ALERT2 Application Layer Protocol Specification

June 2019

Version 1.3

Draft-RevE

ALERT2 Protocol Technical Working Group of the National Hydrologic Warning Council

Revision history

Date	Version	Description	Author
1 Mar 2010	1.0	Original draft for comment	Don Van Wie
31 Aug 2010	1.0	Final Draft Version 1	Ilse Gayl
5 Apr 2011	1.1	Draft Ver 1.1 for TWG comment	Don Van Wie, R. Chris Roark
12 Jun 2012	1.2	Final	Don Van Wie
1 Nov 2018	1.3	Draft Ver 1.3 for TWG comment	Carrie Lery

Release history

Date	Version	Status	Audience	Approval
1 Sep 2010	1.0	Released to web site	Public	TWG, Ilse Gayl
27 Apr 2012	1.2	Released to web site	Public	TWG
09 July 2019	1.3	Released to web site	Public	TWG

Document management register

Document	File reference

Review Status

Reviewer	Date Reviewed	Version Reviewed

Release Signatories

Approval	Name	Signature	Date

Contact details

National Hydrologic Warning Council
2480 W. 26th Ave., Suite 156-B
Denver, CO 80211

ALERT2 Protocol Technical Working Group
of the
National Hydrologic Warning Council

Peter Acton
Ilse Gayl
David Haynes, Chair
Scott Holder
Dan Hurley
Brian Iserman
David Leader
Carrie Lery
James Logan
Steve Malers
Ron Marotto
Mark Moore
Andy Morin
Markus Ritsch
R. Chris Roark
James Slouber
Robert Thompson
Adam Torgerson
Sam Utley
David Van Wie
Don Van Wie

Contents

Figures

1 Introduction

This document contains the technical specifications for Version 1.3 of the Application Layer protocols of the ALERT2 protocol suite. ALERT2 is the next generation successor to the ALERT (Automated Local Evaluation in Real Time) protocol, widely in use for the transmission for hydrologic and meteorologic data used to support flood preparedness and public safety decision making. The ALERT2 protocol suite is optimized for the connectionless transmission of short messages by radio, and offers improved channel efficiency, greater flexibility, error detection and forward error correction, and many other features not available in ALERT.

The need to meet three primary criteria of the existing ALERT community drove the development of the ALERT2 protocol, and in particular, the AirLink Layer. These three criteria are:

1. The protocol must reside in the public domain, and not require proprietary methods or services.
2. The protocol must provide a common air interface, i.e. the "on-the air" modulation and framing is compatible with multiple brands of commercial, off-the-shelf radio transceivers readily available to manufacturers, system integrators and users.
3. The protocol must address the limitations of ALERT – primarily low channel capacity and high data loss – while providing bit and packet error rate performance equal to or better than legacy 300 bps ALERT.

This document is intended primarily for those interested in implementing the ALERT2 application layer protocols in software and hardware.

1.1 Protocol Architecture

The ALERT2 protocol suite has a three-layer architecture.

The Application Layer supports the encoding and decoding of data into and out of formats and structures used by ALERT2 applications. At the Application Protocol Device (APD), data is formed into structures understood by the receiving application software. Similarly, the MANT Protocol and AirLink Protocol devices add information to the Application data that are understood by other MANT and AirLink Protocol devices respectively. Each layer provides independent functionality and operates asynchronously to the others. Physically, all three layers may be integrated into a single device, or separated into three physical devices. When the MANT Protocol and AirLink Protocol are implemented by a single device it is referred to as an Intelligent Network Device (IND) and its interface is by the Application Layer Application Program Interface (API) specification.

The Network and Transport (MANT) layer provides the addressing, port multiplexing, acknowledgement, and other services to logically transport application and network control data across the ALERT2 radio network. When the MANT layer receives an Application Protocol Data Unit (PDU) from the Application Protocol Device, it provides the requested services, adds a header to the Application PDU to form a MANT PDU and forwards the MANT PDU to the AirLink layer, When the MANT layer receives a MANT PDU from the AirLink layer, it inspects the attached MANT Header and provides the appropriate services to the PDU, and sends the Application PDUs to the application port on the Application Protocol Device. The MANT layer exchanges information with other MANT layer devices on the network using MANT PDUs to provide network services, configuration and control.

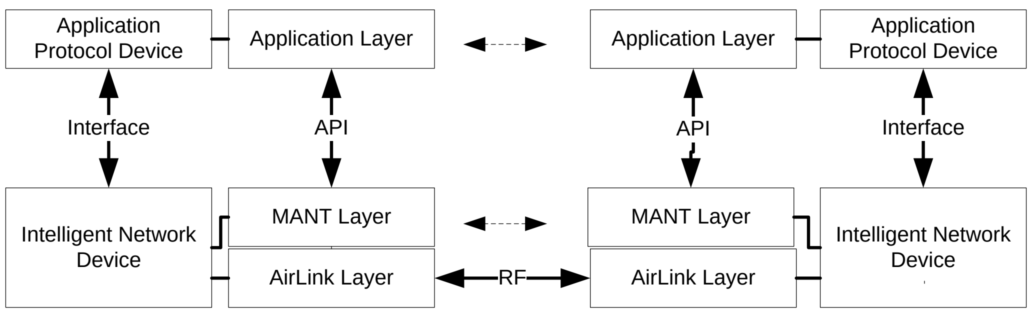

Figure 1-1 ALERT2 Physical and Logical Architecture

The AirLink Protocol modem transmits the PDUs received from the MANT layer since its last transmission. An AirLink frame is created and transmitted at a time determined by the type of media access selected and that method's configuration parameters. The AirLink Frame is created by aggregating all buffered PDUs, adding an AirLink Header, and blocking, scrambling and forward error correcting this aggregate to form an AirLink Frame Payload. The final AirLink Frame is created by pre-pending a preamble and adding a tail. The AirLink Protocol modem controls an FM transceiver's Push to Talk (PTT) as required and transforms the digital data frame into an analog signal sent to the audio input of an FM radio. An AirLink Protocol modem receives and creates MANT PDUs to send to the MANT layer device by reversing the transmission process. When an AirLink Frame is detected on the RF media, the audio waveform is converted to a bit stream, forward error correction decoded and framed into the MANT PDUs.

Figure 1-1 illustrates the flow of data through the protocol layers, and associates them with one possible physical architecture.

1.2 The Application Layer

Version 1.3 of the application layer protocol specifications initially encompasses two protocols: the Self-Reporting Protocol and the ALERT Concentration Protocol. Other Application Protocols such as an ALERT2 Device Control Protocol are likely to be added as the suite evolves.

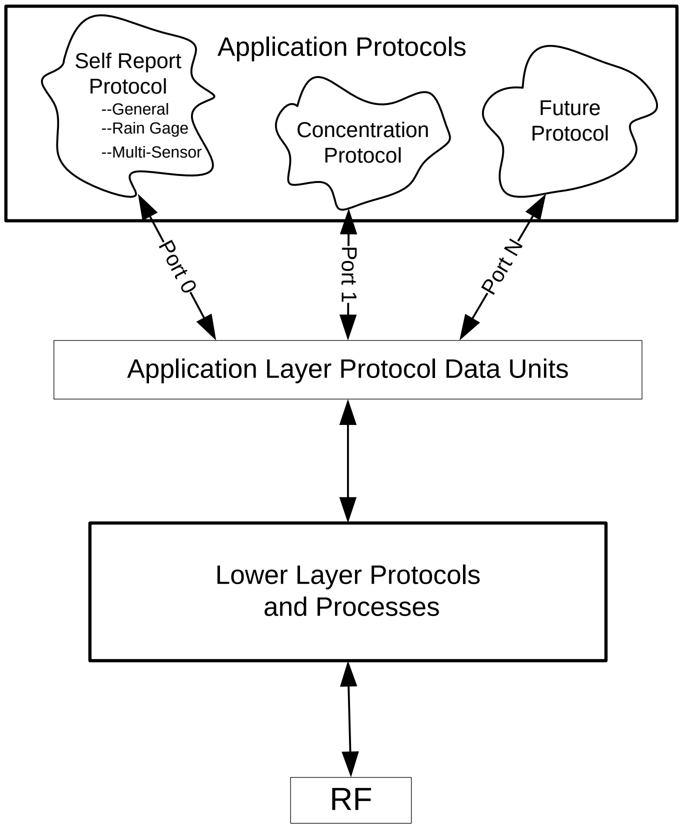

Figure 1-2 Application Layer Protocol Data and Paths

The Self-Reporting Protocol defines the structure and methods by which data, typically but not necessarily sensor data, is encoded for use by the receiving application software. The ALERT Concentration Protocol provides the structure and methods by which legacy ALERT messages can be compressed, time stamped, concatenated and packaged for delivery over the ALERT2 network.

2 ALERT2 Self-Reporting Sensor Protocol

2.1 Self-Reporting Protocol

The Self-reporting Protocol comprises the set of application layer ALERT2 Report Types. All Self-reporting PDUs are directed to Port 0 by the MANT header Port field. Reports of the same or different types may be concatenated into a single Self-reporting Message. Each embedded report uses a Type, Length, Value (TLV) structure; the Value field may recursively embed lower level TLVs, referred to as Data Elements. The following report types are defined in this document:

- Type 1: General Sensor Report
- Type 2: Tipping Bucket Rain Gage Report
- Type 3: Multi-Sensor Report – English Units
- Type 4: Multi-Sensor Report – Metric Units
- Type 5: Multi-Sensor Report – IND Sensor/Status
- Type 7: Time Series Data Report
- Type 250: SET Command
- Type 251: GET Command

Additional Report Types may be developed and added as the need is identified and they are approved by the ALERT2 coordinating body.

A complete Self-reporting Protocol Data Unit consists of the Message plus a header containing the Control Byte and an optional Timestamp; this Application Layer PDU is then sent to, or received from the MANT layer via the API. The general Self-reporting Protocol structure is illustrated below.

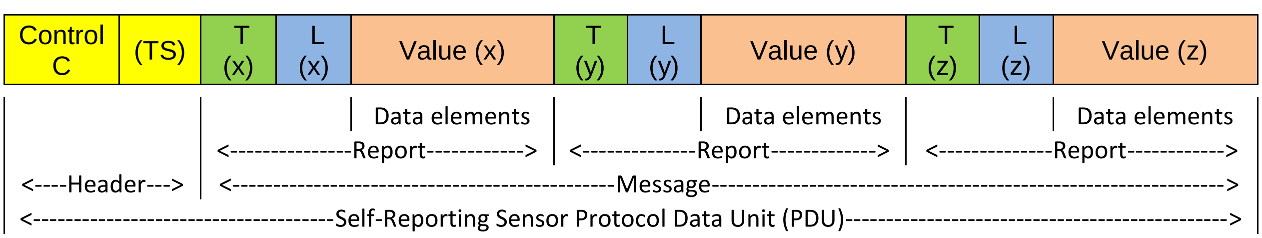

Figure 2-1 Structure and Terminology of a Self-Reporting PDU

2.1.1 Control Byte

The Control Byte consists of 8 bits pre-pended to the start of every Message. It contains flags and reserved bits used to direct the interpretation of the bytes that follow. The definition of the bits is specified below:

Field	Bytes	Bits	Purpose	Notes
Control	1			
		0,1	Version	Current version is 0
		2	Timestamp	If set,16-bit Timestamp follows Control field
		3	Test Flag	If set, report is flagged (as test data)
		4,5,6	APDU ID	Cyclic PDU ID
		7	Extensibility	If set, a second control byte follows

Figure 2-2 Application Control Byte Definition

The bit 3 flag is provided to indicate that a message is being sent as a part of site configuration or servicing and therefore the data may not be a valid part of the dataset. Treatment of the test message is at the discretion of the application decoding software.

Bits 4, 5 and 6 provide an optional Application PDU identifier. Its default value must be set to value 7, indicating it is disabled. When used, the three-bit value is incremented from 0 to 6 then back to 0. One use is to permit the receiving application software to identify duplicate reports when there are multiple paths to the destination.

2.1.2 Timestamp (TS)

The Timestamp must be a 16 bit unsigned integer containing the seconds elapsed since 12:00 AM or 12:00 PM UTC, whichever is more recent. (See discussion of UTC timestamps in the AirLink specification). It shall be the time at which the PDU was exported by the application protocol device (APD). The field is optional; if absent, the third bit of the Control Byte is set to 0. The reason for omitting the timestamp may be that this is an ALOHA media access PDU, because the APD does not carry accurate time, or that the report does not require a timestamp.

The MANT layer provides a timestamp service which can be requested via the Application Layer API. When requested, if the MANT protocol device to which the PDU is sent has accurate time, a 16 bit timestamp is inserted after the Self-Reporting Protocol Control Byte, and the Control Byte timestamp bit is set to 1.

2.1.3 Report Type (T)

The Report Type is a one byte field that identifies the Self-Reporting sub protocol under which this Message is formulated. The following Type values are recognized:

- Type 1: General Sensor Report

- Type 2: Tipping Bucket Rain Gage Report
- Type 3: Multi-Sensor Report – English Units
- Type 4: Multi-Sensor Report – Metric Units
- Type 5: Multi-Sensor Report – IND Layer
- Type 7: Time Series Data Report
- Type 250: SET Command
- Type 251: GET Command

2.1.4 Report Length (L)

The Report Length is the number of bytes in the report payload, starting with the field following the Report Length field and including all fields to, but not including, the next Report Type or, if there is none, to the end of the Report. The Report Length field is extensible: To encode a value greater than 127 requires a 2-byte field. Bit 7 (the highest bit) of the first byte sent (MSB) is set to 1, and the length value is encoded in the following 15 bits. A value of 127 or less is encoded in a single byte (whose high order bit is 0). On decoding, the MSB is read first, and if the high bit contains a 1, the value is read from the following 15 bits. If the high bit is 0, the value is read as the value of that byte. The one byte field may carry a value of 1 to 127 and the two-byte field may contain a number from 0 to 32,767. The format is shown below:

Type	MSB 7 bits Length	[LSB 8 bits Length]	data	…	data

← Length →

Figure 2-3 Extensible Length Field

2.1.5 Report Value (V)

The Report Value is contained in the number of bytes indicated by the Report Length field. The structure of this Value field varies by Report type, and may contain a recursively embedded TLV structure or Data Elements.

2.2 Sensor Report Types: General Sensor Report

This protocol is intended as a general replacement for existing ALERT sensor installations, with the exception of tipping bucket rain gages, for which there is a different report type. Multiple sensor readings may be combined into a single General Sensor Report; channel efficiency increases as multiple sensor reports are combined into a single PDU.

All General Sensor Reports are described by the following recursive TLV protocol. For illustration, 3 sensor readings are shown bundled into the upper level Value field.

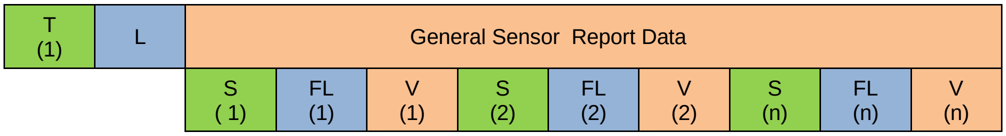

Figure 2-4 General Sensor Data Report

2.2.1 General Sensor Report Sub-Structure

- Report Type (T) — 1 byte application layer Message identifier as defined above
- Report Length (L) — 1 or 2 byte unsigned integer as defined above

- Sensor ID (S) — 1 byte
- Value Format/Length (FL) — 1 byte
- Report Value (V) — as specified in Value Format/Length

Repeating structure for optional multiple Reports

- Sensor ID (S) — optional…
- Value Format/Length (FL) — optional…
- Report Value (V) — optional…

2.2.2 Sensor ID (S)

The values between 0 and 254 of this 1-byte field are used to uniquely identify a sensor at a site. Values 0 through 8 are recommended for use with the common sensor types listed in Figure 2-5, although this is not mandatory. Values 201 through 205 are recommended for use with the sensor/status types as defined in MSR Type 5.

The reserved value of 255 indicates that what follows is a timestamp, carried in the Report Value field. This does not impact the format of the time offsets used in the Tipping Bucket Rain Gage and ALERT Concentration Reports; rather it is intended to precede historical data which follows immediately in the same Message.

ID	Sensor Type
0	Rain
1	Air Temperature
2	Relative Humidity
3	Barometric Pressure
4	Wind Speed
5	Wind Direction
6	Peak Wind Speed

7	Stage
8	Battery Voltage
201	Clock Status
203	IND Temp
204	Messages Received
205	Messages Sent
206	Status Bits

Figure 2-5 Recommended Sensor IDs

2.2.3 Value Format/Length (FL)

A single byte specifies the numeric representation of the data value and its length. The four high-order bits encode the numeric format, and the four low-order bits are used to specify the length of the data field, 0 - 15, in bytes. The following assignments have been made for the format values:

Format Values

- 0x0 Reserved
- 0x1 Unsigned integer
- 0x2 Signed integer, two's complement
- 0x3 Floating point
- 0x4 UTF-8 Encoded Character Array
- 0x5 - 0xC Reserved
- 0xD – 0xF See Figure 2-6 below for details

Format	Length (Bytes)	FL Result	Description
0x0	---	---	Reserved
0x1	1	0x11	1 byte Unsigned Integer
	2	0x12	2 byte Unsigned Integer
	3	0x13	3 byte Unsigned Integer
	4	0x14	4 byte Unsigned Integer
	8	0x18	8 byte Unsigned Integer
0x2	1	0x21	1 byte Signed Integer, two's complement
	2	0x22	2 byte Signed Integer, two's complement
	3	0x23	3 byte Signed Integer, two's complement
	4	0x24	4 byte Signed Integer, two's complement

	8	0x28	8 byte Signed Integer, two's complement
0x3	0x2	0x32	2-Byte floating point value with 'FP2' format as described in *Appendix 3 – FP2 Format*
	0x4	0x34	4-Byte floating point value formatted as IEEE 754 binary32 notation (single-precision)
	0x8	0x38	8-Byte floating point value formatted as IEEE 754 binary64 notation (double-precision)
0x4	0x1 – 0xF	0x41 – 0x4F	1 to 15 Bytes, as indicated in Length field, of UTF-8 encoded characters
0x5 – 0xC	---	---	Reserved
0xD	1	0xD1	1-Byte unsigned integer indicating seconds before transmission
0xE	2	0xE2	2-Byte unsigned integer indicating the number of seconds since 12:00 AM or PM UTC, whichever is more recent
0xF	4	0xF4	4-Byte unsigned integer indicating POSIX time, the number of seconds elapsed since midnight, UTC of January 1, 1970, not counting leap seconds.

Figure 2-6 Format/Length Recognized Combinations (Represented in Hexidecimal)

2.2.4 Report Value (V)

The Report Value is the data content. The number of bytes and format of the Report Value are specified by the preceding byte.

Not all combinations of Format and Length are recognized as valid. The formats described in the previous section shall be recognized. Unrecognized combinations shall be ignored and the Length field used to continue decoding.

2.3 Tipping Bucket Rain Gage Report

This Report specifies a compact structure for self-reporting tipping bucket rain gages. It encapsulates multiple bucket tips in a single message and improves channel utilization by reporting at a pre-set maximum frequency. The Tipping Bucket Rain Gage Report type is ‘2’ in the Self-Reporting Protocol Type Field. Only 1 Tipping Bucket Rain Gage report may follow a 2 in the Type Field. The accumulation period must be 255 seconds or less.

The Type 2 report has two data components. The first is the accumulator value of the tipping bucket at the time of the creation of the APDU. The second is a list of one-byte time offsets, one for each tip that occurred during the report period. The values in the list represent the number of seconds that elapsed between the tip event and the creation of the report. The values shall appear in descending order (i.e., the order in which the tips occurred). If no tips occurred in the reporting period, the list is omitted. By subtracting each member of the tip list from the time of the report, the time of each tip can be determined to one-second accuracy. It is recommended that the accumulator value take precedence over the tip list if there are disparities in the apparent period rainfall.

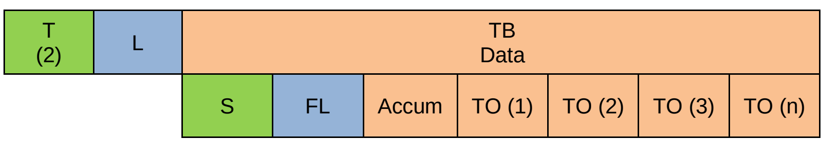

Figure 2-7 Tipping Bucket Rain Gage Report

Report Sub-Structure:

- Report Type (T) — 1 byte application layer message identifier , value 2
- Report Length (L) — 1 or 2 bytes, as defined in 2.1.4
- Sensor ID (S) — 1 byte
- Value Format/Length (FL) — 1 byte, as defined by 2.2.3
- Accumulator value (Accum) — as defined by the FL Field
- Time Offsets (TO) — 1 byte (each)

2.3.1 Sensor ID

As specified in 2.2.2.

2.3.2 Value Format/Length

As specified in 2.2.3.

2.3.3 Accumulator (Accum)

The Accumulator must be a positively increasing tipping bucket count. Its form is specified by the Format/Length field. Accumulators must roll over if they reach the end of their range, which is set by the maximum value of the data length used.

2.3.4 Time Offsets (TO)

Each Time Offset element must be a 1-byte field specifying the number of seconds before the end of reporting cycle that a tip occurred. Each byte from the end of the Accumulator field to the end of the report is a Time Offset representing a rain tip. If there are no tips (e.g., timed report), there will be no Time Offset elements.

2.4 Multi-Sensor Reports

The Multi-Sensor Report Types provide fixed formats for the most common suite of measurements made at existing ALERT sites, with significant length savings over the General Sensor Report if more than 2 sensors are reported in the same transmission. The Multi-Sensor Report format and supported sensor types are shown diagrammatically in Figure 2-6. The Report specifies 8 possible sensors to be included in a single Application PDU, with fixed data size, numerical format, data resolution and units for each measurement. Types 3 and 4 include the same suite of sensors; Type 3 is structured for English units and Type 4 is used for metric. Where a measurement does not fit this format, the General Sensor Report may be used instead.

Implementation of the Multi-Sensor Report Types is optional for encoding systems; however, if implemented it shall adhere to these specifications.

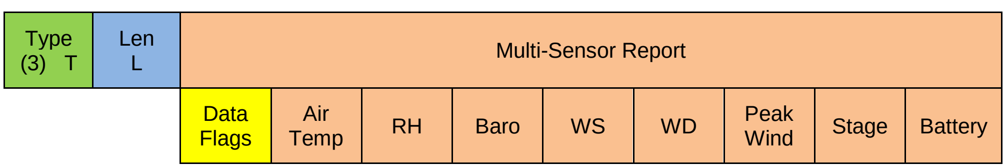

Figure 2-8 Multi-Sensor Report

2.4.1 Report Sub-Structure:

- Report Type (T) — 1 byte application layer message identifier, value 3
- Report Length (L) — 1 byte unsigned integer
- Data Flags — 1 byte representing 8 data flags
- Measurement Suite — Variable number of bytes carrying data from 8 optional sensor types

2.4.2 Report Type (T)

The value 3 indicates that this is a Multi-Sensor Report in United States customary units. The value 4 indicates the Multi-Sensor Report is in metric units. Report Type is formatted as a 1 byte unsigned integer.

2.4.3 Report Length (L)

The Report Length is specified by a 1 byte field, formatted as a 1 byte unsigned integer. The value represents the number of bytes in the report, starting with the field following the Length field.

2.4.4 Data Flags

Each of the 8 bits, when set to 1, indicates the inclusion of one of the 8 defined sensor data fields. The low order bit represents the immediately following data field (Air Temperature) and the high order bit represents the last data field (Battery Voltage).

2.4.5 Type 3 Measurement Suite

The eight included sensor types are listed below. Each field is fixed length, and must contain the number of bytes listed. The value format, measurement resolution and units are derived from the nature and typical accuracy of the measurement.

Sensor	Bytes	Format	Resolution	Units
Air Temperature	2	Signed Integer	0.1	deg F
Relative Humidity	1	Unsigned Integer	1	%
Barometric Pressure	2	Unsigned Integer	0.1	hPa
Wind Speed	1	Unsigned Integer	1	mph
Wind Direction	2	Unsigned Integer	1	deg
Peak Wind	1	Unsigned Integer	1	mph
Stage	2	Signed Integer	0.01	ft
Battery Voltage	1	Unsigned Integer	0.1	V

Figure 2-9 Type 3 Measurement Suite

2.4.6 Type 4 Measurement Suite

The eight included sensor types are listed below. Each field is fixed length, and must contain the number of bytes listed. The value format, measurement resolution and units are derived from the nature and typical accuracy of the measurement.

Sensor	Bytes	Format	Resolution	Units
Air Temperature	2	Signed Integer	0.1	deg C
Relative Humidity	1	Unsigned Integer	1	%
Barometric Pressure	2	Unsigned Integer	0.1	hPa
Wind Speed	2	Unsigned Integer	1	km/hr
Wind Direction	2	Unsigned Integer	1	deg
Peak Wind	2	Unsigned Integer	1	km/hr
Stage	3	Signed Integer	0.001	m
Battery Voltage	1	Unsigned Integer	0.1	V

Figure 2-10 Type 4 Measurement Suite

2.4.7 Type 5 Measurement Suite (MSR-IND)

The Type 5 Multi-Sensor IND Report provides the ability for IND devices on a network to provide important status information, allowing timely detection and resolution of problems when they arise. This report is especially beneficial for backbone devices, such as Repeaters or Base Station Receivers that may not have an Application Layer to send status reports.

Type (5) T	Len L	Multi-Sensor Report								
		Data Flags	Clock Status	Battery	IND Temp	Msgs Rcvd	Msgs Sent	Status1 Bits	Resvd	Resvd

Figure 2-11 Type 5 MSR-IND Report

Each of the 8 bits, when set to 1, indicates the inclusion of one of the 8 defined sensor/status data fields. The low order bit represents the immediately following data field (Clock Status) and the high order bit represents the last data field (Reserved).

The eight included sensor/status types are listed below. Each field is fixed length, and must contain the number of bytes listed. The value format, measurement resolution and units are derived from the nature and typical accuracy of the measurement. The sensor Ids as recommended in section 2.2.2 are inferred by their position in the list.

Sensor	Suggested Sensor ID	Bytes	Format	Resolution	Units
Clock Status	201	1	Unsigned Integer	•	See clock status table
Battery Voltage	8	1	Unsigned Integer	0.1	V
IND Temperature	203	2	Unsigned Integer	0.1	Deg C
Messages Received	204	2	Unsigned Integer	1	Accumulator Value
Messages Sent	205	2	Unsigned Integer	1	Accumulator Value
Status Bits	206	1	Unsigned Integer	-	See status bits table
Reserved	-	-	-	-	-
Reserved	-	-	-	-	-

Figure 2-12 Type 5 Measurement Suite

- The meaning of the Clock Status and the Status Bits field are defined below.
- IND Temperature is a measure of temperature at the device, and may be implemented as internal processor temperature or temperature measured elsewhere on the board. Normal and acceptable ranges for this will vary depending on device manufacturer.

- Messages Sent and Messages Received are accumulator values. These values will roll over, and it is up to the base station software to manage this rollover with the total accrued count. They are not persisted, and are reset when the device is power cycled

The Clock Status sensor reports the following values:

Value	Meaning
0	Clock uncertainty is sufficiently small to allow TDMA (e.g. synchronized to GPS)
2	Clock drifted (e.g. clock was synchronized to GPS in the past)
3	Clock never synchronized (no GPS, no NTP)
4	Clock uncertainty is less than ~1 second, but is too large for TDMA (e.g. synchronized to NTP pool)

Figure 2-13 Clock Synchronization Values

The Status Bits sensor is a bit field, with each bit having the following meaning:

Bit Posn	Meaning
0 (0x01)	Decoder Subsystem Warning or Error
1 (0x02)	Encoder Subsystem Warning or Error
2 (0x04)	GPS Clock Subsystem Warning or Error
3 (0x08)	API Subsystem Warning or Error
4 (0x10)	IO Subsystem Warning or Error
5 (0x20)	User initiated Warning or Error (Misconfiguration, etc.)
6 (0x40)	Device was reboot (set on device power up)
7 (0x80)	Slot overrun: TDMA slot overrun occurred, or slot overrun protection was triggered

Figure 2-14 Status Bit Values

The bits are reset after each read / transmit, so it should be possible to determine if a problem is transient or recurring.

2.5 Time Series Data (TSD) Report Type

The Time Series Data (TSD) Report Type provides a mechanism for reporting sensor data in a common time series format. The TSD report was added to help support technological advances in real-time flood forecasting, modelling and inundation mapping.

Type (7) T	Len L	Time Series Data Report							
		Sensor ID	Interval	Format/ Length	Data(1)	Data(2)	Data(3)	Data(…)	Data(N)

Figure 2-15 Time Series Data Report

2.5.1 Report Sub-Structure

Report Sub-Structure:

- Report Type (T) — 1 byte application layer message identifier, value 7
- Report Length (L) — 1 or 2 bytes, as defined in 2.1.4
- Sensor ID (S) — 1 byte
- Interval — 1 byte, as defined in 2.5.3
- Value Format/Length (FL) — 1 byte, as defined by 2.2.3
- Data(1) – Data(N) — 1 to N Time Series Data Elements

2.5.2 Sensor ID (S)

The Sensor ID indicates the sensor for which the subsequent TSD data elements belong, as pictured in Figure 2-15 above.

For the sake of consistency, the same sensor ID numbering requirements and limitations as in place for a Type 1 GSR message will be employed for Type 7 messages. This includes the use of reserved sensor ID 255 for embedding a 4-byte POSIX timestamp. When such a timestamp is used, then fields would be inserted between the Length and Sensor ID fields of Figure 2-15 above to carry the timestamp. The resulting format is shown in Figure 2-16 below.

Type (7) T	Len L	Time Series Data Report										
		Sensor ID=255	FL = 0xf4	4-byte POSIX value	Sensor ID	Interval	FL	Data(1)	Data(2)	Data(3)	Data(…)	Data(N)

Figure 2-16 TSD with Preceding POSIX Timestamp

Regardless of which timestamp is used (the 2-byte field right after the “Ctl Byte” or the sensor 255 POSIX time) that time will refer to the time of the MOST CURRENT data element carried in the message.

2.5.3 Interval

Interval is a single byte field defined with the two high-order bits representing seconds (00), minutes (01), hours (10), or days (11) and the 6 remaining low order bits providing the actual interval using values 1 – 59. (Values 60-63 are described below, and 0 is invalid.) With this structure it is possible to send ranges from 1-59 seconds, minutes, hours, or days. Sub-second intervals are described below. A bit view of the example (5 minute intervals; 0x45) in Figure 2-17 below would be “01000101”. If the desired interval was to be 30 seconds the byte would be 0x1E, 30, and 00011110. For an hourly time series interval the byte would be 0x81, 129, and 10000001.

Sub-second intervals could be useful for small intervals right after some event, such as collecting a burst of seismic data or bridge vibration. Sub-second intervals are defined by setting the two high order bits to seconds (00) and assigning the lower 6 bit values as 60=0.1 second intervals, 61=0.01 second intervals, 62=0.001 second intervals and 63=0.0001 second intervals. To clarify further, the full 8-bit interval value for each is shown below in hex.

- 0x3F (63) = 0.0001 second intervals
- 0x3E (62) = 0.001 second intervals
- 0x3D (61) = 0.01 second intervals
- 0x3C (60) = 0.1 second intervals

The lower 6 bit values 60-63 for the minutes, hours and days units are reserved. Specifically:

- 0x7C – 0x7F (124-127)
- 0xBC – 0xBF (188 – 192)
- 0xFC – 0xFF (252 – 255)

2.5.4 Format/Length

The Format and Length field follows the same requirements as Format and Length in Type 1 GSR message of the following data.

2.5.5 Data(1) – Data(N) Fields

The Data fields contain values for succeeding evenly spaced intervals (as specified by the Interval field) of time. The oldest data are presented first with the most current data being last in the string. This last field is the one for which the timestamp carried applies to.

2.5.6 Example with Typical Use Cases

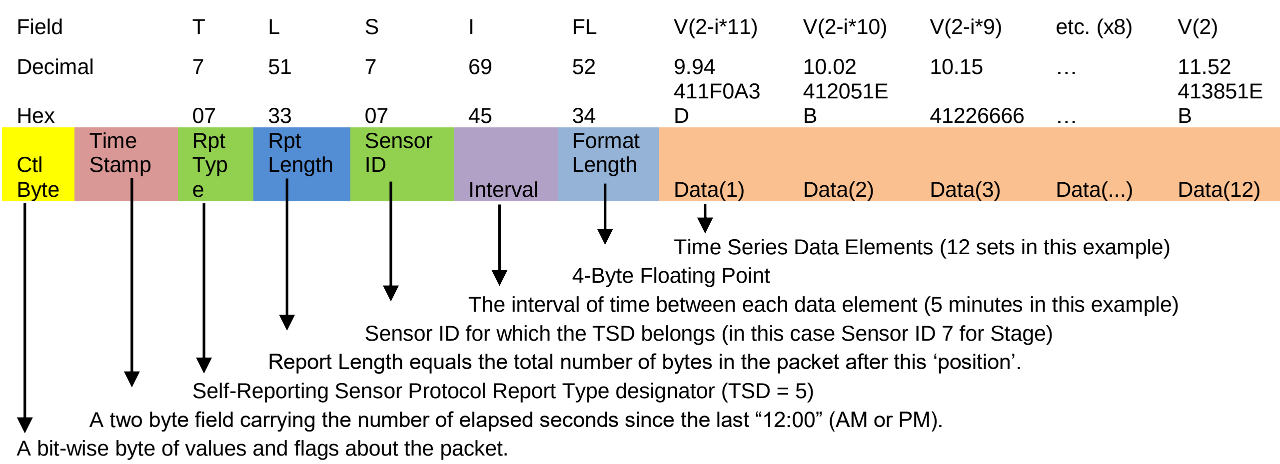

Figure 2-17 TSD Example

The above example represents a TSD report for Stage with 4-byte floating point samples at 5 minute intervals. This TSD report is carrying one hour's worth of 5-minute data samples (12 elements). The last sample (Data12) is the one for which the timestamp carried applies to.

This TSD example can be sent in a 500 millisecond slot assuming typical default values are utilized for related timing configurations of the radio. To have sent this data as individual GSR packets every five minutes would have required 3.72 seconds of actual airtime with six full seconds of airtime having to be allocated.

2.6 SET Command Report Type

The SET Command is intended to instruct a remote site to change the value of one or more sensors. The SET Command provides a standard means to request a remote site to change the value of one or more sensors.   All SET Commands are described by the following recursive TLV protocol. The following illustration shows the setting of 3 sensors bundled into the upper level Value field of the SET Command.

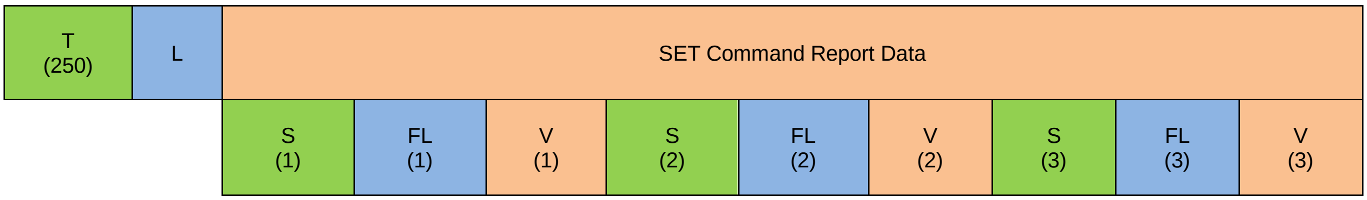

Figure 2-18 SET Command Report

The SET Command is directed to the specific remote using the IND/MANT mechanism for identifying a destination address. This can optionally be extended to a multicast destination where a group of sites recognize the same destination address.

Report Sub-Structure:

- Report Type (T) — 1 byte application layer message identifier , value 250
- Report Length (L) — 1 or 2 bytes, as defined in 2.1.4
- Sensor ID (S) — 1 byte
- Value Format/Length (FL) — 1 byte, as defined by 2.2.3
- Report Value (V) — as specified in Value Format/Length

Repeating structure for optional multiple Reports

- Sensor ID (S) — optional…
- Value Format/Length (FL) — optional…
- Report Value (V) — optional…

2.7 GET Command Report Type

The GET Command provides a standard means to request a remote site to report the value of one or more sensors. All GET Commands are described by the following recursive TLV protocol. For illustration, the GET request for 3 sensor readings are shown bundled into the upper level Value field of the GET Command.

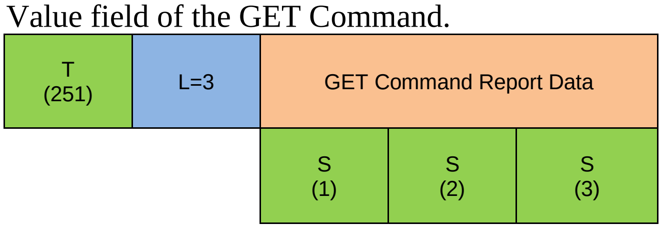

Figure 2-19 GET Command Report

- Report Type (T) 1 byte application layer message identifier , value 251
- Report Length (L) 1 or 2 bytes, as defined in 2.1.4
- Sensor ID/s (V) List of sensor IDs to Get. An empty list indicates all sensors.

Note that only the sensor IDs are required in the GET Command Data field. A GET Command for **all sensors** at a site is formatted by simply setting the length field (L) to zero and eliminating the Data field:

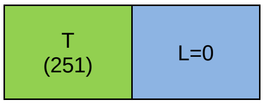

Figure 2-20 GET Command Report – All Sensors

A remote site that receives a GET Command will generate a report for the requested sensor in the Self Report type format that the sensor is configured for. For instance a rain gauge would simply generate a TSR with the current accumulator count; a stage would take a new sample and generate a GSR (or MSR if the sensor was configured for this type of report). It may be beneficial for the responding site to set its destination address to the address of the GET Command sender, but this may not always be the case. Therefore it is recommended this option be configurable in the individual transmitter with the default being “don’t set DA.”

As with the SET Command, the GET Command is directed to the specific remote using the IND/MANT mechanism for identifying a destination address. This can optionally be extended to a multicast destination where a group of sites recognize the same destination address.

3 ALERT Concentration Protocol

3.1 ALERT Concentration PDU

The ALERT Concentration Protocol provides a mechanism to aggregate and compact ALERT messages, then forward them within an ALERT2 frame. Its purpose is to provide the advantages of ALERT2 services to existing ALERT networks; it increases channel efficiency, eliminates contention, performs forward error correction and detects any remaining errors introduced during the ALERT2 transmission. It is used primarily at forwarding repeaters. This specification is Version 00 of the ALERT Concentration Protocol.

ALERT Concentration requires different decoding services than sensor reports and is therefore a separate protocol from the Self-Reporting Protocol. It is directed to Port 1 by the MANT header Port field.

ALERT Concentration is specified to operate on a time cycle of 255 seconds or less, configurable by the user. The Concentration Protocol specifies that ALERT four-byte messages must be received, decoded and re-encoded in a condensed three-byte format. A time offset, specified to mean "seconds before timestamp or export" is created at the moment of receipt and must be appended to the three-byte structure. All reformulated messages created during the cycle must be concatenated into a single message, and formed into an ALERT Concentration PDU by the addition of the header. The Protocol header must contain a control byte and an optional timestamp. The Protocol PDU is shown below

Control C	(TS)	ALERT Concentration					
		ALERT (1)	TO (1)	ALERT (2)	TO (2)	ALERT (n)	TO (n)

Figure 3-1 ALERT Concentration PDU

3.2 ALERT Concentration Header

3.2.1 Control Byte

The Control Byte consists of 8 bits pre-pended to the start of every message. It contains flags and reserved bits used to direct the interpretation of the bytes that follow. The definition of the bits is specified below:

Field	Bytes	Bits	Purpose	Notes
Control	1			
		0,1	Version	Current version is 0
		2	Timestamp	If set,16-bit Timestamp follows Control field
		3	Test Flag	If set, report is flagged (as test data)
		4,5,6	APDU ID	Cyclic PDU ID
		7	Extensibility	If set, a second control byte follows

Figure 3-2 Application Control Byte Definition

The bit 3 flag is provided to indicate that a message is being sent as a part of site configuration or servicing whose data may not be a valid part of the dataset. Treatment of the test message is at the discretion of the application decoding software.

Bits 4, 5 and 6 provide an optional Application PDU identifier. Its default value must be set to value 7, indicating it is disabled. When used, the three-bit value is incremented by one from the values 0 to 6 then back to 0. One use by receiving application software is to identify duplicate reports when there are multiple paths to the destination.

3.2.2 Timestamp

The Timestamp must be a 16 bit unsigned integer representing the seconds elapsed since 12:00 AM or 12:00 PM UTC, whichever is more recent. It must be the time at which the message was exported by the application protocol device (APD). The field is optional; if absent, the third bit of the Control Byte is set to 0. The reason for omitting the timestamp may be that this is an ALOHA media access PDU, because the APD does not carry accurate time, or that the report does not require a timestamp.

The MANT layer provides a timestamp service which can be requested via the Application Layer API. When requested, if the MANT protocol device to which the PDU is sent has accurate time, a 16 bit timestamp is inserted after the Self-Reporting protocol Control Byte, and the Control Byte timestamp bit is set to 1.

3.3 ALERT Concentration Message Substructure

- ALERT Frame (1) 3 byte condensed message…
- Time Offset (TO1)) 1 byte unsigned integer, "seconds before timestamp or export"…
- ALERT Frame (n) optional 3 byte condensed message…
- Time Offset (TO n) optional 1 byte unsigned integer, "seconds before timestamp or export"…

3.3.1 ALERT Concentration Message

There are three generally recognized formats for ALERT data which must be supported in the implementation of the ALERT Concentration Protocol. These are known as ALERT Binary format, ALERT ASCII format and Enhanced IFLOWS (EIF) format; see Appendix 3 for supporting information. Each of these three is decoded to its address and data values, then translated into the same three-byte Concentration format as follows:

Concentration Byte 1	A7	A6	A5	A4	A3	A2	A1	A0
Concentration Byte 2	D10	D9	D8	A12	A11	A10	A9	A8
Concentration Byte 3	D7	D6	D5	D4	D3	D2	D1	D0
Concentration Byte 4	T7	T6	T5	T4	T3	T2	T1	T0

Figure 3-3 ALERT Concentration Data Structure

The completed ALERT Concentration Message field is diagrammed in Figure 3.3, where A(n) refers to the 13 bit ALERT address, and D(n) refers to the 11 bit ALERT data field, and T(n) is a time bit, discussed below.

The ALERT Binary format contains an address of 13 bits and a data value of 11 bits, along with 2 framing bits in each byte, which are discarded. The EIF format also contains 13 address bits and 11 data bits, but contains 2 framing bits in the first byte and 6 bits of CRC checksum. On decoding, the checksum shall be tested and the message must be discarded by the concentration process if it fails. If it succeeds, the address and data bits must be formatted into the concentration format shown above.

The legacy ALERT ASCII format uses each byte to encode one ASCII value, representing a decimal value of 0 to 9. The high order bit is unspecified in ALERT ASCII format; the decoder must recognize and handle either value in the high bit of each byte. When recognized as an ASCII format message, the decimal values must be extracted, re-encoded as binary values, and reformatted to the concentration format shown above.

On decoding, address and data values must be extracted from the concentration data structure; information on the original type of ALERT message is not retained.

3.3.2 Time Offset (TO)

The Time Offset must be a one byte field, formatted as a 1 byte unsigned integer and must be seconds before the packet was exported.

4 Appendix 1 - Examples

4.1 General Sensor Report Example

As an example, we have a water quality site with a pH sensor and a water temperature sensor. This report could be a timed or event-based reading, sent either in a TDMA time slot, or using ALOHA; at this site, the timestamp will be provided externally. This is an operational, not test, measurement, and the PDU ID function is disabled

First Report: Sensor ID = 18 for a pH sensor, with a present value of 8.04 pH. The value is carried in a 4-byte floating point, so the value of the Format/Length Field is 3 in the four high bits and 4 in the four low bits. The value is sent in engineering units.

Second Report: Sensor ID = 19 for the water temperature sensor, with a current value of 63.0 degrees F. The value is carried in a two byte signed integer so the value of the Format/Length Field is 2 in the four high bits and 2 in the four low bits. The unit value is tenths of a degree.

The following formatted message will be transmitted:
Control Byte (C): The 5th, 6th and 7th bits are set to 1; byte value is 0111000 or 0x70
Type = 1, self-reporting sensor report, formatted as a single byte.
Type Length = 10, single byte sum of sizes of all fields following the Length field

Sensor (S) = 18
Format/Length = 0x34 (52), combined value using formula (Format * 16) + Length in a single byte.
Current pH value = 8.04, which has a 4-byte floating point value of 0x 41 00 A3 D7

Sensor (S) = 19
Format/Length = 0x22 (34), combined value using formula (Format * 16) + Length in a single byte.
Current water temperature value = 63.0 F, represented as 630.

The representation of the Self-Reporting Sensor PDU is:

Field	C	T	L	S	FL	V	S	FL	V
Decimal	112	1	10	18	52	8.04	19	34	630
Hex	70	01	0A	12	34	41 00 A3 D7	13	22	02 76

Figure 4-1 Self-Reporting Sensor PDU

4.2 Tipping Bucket Rain Gage Report Example

For an example site, assume a rain gage with rain tips at 12:01:30, 12:01:35, 12:01:40, 12:01:48 and APD hold-off of 20 seconds. The creation of this report is started at the time of the first tip because it is an ALOHA site.

Rain Gage Report: Sensor ID = 0, last transmitted accumulator value = 100. In this example the Accumulator is carried in a 4-byte unsigned integer, so the Format/Length value is 1 in the high order bits and 4 in the low order bits, for a decimal value of 20.

The following formatted message will be transmitted:
Control Byte (C): The PDU ID is enabled with the counter at 5, so the 5^{th} and 7^{th} bits are set to 1: 01010000, 0x50
Type (T) = 2, rain gage report, formatted as a single byte.
Length (L) = 10, single byte sum of sizes of all fields following Length

Sensor (S) = 0
Format/Length (FL) = 0x14 (20), combined value using formula (Format * 16) + Length in a single byte.
Current accumulator value (Accum) = 104, formatted as a four byte value.
Time Offsets (TO(n)) = 20, 15, 10, 2, each formatted as a single byte unsigned integer value.

The representation of the Tipping Bucket Rain Gage PDU is:

Field	C	T	L	S	FL	Accum	TO(1)	TO(2)	TO(3)	TO(4)
Decimal	80	2	10	0	20	104	20	15	10	2
Hex	50	02	0A	00	14	00 00 00 68	14	0F	0A	02

Figure 4-2 Tipping Bucket Rain Gage PDU

4.3 Multi-Sensor Report Example (Type 3)

For this example of a Type 3 Multi-Sensor Report, consider a typical weather station with air temperature, relative humidity, wind speed, and wind direction. A battery voltage is included with this report. The report is generated as a test report by a technician servicing the site and the report is to be marked as such in the database. The APD is requesting timestamp service from the lower layers. Note that 5 reports are being sent in the same length as the two General Sensor reports.

Sensor	Position	Value
Air Temperature	1	23.4 F
Relative Humidity	2	41 %
Wind Speed	4	8 mph
Wind Direction	5	265º
Battery Voltage	8	12.7 V

The following PDU will be formed:
Control Byte (C): The version is 00 in the first two (low) bits. The timestamp is being requested, the test flag is set, and PDU ID is disabled. The third and fourth bits are both 1, as are the 5th, 6th and 7th (binary 01111100, 0x7C, 124 decimal).
Type (T) = 3, Multi-Sensor Report – English Units
Length (L) = 8, including all bytes following the Length field.
Data Flags (DF) = Binary 10011011 = 155 decimal. The flag is set to 1 for each active sensor.
Measurement Suite = 234,41,8,265,127, each formatted according to its specified length and type and concatenated.

Field	C	T	L	DF	AT	RH	WS	WD	BV
Decimal	124	3	8	155	234	41	8	265	127
Hex	7C	03	08	9B	00 EA	29	08	01 09	7F

Figure 4-3 Type 3 Multi-Sensor Report PDU

4.4 Multi-Sensor Report Example (Type 4)

For this example of a Type 4 Multi-Sensor Report, consider the same weather station as in 4.3 with air temperature, relative humidity, wind speed and wind direction. A stage reading is included rather than battery voltage in this report. The report is generated as a test report by a technician servicing the site and the report is to be marked as such in the database. The APD is requesting timestamp service from the lower layers.

Sensor	Position	Value
Air Temperature	1	-15.5 C
Relative Humidity	2	41 %
Wind Speed	4	13 k/h
Wind Direction	5	265º
Stage	7	535.813 m

The following PDU will be formed:
Control Byte (C): The version is 00 in the first two (low) bits. The timestamp is being requested, the test flag is set, and PDU ID is disabled. The third and fourth bits are both 1, as are the 5th, 6th and 7th (binary 01111100, 0x7C, 124 decimal).
Type (T) = 4, Multi-Sensor Report – metric units
Length (L) = 11, including all bytes following the Length field.
Data Flags (DF) = Binary 01011011 = 91 decimal. The flag is set to 1 for each active sensor.
Measurement Suite = -15.5, 41, 13, 265, 535813, each formatted according to its specified length and type and concatenated. Temperature is a 2-byte signed integer, so it is represented as the two's complement of 155.

Field	C	T	L	DF	AT	RH	WS	WD	ST
Decimal	124	4	10	91	-155	41	13	265	535813
Hex	7C	04	0A	5B	FF 65	29	00 0D	01 09	08 2D 05

Figure 4-4 Type 4 Multi-Sensor Report PDU

4.5 Example of Combined Tipping Bucket Rain and General Sensor Report

This example is presented to show that the multiple types of reports can be combined into a single PDU. For an example site, we have a pH sensor, water temperature, and rain gage; the sensor IDs and data values are the same as in the two previous examples, but combined into a single message.

Sensor ID = 0 for rain sensor, with a current accumulator value of 104
Sensor ID = 18 for pH sensor, with a current value of 8.04
Sensor ID = 19 for the water temperature sensor, with a current value of 63.0 degrees C

The following formatted message will be transmitted:
Control: All 0, except PDU ID = 3 (00110000, 0x30) formatted as a one-byte value.
Type = 2, Tipping Bucket Rain Gage Report
Length = 10, single byte sum of sizes of all fields following the Length field
Tipping Bucket ID = 0, Format/Length = 20, Accum = 104, Time offsets = 20, 15, 10, 2
Type = 1, General Sensor Report, formatted as a single byte.
Length = 8, single byte sum of sizes of all fields following the Length field
pH Sensor ID = 18, Format/Length = 18, Current pH value = 804
Water Temperature Sensor ID = 19, Format/Length = 34, Current temperature value = 630

Field	C	T	L	Tipping Bucket Rain Gage							T	L	Sensor			Sensor		
Decimal	48	2	10	0	20	104	20	15	10	2	1	8	18	18	804	19	34	630
Hex	30	02	0A	00	14	00 00 00 68	14	0F	0A	02	01	08	12	12	03 24	13	22	02 76

Figure 4-5 Combined Report PDU

4.6 Multi-Sensor IND Report Example (Type 5)

For this example of a Type 5 Multi-Sensor IND Report, an ALERT2 repeater is reporting its messages count and status. The report is populated with *Messages Received*, *Messages Sent* and *Status Bits* fields. The Status Bits are all zero, indicating the unit is operating normally.

Sensor	Position	Value
Messages Received	3	25,010
Messages Sent	4	25,201
Status Bits	5	0

The following PDU will be formed:
Control Byte (C): The version is 00 in the first two (low) bits. The timestamp is included in the PDU following this control byte, the test flag is reset, and cyclic PDU ID is 1.
Timestamp (TS) = 60 (this report is sent 1 minute after midnight UTC)
Type (T) = 5, Multi-Sensor Report – IND Status
Length (L) = 8, includes all bytes following the Length field.
Data Flags (DF) = Binary 00011100 = 28 decimal. The flag is set to 1 for each populated sensor/status field.
Measurement Suite = 25010, 25201, 0, each formatted according to its specified length and type and concatenated.

Field	C	TS	T	L	MR	MS	S
Decimal	20	60	5	5	25,010	25,201	0
Hex	14	00 3C	05	05	61 B2	62 71	0

Figure 4-6 Type 5 Multi-Sensor Report PDU

4.7 SET Command Report Type Example (Type 250)

An example of setting a rain gauge accumulator value to zero at site 15110:

- MANT Header Byte 0 = 3
  - Version = 0
  - TSSR = 0
  - Add Path Service Req Flat = 1
  - Destination Address (DA) included in header = 1
- MANT Header Byte 1 = 0
  - Port = 0 (App Layer Self Reporting Protocol)
  - Resvd Bits and ACK Bit = 0
- MANT Header Bytes 2-3 = 10 09 (hex)
  - Added Header flag = 0
  - Hop Limit = 1
  - Payload Length = 9
- MANT Header Source Address = 15000 (decimal), 3A 98 (hex)
- MANT Header Destination Address = 15110 (decimal), 3B 06 (hex)
- SET Command PDU Structure:
  - Control Byte (C): The version is 00 in the first two (low) bits. The timestamp is included in the PDU following this control byte (TS bit is set), the test flag is reset, and cyclic PDU ID is 1.
  - Timestamp (TS) = 60 (this report is sent 1 minute after midnight UTC)
  - Report Type (T) = SET (250)
  - Report Length (L) = 3
  - Sensor ID (S) = 0 for Rain
  - Value Format/Length (FL) = upper nibble (F) = unsigned integer / lower nibble (L) = 0x12
  - 2-byte Accumulator Value (V) = zero

Field	MANT	MANT	MANT	MANT	MANT SA	MANT DA	C	TS	T	L	Sensor		
Decimal	3	0	16	9	15000	15110	20	60	250	3	0	17	0
Hex	03	0	10	09	3A 98	3B 06	14	00 3C	FA	03	00	11	00 00

Figure 4-7 SET Command Example

4.8 GET Command Report Type Example (Type 251)

An example of getting 3 sensor values (rain accumulator, stage and battery) from site 15110:

- MANT Header Byte 0 = 3
  - Version = 0
  - TSSR = 0
  - Add Path Service Req Flat = 1
  - Destination Address (DA) included in header = 1
- MANT Header Byte 1 = 0
  - Port = 0 (App Layer Self Reporting Protocol)
  - Resvd Bits and ACK Bit = 0
- MANT Header Bytes 2-3 = 10 08 (hex)
  - Added Header flag = 0
  - Hop Limit = 1
  - Payload Length = 8
- MANT Header Source Address = 15000 (decimal), 3A 98 (hex)
- MANT Header Destination Address = 15110 (decimal), 3B 06 (hex)
- GET Command PDU Structure:
  - Control Byte (C): The version is 00 in the first two (low) bits. The timestamp is included in the PDU following this control byte (TS bit is set), the test flag is reset, and cyclic PDU ID is 2.
  - Timestamp (TS) = 3600 (this report is sent 1 hour after midnight UTC)
  - Report Type (T) = SET (251)
  - Report Length (L) = 3
  - Sensor IDs (S1) = 0 for Rain, S2=7 for Stage, S3=8 for Battery

Field	MANT	MANT	MANT	MANT	MANT SA	MANT DA	C	TS	T	L	Sensor IDs		
Decimal	3	0	16	8	15000	15110	36	3660	251	3	0	7	8
Hex	03	0	10	08	3A 98	3B 06	24	0E 10	FB	03	00	07	08

Figure 4-8 GET Command Example

5 Appendix 2 – ALERT Formats

5.1 ALERT Binary format

Native ALERT binary frames consist of 40 bits; each of the four bytes consists of 2 high order marker bits and 6 low order data bits enclosed by a start bit and stop bit. The marker bits in the first two bytes are 01, the second two bytes are 11. The 24 bits of data consist of 13 bits of address and 11 bits of data, distributed as follows:

0	1	A5	A4	A3	A2	A1	A0
0	1	A11	A10	A9	A8	A7	A6
1	1	D4	D3	D2	D1	D0	A12
1	1	D10	D9	D8	D7	D6	D5

Figure 5-5-1 ALERT Binary Data Structure

The 24 bits of data are extracted and reformulated into a new 4-byte structure:

Byte 1 of the contains the lowest eight bits of the address

Byte 2 contains the 5 high bits of the address in the byte's low bits and the 3 high bits of the data in the new byte's high bits

Byte 3 contains the 8 low bits of the ALERT data field.

5.2 ALERT ASCII format

ALERT ASCII format contains 4 bytes, with each byte containing one ASCII value between 0x30 and 0x39. The first byte contains the units value of the address, and the second byte contains the tens value of the address. Similarly, the third and fourth bytes contain the units and tens values of the data field.

x	0	1	1	Au3	Au 2	Au1	Au 0
x	0	1	1	At 3	At 2	At 1	At 0
x	0	1	1	Du 3	Du 2	Du 1	Du 0
x	0	1	1	Dt 3	Dt 2	Dt1	Dt 0

Figure 5-5-2 ALERT ASCII format

The terms A and D above refer to Address and Data bytes, respectively. The "u" and "t" subscripts indicate unitary or tens values, respectively.

The first byte is the address unit value, and the second byte, multiplied by 10, is added to the unit value for a range of 0 to 99. The third and fourth bytes are processed the same way to yield a data value between 0 and 99. The binary form of these values is transferred to the three byte concentration structure described in the text. Because the address and data values each occupy only the low 7 bits, the second concentration byte will always be 0.

5.3 Enhanced IFLOWS format

Enhanced IFLOWS format messages have a value of 1 in the two high order bits of the first byte. There are no marker bits in the remaining three bytes, and the checksum is contained in the lower 6 bits of the fourth byte. The format is shown below, where A(n) is an address bit, D(n) is a data bit, and C(n) is a CRC checksum bit. The generator polynomial is $x^6 + x^4 + x^3 + 1$.

1	1	A5	A4	A3	A2	A1	A0
D0	A12	A11	A10	A9	A8	A7	A6
D8	D7	D6	D5	D4	D3	D2	D1
C0	C1	C2	C3	C4	C5	D10	D9

Figure 5-5-3 Enhanced IFLOWS Format

Additional information concerning EIF checksum processing can be found at http://www.afws.net/supportsite/iflows/enhanced_iflows_format.htm.

6 Appendix 3 – FP2 Format

For the purposes of succinctly reporting environmental measurement data such as stage and air temperature, the FP2 format has been adopted for the Application Layer specification. The IEEE 754 half-precision format quickly loses precision as the magnitude of the value departs from zero and provides more precision than is commonly needed for values close to zero.

FP2 utilizes 1 bit for indicating the sign of the value, 2 bits for exponent, and 13 bits for encoding the mantissa and for flagging the special cases of ±infinity and NaN. The following two tables outline the bit pattern of FP2 and how they are interpreted.

Name	Bit	Description
Sign (S)	15 (msb)	Specifies the sign of the value. 0 = positive, 1 = negative.
Exponent (E)	14 and 13	Specifies the magnitude of the negative decimal exponent.
Mantissa (M)	12 to 0 (lsb)	Specifies the magnitude of the 13 bit mantissa, 0 to 8191

S	E	M	FP2 Value Is =
0	0	8191	+ infinity
1	0	8191	- infinity
1	0	8190	NaN
0 or 1	0 to 3	0 to 7999	(-1 ^ S) * (10 ^ -E) * M

$Value_{16}$	$Value_{10}$
1F 3F	7,999
C4 D2	-12.34
9F FE	NaN

As shown above, FP2 has a range of -7,999 to 7,999. It has the ability to precisely represent each integer, unlike the IEEE 754 half-precision data type. The following table shows the achievable precision for a given range.

Range	Max Precision
-7.999 to 7.999	0.001
-79.99 to 79.99	0.01
-799.9 to 799.9	0.1
-7,999 to 7,999	1