GRAIL Time Transfer Assembly Algorithm Document

JPL D-61501

Prepared:

Courtney Duncan, GRAIL GPA System Engineer

Concurred:

Charles Dunn, GRAIL Payload Manager

Revisions

2009 July 15, Version 1.2

2009 April 18, Version 1.1

2009 January 26, Version 1.0

Purpose

This document describes the GRAIL TTA and MWA frequency scheme. An accompanying spreadsheet supports the calculations.

GRAIL-A

USO frequency.

4.832000 MHz

X 8 to GPA = 38.656000 MHz

X 12 to MWA = 57.984000 MHz

Ka-Band MWA multiplies by 564 = 32702.976000 MHz (=GRAIL-B - 670032 Hz)

One tone integration is 20.000000 milliseconds of which 19.000000 milliseconds is non-dead time and 1.000000 millisecond is dead time.

The tone integration rate is 50 per second.

Synthesizer Ratio

GPA_refA* 52.5 + 39333 / 524288 =

GPA_refA * 27564453 / 524288 = 2032340040.527344 Hz to 16 digits.

Sample Rate

GPA_refA / 2 = 19.328000 MHz (This is the "P-Code" rate.)

The signal is sampled at 38.656000 MHz and every other sample is discarded.

Chipping Rate

C/A code divider: 20

C/A chip rate = SampRate / 20 = 966400 chips / second

(Details of the ASIC divide by 20 are shown in the accompanying spreadsheet.)

Code Cycle

1023 chips

cycle rate = chip rate / 1023 = 944.672532 code cycles / second (~msec)

Per second, this is exactly 944 cycles with a remainder of 688 chips. Epochs are calculated from this.

Databits

20 code cycles = 20460 chips

databit rate = chip rate / 1023 / 20 = 47.233627 bits / second (~50)

databit length = 21.171 milliseconds.

Data Message

256 databits in a message.

Message length = (1023 * 20 * 256) / chip rate = 5.419868 seconds (~6 sec.)

One 32-bit long in each 120-bit message contains the time code. The high order 14 bits are the “repeat count” (explained below) and the low order 18 bits are the “message count.” Each bit in the message excepting those in the synchronization and end codes are duplicated with an inverse (explained below) such that it takes 240 bits to send the 120-bit message. The message contains an 8 bit synchronization code and an 8 bit end code making a total of 8 + 240 + 8 = 256 bits per message.

Both GRAILs have a databit edge on a one second epoch every 5115 seconds (1 hr. 25 m. 15 s.) A “fortnight” is 256 of these periods. (See **GRAIL-A and GRAIL-B** below.)

Fortnight = 5115 * 256 = 1309440 seconds (15 d. 3 hr. 44 m. 0 s.)

There are 241600 GRAIL-A data messages in a fortnight. This is 0x3afc0 (18 bits wide) meaning that the message count rolls over from 0x3afbf to 0x00000 each fortnight.

GRAIL-B

USO frequency.

4.832099 MHz (=GRAIL-A + 99 Hz)

X 8 to GPA = 38.656792 MHz

X 12 to MWA = 57.985188 MHz

Ka-Band MWA multiplies by 564 = 32703.646032 MHz (=GRAIL-A + 670032 Hz)

One tone integration is 19.999590 milliseconds of which 18.999611 milliseconds is non-dead time and 0.999980 millisecond is dead time.

The tone integration rate is 50.001024 per second.

Synthesizer Ratio

GPA_refB * 57 + 48322 / 524288 =

GPA_refB * 29932738 / 524288 = 2207.000020707123 Hz to 16 digits.

Sample Rate

GPA_refB / 2 = 19.328792 MHz (This is the "P-Code" rate.)

The signal is sampled at 38.656792 MHz and every other sample is discarded.

Chipping Rate

C/A code divider: 19

C/A chip rate = SampRate / 19 = 1017284 chips / second

(Details of the ASIC divide by 19 are shown in the accompanying spreadsheet.)

Code Cycle

1023 chips

cycle rate = chip rate / 1023 = 994.412512 code cycles / second (~msec)

Per second, this is exactly 994 cycles with a remainder of 422 chips. Epochs are calculated from this.

Databits

20 code cycles = 20460 chips

databit rate = chip rate / 1023 / 19 = 49.720626 bits / second (~50)

databit length = 20.112 milliseconds.

Data Message

256 databits in a message.

Message length = (1023 * 19 * 256) / chip rate = 5.148769 seconds (~6 sec.)

One 32-bit long in each 120-bit message contains the time code. The high order 14 bits are the "repeat count" (explained below) and the low order 18 bits are the "message count." Each bit in the message excepting those in the synchronization and end codes are duplicated with an inverse (explained below) such that it takes 240 bits to send the 120-bit message. The message contains an 8 bit synchronization code and an 8 bit end code making a total of 8 + 240 + 8 = 256 bits per message.

Both GRAILs have a databit edge on a one second epoch is every 5115 seconds (1 hr. 25 m. 15 s.) A "fortnight" is 256 of these periods.

Fortnight = 5115 * 256 = 1309440 seconds (15 d. 3 hr. 44 m. 0 s.)

There are 254321 GRAIL-B data messages in a fortnight. This is 0x3E171 (18 bits wide) meaning that the message count rolls over from 0x3E170 to 0x00000 each fortnight.

GRAIL-A and GRAIL-B

Ka-Band Beat Frequency

Difference in USOs: 99 Hz

X 8 to GPA = 792 Hz

X 12 to MWA = 1188 Hz

Ka-Band MWA multiplies by 564 = 670032 Hz.

(For reference: 12 X 564 = 6768)

The tone integration rates of 50.000000 and 50.001024 per second cause GRAIL-B 20 millisecond integrations to lap those of GRAIL-A every **976.161616** seconds (16 m. 16.161616 s.). The two rates lap at one second (that is, fifty 20 millisecond laps) every 48808.080808 seconds (13 h. 33 m. 28.080808 s.) but do not occur together on an integer second for 241,600,000 seconds (7.7 years). Note: This number is not related to a Fortnight.

Carrier Offsets

GRAIL-A uses a scheme based on its USO to sample the signal from GRAIL-B generated from the scheme based on its USO. This results in a baseband carrier offset from subharmonic sampling as follows:

Carrier Offset = GR-B-tx – GR-A-samp * int(GR-B-tx / GR-A-samp)

= 2207.00002070712 – 19.328000 * int(2207.00002070712 / 19.328000)

= 3.60802070712271 MHz for the S-Band signal.

The virtual local oscillator is 114 * GR-A-samp = 2203.392000 MHz.

GRAIL-B uses a scheme based on its USO to sample the signal from GRAIL-A generated from the scheme based on its USO. This results in a baseband carrier offset from subharmonic sampling as follows:

Carrier Offset = GR-A-tx – GR-B-samp * int(GR-A-tx / GR-B-samp)

= 2032.34004052734 – 19.328396 * int(2032.34004052734 / 19.328396)

= 2.85846052734405 MHz for the S-Band signal.

The virtual local oscillator is 105 * GR-B-samp = 2029.481580 MHz.

Fortnights

GRAIL-A has a databit edge occur on a one-second edge every 1023 seconds (17 m. 3 s.) but GRAIL-B only has this every 5115 seconds = 5 * 1023 (1 h. 25 m. 15 s.). There are 256 of these 5115 second periods in a fortnight because this is how often GRAIL-A and GRAIL-B data messages start at the same time and on an integer second.

Fortnights are therefore calculated differently on each side but the number of seconds in a fortnight is the same (1309440).

GPA-A and GPA-B send data messages every 5.419868 and 5.148769 seconds respectively. GPA-B sends 254321 – 241600 = 12721 more messages in a fortnight than GPA-A does, message lapping it about every 102.935 seconds. 12721 is prime so there is no shorter period of time in which both data messages start on an integer second than a fortnight.

Note that a colloquial fortnight is two weeks. A GRAIL fortnight is 15 days, 3 hours 44 minutes and 0 seconds and is referred to as a “GRAIL fortnight,” a “fort-time,” or, in context of a GRAIL discussion, a “fortnight.”

GRAIL Ambiguity

The 14-bit repeat count counts fortnights. This means that the fundamental ambiguity of the Time Code is

2^14 * 1309440 = 21453864960 seconds = 679.8 Julian years.

At that time, the fortnight counter wraps from 0x3FFF to 0x0000.

Note that 21453864960 takes 35 bits to represent (~5 X 2^32) so problems in the seconds-since counters will more likely occur after only 136 years, as is the case with GPS which counts seconds since 1980. (i.e. 2116)

Contents of Data Message

The data message is 256 bits long. All but the first ten and last eight bits are coded in order to preserve a maximum number of bit edge transitions for tracking. The encoding of one bit with two is as follows:

0b0 encodes to 0b01

0b1 encodes to 0b10

The first 12 bits of the message are

0x8BX where 0x8B is the synch byte (as in GPS) and

X is 0bppgg where g is

0b01 for GRAIL-1 and

0b10 for GRAIL-2 and

pp is set to 0b10 or 0b00 to provide correct parity for the entire message.

(Note that the one-bit GRAIL flag is encoded into two bits per the encoding scheme.)

The next 20 bits are ten encoded zeros, that is, 0x555555. Therefore,

Message[0] = 0x8BX55555, with X defined above.

Continuing,

Message[1,2] = 14 bit fortnight count and 18 bit message index encoded into 64 bits.

Message[3,6] = 64 bit double encoded into 128 bits. This double represents the time offset measured at the transmitting GPA. This is used by the receiving GPA to compute the absolute clock offset and range as discussed below.

Message[7] contains a 12 bit status word encoded into 24 bits followed by the 8 bit end flag 0xF0. Currently, the status word (Status2) is unused and is set to 0xFFF, therefore

Message[7] = 0xCCCCCCF0.

Note that with the encoding pattern, it is impossible for more than two ones or two zeros to occur in a row. Therefore, the synch pattern 0x8B and the end flag 0xF0 cannot occur in the data and, apart from the possibility of bit reception errors, there is no possibility of false synch.

In the ongoing data stream, the end flag 0xF0 is immediately followed by the next synch code 0x8B.

Calculation of "seconds since" from the Time Code

Both sides keep time in message indexes and fortnights as described above.

For GRAIL-A a 256 databit message is 5.419868 seconds long and there are 241600 of them in a fortnight.

For GRAIL-B a 256 databit message is 5.148769 seconds long and there are 254321 of them in a fortnight.

GRAIL-A and GRAIL-B both have 256 * 20 * 1023 = 5,237,760 chips per data message where

256 is the number of bits per message,

20 is the number of code cycles in a databit, and

1023 is the number of chips in a code cycle.

The number of "seconds since" the time origin is calculated by converting the time code in the message to chips and then to seconds using the side-specific chipping rate. The durations of chips, bits, and messages are different on the two sides, but the length of the second is the same.

Note: Due to the start-up strategy, the origin of GPA time will ordinarily be one fortnight and 42 seconds previous to the first GPA of the pair to boot up initially.

Calculate GRAIL-A seconds-since at end-of-message from its data message as

Fortnights * 1,309,440 + (message index) * 5,237,760 / 966,400

Calculate GRAIL-B seconds-since at end-of-message from its data message as

Fortnights * 1,309,440 + (message index) * 5,237,760 / 1,017,284

Where

1,309,440 is the number of seconds in a fortnight;

5,237,760 is the number of chips per data message;

966,400 is the number of chips per second on GRAIL-A: 19,328,000 / 20;

1,017,284 is the number of chips per second on GRAIL-B: 19,328,396 / 19;

19,328,000 is the sample frequency of GRAIL-A; and

19,328,396 is the sample frequency of GRAIL-B.

Calculation of Clock Offset and Range

When the GPAs are tracking each other they each compute a pseudorange between the two systems. The pseudorange is the difference between "seconds since" calculated from the received message (above) and the corresponding "seconds since" at the receiving instant in the receiving GPA.

At GPA A,

prAB = TimeA – TimeB at the GPA A measurement epoch.

For GPA B,

prBA = TimeB – TimeA at the GPA B measurement epoch.

A second-order effect due to drift between the clocks during the signal time-of-flight is neglected.

In the unlikely case that the clocks are exactly synchronized,

prAB = prBA = range / C (seconds) (C is the speed of light) and

offset = 0.

Each GPA calculates its own pseudorange and transmits this value in the data message to the other GPA so that both GPAs have both pseudoranges from which they can perform range and clock offset calculations.

In the general case, both the range and the clock offset are calculated from the two pseudoranges:

range / C = (prAB + prBA) / 2

and

offsetAB = -offsetBA = (prAB – prBA) / 2.

Nudge

"Nudging" is the process of correcting the present receiver's time by some amount to bring it into synchronization with another source, in this case the other GPA. (This concept is inherited from GPS where the receiver is synchronized to GPS time after a point position solution with clock offset has been calculated successfully.) Typically, both integer and fractional parts of the time are changed, so the 1 PPS output phase will typically change when the nudge occurs.

Synchronization of the two-GPA system involves a nudge on one end and a possible integer time count correction on the other described below. The amount to nudge or correct comes from the clock offset calculation described above.

When this is complete, the 1 PPS output signals of both GPAs are synchronized. The offset between them subsequently should be the same as subsequently calculated clock offsets.

(The cases of the system wrapping around to zero time after 679.8 years and causing possible confusion to the initial synchronization process or of the seconds-since integer part counter wrapping 2^32 are not addressed in the code.)

There is no clock “steering” on GRAIL since the standard is the USO outside of the GPA. The clock offset is therefore allowed to drift to arbitrary size. The synchronization requirement is 100 nanosecond *knowledge* so the calculated clock offset must be accurate to 100 nanoseconds.

The clock offset immediately after a nudge should not be much larger than one sample time, about 50 nanoseconds but this offset will grow with time since the USOs are not expected to be exactly on their nominal frequencies.

How Time Transfer Works

In GPS, the raw count of seconds in a receiver is adjusted to “seconds since 1980” from the navigation solution and adjusted to UTC for display purposes using leap seconds from the GPS navigation message.

GRAIL GPAs have no concept of “UTC” or “GPS Time” but they do count integer seconds and establish their own ad hoc time origin in order to synchronize their clocks.

When a GPA boots up, it starts counting the seconds of its internally generated 1 PPS at the number 42 (heritage from Black Jack). After “Beep” mode and loading operating software, it begins transmitting ranging code on its S-Band transmitter. The ranging code is calculated from the GPA raw seconds count as described below.

Each GPA calculates the ranging code it transmits from its own “seconds since” count and calculates the “seconds since” of the other GPA from the ranging code it receives.

The GPA with the larger seconds since count, that is, the one that was booted first or already has its clock set, is considered the standard in the following clock synchronization scenarios. Consider the present GPA to be “A” and the other GPA to be “B.”

Two conditions are being satisfied by the synchronization algorithm:

1. The pulse per second (1 PPS) phase will be the same on both ends (within a sample time ~ 50 nanoseconds).
2. Both clocks will read a time greater than a fortnight.

First, form dt = TimeB – TimeA. This is the amount by which we want to change (or “nudge”) TimeA to match TimeB.

Case 1

If dt < 0, that is, TimeB < TimeA *and* TimeA is already greater than a fortnight, we do nothing.

GPA B is in Case 3 and will perform all of the correction.

Case 2

If dt < 0, that is, TimeB < TimeA *and* TimeA is less than a fortnight, we increase TimeA by a fortnight.

GPA B is now in Case 3 (if it has not yet calculated its own dt) or Case 4 (if it has) and will perform the remainder of the correction.

Case 3

If dt >= 0, that is, TimeB >= TimeA, and *either* dt *or* TimeA is greater than a fortnight, we adjust our time by dt.

GPA B was in Case 1 when we made our dt calculation.

Case 4

If dt >= 0, that is, TimeB >= TimeA, but dt is less than a fortnight *and* TimeA is less than a fortnight, we increase dt by a fortnight and adjust our time by dt.

GPA B was in Case 2 when we made our calculation and will also correct its own clock by exactly a fortnight.

Clocks are now synchronized.

The upshot of this approach is that if during science operations, one GPA restarts, the other GPA maintains the ongoing seconds-since count and 1 PPS phase and the restarted GPA synchronizes to it. In the case where both receivers start or restart, the one that starts counting first sets the initial seconds-since count and 1 PPS phase.

It is unlikely that during a mapping phase both GPAs will be down at the same time, so it is likely that, once established, there will be a monotonic system seconds-since count for each mapping phase.

Acronymns

DOWO	Dual One Way [Clock] Offset
DOWR	Dual One Way Range
EM	Engineering Model
IF	Intermediate Frequency
LO	Local Oscillator
GPA	Gravity Processor Assembly
GRAIL	Gravity and Interior Laboratoy
MWA	Microwave Assembly (32 GHz to 670 KHz conversion)
"Seconds Since"	Time elapsed since the "origin of time"
TTA	Time Transfer Assembly
USO	Ultra Stable Oscillator