ENF0000AN003

Accelerometer Guide

Application Note

Version: 1.03

NOVATEL WIRELESS

02 December, 2013

General

TERMS OF USE OF NEW MATERIALS - PLEASE READ CAREFULLY

From time to time, Novatel Wireless M2M, in its sole discretion, may make available for download on its website (www.novatelwireless.com), or may transmit via mail or email, updates or upgrades to, or new releases of, the firmware, software or documentation for its products (collectively, 'New Materials'). Use of such New Materials is subject to the terms and conditions set forth below, and may be subject to additional terms and conditions as set forth in Novatel Wireless M2M's Technical Support Policy (posted on its website) and/or any written agreement between the user and Novatel Wireless M2M.

All New Materials are provided AS IS. Novatel Wireless M2M makes no warranty or representation with respect to the merchantability, suitability, functionality, accuracy or completeness of any such New Materials. The user of such New Materials assumes all risk (known or unknown) of such use. Novatel Wireless M2M reserves all rights in such New Materials. The user shall have only a revocable and limited license to use such New Materials in connection with the products for which they are intended. Distribution or modification of any New Materials without Novatel Wireless M2M's consent is strictly prohibited.

IN NO EVENT WILL Novatel Wireless M2M BE RESPONSIBLE FOR ANY INCIDENTAL, INDIRECT, CONSEQUENTIAL OR SPECIAL DAMAGES AS A RESULT OF THE USE OF ANY NEW MATERIALS. Novatel Wireless M2M'S MAXIMUM LIABILITY FOR ANY CLAIM BASED ON THE NEW MATERIALS SHALL NOT EXCEED FIFTY U.S. DOLLARS ($50).

Version Verification

Please ensure you have the latest version of this document by downloading it from www.novatelwireless.com

Copyright

251 Renner Pkwy

Richardson, TX 75080 USA

Phone: (972) 633-4400

Fax: (972) 633-4444

Email: info@nvtl.com

www.novatelwireless.com

Table of Contents

1

Introduction

Accelerometer

Accelerometer Calibration

Detect Any Motion

Accelerometer

This document covers Novatel Wireless M2M Spider MT devices with an installed accelerometer, with the exception of the MT 3000 and the MT 3050 product lines which are covered in their applicable User Guides.

Overview:

This document explains the practical uses of the Accelerometer, which include features, calibration, and sample event scripting.

Features:

The Accelerometer allows users to easily integrate motion-based functions into the Novatel Wireless M2M event engine. The following motion features are supported:

- Detect general motion (on any axis)
- Acceleration in forward or reverse direction
- Rapid acceleration or de-acceleration

Accelerometer Calibration

Auto-Calibration

Novatel Wireless M2M introduced an accelerometer feature in PKG 51 to configure the device to auto-calibrate using the following AT Commands:

- AT$AUTOCFG - Used to configure the auto-calibration parameters
- AT$ACCAC - Used to enable/disable auto-calibration

By default auto-calibration is enabled, if auto-calibration has been disabled, then enable it and save this configuration into non-volatile memory by entering the following commands:

```
AT$ACCAC=1
```

```
AT&W
```

You can query the state of auto calibration by sending the following command:

```
AT$ACCAC?
```

For new units (that do not have a previous calibration), you can configure automatic calibration by performing the following steps:

1. Park the vehicle on a flat surface.
2. Install the device in a vehicle.
3. Power on the vehicle and device.
   The device will begin to auto-calibrate.
4. Wait at least 30 seconds for the static calibration to finish before moving the vehicle

When calibrating a device using AT$ACCAC=1, the device must get GPS lock before proceeding with calibration.

5. Drive the vehicle in a forward direction to a speed exceeding the threshold set in AT$AUTOCFG
6. Decelerate in a straight line.

The device will repeat attempts to calibrate until a successful calibration has been accomplished. After the device calibrates, it will not attempt to calibrate again until external power has been completely lost and re-applied again. At this point, it will automatically calibrate again as long as AT$ACCAC=1 is stored in the device.

For vehicles that already have a calibration prior to setting the AT$ACCAC=1 command, you should erase the old calibration data by sending AT$ACCMGC=0,3.

Novatel Wireless M2M strongly recommends using the default settings of AT$AUTOCFG. See the AT Command Manual for your device for details.

Detect Any Motion

Any motion monitors all 3 axes and looks for any motion to exceed threshold and count. It is enabled by default and configured using AT$ACCAM command.

Threshold: is an integer value from 0-255, with 0 being most sensitive and 255 being least sensitive.

Count: is the number of times in a row the threshold is exceeded (based on sampling) before declaring motion.

Hold Time: is the amount of time in seconds of no motion before declaring motion is cleared. This uses input event 150 to trigger an event.

If AT$ACCAM settings are changed from the default, then the accelerometer will be removed from the GPS filtering function.

Sample script:

```
AT$ACCAM=1,10,1,120              // Any motion enabled, threshold of 10, count of 1 and hold
                                 time of 120
AT$EVENT=10,0,150,1,1            // if motion start is detected
AT$EVENT=10,3,40,,10,3802087     // send UPD message to IP in $FRIENDS list
AT$EVENT=10,0,150,0,0            // if motion clear event is detected
AT$EVENT=10,3,40,,10,3802087     // send UPD message to IP in $FRIENDS list
```

Detect Motion Of Specific Axis

Detecting specific motion events, such as rapid acceleration, require a finer level of detail to determine a change in states. An explanation of the AT commands and settings follows:

AT$ACCCFG - Accelerometer Configuration Settings

Range: is the overall range the accelerometer looks for in determining motion, +/- 4 Gs is the default.

Bandwidth: is how often a reliable reading can be taken. This is measured in hertz (25 is default, which is approx. every 40 mSecs)

Mode: defines the state of the accelerometer (awake or sleep).

Wakeup pause: how often the accelerometer wakes up to check for motion. Default is every 20 mSeconds.

AT$ACCSAM - Accelerometer Sampling

AT$ACCSAM configures sampling and coefficient of the accelerometer.

Sample rate is the number of sample reads per second of the accelerometer. The default is 25.

Coefficient, similar to FIR (Finite Impulse Response), is the number of samples. The reading is averaged over to “smooth” out the spikes in acceleration. The lower the number, the faster the filter. The default is 10.

Additionally, you can configure triggers for each individual axis. You can set up to 2 filters for each axis. (4 for the X axis). You can set up filters for opposing directions (like front and back or left and right), or for 2 triggers in the same direction, such as forward motion or rapid acceleration.

AT$ACCFLT - Accelerometer Filter Parameters

The filters are defined in AT$ACCFLT command.

Filter number: 1-8 (X1, X2, Y1, Y2, Z1, Z2, X3, X4).

Threshold: The amount of acceleration that must be exceeded before the filter is considered exceeded (plus hysteresis).

Duration: The number of samples above the threshold before the exceeded threshold event is valid.

Hysteresis: The number of milliGs (default 25) that the acceleration must exceed if velocity is increasing, or decline if decreasing, to determine a change in state (acceleration exceeded or acceleration not exceeded.)

AT$ACCEL? - Accelerometer Filter Parameters

AT$ACCEL? allows the user to query the current running average of acceleration on each axis

Sample Script:

```
AT$ACCCFG=0,1,0,0          // range of +/- 2G, bandwidth 50, Accelerometer normal, wakeup pause=20 mSeconds
AT$ACCSAM=25,10            // Sample rate of 25 , filter coefficient of 10
AT$ACCFLT=1, 250,5,25,10   //set X axis filter forward direction, 250 milliG threshold, 5 samples, 25 miliG hys. 10 coef.
AT$ACCFLT=2,-250,5,25,10   //set X axis filter reverse direction, 250 milliG threshold, 5 samples, 25 miliG hys. 10 coef.
AT$ACCFLT=3, 250,5,25,10   //set Y axis filter left turn, 250 milliG threshold, 5 samples, 25 miliG hys. 10 coef.
AT$ACCFLT=4,-250,5,25,10   //set Y axis filter right turn, 250 milliG threshold, 5 samples, 25 miliG hys. 10 coef.
AT$EVENT=55,0,148,1,1      // Accelerometer Filter 1 limit exceeded acceleration
AT$EVENT=55,3,40,20,3801087 // send UPD message to IP in $FRIENDS list
AT$EVENT=56,0,148,0,0      // Filter 1 motioned cleared
AT$EVENT=56,3,40,21,3801087 // send UPD message to IP in $FRIENDS list
AT$EVENT=57,0,149,1,1      // Accelerometer Filter 2 limit exceeded deceleration
```

```
AT$EVENT=57,3,40,22,3801087          // send UPD message to IP in $FRIENDS list
AT$EVENT=58,0,149,0,0                // Filter 2 motioned cleared
AT$EVENT=58,3,40,23,3801087          // send UPD message to IP in $FRIENDS list
AT$EVENT=59,0,151,1,1                // Accelerometer Filter 3 limit exceeded
                                     left turn
AT$EVENT=59,3,40,23,3801087          // send UPD message to IP in $FRIENDS list
AT$EVENT=60,0,151,0,0                // Accelerometer Filter 3 motion cleared
AT$EVENT=60,3,40,23,3801087          // send UPD message to IP in $FRIENDS list
AT$EVENT=61,0,152,1,1                // Accelerometer Filter 4 limit exceeded right turn
AT$EVENT=61,3,40,23,3801087          // send UPD message to IP in $FRIENDS list
AT$EVENT=62,0,152,0,0                // Accelerometer Filter 4 motion cleared
AT$EVENT=62,3,40,23,3801087          // send UPD message to IP in $FRIENDS list
```

Please refer to the AT Command Reference Document applicable to the device for detailed information about accelerometer AT commands.

2

Accelerometer Event Logging

Accelerometer Filter Configuration

Novatel Wireless M2M devices with an accelerometer have 8 filters that can be used to trigger event processing. The 8 filters and their associated input events are as follows:

Filter number	Input event	Axis
1	148	X
2	149	X
3	151	Y
4	152	Y
5	153	Z
6	154	Z
7*	203	X
8*	204	X

* Not available in older firmware versions. To verify the filters supported by your device enter the command:

```
AT$ACCFLT=?
```

The response will indicate which filter numbers are available.

The filter parameters are set by the $ACCFLT command:

```
AT$ACCFLT=<Filter No>,<Threshold>,<Duration>,<Hysteresis>,<Coef>
```

The filter configuration allows the user to trigger the input event when the filtered acceleration value exceeds the threshold for that filter.

The parameters that control the filter are as follows:

Threshold	The trigger point for the filter (modified by the hysteresis)
Duration	How long the filter value has to exceed the threshold before the event is triggered
Hysteresis	Modifies the threshold (see below)
Coefficient	Controls the low pass filtering applied to the raw acceleration measurement

The hysteresis modifies the threshold as follows:

- When the event is not set, the filtered acceleration value must exceed the <threshold> + <hysteresis> to set the event.
- When the event is set, the filtered acceleration value must go below the <threshold> - <hysteresis> to clear the event. The hysteresis can be set to zero to disable it.

The duration is similar to the hysteresis but is based on time. The filtered acceleration value must exceed the <threshold> + <hysteresis> for duration number of samples before the event will be set. For clearing the event, the filtered acceleration value must fall below the <threshold> - <hysteresis> for duration samples. The duration can be disabled by setting it to 1.

The coefficient controls low pass filtering. The coefficient controls how much of the change in acceleration is applied to the filtered value. Setting the coefficient to 1 effectively disables the low pass filter.

Accelerometer Event Log Configuration

Accelerometer event logging consists of several parts:

- An input event
- An output event
- AT$ACCFLT *
- AT$ACCELC

Event logging is configured with the $ACCELC command:

```
AT$ACCELC: <logging filter>, <trigger filter>, <magnitude
threshold>, <seconds before>, <seconds after>
```

* Not available in older firmware versions. To verify the filters supported by your device enter the

command:

AT$ACCFLT=?

The parameters are:

Logging filter

This parameter sets the accelerometer filter that is used as the source of accelerometer data that is logged. The logging filter parameter is the number of the filter for which to save the acceleration data. The acceleration data is processed by the filter settings from the $ACCFLT command for that filter. The filtered acceleration data is saved when the output event (output event #154) is triggered. This output event can be used the same way as any other output event. When the output event is triggered, the device starts an internal timer for 'seconds after' amount of time. When this timer expires, acceleration and GPS data that was collected over the time interval of 'seconds before' + 'seconds after' is written to a file in the Flash File System (FFS). If the output event is triggered again during the 'seconds after' interval, it is ignored by the accelerometer event logging.

Trigger filter

This parameter sets the accelerometer filter that is used for the magnitude trigger. The trigger filter parameter specifies the number of the filter that will be used for a magnitude trigger. The magnitude trigger is done by comparing the magnitude of the X and Y acceleration values to the magnitude threshold. The units of the magnitude threshold is milli-G (acceleration in G x 1000). When this threshold is exceeded, the input event for the trigger filter (see table 1) is set. One difference between this input event and a normal input event is that this input event can be used only as an occurrence event type. This input event is only set but never cleared. The filter that is selected for use as the 'trigger filter' cannot be used for the normal filter triggering. The input event for that filter becomes the input event for the magnitude trigger.

Magnitude threshold

This parameter sets the threshold used for the trigger filter when determining if the input event should be triggered.

Seconds before

This is the number of seconds before the output event is triggered to store accelerometer & GPS data.

Seconds after

This is the number of seconds after the output event is triggered to store accelerometer & GPS data.

The X and Y magnitude is calculated by the following equation:

MagXY = square root (X2 + Y2)

Where:

- X is the value of the acceleration in the X axis.
- Y is the value of the acceleration in the Y axis.
- MagXY is the magnitude of the X and Y acceleration values.

Since the MagXY calculation is dependent on the correct accelerometer orientation calibration, the input event for the magnitude trigger will not be set unless the orientation calibration has been completed. For more information on the orientation calibration, see the AT$ACCAC command.

Typical Usage

The logging filter and the trigger filter should normally be the same. This means that the acceleration data used for generating the input event trigger will be the same data that is logged when the output event is triggered. It is possible to use other input events to trigger the event logging output event. In this case, the user can disable the magnitude trigger by setting the trigger filter to 0 (using the AT command AT$ACCELC). If the magnitude trigger is 0 (disabled), all the acceleration filters are available for their normal use.

Example 1

The most common use of the trigger and/or logging filter would be to re-purpose either filter 5 or filter 6 (the Z axis filters) for use as the trigger or logging filter. A typical usage might be as follows:

```
AT$ACCFLT= 6,,1,0,1

AT$ACCELC= 6, 6, 1750,
15, 10

AT$EVENT= 10, 1, 154,
1, 1

AT$EVENT= 10, 3, 154,
10, 0
```

This set of parameters will configure the device to do the following:

Parameters for the AT$ACCFLT command. (AT$ACCFLT= 6,,1,0,1)

Set 6 as the logging filter number.

Do not set the Threshold.

Set the filter duration to 1.

Set the hysteresis to 0.

Set the coefficient to 1.

This effectively disables the filtering of the data.

Parameters for the AT$ACCELC command. (AT$ACCELC= 6, 6, 1750, 15, 10)

AT$ACCELC=<Filter No>,<Trigger Filter No>,<Magnitude Threshold>,<Seconds Before>,<Seconds After>

Set 6 as the trigger filter number.

The filter parameters (set using the previous command) apply to the acceleration data used to compare to the magnitude threshold. A duration of 2 means the MagXY must exceed the magnitude threshold for 2 consecutive samples before the input event is set.

Set the magnitude threshold to 1750 milli-Gs.

When the MagXY exceeds this value, the input event for filter #6 will be set (input event 154).

Set the 'seconds before' to 15. When the output event is triggered, the data for the previous 15 seconds will be saved in the event log file.

Set the 'seconds after' to 10. When the output event is triggered, the data for the following 10 seconds will be saved in the event log file.

This will result in approximately 25 seconds of accelerometer and GPS data being saved into the log file.

Example 2

```
AT$ACCFLT= 5,,2,0,1

AT$ACCELC= 5, 5, 1500,
30, 30

AT$EVENT= 10, 1, 153,
1, 1

AT$EVENT= 10, 3, 154,
10, 0
```

This set of parameters will configure the device to do the following:

Parameters for the AT$ACCFLT command. (AT$ACCFLT= 5,,2,0,1)

AT$ACCFLT=<Filter No>,<Threshold>,<Duration>,<Hysteresis>,<Coef>

Set 5 as the logging filter number.

Do not set the Threshold.

Set the filter duration to 2.

Set the hysteresis to 0.

Set the coefficient to 1.

Parameters for the AT$ACCELC command. (AT$ACCELC= 5, 5, 1500, 30, 30)

AT$ACCELC=<Filter No>,<Trigger Filter No>,<Magnitude Threshold>,<Seconds Before>,<Seconds After>

Set 5 as the trigger filter number.

The filter parameters (set using the previous command) apply to the acceleration data used to compare to the magnitude threshold. A duration of 2 means the MagXY must exceed the magnitude threshold for 2 consecutive samples before the input event is set.

Set the magnitude threshold to 1500 milli-Gs.

When the MagXY exceeds this value, the input event for filter #5 will be set (input event 153).

Set the 'seconds before' to 30. When the output event is triggered, the data for the previous 30 seconds will be saved in the event log file.

Set the 'seconds after' to 30. When the output event is triggered, the data for the following 30 seconds will be saved in the event log file.

This will result in approximately 60 seconds of accelerometer and GPS data being saved into the log file.

Example 3

```
AT$ACCFLT= 6,,1,0,5

AT$ACCELC= 6, 0, , 30,
30

AT$EVENT= 10, 0, 21, 1,
1

AT$EVENT= 10, 3, 154,
10, 0
```

This set of parameters will configure the device to do the following:

Parameters for the AT$ACCFLT command. (AT$ACCFLT= 6,,1,0,5)

AT$ACCFLT=<Filter No>,<Threshold>,<Duration>,<Hysteresis>,<Coef>

Set 6 as the logging filter number.

Do not set the Threshold.

Set the filter duration to 1.

Set the hysteresis to 0.

Set the coefficient to 5.

Using the coefficient of 5 will apply some low pass filtering to the logged data.

Parameters for the AT$ACCELC command. (AT$ACCELC= 6, 0, , 30, 30)

AT$ACCELC=<Filter No>,<Trigger Filter No>,<Magnitude Threshold>,<Seconds Before>,<Seconds After>

Set 0 as the trigger filter number.

This disables the magnitude triggering.

Since the magnitude triggering is disabled, the threshold is ignored.

Set the 'seconds before' to 10. When the output event is triggered, the data for the previous 10 seconds will be saved in the event log file.

Set the 'seconds after' to 0. No data after the trigger will be saved in the event log file.

The input event used to trigger this event group is Geofence #1. (Set up of the geofence is not shown here.) This means when the geofence event is set, the accelerometer and GPS data logging will be triggered. This will result in approximately 10 seconds of accelerometer and GPS data being saved into the log file when the geofence area is entered.

Event Log File

When the event logging output event is triggered, a event log file will be created. The event log files will be named 'accevt_0', 'accevt_1', ...'accevt_4'. Up to 5 event log files can be created. If all 5 files are present and the output event is triggered, no new log files will be generated, and the existing files will remain unmodified. It is up to the user to delete the files to make room for new event log files.

The user can retrieve the event log files from the modem in two ways: with FTP or with the $FFS commands. The $FFS commands can be used to list the files currently on the device:

```
AT$FFS=4

$FFS:

7188 accevt_0
```

```
7274 accevt_1

7378 accevt_2

7360 accevt_3

OK
```

This listing shows 4 files on the modem, so there is room for one more. The files can be read with the $FFS command:

```
AT$FFS=1,"accevt_0",0,64

$FFS:

2000000114210400B13202000200000030000009A0000009A0000000C061B0E2831
000014000003504554455F4C4F434F000000000000000000000000C000002

OK
```

This command reads the first 64 bytes from the file ‘accevt_0’. To read the next 64 bytes, use the following:

```
AT$FFS=1,"accevt_0",64,64

$FFS:

5E01000006060F0F19040000040000040A000101C800000603C5FFFF140155FC6DFE
2CC5FFFF130153FC75FE56C5FFFF130153FC75FE80C5FFFF130153FC75FE

OK
```

The entire file can be read this way. To delete a file after reading it, use the following:

```
AT$FFS=2,"accevt_0"

OK
```

When the output event is triggered, the event log file will use an available name for the event data.

To use FTP to send the files to an FTP server, see the descriptions of the AT$FTP commands in the AT command document for your device.

Event Log Data File Format

The event log uses a data format based on the tag-length-value (TLV) form. Each data item in the data file is preceded by the tag and length of the item. The tag indicates which data item follows, and the length specifies the number of bytes following the tag and length. The tag and length are packed into four bytes—one byte for the tag and three bytes for the length. The three bytes are the length (LSB first), and the next byte is the tag. All values in the log file are 'little-endian'.

The tags that have been defined are:

Tag name	Tag value	Data length	struct
GPS data	7	22	gps_bin_data_t
Acceleration data	6	10 * N	acc_bin_data_t
Trigger info	1	32	evt_trigger_info_t
Event log configuration	2	12	evt_log_param_hdr_t
Filter log configuration	4	4	evt_log_filt_t
Trigger log configuration	5	4	evt_log_filt_t
Modem ID	3	20	modem_id_t
Debug tags	128 – 254	various	-
End tag	255	0	-

For the acceleration data item, there can be multiple records per data item. The length of the data item should always be a multiple of the data length (10 bytes for each acceleration data record). Also note that the end tag has a length of 0; no data will follow this tag. The tag-length bytes for the end tag will be the last bytes in the file.

Any software that processes the event logging data files should not depend on the order the tags appear in the file or on specific tags being present. This will help prevent future problems if there are changes (or tags are deprecated or added) to the data files.

The debug tags are reserved for Novatel Wireless use. You should use the data length to skip over those bytes. These tags will not normally be present in the event data log file, but Novatel Wireless engineering may enable some debug info to troubleshoot issues.

'C' Struct Definitions For The Data Records

GPS Logging Data Record

```
typedef struct
{
    int32_t tick;                   // millisecs before/after trigger
    uint32_t ulTime;                // GPS time
    int32_t lLatitude;              // latitude
    int32_t lLongitude;             // longitude
    uint16_t usSpeed;               // GPS speed
    uint16_t usHeading;             // heading
    int8_t cStatus;                 // GPS fix status
    int8_t spare;                   // pad to make even # of bytes
} gps_bin_data_t;
```

Accelerometer Logging Data Record

```
typedef struct
{
    int32_t tick;                   // millisecs before/after trigger
    int16_t x;                      // Filtered & transformed X
    int16_t y;                      // Filtered & transformed Y
    int16_t z;                      // Filtered & transformed Z
} acc_bin_data_t;
```

Trigger Info Record

```
typedef struct
```

```
{
    int32_t gps_date;          // GPS date at trigger (MMDDYY)
    int32_t gps_time;          // GPS time at trigger (HHMMSS)
    int32_t parm1;             // parm1 from event group
    int32_t parm2;             // parm2 from event group
    int32_t in_evt;            // input event from event group
    int32_t out_evt;           // output event from event group
    uint8_t rtc_year;          // years - [00,99]
    uint8_t rtc_month;         // months - [01,12]
    uint8_t rtc_day;           // day of the month - [1,31]
    uint8_t rtc_hour;          // hours after the midnight - [0,23]
    uint8_t rtc_min;           // minutes after the hour - [0,59]
    uint8_t rtc_sec;           // seconds after the minute - [0,59]
    uint8_t spare1;
    uint8_t spare2;
} evt_trigger_info_t;
```

Event Logging Parameters Record

```
typedef struct
{
    int32_t thresh;            // magnitude threshold used
    uint8_t logFiltNum;        // filter # of log filter
    uint8_t trigFiltNum;       // filter # of trigger filter
    uint8_t prior;             // seconds prior to trigger
    uint8_t after;             // accelerometer sample rate
    uint8_t sampleRate;        // seconds after trigger
    uint8_t accCalState;       // accelerometer orientation cal
                               state
```

```
    uint8_t spare1;

    uint8_t spare2;

} evt_log_param_hdr_t;
```

Filter Parameters Record

Used for both the trigger and logging filter records.

```
typedef struct

{

    uint16_t hyst;                  // filter hysteresis

    uint8_t duration;               // filter duration
    uint8_t coef;                   // filter coefficient

} evt_log_filt_t;
```

Modem ID

```
typedef struct

{

    uint8_t modem_id[20];           // up to 20 chars, not always nul
                                    term

} modem_id_t;
```

Notes On The ‘C’ Structs:

The ‘sizeof’ for the AccEvtData_t would return 12, due to the padding most C compilers add to the struct. The data file implementation intentionally omits these padding bytes to reduce the size of the data file, since there are generally many accelerometer data records in the file. Any parsing of this data file must take the missing padding into account. The ‘sizeof’ for the AccGpsBinData_t struct would return 24; the extra two bytes of padding are omitted from the data file for this struct as well. For the other structs, the padding is left in place.

Example Log Files

Hexadecimal

This is an example of a log file dumped in hexadecimal, with some of the repetitive data omitted:

```
0000000:  20 00 00 01 04 fa 03 00 93 6e 03 00 02 00 00 00  ........n......
0000010:  03 00 00 00 9a 00 00 00 9a 00 00 00 0c 06 1a 16  ................
0000020:  31 0f 00 00 14 00 00 03 50 45 54 45 5f 4c 4f 43 1  ........PETE_LOC
0000030:  4f 00 00 00 00 00 00 00 00 00 00 00 0c 00 00 02 O  ................
0000040:  5e 01 00 00 06 06 0f 0f 19 04 00 00 04 00 00 04  ^...............
0000050:  0a 00 01 01 c8 00 00 06 8b c4 ff ff 85 fe 18 00  ................
0000060:  a0 03 b4 c4 ff ff 38 fe f7 ff 88 03 ec c4 ff ff  ......8.........
0000070:  52 fe 0b 00 78 03 15 c5 ff ff 95 fe 33 00 0b 04  R...x.......3...
0000080:  a4 c5 ff ff a5 fe 17 00 fd 03 0a c6 ff ff 6f fe  ..............o.
0000090:  48 00 32 03 34 c6 ff ff 8a fe 5a 00 d4 03 5d c6  H.2.4.....Z...].
00000a0:  ff ff 6a fe 1b 00 d0 03 87 c6 ff ff 78 fe 06 00  ..j.........x...
00000b0:  8a 03 b0 c6 ff ff 39 fe 23 00 23 03 da c6 ff ff  ......9.#.#.....
00000c0:  58 fe 27 00 89 03 1a c7 ff ff 71 fe 32 00 7e 03  X.'.......q.2.~.
00019a0:  56 00 01 00 16 00 00 07 14 dd ff ff 8a 6e 03 00  V............n..
00019b0:  c3 90 f7 01 aa 98 40 fa a3 00 62 00 01 00 16 00  ......@...b.....
00019c0:  00 07 1e e1 ff ff 8b 6e 03 00 eb 90 f7 01 b3 98  .......n........
00019d0:  40 fa 99 00 6e 00 01 00 16 00 00 07 d0 e4 ff ff  @...n...........
00019e0:  8c 6e 03 00 0f 91 f7 01 ba 98 40 fa 8f 00 77 00  .n........@...w.
00019f0:  01 00 16 00 00 07 c3 e8 ff ff 8d 6e 03 00 31 91  ...........n..1.
0001a00:  f7 01 c1 98 40 fa 79 00 75 00 01 00 16 00 00 07  ....@.y.u.......
0001a10:  c8 ec ff ff 8e 6e 03 00 4c 91 f7 01 c6 98 40 fa  .....n..L.....@.
0001bc0:  00 00 00 00 01 00 16 00 00 07 e9 2e 00 00 9f 6e  ...............n
0001bd0:  03 00 a8 91 f7 01 df 98 40 fa 00 00 00 00 01 00  ........@.......
0001be0:  16 00 00 07 dc 32 00 00 a0 6e 03 00 a8 91 f7 01  .....2...n......
0001bf0:  df 98 40 fa 00 00 00 00 01 00 16 00 00 07 dc 36  ..@............6
```

```
0001c00:    00 00 a1 6e 03 00 a8 91 f7 01 df 98 40 fa 00 00    ...n........@...
0001c10:    00 00 01 00 16 00 00 07 d8 3a 00 00 a2 6e 03 00    .........:...n..
0001c20:    a8 91 f7 01 df 98 40 fa 00 00 00 00 01 00 00 00    ......@.........
0001c30:    00 ff                                              ..
```

By Data Item

Here is the same file dumped by data item. Note that the parts omitted are not exactly the same.

```
Read 7218 bytes from accevt_0

Dump item: tag = 1 length = 32
20 00 00 01                                                   | ................|

04 fa 03 00 93 6e 03 00 02 00 00 00 03 00 00 00               |.....n..........|
9a 00 00 00 9a 00 00 00 0c 06 1a 16 31 0f 00 00               |............1...|


Dump item: tag = 3 length = 20
14 00 00 03                                                   | ................|

50 45 54 45 5f 4c 4f 43 4f 00 00 00 00 00 00 00               |PETE_LOCO.......|
00 00 00 00                                                   |................|


Dump item: tag = 2 length = 12
0c 00 00 02                                                   | ................|

5e 01 00 00 06 06 0f 0f 19 04 00 00                           |^...............|


Dump item: tag = 4 length = 4
04 00 00 04                                                   | ................|

0a 00 01 01                                                   |................|


Dump item: tag = 6 length = 200
```

```
c8 00 00 06                                                  |................|

8b c4 ff ff 85 fe 18 00 a0 03 b4 c4 ff ff 38 fe              |..............8.|

f7 ff 88 03 ec c4 ff ff 52 fe 0b 00 78 03 15 c5              |........R...x...|
ff ff 95 fe 33 00 0b 04 a4 c5 ff ff a5 fe 17 00              |....3...........|

fd 03 0a c6 ff ff 6f fe 48 00 32 03 34 c6 ff ff              |......o.H.2.4...|

8a fe 5a 00 d4 03 5d c6 ff ff 6a fe 1b 00 d0 03              |..Z...]...j.....|
87 c6 ff ff 78 fe 06 00 8a 03 b0 c6 ff ff 39 fe              |....x.........9.|
23 00 23 03 da c6 ff ff 58 fe 27 00 89 03 1a c7              |#.#.....X.'.....|

ff ff 71 fe 32 00 7e 03 44 c7 ff ff 59 fe 44 00              |..q.2.~.D...Y.D.|

eb 03 6d c7 ff ff 4f fe 27 00 fa 03 97 c7 ff ff              |..m...O.'.......|

81 fe 60 00 dd 03 c0 c7 ff ff 42 fe 53 00 ed 03              |..`.......B.S...|
ea c7 ff ff 81 fe 28 00 de 03 21 c8 ff ff 74 fe              |......(...!...t.|
30 00 bf 03 4b c8 ff ff aa fe 50 00 c4 03 74 c8              |0...K.....P...t.|
ff ff 06 fe 1e 00 4f 03                                      |......O.........|
```

Dump item: tag = 6 length = 200

```
c8 00 00 06                                                  |................|
ec 37 00 00 f1 ff 07 00 eb 03 16 38 00 00 f1 ff              |.7.........8....|

07 00 eb 03 40 38 00 00 ef ff fe ff ee 03 69 38              |....@8........i8|

00 00 f1 ff 0a 00 e4 03 93 38 00 00 f1 ff 07 00              |.........8......|

eb 03 bc 38 00 00 ef ff 03 00 e1 03 f4 38 00 00              |...8.........8..|

f1 ff 07 00 eb 03 1d 39 00 00 f8 ff 05 00 ea 03              |.......9........|
47 39 00 00 f1 ff 07 00 eb 03 70 39 00 00 f1 ff              |G9........p9....|

0a 00 e4 03 9a 39 00 00 f1 ff 07 00 eb 03 c3 39              |.....9.........9|
00 00 f1 ff 07 00 eb 03 0d 3a 00 00 f1 ff 07 00              |.........:......|

eb 03 37 3a 00 00 f1 ff 07 00 eb 03 c1 3a 00 00              |..7:.........:..|
ef ff 00 00 e7 03 eb 3a 00 00 ef ff 00 00 e7 03              |.......:........|

14 3b 00 00 f1 ff 07 00 eb 03 3e 3b 00 00 f1 ff              |.;........>;....|

04 00 f2 03 67 3b 00 00 e7 ff 00 00 ef 03 91 3b              |....g;.........;|
```

```
00 00 e7 ff 02 00 e8 03                                 |................|
```

```
Dump item: tag = 7 length = 22
16 00 00 07                                             |................|

0a c6 ff ff 84 6e 03 00 4b 8f f7 01 61 98 40 fa         |.....n..K...a.@.|
4f 01 20 00 01 00                                       |O. ..............|
```

```
Dump item: tag = 7 length = 22
16 00 00 07                                             |................|

d8 3a 00 00 a2 6e 03 00 a8 91 f7 01 df 98 40 fa         |.:...n........@.|
00 00 00 00 01 00                                       |................|
```

```
Dump item: tag = 255 length = 0
```

Readable Format

Here is the same file dumped in readable format:

```
Read 7218 bytes from accevt_0
```

Trigger info:	Value
GPS date(YYMMDD):	12/06/26
GPS time(HHMMSS):	22:49:15
Event parm1:	2
Event parm2:	3
Input event:	154
Output event:	154
RTC date (YYMMDD):	12/06/26
RTC time (HHMMSS):	22:49:15
Modem ID:	'PETE_LOCO'

Event configuration:	Value
Threshold:	350
Logging filter #:	6
Trigger filter #:	6

Event configuration:	Value
Seconds prior:	15
Seconds after:	15
Sample rate:	25
Acc Cal state:	4

Logging filter configuration:	Value
Duration:	1
Hysteresis:	10
Coefficient:	1

Accelerometer data:

Time (ms)	X	Y	Z (mG)	XYmag
-15221	-379	24	928	380
-15180	-456	-9	904	456
-15124	-430	11	888	430
...				
-535	-122	13	989	123
-494	-132	0	1008	132
-438	-122	10	997	122
-397	-129	-6	979	129
-355	-139	-18	991	140
-314	-114	5	1002	114
-272	-134	2	984	134
-231	-146	6	1003	146
-175	-211	12	998	211
-134	-249	-16	1010	250
0	-358	-13	986	358
42	-346	-17	966	346
83	-333	-6	972	333
125	-331	0	975	331
166	-320	4	995	320
208	-314	-4	990	314
249	-307	-6	989	307
309	-272	0	992	272
351	-250	0	975	250
392	-240	1	984	240

Time (ms)	X	Y	Z (mG)	XYmag
434	-225	-5	989	225
475	-200	-8	996	200
517	-80	0	1026	80
…				
14861	-15	7	1003	17
14903	-15	7	1003	17
15041	-17	0	999	17
15083	-17	0	999	17
15124	-15	7	1003	17
15166	-15	4	1010	16
15207	-25	0	1007	25
15249	-25	2	1000	25

GPS data:

Time (ms)	GPS time (HHMMSS)	Lat (deg)	Long (deg)	Speed (KPH)	Hdg (deg)	Status
-14838	22:48:00	+33001291	-096429983	33.5	3.2	1
-13855	22:48:01	+33001379	-096429975	30.0	4.8	1
-12913	22:48:02	+33001450	-096429964	26.3	6.2	1
-11898	22:48:03	+33001517	-096429950	23.8	6.9	1
-10924	22:48:04	+33001575	-096429936	21.4	8.4	1
-9983	22:48:05	+33001627	-096429921	17.4	8.6	1
-8940	22:48:06	+33001667	-096429910	16.3	9.8	1
-7906	22:48:07	+33001707	-096429901	15.3	11.0	1
-6960	22:48:08	+33001743	-096429894	14.3	11.9	1
-5949	22:48:09	+33001777	-096429887	12.1	11.7	1
-4920	22:48:10	+33001804	-096429882	10.5	12.6	1
-3881	22:48:11	+33001830	-096429874	8.9	12.6	1
-2945	22:48:12	+33001851	-096429868	7.3	12.7	1
-1929	22:48:13	+33001869	-096429863	6.7	13.3	1
-969	22:48:14	+33001884	-096429859	5.6	13.6	1
37	22:48:15	+33001894	-096429856	3.7	10.9	1
1066	22:48:16	+33001896	-096429857	1.1	16.7	1
2012	22:48:17	+33001896	-096429857	0.0	0.0	1
3023	22:48:18	+33001896	-096429857	0.0	0.0	1
4048	22:48:19	+33001896	-096429857	0.0	0.0	1
5072	22:48:20	+33001896	-096429857	0.0	0.0	1
6032	22:48:21	+33001896	-096429857	0.0	0.0	1

Time (ms)	GPS time (HHMMSS)	Lat (deg)	Long (deg)	Speed (KPH)	Hdg (deg)	Status
7043	22:49:22	+33001896	-096429857	0.0	0.0	1
8072	22:49:23	+33001896	-096429857	0.0	0.0	1
9018	22:49:24	+33001896	-096429857	0.0	0.0	1
10029	22:49:25	+33001896	-096429857	0.0	0.0	1
11058	22:49:26	+33001896	-096429857	0.0	0.0	1
12009	22:49:27	+33001896	-096429857	0.0	0.0	1
13020	22:49:28	+33001896	-096429857	0.0	0.0	1
14044	22:49:29	+33001896	-096429857	0.0	0.0	1
15064	22:49:30	+33001896	-096429857	0.0	0.0	1

-- End Tag ---