A DESCRIPTION OF THE ALERT2 PROTOCOL

Don Van Wie
Telos Services

October 2011

Table of Contents

FIGURES

TABLES

Telos Services
206 Hazelwood Dr.
Nederland CO 80466

Don.vanwie@gmail.com

A DESCRIPTION OF THE ALERT2 PROTOCOL

INTRODUCTION

WHAT IS ALERT2

ALERT2 is a new protocol optimized for the transport of real-time data over radio telemetry networks. It is the intended successor to the ALERT (Automated Local Evaluation in Real Time) protocol introduced in the 1970s. It offers a 7- to 10-fold increase in net data rate (or channel capacity), detects all errors introduced in transmission and corrects the great majority of them. The new protocol comprises multiple sub-protocols, with the flexibility to add new ones as needs emerge. It provides greater "data space" that expands the range of sensor identifiers and data resolution. It can be used in either ALOHA or TDMA environments, the latter providing the opportunity to eliminate data contention altogether.

ALERT CONSTRAINTS

ALERT, developed in the early 1970s, was originally intended for the collection of real-time rainfall data to support analysis and decision-making. Its success led its use to be extended to other purposes, including meteorologic and hydrologic parameters, and even some control applications. The original ALERT has some inherent limitations that restrict its suitability for large or multipurpose systems.

- It is an ALOHA system, meaning that every gauging site reports independently, without regard to other traffic on the channel. For its original purpose of collecting rainfall data, the occasional loss of data to channel contention is tolerable because the transmitted data is an accumulator value incremented by bucket tips. Missing tips are readily recovered by comparing the current value to the previously received value. However, as it is extended to analog measurements, there is no way to recover missing data. As traffic levels increase, contention losses increase; in some systems, data losses exceeding 50% have been documented during large rainfall events.

- The ALERT data payload is constrained to 24 bits: 13 bits of identifier and 11 bits of data. This has proven restrictive both in terms of sensor identifiers (only 8191) and data resolution (0-2047).

- It operates at 300 baud, but with a 200 msec tone preamble and other overhead, the data throughput is approximately 72 bits per second. Systems with this low channel efficiency will not continue to be licensable under FCC regulations for very much longer.

- While the simplicity of the ALERT protocol is advantageous, the flip side of the coin is that it lacks the flexibility to be adapted to broader uses that are in demand in the hydromet data domain, such as transmitting engineering units, two-way communications, message routing, and control applications.

Positive attributes of the ALERT protocol are that it is an open standard and uses audio frequency shift keying (AFSK) that can be implemented in virtually any commercial off-the-shelf (COTS) radio. Open standards and compatibility with COTS transceivers have been requirements in the development of ALERT2 protocols.

PURPOSE AND RATIONALE OF ALERT2

ALERT2 was conceived by the ALERT community a decade ago. The limitations of ALERT in larger systems were becoming a serious issue. Southern California was running out of ID space.

Narrowbanding was on the horizon, and the need to improve spectrum efficiency was becoming clear. There was discussion of going to a newer technology, but there was a question of feasibility.

One great feature of the lumbering ALERT technology is that, at its slow speed, it exerts more “energy per bit,” and therefore works better than many faster protocols over long and indirect radio paths. The ALERT community has a great deal of infrastructure investment in repeaters and networks that take advantage of this robustness. Replacing ALERT with equipment that could not deliver the same performance over existing radio paths was a non-starter; any new, faster protocol would need to operate as well or better than ALERT in existing installations.

By the standards of the major radio and radio-modem manufacturers, the ALERT market is tiny. Manufacturers of ALERT equipment have no real influence in what is produced or what is discontinued. They have had to scramble more than once to develop work-arounds when specific equipment was suddenly no longer available. This led to a second requirement: Any new protocol would need to be operable with commercially available, off-the-shelf radios, using modulation technologies that the ALERT community could be reasonably assured would be available from multiple manufacturers.

The life cycle of ALERT equipment has been often 15 years or more; there is actually some in service that is over 30 years old! Proprietary product life cycles are typically much shorter than this, with ‘captive’ customers forced to replace entire fleets when products are made obsolete. This led to another requirement for any new protocol: It would need to reside in the public domain so that multiple manufacturers would have access to the methods and technologies to continue production so long as there was a market.

A feasibility study was undertaken in 2002 (“A New ALERT Protocol: Feasibility Study of a New Air Interface and Physical Layer Packet Definition for the ALERT User Community”, R. Chris Roark and Donald Van Wie, February 2003). It assessed a number of readily available technologies, but all were proprietary, and none were well adapted to the short messaging used in ALERT. The study concluded that changing the modulation scheme and implementing forward error correction (FEC) could produce performance comparable to ALERT at much higher throughput. The proposed FEC techniques were developed for NASA’s communications with deep space probes; additional data contained in the message permits receiving computers to compensate for the decreased signal-to-noise ratio by adding “coding gain.”

Following the feasibility study, the ALERT community decided to work toward the development of prototype software and hardware that would meet the constraints enumerated above, and this became the start of ALERT2. The ALERT Users Group (AUG), with additional funding from NOAA/NWS and the Southwest Association of ALERT Systems, awarded contracts to Chris Roark of Blue Water Design, LLC to complete the design of a variable-length, FEC encoded protocol structure. This work concluded with the delivery of working prototypes of an ALERT2 encoder and decoder in 2008. All of the design and software work products are in the public domain and the prototypes are available from the AUG. The final report of this effort, “Prototype Reference Design of Open Source High Bit Rate RF Modem” is available to NHWC members at the NHWC website, http://www.hydrologicwarning.org/. Blue Water Design has continued developing enhanced ALERT2 equipment, which is now installed in the production systems of several large ALERT networks.

The ALERT2 protocol comprises much more than a new modulation technique and FEC. Through the work of the NHWC Technical Working Group, there is now a specification for a modern multi-layer protocol that is optimized for the types of short messaging used in ALERT. Compliant ALERT2 encoders have the capability to capture ALERT messages and retransmit them in ALERT2 format from existing repeater sites. Thus, a transition path from the old to the new technology is built into ALERT2.

In the intervening decade from the inception of ALERT2, the FCC has begun demanding higher spectrum efficiency. The first round of “narrowbanding” cut the channel bandwidth in half, from 25 kHz to 12.5 kHz. ALERT technology was able to accommodate this requirement by retrofitting existing ALERT devices with new transmitters. In 2013, it will no longer be legal to operate any of the existing equipment at the 25 kHz channel spacing. Also in 2013, the next narrowbanding step begins: All transmitters placed in service must be capable of operating in a channel cut in half again to 6.25 kHz. This is in preparation for the next requirement, as yet without a deadline, that all telemetry equipment operate within a 6.25 kHz channel. This will not be possible with existing ALERT technology.

Current Validity of ALERT2 Assumptions

The requirements expressed by the ALERT community in 2000 are still legitimate today. A decade later, the question arises whether something has since appeared in the public domain which could also meet these requirements and eliminate the need to develop and maintain a separate protocol. A brief review of the current market indicates that there may be other physical layer technologies coming available which could provide even greater channel efficiency, but these come at the cost of higher bit error rates and may not perform adequately over difficult radio paths. Further, these technologies do not provide a complete protocol adapted for the needs of the ALERT community.

With a layered protocol, the ALERT2 physical (AirLink) layer can be adapted to emerging technologies, while retaining the coding gain, network management and application functionality of ALERT2.

The ALERT2 Protocols

The ALERT2 Protocol standards are developed and maintained through the ALERT2 Technical Working Group of the National Hydrologic Warning Council. Working drafts of the current standards for each of the three layers are available for public review and comment at http://www.hydrologicwarning.org/. The information in this document, and many of the figures, come primarily from these drafts. This document is intended as an overview, and may not remain current as the standards are finalized and evolve over time. For definitive information, or for detail that goes beyond the scope of this discussion, the reader should consult the appropriate standard.

General Architecture

ALERT2 is a layered protocol with three logical levels: The AirLink Physical Layer, the Network and Transport (MANT) Layer, and the Application Layer. The Application Layer specifies the structure and format of the end user data. The MANT Layer is concerned with the logical transport of data – how data is managed and routed through the network. The AirLink Layer defines the physical properties associated with the transmission of the message; the conversion of a stream of data bits to a modulated RF signal and back again. It also manages forward error correction encoding/decoding and media access.

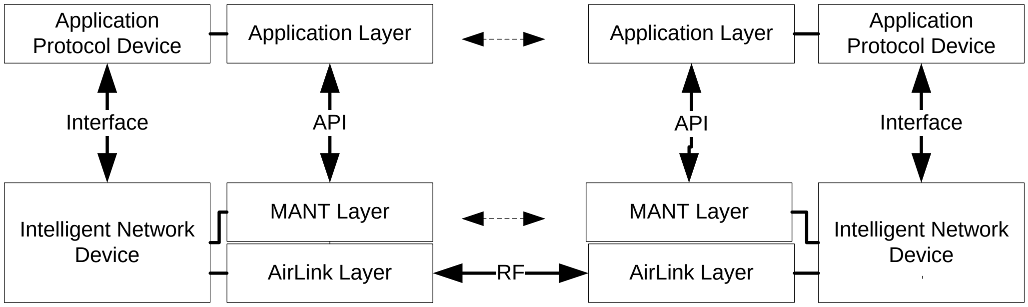

Figure 1 - ALERT2 Architecture

In most ALERT2 installations, (and primarily for historical reasons) the Application layer and the MANT layer are executed in different pieces of hardware; that configuration is shown in Figure 1. The Application Protocol Device (APD) typically has attached sensors, manages the capture of environmental data and exports it as Application protocol data units (PDUs). The MANT and AirLink layers, for now, typically reside in a separate piece of hardware with the transceiver; together these are called the Intelligent Network Device (IND). INDs include ALERT2 encoders, decoders and ALERT2 repeaters.

The layered approach facilitates the management of a more complex modern protocol. Each layer on the transmission end can be perceived as interacting only with its corresponding layer at the receiving end. For example, the physical and MANT layers operate independently of whatever is agreed to comprise the Application protocol. They are responsible only for delivering an application “payload” to the application layer processor on the receiving end, and changes or additions to the application layer protocols do not require changes to how messages are routed or transmitted.

MESSAGE STRUCTURE

ALERT2 uses a variable length block structure. The transmitted message is called a ‘frame,’ each frame consists of a fixed length initial block, followed, if the message length requires, by one or more fixed length follow-on blocks, or by a final partial block. Payload data is distributed across blocks by the ALERT2 encoder; this is transparent to the Application Layer. The complete frame comprises the preamble, synchronization bits, the first block and the combined follow-on blocks, all sent in a single transmission. The logical structure of an AirLink Frame is shown in Figure 2; it consists of a preamble, a frame payload, and an optional short carrier-only tail. The packet payload is FEC encoded and decoded in two segments: the fixed length first block, and all the remaining follow-on blocks, if any.

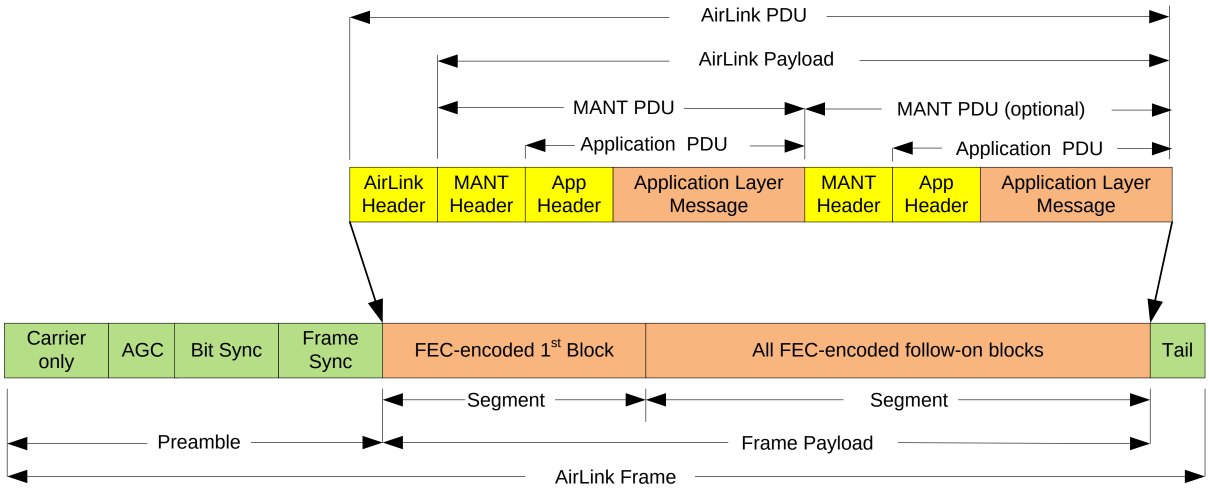

Figure 2 - Message structure and terminology

Independently of the blocking and segmentation used for FEC encoding and decoding, the AirLink payload has a logical data structure built from the sequence of layer processes. It begins with the Application Layer message, such as reports from one or more sensors. An application layer header is pre-pended to it which carries information needed for its downstream processing and decoding. Together, the header and data make up the Application Protocol Data Unit (PDU), which is passed to the MANT layer. The MANT layer performs certain network services, including addressing, hop limits, and pass listing, and pre-pends a six-byte header of its own to form the MANT layer PDU. This is in turn passed to the AirLInk layer which aggregates messages arriving between transmissions, and pre-pends its own header to form the AirLink PDU. The FEC encoding is then done, which lengthens and transforms the AirLink PDU into the Frame Payload. The AirLink initiates the transmission at the appropriate time, sending the preamble followed by the Frame Payload.

The receiving end is the reverse process; the appropriate header is read and the actions it directs for that layer are taken. The header is stripped off and the packet is presented to the next layer above. The application layer receives an Application PDU, along with meta-data such as timestamps and routing information.

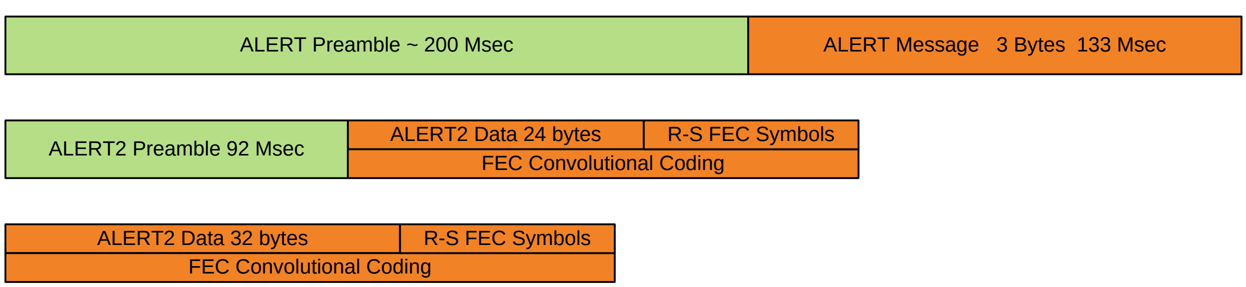

Figure 3 - ALERT and ALERT2 comparison

The representations of ALERT and ALERT2 messages in Figure 3 are drawn at the same time scale. An ALERT2 first block message carries 24 bytes of payload in a little more than half the time it takes ALERT to transmit 24 bits (3 bytes) of useful data. Forward error correction (FEC) increases the size of the data stream by 200% to 240%, but that data is transmitted at a 16-fold greater rate, for a net improvement of 4- to 5-fold over ALERT. The shorter preamble of ALERT2 further improves the relative performance to a factor of 7 to10-fold.

The 228 msec first block contains 2 bytes of AirLink header, 6 bytes of MANT header, and 16 bytes of Application PDU. FEC encoding expands these 24 bytes to 80, with an additional 2-byte convolutional code tail. The 2-byte header identifies the version and the number of bytes that follow. Since the first block is fixed length, any AirLink PDU shorter than 24 bytes will result in the same length transmission.

Full follow-on blocks are 160 msec long and carry 32 bytes of data payload. Partial blocks have a minimum overhead of 53.3 msec, with an additional 3.33 msec for each byte of data payload. The longest payload that can be carried in a single frame in the current implementation is 1023 bytes.

AIRLINK – THE PHYSICAL LAYER

AIRLINK PROCESSES

At the transmitter, the AirLInk Layer receives MANT PDUs from the layer above, and exports an audio data stream to an FM radio it controls. At the receiver, the AirLink extracts AirLink frames, demodulates and converts them back to error-corrected MANT PDUs that are passed to the MANT layer.

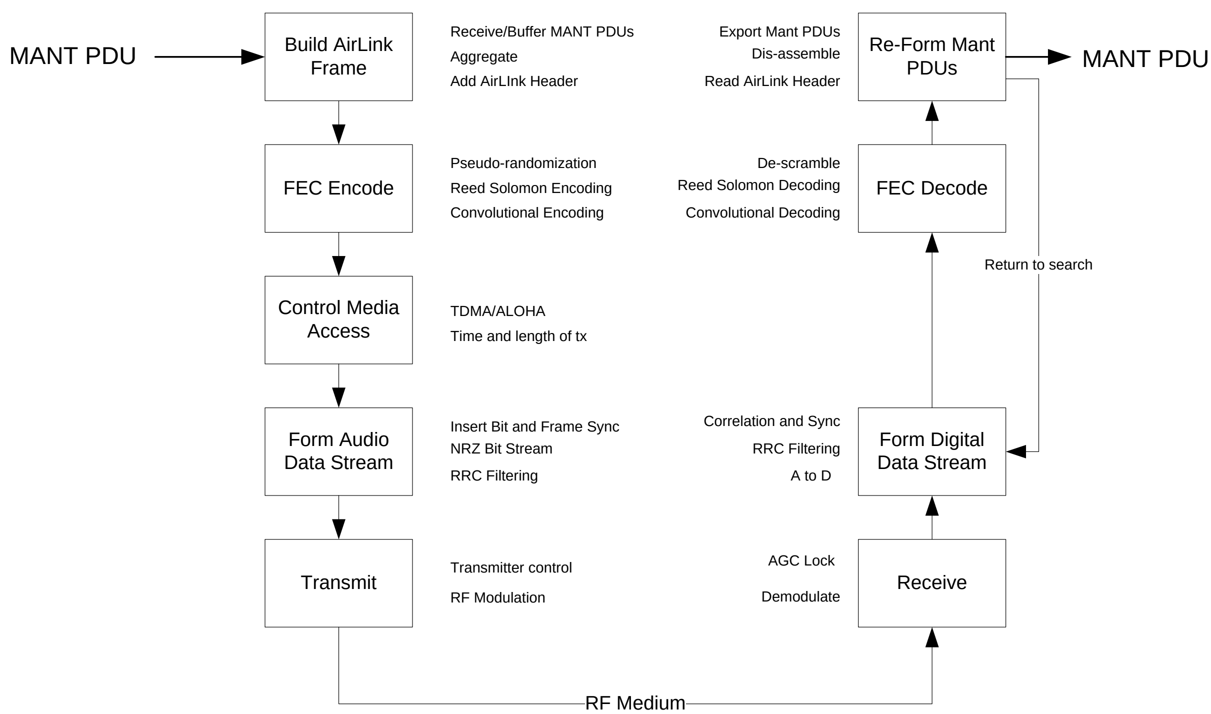

Figure 4 - Block Diagram of AirLink Processes

The AirLink Layer receives, buffers and aggregates MANT messages, builds the AirLink PDU, performs FEC encoding, controls the timing of transmission, adds bit and frame synchronization, converts the data stream to an audio signal and operates the transmitter. These processes are summarized in Figure 4.

The current implementation of the ALERT2 receiver operates in open squelch mode in order to maximize its sensitivity and eliminate the part of the preamble time required for carrier-detect. When bit and frame sync have been established, the demodulated bytes are streamed to the FEC decoder. As soon as a number of bytes equal to the size of the first segment is received, it is FEC decoded and the AirLink header is read to determine the total AirLink PDU length; this information is fed back to the AirLink layer so it can shut down the demodulation process at the appropriate time and return to monitoring.

The AirLink uses FM frequency shift keying. The digital signal is converted to a continuous bit stream which is shaped by filtering into an audio signal that is presented to an FM transmitter at the equivalent of 4800 bits per second.

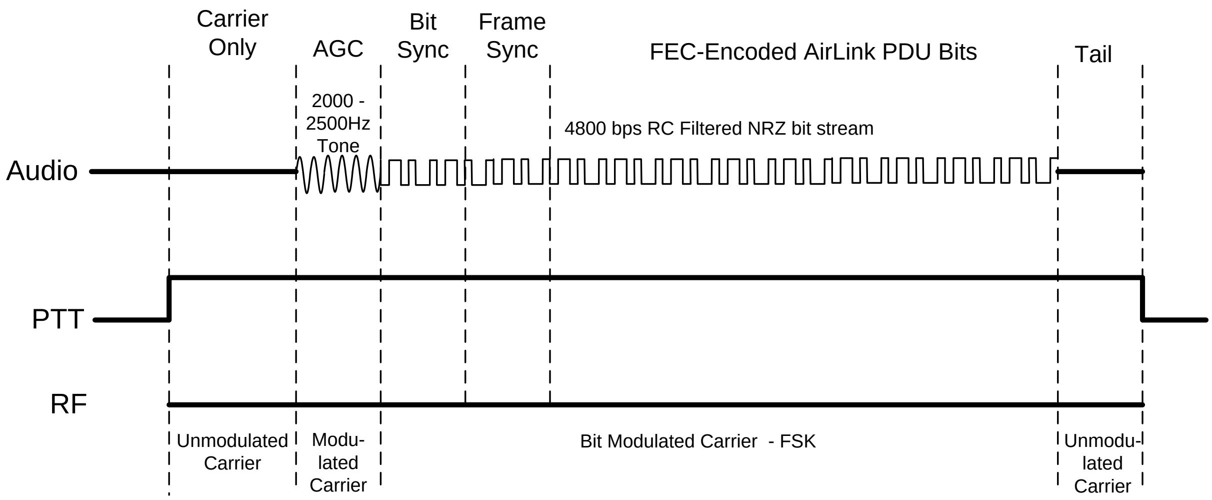

Figure 5 - Transmission of a frame

Figure 5 depicts the Audio, PTT and RF behavior during the transmission of an ALERT2 frame. PTT is raised to begin unmodulated transmission. After a carrier-only period, a 15-msec AGC tone is transmitted that permits the receiver to lock on to the signal. This is followed by a bit sync-correlation pattern that the receiver identifies as an incoming message. Frame sync permits the receiving processor to lock on to the byte pattern, and the bit stream follows. When the complete message has been sent, the audio is stopped and an unmodulated carrier tail is transmitted for another 5 msec. The PTT line is then dropped and the RF transmission ends.

PREAMBLE

Transmission of the first block begins with a minimum 10 msec of carrier-only; field results indicate a longer preamble may be needed in some radio environments, and it has been extended to 60 milliseconds in the current implementations. The optimum carrier-only period will be determined with further testing and field experience; the 60-msec value has been used in these time calculations. The preamble includes 15 msec of AGC lock tone, followed by 6 bytes of bit synchronization and 4 bytes of frame sync bit pattern that are not FEC encoded.

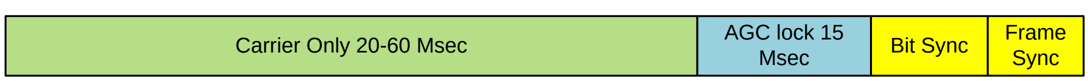

Figure 6 - ALERT2 Preamble

AirLink Header

The AirLink Header is used to direct the decoding process. The first 4 bits carry the version number; the next four are reserved for future use, with 10 bits specifying the AirLink Payload length that follows. The length of the payload is needed to parse the data packet out of the white noise environment.

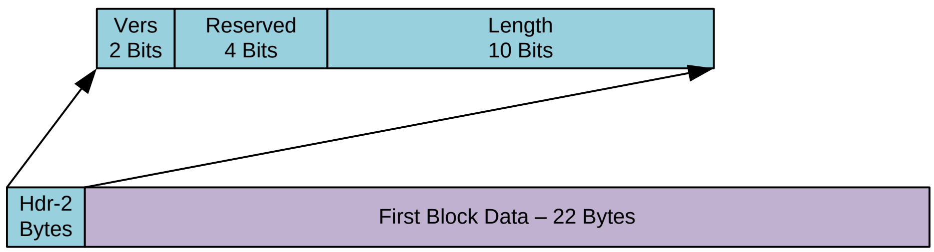

Figure 7 - ALERT2 1st Block Data StructureALERT2 1st Block Data Structure

Forward Error Correction

FEC provides the additional coding gain needed to achieve bit error rates comparable to ALERT at the 16-fold higher ALERT2 data rate. Additional overhead – about 240% of payload in the first block, and 200% of payload in full follow-on blocks – carries information generated by FEC encoding that permits detection of all, and correction of most, errors at the downstream decoder. Two forms of error correction – Reed-Solomon block codes and convolutional coding - are used together to optimize performance. This concatenated coding was developed to support deep space communications, and is now used extensively in digital video as well as earth-satellite communications. Convolutional coding is best at correcting the types of errors introduced by white noise. The Reed-Solomon FEC is able to correct "burst" noise that affects multiple bits in sequence. Up to 8 bytes in error can be corrected in any block by the R-S process. A key feature of the ALERT2 FEC process is that essentially all data exported by the decoder is assured to be correct; error detection is at least as good as a 16-bit CRC. Non-correctable blocks occur infrequently (early data suggests 1% - 3% of blocks in error are uncorrectable and must be discarded). In a month of operation in a production ALERT2 system, 1 block was discarded out of 115,000 received.

The first block is FEC encoded independently while the follow-on blocks are combined for convolutional encoding. Because the AirLink parameters in the first block are essential to decoding the follow-on blocks, the first block carries a higher FEC overhead and is limited in length.

In the encoding process for the first block, 16 bytes of R-S symbols are added. The data portion of the payload is next scrambled so that errors affecting adjacent bits over the air are distributed across the packet when it is unscrambled. This increases the efficiency of error correction. The scrambled payload is passed to the convolutional encoder, whose output is twice the number of input bits, plus a 2-byte tail.

In the follow-on blocks, 16 bytes of R-S symbols are added to each 32 bytes of payload. A partial block contains 16 bytes of R-S symbols and 1 to 31 bytes of payload. All follow-on blocks are aggregated, scrambled and passed to the convolutional encoder, which doubles the length and adds 2 bytes of "tail" to the package. Data payload accounts for 1/3 of the bits in a full follow-on block, yielding an effective data throughput of 1600 bits per second.

The decoding process is the reverse of the encoding. The first block is passed through the Viterbi decoder, then unscrambled to reconstruct the data blocks with the R-S symbols. The R-S decoder uses the symbol information to correct up to 8 bytes of errors in each block, and the output is the original AirLink payload.

Media Access - ALOHA and TDMA

When each site randomly and independently initiates packet transmissions, as happens with ALERT, the media access method is known as ALOHA. Packet losses occur when two packets are transmitted partially or wholly at the same time. In RF networks at least one and sometimes both of the contending packets are lost, or the surviving packet may be corrupted. At high traffic rates, the number of collisions can become so great that data throughput decreases as the input traffic increases. During large storms, data losses as high as 55% have been documented, while losses exceeding 80% are expected in severe case situations in some systems.

ALERT2 can be used in ALOHA mode with losses that are somewhat less severe than ALERT. The average ALERT2 message length is about two thirds that of an ALERT message. Perhaps more important, because the data is received error free, and multiple tips can be combined in a single packet, it is not necessary to transmit at every tip of the bucket. Gages can be programmed to report no more frequently than once per minute, for example, which will greatly reduce peak traffic and reduce losses during critical events.

The preferred media access method for ALERT2 is Time Division Multiple Access (TDMA). Each transmitter is assigned a specific time (a slot) during which it may transmit in the cycle (a TDMA frame) that includes all transmitters on the channel. For example, consider a system with 30 gage sites and a repeater on a single channel. Each gage could be assigned a ½ second long slot and the repeater a 5-second slot in a 20-second frame. No gage would be allowed to transmit more than once every 20 seconds, but multiple individual rain tips can be transmitted in a single packet without loss of time resolution (see Application Layer). ALERT2 TDMA requires each IND to ensure that its transmission occurs entirely within its assigned slot. The IND must take into account the clock drift rate, the time since last clock correction and the length of its message before it transmits. The needed clock accuracy is usually achieved by regularly syncing the clock to the time obtained from a GPS receiver.

A ½ second slot once a minute is long enough for a gage site with a clock accuracy of +/- 50 milliseconds to report a first block and a full follow-on block. This is sufficient capacity to report each tip of a .01" tipping bucket at a rain rate of 10 inches per hour. In addition, the report could include at least eight sensor reports, such as a PT and a full weather suite. A repeater with a 5 second slot can concatenate and retransmit approximately 35 ALERT2 single block frames. Note that a repeater may be operated without contention in its own time slot on the same channel as the gages; in many cases it may be possible to operate on a single frequency where ALERT demanded two.

Design of the TDMA frame and the definition of slots is quite flexible. These are controlled by the user in the configuration of the IND and remain static. The minimum practical ALERT2 slot is 500 msec, which can be extended in increments of 250 msec. The frame length is user set; as a practical matter, it is generally chosen to be a multiple or sub-multiple of 60 seconds. An IND may be assigned multiple adjacent slots in order to increase data capacity, or may be assigned multiple slots within a single frame in order to reduce latency. Figure 8 below illustrates this. This example represents a 10-second frame with ½-second slots. Devices B through M are assigned a single slot per frame. Device A is assigned two slots 5 seconds apart. Device N, perhaps a weather station with multiple sensors, is assigned 2 adjacent ½-second slots, while device O, a repeater, effectively has a 2-second slot.

Figure 8 - TDMA slot allocation example

There must be room within a slot for clock drift between time corrections. The transmission window is therefore shorter than the slot, and the transmission is centered in the slot so that a buffer is left at the beginning and the end of each transmission. Longer slots are more efficient as a smaller percentage of the slot is lost to buffer. Figure 9 illustrates a case where the allowable drift has been sent to +/- 50 msec.

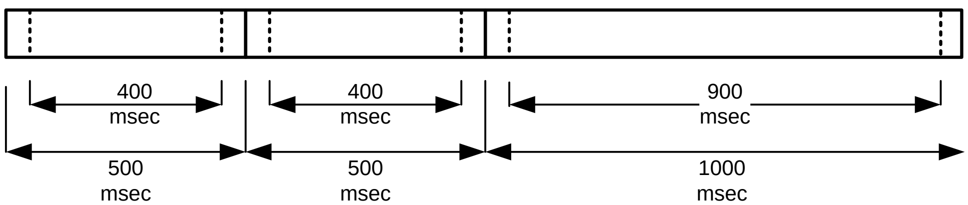

Figure 9 - TDMA slot buffering example

The IND has the task of staying within the slot limits. It will not send messages for which the PTT time would be longer than the configured transmit window. The user, therefore, must be cognizant of the longest time that may be needed to accommodate transmissions from each device. The clock's correction frequency must be set with regard to the expected drift rate, and if a correction cannot be made within a limited time period, the device will use a random offset from its scheduled slot to minimize the contention with any single device.

The Network Transport (MANT) Layer

Overview

The MANT Layer provides a variety of logical services in preparing, transporting and delivering application layer packets to the appropriate application layer processor. It can provide time services, track routing, employ pass-listing, invoke echo suppression and enforce hop limits. The MANT header is designed to support both existing and anticipated new functionalities. The Version field of three bits permits new versions to be introduced with backward compatibility.

MANT Header

To prepare for transmission, the MANT layer receives a PDU from the Application Layer and adds a 6-byte header. Twelve bits of the MANT header carry the payload length – the number of bytes that follow the MANT header. This provides information that permits aggregation of multiple PDUs into a single transmitted frame, i.e., with a single preamble and AirLink header.

Source Address

Two bytes of the MANT header carry the Source Address. This is the identifier of the IND where the message originated. It also serves as a Site ID, uniquely identifying a specific measurement suite. The orderly assignment of Source Addresses is important to prevent duplicate use within an inter-operating area. Several management methods have been proposed, including assignment of regional number blocks, distribution through a coordinating body such as an NHWC committee, or deployment of an automated application system maintained by a neutral body. Specification of this aspect of the ALERT2 protocol will occur in the near future.

VER	Prot ID	TS	AP	DA	Port	Res 3	ACK	AH	Hop Lim	Payload Length 12	Source Address 16

Figure 10 - MANT Header required fields

[Destination Address 16]	[MANT PDU ID 8]	[# of SAs]	[Additional SAs (# * 16)]

Figure 11 - MANT Header optional fields

Protocol ID

The Protocol ID provides 3 bits to identify different types of service. For example, the initial ID of 0 indicates "best effort" service: the message, like ALERT, is simply transmitted with no acknowledgement or retries. Protocol ID 1 is acknowledged message service in which the destination IND 'ACKs' receipt of a message and the source IND retries if an ACK is not received.

Destination Address

Another flag (DA in Figure 10) is used to indicate that a destination address follows. This will permit ALERT2 frames to send information (e.g. control commands) to specific remote devices. Commands may be directed to a repeater's IND MANT layer to remotely enter or modify pass lists or control other repeater or originating modem functions. Control messages may also be directed to the application layer of a remote device that operates equipment such as warning signs, gates or valves. Combined with the acknowledged message service, ALERT2 will offer a robust, two way control capability.

Timestamps

Wherever possible, PDUs are time stamped by the application layer at the source so the exact time of measurement is known. Most APDs do not have accurate clocks, however, while most of the INDs do. For this reason, the MANT, residing in the IND, inspects the Application PDU header and inserts a timestamp if requested. If the IND does not have accurate time – such as an ALERT2 ALOHA site with no GPS – the MANT sets a timestamp request flag (TS, above). The MANT at the next site processing this frame inspects the timestamp request field; if the request is set and the site is able, it will insert the timestamp in the APDU and clear the TS flag.

The ALERT2 protocol uses a 16-bit timestamp that encodes seconds and rolls over at noon and midnight, UTC. On decoding, the timestamp is integrated with the full date/time of arrival to calculate and record the time of origin of the message to within one second.

Path Added and Echo Suppression

The next flag, Path Added, is used to request that the Source Address of the IND be added to the MANT header at each hop the frame transits. When invoked, the complete path taken is contained in the message received at a base station. This provides diagnostic capability for the network manager, while eliminating overhead when it is not required.

Echo Suppression is a MANT service that can be enabled when the IND is configured through the API. When enabled, the repeater IND examines the list of added Source Addresses in the MANT header. If it finds its own address, it does not repeat the frame. Using the Path Added feature is recommended when using echo suppression.

Hop Limit

The Hop Limit field is a three bit value which is set when configuring the originating IND. The field is read at each repeater. If it is '7', it is ignored and the frame is repeated (hop limit is disabled). If it is between 1 and 6, the field's value is decremented and the frame is repeated. If the count is found to be zero, the message is not retransmitted. This provides a means to prevent the system being jammed while a frame is passed back and forth indefinitely between two repeaters, and can also limit the propagation of messages beyond the intended area of operation. A value of 0 inserted in the field at the originating modem will permit only a direct path transmission to the receiver.

Port

The Port field identifies the destination application process to which the message should be directed. For example, an ALERT Concentration PDU is a different protocol from ALERT2 Self-Report PDUs and will be decoded by a different application process. It is therefore directed to a different port. Similarly, the Control protocol to remotely configure INDs operates at the MANT level, and has a unique port assignment.

Pass Listing

Pass Listing is a MANT service. The repeater IND has the capability for individual pass, range pass, individual reject and range reject based on the Source Address in the MANT header. Pass listing can also be based on the Destination Address; this might be done to prevent propagation of reports from neighboring systems. Multiple lists can be built and stored, and more than one list can be enabled at one time. For example a repeater could be configured with a Source Address range pass list, with some addresses in that range blocked by a Source Address individual reject list. The Pass Listing Service uses the most restrictive interpretation when multiple lists are involved; therefore an individual pass list cannot be used to pass IDs within a reject range.

Added Header, ACK and Reserved bits

These fields are provided for future use. The ACK field is used to acknowledge receipt of a message when end-to-end service is implemented. Added Header may be used in the future to indicate another header follows; this would be used to support additional services not included at this time.

Application Layer Protocols

The task of the originating application layer is to put data into formats and structures that are suitable for transport and recognizable to the appropriate application layer protocol at the receiving end. The receiving application layer is tasked with extracting the data encoded in the PDU and making it available to the appropriate process (such as a base station data collector). There are multiple Application Layer protocols, each performing different data functions and targeted to different application processors.

Figure 12 shows three protocols; more may be developed and added in the future. The first is the Self-Report protocol which replaces the standard ALERT functionality of reporting environmental parameters in real time. It has 3 sub-protocols called Types, each of which is specialized for a particular reporting function. The ALERT Compatibility protocol, also known as the ALERT Concentrator, condenses legacy ALERT messages and forwards them in ALERT2 frames. In the future, a Device Control protocol will support sending commands to IND or application devices and receiving acknowledgement from the device.

Each protocol's PDUs are directed to a specific logical port that is identified in the MANT header. Communication across the port may be bi-directional.

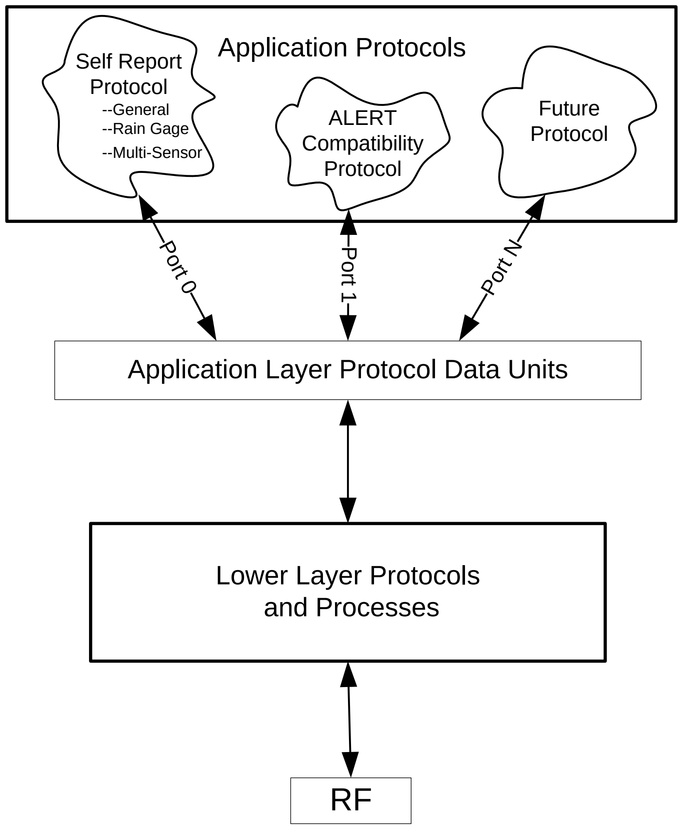

Figure 12 - Application Protocol relationships

SELF REPORT PROTOCOL

The Self-Report protocol for sensor data has three sub-protocols, or types: The General Sensor report, the Tipping Bucket Rain Gage report and the Multi-Sensor report.

THE CONTROL BYTE

Each Self-Report application PDU begins with a one-byte header or control byte. Bits in the control byte work as flags to specify the treatment of the data that follows. The definition of the application layer control byte, low order to high order bits, is presented in Table 1, below.

Application Layer Control Byte		
Bit	**Purpose**	**Notes**
0,1	Version	Current version is 0
2	Timestamp	If set, a16-bit Timestamp follows Control field
3	Test Flag	If set, report is flagged; for marking data as test, out of service, etc
4,5,6	APDU ID	Cyclic counter: to detect duplicate or missing reports
7	Extensibility	If set, a second control byte follows (for future use)

Table 1 - Application Control byte usage

All Application layer data that follows the Control byte is structured in a “Type-Length-Value” (TLV) pattern. A Value at an upper level may comprise another, embedded TLV in the next lower layer, in a recursive message building process.

TIMESTAMP

The value indicates the number of elapsed seconds, since the most recent midnight or noon, in UTC. Together with the full date and time available at the receiver, the time (to the second) of the measurement can be determined, regardless of latencies in the transmission introduced by TDMA. If possible, the application layer inserts the timestamp immediately following the Control byte, and sets the Timestamp flag in the Control byte. If accurate time is not available, the Timestamp bit in the Control byte is cleared, and MANT services will provide the Timestamp, if requested and possible.

TEST FLAG

This 1-bit flag can be used to mark frames that may warrant special treatment by the receiving application. As the name implies, it was created for the use of technicians testing or configuring a site so that the path could be tested and the data could be recorded, but marked as “test” or “out-of-service” reports.

APPLICATION PDU IDENTIFIER

The Application PDU Identifier uses 3 bits in the Control byte. Its default value is 7, meaning that it is disabled. When enabled, it is incremented with each exported PDU and the value cycles between 0 and 6. It is not changed by any downstream process. This provides a means for the base station software to identify duplicate reports, as when a frame has two different paths by which it reaches its destination. Incidentally, duplicate reports are much less of an issue with ALERT2; corrupted reports are either corrected or rejected and duplicate reports will match. Another use of the APDU ID is to determine if a report from a site was missed, as indicated by a gap in the cyclic APDU ID count. The APU ID should not normally be disabled.

TYPE 1: GENERAL SENSOR REPORT

The General Sensor report was developed to concatenate a series of TLVs for individual sensors within the “value” field of the uppermost TLV. This top level consists of the Type field (a “1” in this case), the Type’s length field, and the sensor data field. The length field contains the count of all bytes that follow the length field, to the next type, or if there is none, to the end of the PDU. The length field is extended to two bytes if the message is more than 127 bytes long. There may be one to many sensors in the report; the theoretical limit of 32,767 bytes is constrained by practical considerations such as duration of the transmitted message. A block diagram of the General Sensor report is shown in Figure 13.

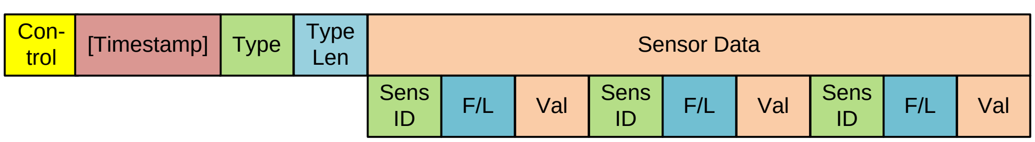

Figure 13 - General Sensor report structure

Unlike ALERT, the Sensor ID only distinguishes a sensor at a site, not across the system. The complete “address” of a sensor is the Source Address (1 to 65,535) and the Sensor ID. In ALERT, it is typical to assign a block of ten or more Sensor IDs to a single site. Given the ALERT limit of 8191 addresses, ALERT addressing can only support about 800 sites; ALERT2’s capacity is at least 80 times greater. At the time of this writing, sensor IDs are not tied by the protocol to any particular type of sensor. However, a convention has been established that reserves IDs 1 through 11 for the most common sensor types, and a list like this may become mandatory. The current convention is presented in Table 2, below:

ID	Sensor Type
0	Reserved (Mandatory)
1	Rain
2	Stage
3	Battery
4	Wind Speed
5	Wind Direction
6	Peak Wind Speed
7	Air Temperature
8	Relative Humidity
9	Air Pressure
10	Status (8 on/off bits)
11	Flow Velocity
255	Timestamp (Mandatory)

Table 2 - Sensor ID convention

The Format/Length (F/L) field consists of 2 parts. The upper 4 bits specify the format; the values 1, 2 and 3 specify an unsigned integer, signed integer and floating point value, respectively. The value of the lower four bits indicates the length, from 0 to 15 bytes. For example, a '3' in the upper 4 bits and a '4' in the lower four bits would specify a 4-byte IEEE floating point single. The value of the byte is 52 decimal or 34 hexadecimal.

Type 2: Tipping Bucket Rain Gage Report

Type 2 was developed to allow multiple tips to be efficiently reported in a single message. In ALERT, a transmission occurs for each tip, which has two undesirable impacts. First, it forces the transmission rate to spike during rain events, with high contention losses at the most critical times. Second, in order to limit those losses, rain measurements are typically limited to 1 mm accuracy. The evaporation from partially full buckets between events leads to a significant under-count of rain totals when the data is used for climatologic purposes.

The Control, Timestamp, Type and TypeLength fields are the same for all three Self-Report types. The top level value field contains the Rain Gage Report. This report carries the most recent accumulator value from the gage (a no-rain status report would end there). Each tip since the previous report is indicated by a single byte Time Offset value representing the number of seconds the tip occurred before transmission. If a 2-byte accumulator is used, the report length will be 9 bytes plus 1 additional byte per tip.

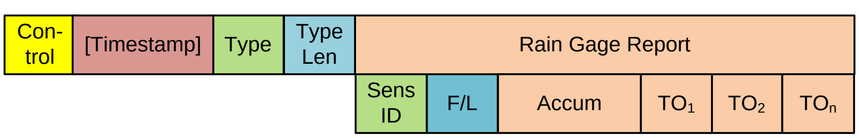

Figure 14 - Rain gage report structure

In ALERT2 ALOHA, the first rain tip starts a count-down timer set to the length of the reporting cycle, and the first time offset carries that value. Subsequent tips will generate timestamps that decrease to 0, at which time the report is transmitted and the timer is shut off until a new tip starts another cycle.

Type 3: Multi-Sensor Report

Type 3 of the Self-Report protocol provides an efficient format for the most common suite of measurements made at existing ALERT sites. One-byte sensor IDs are replaced by a single bit in a Data Flags byte; Format/Length fields are eliminated because the numeric formats are predefined. Where a measurement does not fit this format, the General Sensor Report may be used instead.

Con-trol	[Timestamp]	Type	Type Len	Multi-Sensor Report

Data Flag	A Temp	RH	Baro P	WS	WD	PW	Stage	BV

Figure 15 - Multi-sensor report structure

The proposed sensors, their data format, resolution and units are shown in Table 3, below. These may change before becoming final; also, a different suite, and particularly different units, may be adopted in non-interacting regions.

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

Table 3 - Proposed multi-sensor report configuration

Many ALERT2 sites will consist of a tipping bucket rain gage along with one or more analog sensors such as stage, temperature or relative humidity sensors. Type 1, 2 and 3 messages can be combined in a single frame with a single Header and Timestamp field.

Concentration Protocol

The Concentration protocol was the first to be implemented and is central to the migration path from ALERT to ALERT2. The Concentration protocol is typically implemented at an existing ALERT repeater site, where the received ALERT messages are decoded to a serial stream and sent to the ALERT2 modulator and encoder. All of the ALERT messages arriving during the TDMA frame are aggregated into a single message. TDMA ALERT2 messages are usually transmitted on a separate radio channel from the ALOHA ALERT messages so that data contention losses and corruption errors are eliminated on the repeater output path.

An ALERT message is 40 bits long and contains 24 bits of sensor data. The Concentrator extracts just these 24 bits then adds a 1-byte Time Offset field that contains the number of seconds it was held before it was exported. Figure 16 illustrates the concentration process.

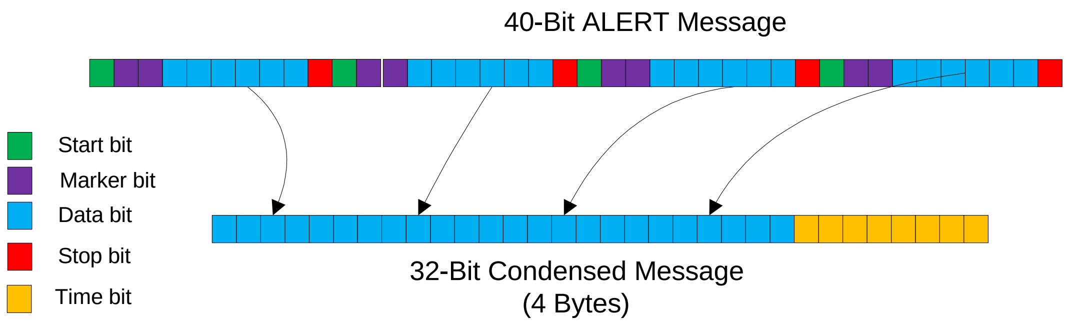

Figure 16 - ALERT Concentrator report construction

If there is a timestamp, the time offsets are subtracted from the timestamp to recover the time of the tip. If there is no timestamp, e.g., it is a directly reporting ALOHA site, the time offsets will be subtracted from the receive time at the destination.

The Concentrator Protocol serves a single purpose, and therefore does not require a TLV structure. The Application PDU consists of the Control byte and the Timestamp, followed directly by concatenated 4-byte ALERT and Time Offset values.

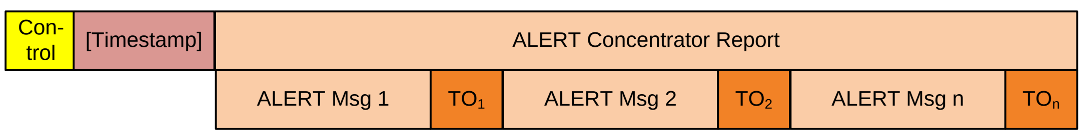

Figure 17 - ALERT Concentration Protocol structure

The efficiency of the Concentrator protocol increases as the traffic rate goes up. The first message, carried in a 228-msec first block, results in a 32% reduction in air time per message. However, up to three full ALERT messages are carried in the same first block, so at this traffic rate the air time per message drops to 76 msec, more than a 4-fold increase in efficiency. When the traffic is sufficient to fill a second block to capacity, 11 ALERT messages are carried in a 388-msec message at an air time of 35 msec per message, more than a 9-fold improvement in channel efficiency. A single 2-second slot in a 15-second frame has a theoretical capacity of over 20,000 ALERT messages per hour.

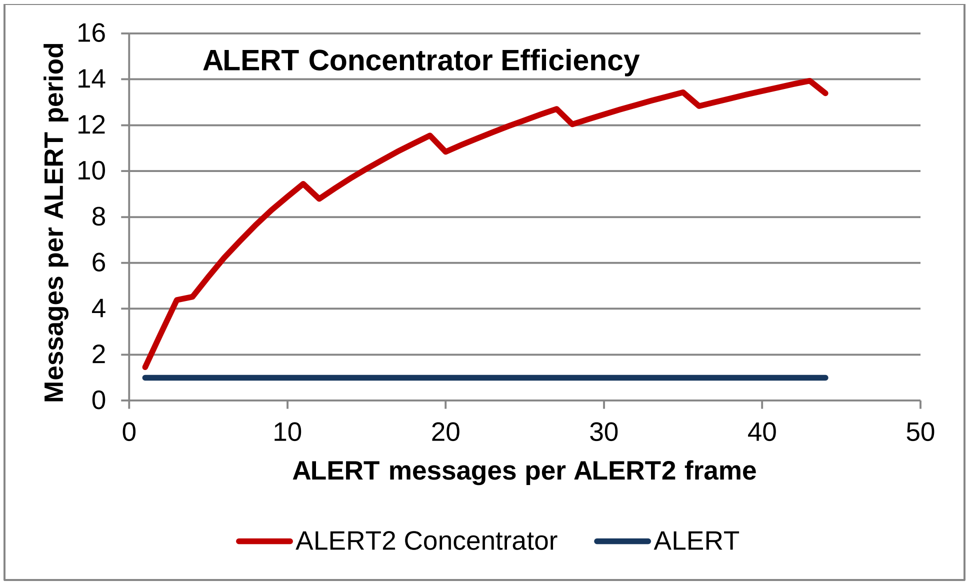

Figure 18 - Concentrator efficiency as a function of traffic level

The figure above illustrates the improvement in efficiency with increasing traffic. The sawtooth pattern results from the addition of a new block, with its 16-byte R-S FEC overhead with the first byte of data.

SUMMARY

ALERT has served the hydrologic warning community well for decades. Pressure to use valuable RF spectrum more efficiently and expectations of information density, precision and quality have increased dramatically in this time. The ALERT2 protocol will enable the hydrologic warning community to retain the benefits of ALERT:

- robust paths due to high energy per bit
- low power requirements at remote sites
- redundant receive point capability with broadcast
- open standard that encourages competition

With ALERT2 come new benefits:

- Increased capacity due to higher speed
- Contention-free transmission using TDMA, eliminating data losses
- Error-free data
- Much larger ID space
- Much more complete information carried (engineering units, flexible messages)
- Potential to add functionality due to message design
- NHWC framework that provides rigorous standards and an evolutionary path

We believe ALERT2 will greatly improve the data collection capabilities of the ALERT community, and that its pragmatic benefits may lead it to be adopted by wider domains.