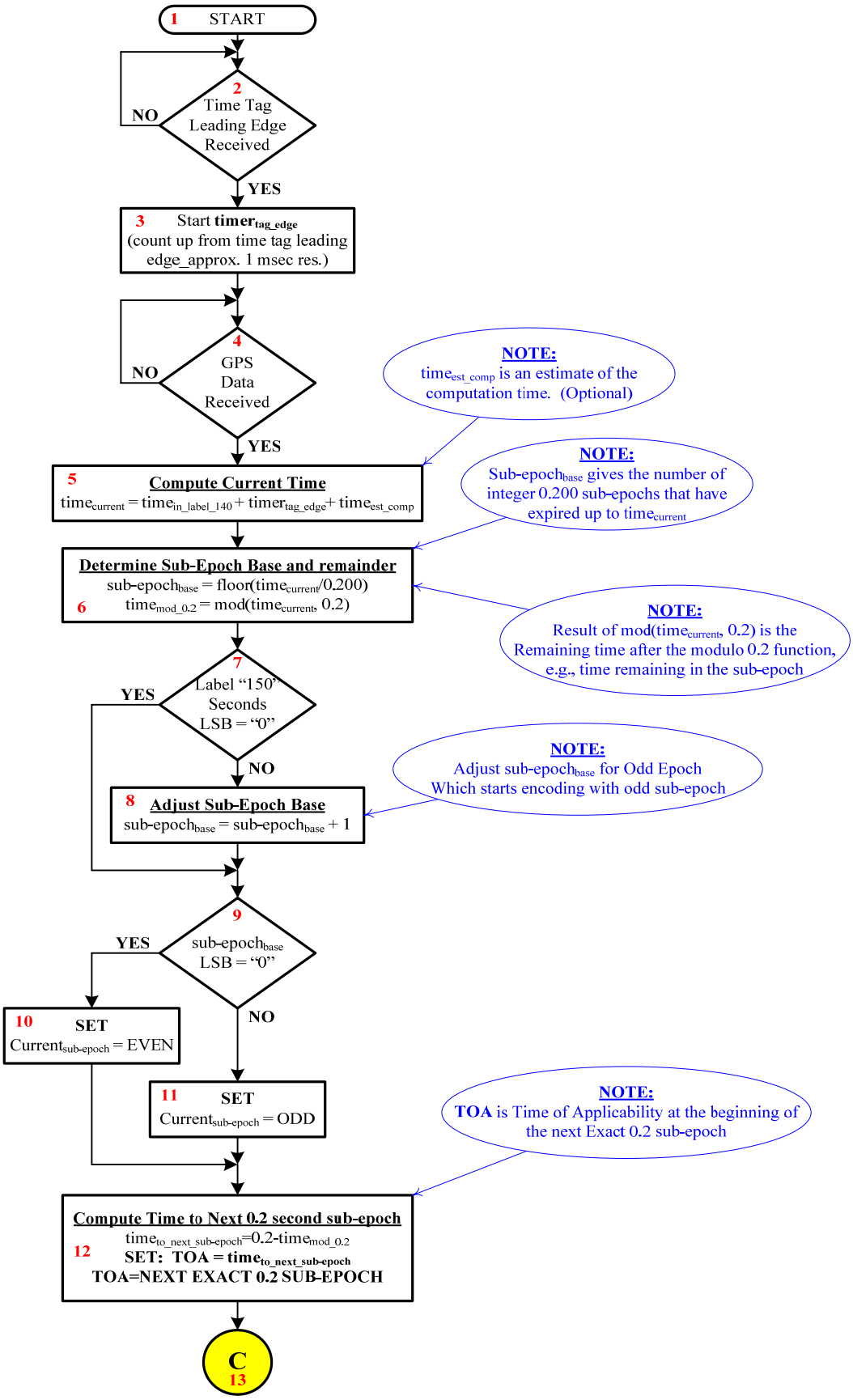
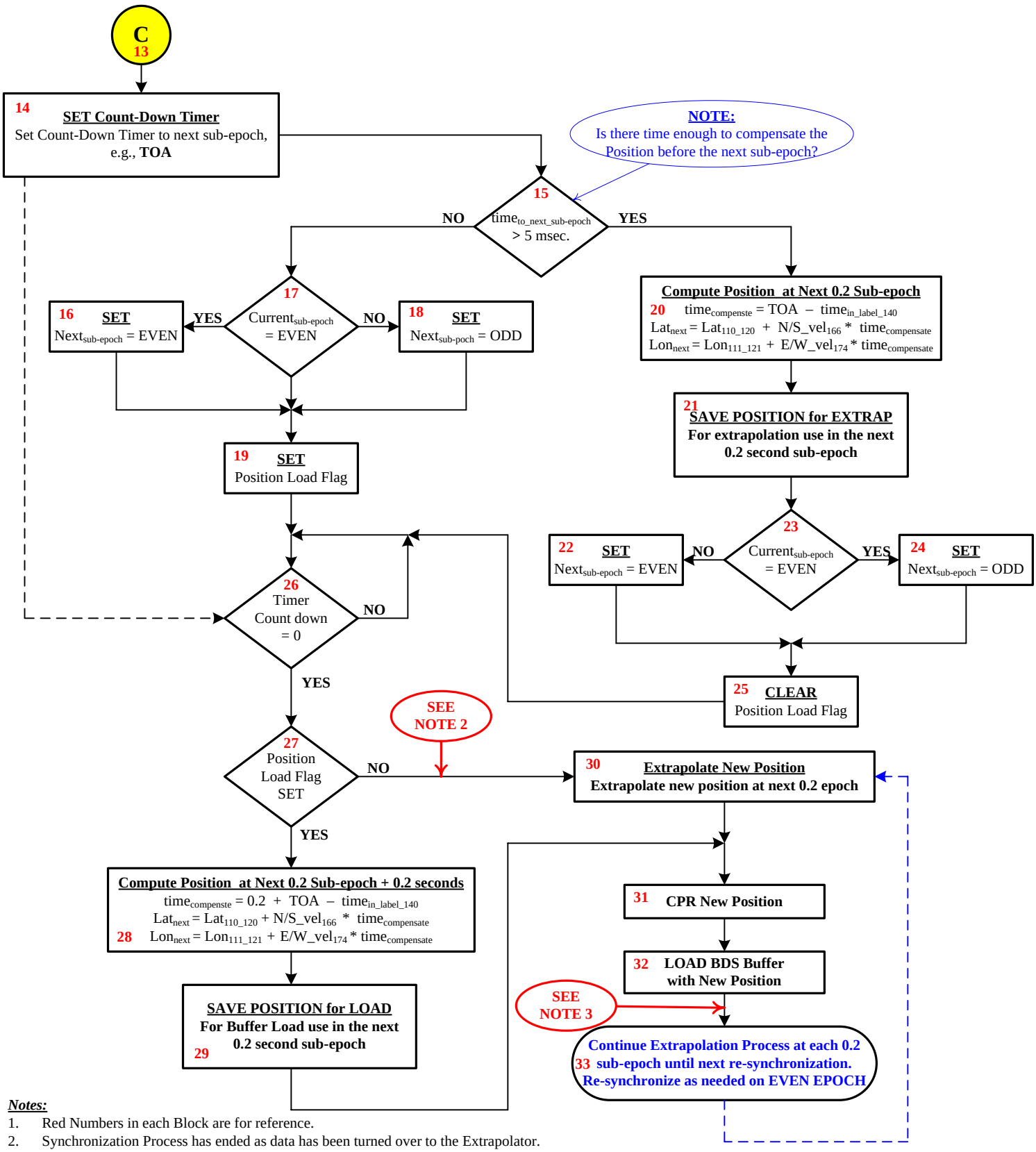



**Figure S-2a: Time Mark __ UTC Synchronization Flow Diagram (Part 1)**




***Notes:***

1. Red Numbers in each Block are for reference.
2. Synchronization Process has ended as data has been turned over to the Extrapolator.
3. Synchronization Process has ended as data has been computed forward, CPR encoded, loaded into the appropriate register and available for transmission .

**Figure S-2b: Time Mark __ UTC Synchronization Flow Diagram (Part 2)**

Example: let **time**$_{\mathbf{current}}$ = 0.680 seconds

Then, **sub-epoch**$_{\mathbf{base}}$ = 3 which means that 3 sub-epochs of 0.2 seconds have been used up to get to **time**$_{\mathbf{current}}$

Next, we need to know the amount of time remaining to **time**$_{\mathbf{current}}$ after the **sub-epoch**$_{\mathbf{base}}$. That is, how far into the next sub-epoch is **time**$_{\mathbf{current}}$? This parameter is referred to as **time**$_{\mathbf{mod_0.2}}$ and is defined in the following Equation 3.

**Equation 3:** **time**$_{\mathbf{mod_0.2}}$ = **mod**(timecurrent,0.2)

where: **mod** represents the modulus function of **time**$_{\mathbf{current}}$ divided by 0.200. Effectively, the function returns the remainder of **time**$_{\mathbf{current}}$ divided by 0.200.

Example: **time**$_{\mathbf{mod_0.2}}$ = **mod**(0.680,0.200) = 0.080 seconds

## S-2.5 Determine Current Sub-Epoch Base (Block #7 and 8)

The **sub-epoch**$_{\mathbf{base}}$ was computed in S-2.4 and is used to establish proper synchronization in any EPOCH and maintain the proper Even/Odd encoding distribution. Figure S-3 illustrates how the **sub-epoch**$_{\mathbf{base}}$ aligns with the Even/Odd encoding distribution for the EVEN Epoch.

Now, referring to Figure S-2a, Block #7 determines whether the time is in the EVEN or ODD Epoch based on the least significant bit (LSB) of the "Seconds" provided in Label **"150"**. If time is in the EVEN epoch, then the **sub-epoch**$_{\mathbf{base}}$ is either even or odd and aligns with the expected Even/Odd distribution for an EVEN epoch as illustrated in Figure S-3. However, if Label **"150"** indicates that time is in an ODD epoch, **sub-epoch**$_{\mathbf{base}}$ needs to be increased by "1" as done in Block #8 of Figure S-2a. **Sub-epoch**$_{\mathbf{base}}$ **+ 1** will then be either odd or even and align with the expected Odd/Even distribution for an ODD epoch as illustrated in Figure S-3.

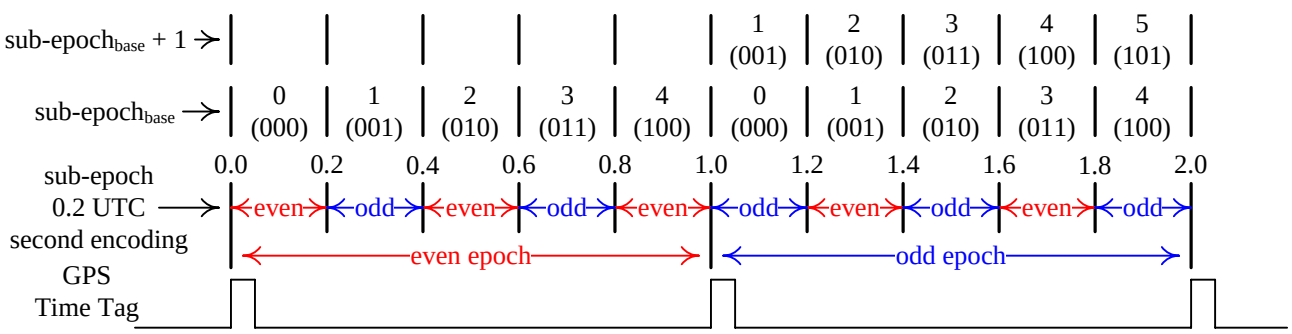



**Figure S-3: Exact 0.2 UTC sub-epoch Encoding with sub-epoch**$_{\mathbf{base}}$ **and sub-epoch**$_{\mathbf{base}}$ **+ 1**

## S-2.6 Determine Current Sub-Epoch Even or Odd Encoding (Block #9 through 11)

Figure 2a, Block #9 then checks the results out of Block #7 or #8 to see if the LSB of the resultant **sub-epoch**$_{\mathbf{base}}$ is either "0" or "1". If "0", then the **Current**$_{\mathbf{sub\text{-}epoch}}$ is **SET** to **"EVEN"** as indicated by Block #10. Otherwise, the resultant **sub-epoch**$_{\mathbf{base}}$ is "1" and the **Current**$_{\mathbf{sub\text{-}epoch}}$ is **SET** to **"ODD"** as indicated in Block #11.

Example: Consider the case where the **sub-epoch**$_{\mathbf{base}}$ = 3 as in Section S-2.4.

Then, for an EVEN epoch, "1" is not added as per Block #7. The **sub-epoch**$_{\mathbf{base}}$ remains at three and Block #9 and #11 will result in the **Current**$_{\mathbf{sub\text{-}epoch}}$ being set to ODD. Review of Figure S-3 confirms that the sub-epoch encoding should be ODD for the given sub-epoch 3 in the EVEN epoch.

Then, for an ODD epoch, "1" is added as per Block #7 and #8. The **sub-epoch$_{base}$** increases from 3 to 4 which results in Block #9 and #10 setting the **Current$_{sub\text{-}epoch}$** to EVEN. Review of Figure S-3 confirms that the sub-epoch encoding should be EVEN for the given sub-epoch 3 in the ODD epoch.

**S-2.7 Compute time to Next Exact 0.2 second Sub-epoch (Block #12)**

Block #12 establishes the exact amount of time remaining to the beginning of the next exact 0.2 second sub-epoch. **Time$_{to_next_sub\text{-}epoch}$** is defined in Equation 4 as follows:

**Equation 4:** $\mathbf{time_{to_next_sub\text{-}epoch}} = 0.2 - \text{time}_{mod_0.2}$

where: **time$_{mod_0.2}$** is as computed or defined in section S-2.4 Equation 3.

Next, for convenience purposes, the Time of Applicability (TOA) at the next exact 0.2 second sub-epoch is set to be equal to the **time$_{to_next_sub\text{-}epoch}$** as per the following Equation 5.

**Equation 5:** $\mathbf{TOA} = \mathbf{time_{to_next_sub\text{-}epoch}}$

**S-2.8 SET Count Down Timer to next sub-epoch (Block #14)**

Figure S-2a Block #14 simply loads a count-down timer with the value of **TOA** and initiates the timer to count down. Action that is necessary upon terminal count down of the counter is addressed in section S-2.11, for Block #26.

**S-2.8 Determine if there is sufficient Computation Time (Block #15)**

Figures S-2b, Block #15 is necessary to determine if there is sufficient time to compensate the Current Position prior to the beginning of the next exact 0.2 second sub-epoch. The time of 5 milliseconds shown in Block #15 is ***ARBITRARY*** and is based on the particular vendor design as the vendor needs to be able to establish how much time is required to complete the position compensation.

a. Sufficient Time Available:

If there is sufficient time available, then position is compensated to be applicable at the beginning of the next exact 0.2 second sub-epoch. Then once the next exact 0.2 second sub-epoch is reached, the up-dated position is extrapolated forward to the next exact 0.2 second sub-epoch to provide the updated position. The updated position is then CPR encoded and loaded into the appropriate register in order to be available for randomized transmission when required.

The key thing here is that position is only compensated to be accurate at the next 0.2 second sub-epoch. After that, further compensation is left to the Extrapolation process. ***Note that the compensation is done prior to the Count Down Timer expiration in Figure S-2b Block #26.***

This entire process is covered in detail in Section S-2.9 and Figure S-2b Block #20 through #25.

b. Insufficient Time Available:

If there is insufficient time to compensate the position prior to the next exact 0.2 second sub-epoch, then the position must be compensated after the Count Down Timer expiration in Figure S-2b Block #26. In this case, the position will be compensated directly to the ***next exact 0.2 second sub-epoch that occurs AFTER the Count Down Timer has expired***. The compensated position is then saved, CPR encoded and loaded into the appropriate register in order to be available for randomized transmission when required. As such, ***the compensated position DOES NOT go through the extrapolation process in this case as it has already been extrapolated forward***.

The entire computational process discussed in this subparagraph is covered in detail in Section S-2.11.2 and Figure S-2b Block #27 through #29, Block #31 and Block #32. However, prior to the computational process which takes place after Count-Down Timer expiration, the process must be set up via decision logic prior to Timer expiration as discussed in Section S-2.10 and Figure S-2b Block #16 through Block #19

## S-2.9 Sufficient Computation Time (Block #20 through #25)

The following sub-sections apply if there is Sufficient Computation Time as discussed previously in sub-section 2-2.8.a.

### S-2.9.1 Compute Position at Next 0.2 Sub-epoch (Block #20)

Figure S-2b Block #20 compensates the position information to be valid or applicable at the next exact 0.2 second sub-epoch when there is time to do the compensation prior to the start of ***that*** next exact 0.2 second sub-epoch. First, the amount of time to be compensated for is computed by the following Equation 6.

**Equation 6:** $\mathbf{time_{compensate} = TOA - time_{in_label_140}}$

where: **TOA** is as previously computed in section S-2.7 and Equation 5.

$\mathbf{time_{in_label_140}}$ is as previously discussed in section S-2.3

Next, the Next Latitude Position or better defined as the Latitude Position at the beginning of the next exact 0.2 second sub-epoch is computed as shown in the following Equation 7.

**Equation 7:** $\mathbf{Lat_{next} = Lat_{110_120} + N/S_vel_{166} * time_{compensate}}$

where: $\mathbf{Lat_{110_120}}$ is the current latitude position value provided in ARINC Labels “110” and “120” as received from the GPS receiver during the same time frame as the UTC time words were received from the GPS receiver.

$\mathbf{N/S_vel_{166}}$ is the current North/South velocity value provided in ARINC Label “166” as received from the GPS receiver during the same time frame as the UTC time words were received from the GPS receiver.

$\mathbf{time_{compensate}}$ is as defined in Equation 6

Next, the Next Longitude Position or better defined as the Longitude Position at the beginning of the next exact 0.2 second sub-epoch is computed as shown in the following Equation 8.

**Equation 8 :** $\mathbf{Lon_{next} = Lon_{111_121} + E/W_vel_{174} * time_{compensate}}$

where: $\mathbf{Lon_{111_121}}$ is the current longitude position value provided in ARINC Labels “111” and “121” as received from the GPS receiver during the same time frame as the UTC time words were received from the GPS receiver.

$\mathbf{E/W_vel_{174}}$ is the current East/West velocity value provided in ARINC Label “174” as received from the GPS receiver during the same time frame as the UTC time words were received from the GPS receiver.

$\mathbf{time_{compensate}}$ is as defined in Equation 6

### S-2.9.2 Save Position for Extrapolation (Block #21)

Figure S-2b Block #21 saves the $\mathbf{Lat_{next}}$ and $\mathbf{Lon_{next}}$ values computed in section S-2.9.1 to be used by the next extrapolation process which will come after the beginning of the next exact 0.2 second sub-epoch.

### S-2.9.3 Determine Next sub-epoch Even/Odd Encoding (Block #22 through #24)

Figure S-2b Block #22 through #24 then establish the Even/Odd encoding format that should be used by the Extrapolation Process which occurs after the beginning of the next exact 0.2 second sub-epoch.

Basically, ***IF*** the current sub-epoch is **ODD** (see figure S-3), ***THEN*** the $\mathbf{Next_{sub\text{-}epoch}}$ is **EVEN**.

Conversely, ***IF*** the current sub-epoch is **EVEN** (see figure S-3), ***THEN*** the $\mathbf{Next_{sub\text{-}epoch}}$ is **ODD**.

### S-2.9.4 Clear Position Load Flag (Block #25)

Remembering that this entire section, e.g., S-2.9, deals with the condition where there is sufficient time to compensate position prior to the beginning of the next exact 0.2 second epoch, it is important to note that position has been compensated to the beginning of the next exact 0.2 second epoch. When that happens, the Extrapolation Process will then properly extrapolate the position to the next exact 0.2 second epoch and then proceed to the CPR encoding.

**THEREFORE:** **CLEAR** the **Position Load Flag** which may have been set in prior synchronization processes in section S-2.10.2 (Block #19).

**THEN:** **PROCEED** to section S-2.11 (Block #26)

### S-2.10 Insufficient Computation Time Process Set Up (Block # 16 through 19)

As previously discussed, Section S-2.8.b, indicates that computational process must be properly set up prior to Count-Down Timer expiration when the position computation must be done after the beginning of the next exact 0.2 second sub-epoch. This setup process is now addressed in detail in the following sub-sections in regard to Figure S-2b Block #16 through Block #19.

### S-2.10.1 Determine the Next sub-epoch Even/Odd Encoding (Block # 16 through 18)

Figure S-2b Block #16 through #18 are used to establish the Even/Odd encoding format that should be used by the CPR Encoding algorithm after receiving updated Position data directly from the synchronization process as opposed to having come from the extrapolator.

Basically, ***IF*** the current sub-epoch is **ODD** (see figure S-3), ***THEN*** the $\mathrm{Next_{sub\text{-}epoch}}$ is **ODD**.

Conversely, ***IF*** the current sub-epoch is **EVEN** (see figure S-3), ***THEN*** the $\mathrm{Next_{sub\text{-}epoch}}$ is **EVEN**.

Note that the EVEN/ODD encoding is different from that in section S-2.9.3 specifically because the position will have been extrapolated an additional 0.2 milliseconds such that the extrapolator at the beginning of the 0.2 second sub-epoch is ***by-passed***.

### S-2.10.2 SET the Position Load Flag (Block #19)

Figure S-2b Block #19 is used to ***SET*** the Position Load Flag to indicate that the Position Needs to be Compensated and Loaded after the Count-Down Time has expired, thereby indicating the beginning of the next exact 0.2 second sub-epoch.

### S-2.11 Timer Count Down and Position Flag Status (Block # 26 through #27)

Count-Down Timer expiration is determined by Figure S-2b Block #26 and indicates the start of the next exact 0.2 second sub-epoch. At that time, there are two choices that have been set up by the previous processing that need to be resolved as discussed in the following sub-sections.

### S-2.11.1 Position Load Flag NOT SET (Block # 27 )

If the Position Load Flag is ***NOT SET***, then the position has been extrapolated to the beginning of the next exact 0.2 second sub-epoch. As such, the position is ready to be extrapolated to the next exact 0.2 second sub-epoch by the extrapolator which executes shortly after the beginning of the 0.2 second sub-epoch.

As indicated by ***Note 2*** in ***Figure S-2b***, the ***Synchronization Process is DONE*** at this point and the Extrapolation Process will take over in the next 0.2 second sub-epoch as shown in Figure S-

2b Block #30. After the Extrapolation, the Position is CPR Encoded as shown in Figure S-2b Block #31 and loaded into the appropriate register as shown in Figure S-2b Block #32.

### S-2.11.2 Position Load Flag SET (Block # 27 )

#### S-2.11.2.1 Compute Position to be Loaded (Block # 28 )

If the Position Load Flag ***IS SET***, then the Position data must be extrapolated to the next exact 0.2 second sub-epoch **PLUS** an additional 200 milliseconds. First, the amount of time to be compensated for is computed by the following Equation 9.

**Equation 9:** $\mathbf{time_{compensate}} = 0.2 + \mathbf{TOA} - \mathbf{time_{in_label_140}}$

where: **TOA** is as previously computed in section S-2.7 and Equation 5.

$\mathbf{time_{in_label_140}}$ is as previously discussed in section S-2.3

Next, the Next Latitude Position is computed as shown in section S-2.9.1 Equation 7 and in Figure S-2b Block #28.

Next, the Next Longitude Position is computed as shown in section S-2.9.1 Equation 8 and in Figure S-2b Block # 28

#### S-2.11.2.2 Save Position for Load (Block #29 )

The new position computed in section S-2.11.2.1, e.g., Block #28, is then saved for extrapolation to a new position update in subsequent sub-epochs that occur **AFTER** the synchronization process.

#### S-2.11.2.3 CPR Encode New Position Data (Block # 31 )

The new position computed in section S-2.11.2.1, e.g., Block #28, is then CPR Encoded as shown in Figure S-2b, Block #31.

Note that the position data was not routed to the Extrapolator (Figure S-2b, Block #30) as was previously done in section S-2.11.1.

#### S-2.11.2.4 Load BDS Buffer (Block # 32 )

Last, but not least, the CPR encoded (section S-2.11.2.3) position data is loaded into the appropriate register as shown in Figure S-2b Block #32. At this point, the data is ready for random access for transmission by the ADS-B Transmitting Subsystem.

As indicated by ***Note 3*** in ***Figure S-2b***, the ***Synchronization Process is DONE*** at this point as data has been extrapolated to the next exact 0.2 sub-epoch, CPR encoded, and loaded into the appropriate register to be accessed for transmission.

### S-2.12 Established Processing Loop (Block # 30 through #33)

Figure S-2b Block #30 through Block #33 represent blocks that should already be part of the position extrapolation function used by the ADS-B Transmitting Subsystem.

Block #30 represents the Extrapolation function which extrapolates position to the next exact 0.2 second sub-epoch.

Block #31 represents the CPR encoding process which encodes the position data into the format necessary for ADS-B transmission.

Block #32 represents the storage of the CPR encoded position data for subsequent access by the transmitting function of the ADS-B Transmitting Subsystem.

Block #33 represents the requirement that the extrapolation process is continued at subsequent 0.2 second sub-epochs until the next synchronization process. Although, Block #33 infers that the re-synchronization process should start at the beginning of an EVEN epoch, previous discussion in the first paragraph of section S-1 indicates that the re-synchronization process can be started at the beginning of an ODD epoch. As such, the synchronization process detailed in this document is designed to work in both cases and will work starting with either the EVEN or ODD epoch. Actually, the synchronization process detailed herein establishes proper synchronization when starting anywhere within an EVEN or an ODD epoch.

**S-3 Summary / Conclusion**

The Synchronization Method described in this document should ensure that proper synchronization is established to the beginning of an ODD or EVEN epoch as defined by UTC time provided in ARINC Labels "150" and "140". Likewise, the Method ensures that the EVEN/ODD sub-epoch distribution is properly initiated during the synchronization process. As such, the Method can be applied at any time and still insure proper synchronization and sub-epoch encoding distribution.

As a disclaimer, the Synchronization Method described in this document represents an acceptable method of achieving the synchronization task. Other methods that achieve equivalent results should be considered as being equally acceptable.

## END OF APPENDIX "S"