AltOS Telemetry

Packet Definitions

1 July 2011

Keith Packard <keithp@keithp.com>

Revision History		
Revision 0.1	1 July 2011	
Initial content		

Table of Contents

1. Packet Format Design ... 1
2. Packet Formats ... 2
   2.1. Packet Header ... 2
   2.2. TeleMetrum v1.x, TeleMini and TeleNano Sensor Data ... 2
   2.3. TeleMega Sensor Data ... 3
   2.4. TeleMetrum v2 Sensor Data ... 4
   2.5. Configuration Data ... 5
   2.6. GPS Location ... 6
   2.7. GPS Satellite Data ... 7
   2.8. Companion Data ... 8
3. Data Transmission ... 8
   3.1. Modulation Scheme ... 8
   3.2. Error Correction ... 9
4. TeleDongle serial packet format ... 9
5. History and Motivation ... 10

1. Packet Format Design

AltOS telemetry data is split into multiple different packets, all the same size, but each includs an identifier so that the ground station can distinguish among different types. A single flight board will transmit multiple packet types, each type on a different schedule. The ground software need look for only a single packet size, and then decode the information within the packet and merge data from multiple packets to construct the full flight computer state.

Each AltOS packet is 32 bytes long. This size was chosen based on the known telemetry data requirements. The power of two size allows them to be stored easily in flash memory without having them split across blocks or leaving gaps at the end.

All packet types start with a five byte header which encodes the device serial number, device clock value and the packet type. The remaining 27 bytes encode type-specific data.

2. Packet Formats

This section first defines the packet header common to all packets and then the per-packet data layout.

2.1. Packet Header

Table 1. Telemetry Packet Header

Offset	Data Type	Name	Description
0	uint16_t	serial	Device serial Number
2	uint16_t	tick	Device time in 100ths of a second
4	uint8_t	type	Packet type

Each packet starts with these five bytes which serve to identify which device has transmitted the packet, when it was transmitted and what the rest of the packet contains.

2.2. TeleMetrum v1.x, TeleMini and TeleNano Sensor Data

Table 2. Sensor Packet Type

Type	Description
0x01	TeleMetrum v1.x Sensor Data
0x02	TeleMini Sensor Data
0x03	TeleNano Sensor Data

TeleMetrum v1.x, TeleMini and TeleNano share this same packet format for sensor data. Each uses a distinct packet type so that the receiver knows which data values are valid and which are undefined.

Sensor Data packets are transmitted once per second on the ground, 10 times per second during ascent and once per second during descent and landing

Table 3. Sensor Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	state	Flight state
6	int16_t	accel	accelerometer (TM only)
8	int16_t	pres	pressure sensor
10	int16_t	temp	temperature sensor
12	int16_t	v_batt	battery voltage
14	int16_t	sense_d	drogue continuity sense (TM/Tm)
16	int16_t	sense_m	main continuity sense (TM/Tm)
18	int16_t	acceleration	m/s^2 * 16
20	int16_t	speed	m/s * 16
22	int16_t	height	m

24	int16_t	ground_pres	Average barometer reading on ground
26	int16_t	ground_accel	TM
28	int16_t	accel_plus_g	TM
30	int16_t	accel_minus_g	TM

2.3. TeleMega Sensor Data

Table 4. TeleMega Packet Type

Type	Description
0x08	TeleMega IMU Sensor Data
0x09	TeleMega Kalman and Voltage Data

TeleMega has a lot of sensors, and so it splits the sensor data into two packets. The raw IMU data are sent more often; the voltage values don't change very fast, and the Kalman values can be reconstructed from the IMU data.

IMU Sensor Data packets are transmitted once per second on the ground, 10 times per second during ascent and once per second during descent and landing

Kalman and Voltage Data packets are transmitted once per second on the ground, 5 times per second during ascent and once per second during descent and landing

The high-g accelerometer is reported separately from the data for the 9-axis IMU (accel/gyro/mag). The 9-axis IMU is mounted so that the X axis is "across" the board (along the short axis0, the Y axis is "along" the board (along the long axis, with the high-g accelerometer) and the Z axis is "through" the board (perpendicular to the board). Rotation measurements are around the respective axis, so Y rotation measures the spin rate of the rocket while X and Z rotation measure the tilt rate.

The overall tilt angle of the rocket is computed by first measuring the orientation of the rocket on the pad using the 3 axis accelerometer, and then integrating the overall tilt rate from the 3 axis gyroscope to compute the total orientation change of the airframe since liftoff.

Table 5. TeleMega IMU Sensor Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	orient	Angle from vertical in degrees
6	int16_t	accel	High G accelerometer
8	int32_t	pres	pressure (Pa * 10)
12	int16_t	temp	temperature (°C * 100)
14	int16_t	accel_x	X axis acceleration (across)
16	int16_t	accel_y	Y axis acceleration (along)
18	int16_t	accel_z	Z axis acceleration (through)
20	int16_t	gyro_x	X axis rotation (across)
22	int16_t	gyro_y	Y axis rotation (along)

24	int16_t	gyro_z	Z axis rotation (through)
26	int16_t	mag_x	X field strength (across)
28	int16_t	mag_y	Y field strength (along)
30	int16_t	mag_z	Z field strength (through)

Table 6. TeleMega Kalman and Voltage Data Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	state	Flight state
6	int16_t	v_batt	battery voltage
8	int16_t	v_pyro	pyro battery voltage
10	int8_t[6]	sense	pyro continuity sense
16	int32_t	ground_pres	Average barometer reading on ground
20	int16_t	ground_accel	Average accelerometer reading on ground
22	int16_t	accel_plus_g	Accel calibration at +1g
24	int16_t	accel_minus_g	Accel calibration at -1g
26	int16_t	acceleration	m/s^2 * 16
28	int16_t	speed	m/s * 16
30	int16_t	height	m

2.4. TeleMetrum v2 Sensor Data

Table 7. TeleMetrum v2 Packet Type

Type	Description
0x0A	TeleMetrum v2 Sensor Data
0x0B	TeleMetrum v2 Calibration Data

TeleMetrum v2 has higher resolution barometric data than TeleMetrum v1, and so the constant calibration data is split out into a separate packet.

TeleMetrum v2 Sensor Data packets are transmitted once per second on the ground, 10 times per second during ascent and once per second during descent and landing

TeleMetrum v2 Calibration Data packets are always transmitted once per second.

Table 8. TeleMetrum v2 Sensor Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	state	Flight state
6	int16_t	accel	accelerometer
8	int32_t	pres	pressure sensor (Pa * 10)
12	int16_t	temp	temperature sensor (°C * 100)

14	int16_t	acceleration	m/s² * 16
16	int16_t	speed	m/s * 16
18	int16_t	height	m
20	int16_t	v_batt	battery voltage
22	int16_t	sense_d	drogue continuity sense
24	int16_t	sense_m	main continuity sense
26	pad[6]	pad bytes	

Table 9. TeleMetrum v2 Calibration Data Packet Contents

Offset	Data Type	Name	Description
5	pad[3]	pad bytes	
8	int32_t	ground_pres	Average barometer reading on ground
12	int16_t	ground_accel	Average accelerometer reading on ground
14	int16_t	accel_plus_g	Accel calibration at +1g
16	int16_t	accel_minus_g	Accel calibration at -1g
18	pad[14]	pad bytes	

2.5. Configuration Data

Table 10. Configuration Packet Type

Type	Description
0x04	Configuration Data

This provides a description of the software installed on the flight computer as well as any user-specified configuration data.

Configuration data packets are transmitted once per second during all phases of the flight

Table 11. Configuration Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	type	Device type
6	uint16_t	flight	Flight number
8	uint8_t	config_major	Config major version
9	uint8_t	config_minor	Config minor version
10	uint16_t	apogee_delay	Apogee deploy delay in seconds
12	uint16_t	main_deploy	Main deploy alt in meters
14	uint16_t	flight_log_max	Maximum flight log size (kB)
16	char	callsign[8]	Radio operator identifier
24	char	version[8]	Software version identifier

2.6. GPS Location

Table 12. GPS Packet Type

Type	Description
0x05	GPS Location

This packet provides all of the information available from the GPS receiver—position, time, speed and precision estimates.

GPS Location packets are transmitted once per second during all phases of the flight

Table 13. GPS Location Packet Contents

Offset	Data Type	Name	Description
5	uint8_t	flags	See GPS Flags table below
6	int16_t	altitude	m
8	int32_t	latitude	degrees * 107
12	int32_t	longitude	degrees * 107
16	uint8_t	year	
17	uint8_t	month	
18	uint8_t	day	
19	uint8_t	hour	
20	uint8_t	minute	
21	uint8_t	second	
22	uint8_t	pdop	* 5
23	uint8_t	hdop	* 5
24	uint8_t	vdop	* 5
25	uint8_t	mode	See GPS Mode table below
26	uint16_t	ground_speed	cm/s
28	int16_t	climb_rate	cm/s
30	uint8_t	course	/ 2
31	uint8_t	unused[1]	

Packed into a one byte field are status flags and the count of satellites used to compute the position fix. Note that this number may be lower than the number of satellites being tracked; the receiver will not use information from satellites with weak signals or which are close enough to the horizon to have significantly degraded position accuracy.

Table 14. GPS Flags

Bits	Name	Description
0-3	nsats	Number of satellites in solution

4	valid	GPS solution is valid
5	running	GPS receiver is operational
6	date_valid	Reported date is valid
7	course_valid	ground speed, course and climb rates are valid

Here are all of the valid GPS operational modes. Altus Metrum products will only ever report *N* (not valid), *A* (Autonomous) modes or *E* (Estimated). The remaining modes are either testing modes or require additional data.

Table 15. GPS Mode

Mode	Name	Description
N	Not Valid	All data are invalid
A	Autonomous mode	Data are derived from satellite data
D	Differential Mode	Data are augmented with differential data from a known ground station. The SkyTraq unit in TeleMetrum does not support this mode
E	Estimated	Data are estimated using dead reckoning from the last known data
M	Manual	Data were entered manually
S	Simulated	GPS receiver testing mode

2.7. GPS Satellite Data

Table 16. GPS Satellite Data Packet Type

Type	Description
0x06	GPS Satellite Data

This packet provides space vehicle identifiers and signal quality information in the form of a C/N1 number for up to 12 satellites. The order of the svids is not specified.

GPS Satellite data are transmitted once per second during all phases of the flight.

Table 17. GPS Satellite Data Contents

Offset	Data Type	Name	Description
5	uint8_t	channels	Number of reported satellite information
6	sat_info_t	sats[12]	See Per-Satellite data table below
30	uint8_t	unused[2]	

Table 18. GPS Per-Satellite data (sat_info_t)

Offset	Data Type	Name	Description
0	uint8_t	svid	Space Vehicle Identifier

1	uint8_t	c_n_1	C/N1 signal quality indicator

2.8. Companion Data

Table 19. Companion Data Packet Type

Type	Description
0x07	Companion Data

When a companion board is attached to TeleMega or TeleMetrum, it can provide telemetry data to be included in the downlink. The companion board can provide up to 12 16-bit data values.

The companion board itself specifies the transmission rate. On the ground and during descent, that rate is limited to one packet per second. During ascent, that rate is limited to 10 packets per second.

Table 20. Companion Data Contents

Offset	Data Type	Name	Description
5	uint8_t	board_id	Type of companion board attached
6	uint8_t	update_period	How often telemetry is sent, in 1/100ths of a second
7	uint8_t	channels	Number of data channels supplied
8	uint16_t[12]	companion_data	Up to 12 channels of 16-bit companion data

3. Data Transmission

Altus Metrum devices use Texas Instruments sub-GHz digital radio products. Ground stations use parts with HW FEC while some flight computers perform FEC in software. TeleGPS is transmit-only.

Table 21. Altus Metrum Radio Parts

Part Number	Description	Used in
CC1111	10mW transceiver with integrated SoC	TeleDongle v0.2, TeleBT v1.0, TeleMetrum v1.x, TeleMini
CC1120	35mW transceiver with SW FEC	TeleMetrum v2, TeleMega
CC1200	35mW transceiver with HW FEC	TeleDongle v3.0, TeleBT v3.0
CC115L	14mW transmitter with SW FEC	TeleGPS

3.1. Modulation Scheme

Texas Instruments provides a tool for computing modulation parameters given a desired modulation format and basic bit rate.

While we might like to use something with better low-signal performance like BPSK, the radios we use don't support that, but do support Gaussian frequency shift keying (GFSK). Regular frequency shift keying (FSK) encodes the signal by switching the carrier between two frequencies. The Gaussian version is essentially the same, but the shift between frequencies gently follows a gaussian curve, rather than switching immediately. This tames the bandwidth of the signal without affecting the ability to transmit data.

For AltOS, there are three available bit rates, 38.4kBaud, 9.6kBaud and 2.4kBaud resulting in the following signal parmeters:

Table 22. Modulation Scheme

Rate	Deviation	Receiver Bandwidth
38.4kBaud	20.5kHz	100kHz
9.6kBaud	5.125kHz	25kHz
2.4kBaud	1.5kHz	5kHz

3.2. Error Correction

The cc1111 and cc1200 provide forward error correction in hardware; on the cc1120 and cc115l that's done in software. AltOS uses this to improve reception of weak signals. As it's a rate 1/2 encoding, each bit of data takes two bits when transmitted, so the effective data rate is half of the raw transmitted bit rate.

Table 23. Error Correction

Parameter	Value	Description
Error Correction	Convolutional coding	1/2 rate, constraint length m=4
Interleaving	4 x 4	Reduce effect of noise burst
Data Whitening	XOR with 9-bit PNR	Rotate right with bit 8 = bit 0 xor bit 5, initial value 111111111

4. TeleDongle serial packet format

TeleDongle does not do any interpretation of the packet data, instead it is configured to receive packets of a specified length (32 bytes in this case). For each received packet, TeleDongle produces a single line of text. This line starts with the string "TELEM " and is followed by a list of hexadecimal encoded bytes.

```
TELEM 224f01080b05765e00701f1a1bbeb8d7b60b070605140c000600000000000000003fa988
```

The hexadecimal encoded string of bytes contains a length byte, the packet data, two bytes added by the cc1111 radio receiver hardware and finally a checksum so that the host software can validate that the line was transmitted without any errors.

Table 24. TeleDongle serial Packet Format

Offset	Name	Example	Description
0	length	22	Total length of data bytes in the line. Note that this includes the added RSSI and status bytes
1 ·· length-3	packet	4f ·· 00	Bytes of actual packet data
length-2	rssi	3f	Received signal strength. dBm = rssi / 2 - 74

length-1	lqi	a9	Link Quality Indicator and CRC status. Bit 7 is set when the CRC is correct
length	checksum	88	(0x5a + sum(bytes 1 ‥ length-1)) % 256

5. History and Motivation

The original AltoOS telemetry mechanism encoded everything available piece of information on the TeleMetrum hardware into a single unified packet. Initially, the packets contained very little data—some raw sensor readings along with the current GPS coordinates when a GPS receiver was connected. Over time, the amount of data grew to include sensor calibration data, GPS satellite information and a host of internal state information designed to help diagnose flight failures in case of a loss of the on-board flight data.

Because every packet contained all of the data, packets were huge—95 bytes long. Much of the information was also specific to the TeleMetrum hardware. With the introduction of the TeleMini flight computer, most of the data contained in the telemetry packets was unavailable. Initially, a shorter, but still comprehensive packet was implemented. This required that the ground station be pre-configured as to which kind of packet to expect.

The development of several companion boards also made the shortcomings evident—each companion board would want to include telemetry data in the radio link; with the original design, the packet would have to hold the new data as well, requiring additional TeleMetrum and ground station changes.

