The Altus Metrum System

An Owner's Manual for TeleMetrum and TeleDongle Devices

Bdale Garbee

Keith Packard

Bob Finch

Anthony Towns

The Altus Metrum System: An Owner's Manual for TeleMetrum and TeleDongle Devices

Bdale Garbee
Keith Packard
Bob Finch
Anthony Towns
Copyright © 2010 Bdale Garbee and Keith Packard

Acknowledgements

Thanks to Bob Finch, W9YA, NAR 12965, TRA 12350 for writing "The Mere-Mortals Quick Start/Usage Guide to the Altus Metrum Starter Kit" which has turned into the Getting Started chapter in this book. Bob was one of our first customers for a production TeleMetrum, and the enthusiasm that led to his contribution of this section is immensely gratifying and highy appreciated!

And thanks to Anthony (AJ) Towns for contributing the AltosUI graphing and site map code and documentation. Free software means that our customers and friends can become our collaborators, and we certainly appreciate this level of contribution.

Have fun using these products, and we hope to meet all of you out on the rocket flight line somewhere.

Bdale Garbee, KB0G
NAR #87103, TRA #12201

Keith Packard, KD7SQG
NAR #88757, TRA #12200

Table of Contents

Chapter 1. Introduction and Overview

Welcome to the Altus Metrum community! Our circuits and software reflect our passion for both hobby rocketry and Free Software. We hope their capabilities and performance will delight you in every way, but by releasing all of our hardware and software designs under open licenses, we also hope to empower you to take as active a role in our collective future as you wish!

The focal point of our community is TeleMetrum, a dual deploy altimeter with fully integrated GPS and radio telemetry as standard features, and a "companion interface" that will support optional capabilities in the future.

Complementing TeleMetrum is TeleDongle, a USB to RF interface for communicating with TeleMetrum. Combined with your choice of antenna and notebook computer, TeleDongle and our associated user interface software form a complete ground station capable of logging and displaying in-flight telemetry, aiding rocket recovery, then processing and archiving flight data for analysis and review.

More products will be added to the Altus Metrum family over time, and we currently envision that this will be a single, comprehensive manual for the entire product family.

Chapter 2. Getting Started

The first thing to do after you check the inventory of parts in your "starter kit" is to charge the battery by plugging it into the corresponding socket of the TeleMetrum and then using the USB A to mini B cable to plug the Telemetrum into your computer's USB socket. The TeleMetrum circuitry will charge the battery whenever it is plugged in, because the TeleMetrum's on-off switch does NOT control the charging circuitry. When the GPS chip is initially searching for satellites, TeleMetrum will consume more current than it can pull from the usb port, so the battery must be attached in order to get satellite lock. Once GPS is locked, the current consumption goes back down enough to enable charging while running. So it's a good idea to fully charge the battery as your first item of business so there is no issue getting and maintaining satellite lock. The yellow charge indicator led will go out when the battery is nearly full and the charger goes to trickle charge. It can take several hours to fully recharge a deeply discharged battery.

The other active device in the starter kit is the TeleDongle USB to RF interface. If you plug it in to your Mac or Linux computer it should "just work", showing up as a serial port device. Windows systems need driver information that is part of the AltOS download to know that the existing USB modem driver will work. If you are using Linux and are having problems, try moving to a fresher kernel (2.6.33 or newer), as the USB serial driver had ugly bugs in some earlier versions.

Next you should obtain and install the AltOS utilities. These include the AltosUI ground station program, current firmware images for TeleMetrum and TeleDongle, and a number of standalone utilities that are rarely needed. Pre-built binary packages are available for Debian Linux, Microsoft Windows, and recent MacOSX versions. Full sourcecode and build instructions for some other Linux variants are also available. The latest version may always be downloaded from http://altusmetrum.org/AltOS.

Both Telemetrum and TeleDongle can be directly communicated with using USB ports. The first thing you should try after getting both units plugged into to your computer's usb port(s) is to run 'ao-list' from a terminal-window to see what port-device-name each device has been assigned by the operating system. You will need this information to access the devices via their respective on-board firmware and data using other command line programs in the AltOS software suite.

To access the device's firmware for configuration you need a terminal program such as you would use to talk to a modem. The software authors prefer using the program 'cu' which comes from the UUCP package on most Unix-like systems such as Linux. An example command line for cu might be 'cu -l /dev/ttyACM0', substituting the correct number indicated from running the ao-list program. Another reasonable terminal program for Linux is 'cutecom'. The default 'escape' character used by CU (i.e. the character you use to issue commands to cu itself instead of sending the command as input to the connected device) is a '~'. You will need this for use in only two different ways during normal operations. First is to exit the program by sending a '~.' which is called a 'escape-disconnect' and allows you to close-out from 'cu'. The second use will be outlined later.

Both TeleMetrum and TeleDongle share the concept of a two level command set in their firmware. The first layer has several single letter commands. Once you are using 'cu' (or 'cutecom') sending (typing) a '?' returns a full list of these commands. The second level are configuration sub-commands accessed using the 'c' command, for instance typing 'c?' will give you this second level of commands (all of which require the letter 'c' to access). Please note that most configuration options are stored only in DataFlash memory, and only TeleMetrum has this memory to save the various values entered like the channel number and your callsign when powered off. TeleDongle requires that you set these each time you plug it in, which ao-view can help with.

Try setting these config ('c' or second level menu) values. A good place to start is by setting your call sign. By default, the boards use 'N0CALL' which is cute, but not exactly legal! Spend a few minutes getting comfortable with the units, their firmware, and 'cu' (or possibly 'cutecom'). For instance, try to send (type)

a 'c r 2' and verify the channel change by sending a 'c s'. Verify you can connect and disconnect from the units while in your terminal program by sending the escape-disconnect mentioned above.

Note that the 'reboot' command, which is very useful on TeleMetrum, will likely just cause problems with the dongle. The *correct* way to reset the dongle is just to unplug and re-plug it.

A fun thing to do at the launch site and something you can do while learning how to use these units is to play with the rf-link access of the TeleMetrum from the TeleDongle. Be aware that you *must* create some physical separation between the devices, otherwise the link will not function due to signal overload in the receivers in each device.

Now might be a good time to take a break and read the rest of this manual, particularly about the two "modes" that the TeleMetrum can be placed in and how the position of the TeleMetrum when booting up will determine whether the unit is in "pad" or "idle" mode.

You can access a TeleMetrum in idle mode from the Teledongle's USB connection using the rf link by issuing a 'p' command to the TeleDongle. Practice connecting and disconnecting ('~~' while using 'cu') from the TeleMetrum. If you cannot escape out of the "p" command, (by using a '~~' when in CU) then it is likely that your kernel has issues. Try a newer version.

Using this rf link allows you to configure the TeleMetrum, test fire e-matches and igniters from the flight line, check pyro-match continuity and so forth. You can leave the unit turned on while it is in 'idle mode' and then place the rocket vertically on the launch pad, walk away and then issue a reboot command. The TeleMetrum will reboot and start sending data having changed to the "pad" mode. If the TeleDongle is not receiving this data, you can disconnect 'cu' from the Teledongle using the procedures mentioned above and THEN connect to the TeleDongle from inside 'ao-view'. If this doesn't work, disconnect from the TeleDongle, unplug it, and try again after plugging it back in.

Eventually the GPS will find enough satellites, lock in on them, and 'ao-view' will both auditorially announce and visually indicate that GPS is ready. Now you can launch knowing that you have a good data path and good satellite lock for flight data and recovery. Remember you MUST tell ao-view to connect to the TeleDongle explicitly in order for ao-view to be able to receive data.

Both RDF (radio direction finding) tones from the TeleMetrum and GPS trekking data are available and together are very useful in locating the rocket once it has landed. (The last good GPS data received before touch-down will be on the data screen of 'ao-view'.)

Once you have recovered the rocket you can download the eeprom contents using either 'ao-dumplog' (or possibly 'ao-eeprom'), over either a USB cable or over the radio link using TeleDongle. And by following the man page for 'ao-postflight' you can create various data output reports, graphs, and even kml data to see the flight trajectory in google-earth. (Moving the viewing angle making sure to connect the yellow lines while in google-earth is the proper technique.)

As for ao-view.... some things are in the menu but don't do anything very useful. The developers have stopped working on ao-view to focus on a new, cross-platform ground station program. So ao-view may or may not be updated in the future. Mostly you just use the Log and Device menus. It has a wonderful display of the incoming flight data and I am sure you will enjoy what it has to say to you once you enable the voice output!

FAQ

The altimeter (TeleMetrum) seems to shut off when disconnected from the computer. Make sure the battery is adequately charged. Remember the unit will pull more power than the USB port can deliver before the GPS enters "locked" mode. The battery charges best when TeleMetrum is turned off.

It's impossible to stop the TeleDongle when it's in "p" mode, I have to unplug the USB cable? Make sure you have tried to "escape out" of this mode. If this doesn't work the reboot procedure for the TeleDongle *is* to simply unplug it. 'cu' however will retain it's outgoing buffer IF your "escape out" ('~~') does not work. At this point using either 'ao-view' (or possibly 'cutemon') instead of 'cu' will 'clear' the issue and allow renewed communication.

The amber LED (on the TeleMetrum/altimeter) lights up when both battery and USB are connected. Does this mean it's charging? Yes, the yellow LED indicates the charging at the 'regular' rate. If the led is out but the unit is still plugged into a USB port, then the battery is being charged at a 'trickle' rate.

There are no "dit-dah-dah-dit" sound like the manual mentions? That's the "pad" mode. Weak batteries might be the problem. It is also possible that the unit is horizontal and the output is instead a "dit-dit" meaning 'idle'.

It's unclear how to use 'ao-view' and other programs when 'cu' is running. You cannot have more than one program connected to the TeleDongle at one time without apparent data loss as the incoming data will not make it to both programs intact. Disconnect whatever programs aren't currently being used.

How do I save flight data? Live telemetry is written to file(s) whenever 'ao-view' is connected to the TeleDongle. The file area defaults to ~/altos but is easily changed using the menus in 'ao-view'. The files that are written end in '.telem'. The after-flight data-dumped files will end in .eeprom and represent continuous data unlike the rf-linked .telem files that are subject to the turnarounds/data-packaging time slots in the half-duplex rf data path. See the above instructions on what and how to save the eeprom stored data after physically retrieving your TeleMetrum. Make sure to save the on-board data after each flight, as the current firmware will over-write any previous flight data during a new flight.

Chapter 3. Specifications

- Recording altimeter for model rocketry.

- Supports dual deployment (can fire 2 ejection charges).

- 70cm ham-band transceiver for telemetry downlink.

- Barometric pressure sensor good to 45k feet MSL.

- 1-axis high-g accelerometer for motor characterization, capable of +/- 50g using default part.

- On-board, integrated GPS receiver with 5hz update rate capability.

- On-board 1 megabyte non-volatile memory for flight data storage.

- USB interface for battery charging, configuration, and data recovery.

- Fully integrated support for LiPo rechargeable batteries.

- Uses LiPo to fire e-matches, support for optional separate pyro battery if needed.

- 2.75 x 1 inch board designed to fit inside 29mm airframe coupler tube.

Chapter 4. Handling Precautions

TeleMetrum is a sophisticated electronic device. When handled gently and properly installed in an airframe, it will deliver impressive results. However, like all electronic devices, there are some precautions you must take.

The Lithium Polymer rechargeable batteries used with TeleMetrum have an extraordinary power density. This is great because we can fly with much less battery mass than if we used alkaline batteries or previous generation rechargeable batteries... but if they are punctured or their leads are allowed to short, they can and will release their energy very rapidly! Thus we recommend that you take some care when handling our batteries and consider giving them some extra protection in your airframe. We often wrap them in suitable scraps of closed-cell packing foam before strapping them down, for example.

The TeleMetrum barometric sensor is sensitive to sunlight. In normal mounting situations, it and all of the other surface mount components are "down" towards whatever the underlying mounting surface is, so this is not normally a problem. Please consider this, though, when designing an installation, for example, in a 29mm airframe with a see-through plastic payload bay.

The TeleMetrum barometric sensor sampling port must be able to "breathe", both by not being covered by foam or tape or other materials that might directly block the hole on the top of the sensor, but also by having a suitable static vent to outside air.

As with all other rocketry electronics, TeleMetrum must be protected from exposure to corrosive motor exhaust and ejection charge gasses.

Chapter 5. Hardware Overview

TeleMetrum is a 1 inch by 2.75 inch circuit board. It was designed to fit inside coupler for 29mm airframe tubing, but using it in a tube that small in diameter may require some creativity in mounting and wiring to succeed! The default 1/4 wave UHF wire antenna attached to the center of the nose-cone end of the board is about 7 inches long, and wiring for a power switch and the e-matches for apogee and main ejection charges depart from the fin can end of the board. Given all this, an ideal "simple" avionics bay for TeleMetrum should have at least 10 inches of interior length.

A typical TeleMetrum installation using the on-board GPS antenna and default wire UHF antenna involves attaching only a suitable Lithium Polymer battery, a single pole switch for power on/off, and two pairs of wires connecting e-matches for the apogee and main ejection charges.

By default, we use the unregulated output of the LiPo battery directly to fire ejection charges. This works marvelously with standard low-current e-matches like the J-Tek from MJG Technologies, and with Quest Q2G2 igniters. However, if you want or need to use a separate pyro battery, you can do so by adding a second 2mm connector to position B2 on the board and cutting the thick pcb trace connecting the LiPo battery to the pyro circuit between the two silk screen marks on the surface mount side of the board shown here [insert photo]

We offer two choices of pyro and power switch connector, or you can choose neither and solder wires directly to the board. All three choices are reasonable depending on the constraints of your airframe. Our favorite option when there is sufficient room above the board is to use the Tyco pin header with polarization and locking. If you choose this option, you crimp individual wires for the power switch and e-matches into a mating connector, and installing and removing the TeleMetrum board from an airframe is as easy as plugging or unplugging two connectors. If the airframe will not support this much height or if you want to be able to directly attach e-match leads to the board, we offer a screw terminal block. This is very similar to what most other altimeter vendors provide and so may be the most familiar option. You'll need a very small straight blade screwdriver to connect and disconnect the board in this case, such as you might find in a jeweler's screwdriver set. Finally, you can forego both options and solder wires directly to the board, which may be the best choice for minimum diameter and/or minimum mass designs.

For most airframes, the integrated GPS antenna and wire UHF antenna are a great combination. However, if you are installing in a carbon-fiber electronics bay which is opaque to RF signals, you may need to use off-board external antennas instead. In this case, you can order TeleMetrum with an SMA connector for the UHF antenna connection, and you can unplug the integrated GPS antenna and select an appropriate off-board GPS antenna with cable terminating in a U.FL connector.

Chapter 6. System Operation

Firmware Modes

The AltOS firmware build for TeleMetrum has two fundamental modes, "idle" and "flight". Which of these modes the firmware operates in is determined by the orientation of the rocket (well, actually the board, of course...) at the time power is switched on. If the rocket is "nose up", then TeleMetrum assumes it's on a rail or rod being prepared for launch, so the firmware chooses flight mode. However, if the rocket is more or less horizontal, the firmware instead enters idle mode.

At power on, you will hear three beeps ("S" in Morse code for startup) and then a pause while TeleMetrum completes initialization and self tests, and decides which mode to enter next.

In flight or "pad" mode, TeleMetrum turns on the GPS system, engages the flight state machine, goes into transmit-only mode on the RF link sending telemetry, and waits for launch to be detected. Flight mode is indicated by an audible "di-dah-dah-dit" ("P" for pad) on the beeper, followed by beeps indicating the state of the pyrotechnic igniter continuity. One beep indicates apogee continuity, two beeps indicate main continuity, three beeps indicate both apogee and main continuity, and one longer "brap" sound indicates no continuity. For a dual deploy flight, make sure you're getting three beeps before launching! For apogee-only or motor eject flights, do what makes sense.

In idle mode, you will hear an audible "di-dit" ("I" for idle), and the normal flight state machine is disengaged, thus no ejection charges will fire. TeleMetrum also listens on the RF link when in idle mode for packet mode requests sent from TeleDongle. Commands can be issued to a TeleMetrum in idle mode over either USB or the RF link equivalently. Idle mode is useful for configuring TeleMetrum, for extracting data from the on-board storage chip after flight, and for ground testing pyro charges.

One "neat trick" of particular value when TeleMetrum is used with very large airframes, is that you can power the board up while the rocket is horizontal, such that it comes up in idle mode. Then you can raise the airframe to launch position, use a TeleDongle to open a packet connection, and issue a 'reset' command which will cause TeleMetrum to reboot, realize it's now nose-up, and thus choose flight mode. This is much safer than standing on the top step of a rickety step-ladder or hanging off the side of a launch tower with a screw-driver trying to turn on your avionics before installing igniters!

GPS

TeleMetrum includes a complete GPS receiver. See a later section for a brief explanation of how GPS works that will help you understand the information in the telemetry stream. The bottom line is that the TeleMetrum GPS receiver needs to lock onto at least four satellites to obtain a solid 3 dimensional position fix and know what time it is!

TeleMetrum provides backup power to the GPS chip any time a LiPo battery is connected. This allows the receiver to "warm start" on the launch rail much faster than if every power-on were a "cold start" for the GPS receiver. In typical operations, powering up TeleMetrum on the flight line in idle mode while performing final airframe preparation will be sufficient to allow the GPS receiver to cold start and acquire lock. Then the board can be powered down during RSO review and installation on a launch rod or rail. When the board is turned back on, the GPS system should lock very quickly, typically long before igniter installation and return to the flight line are complete.

Ground Testing

An important aspect of preparing a rocket using electronic deployment for flight is ground testing the recovery system. Thanks to the bi-directional RF link central to the Altus Metrum system, this can be accomplished in a TeleMetrum-equipped rocket without as much work as you may be accustomed to with other systems. It can even be fun!

Just prep the rocket for flight, then power up TeleMetrum while the airframe is horizontal. This will cause the firmware to go into "idle" mode, in which the normal flight state machine is disabled and charges will not fire without manual command. Then, establish an RF packet connection from a TeleDongle-equipped computer using the P command from a safe distance. You can now command TeleMetrum to fire the apogee or main charges to complete your testing.

In order to reduce the chance of accidental firing of pyrotechnic charges, the command to fire a charge is intentionally somewhat difficult to type, and the built-in help is slightly cryptic to prevent accidental echoing of characters from the help text back at the board from firing a charge. The command to fire the apogee drogue charge is 'i DoIt drogue' and the command to fire the main charge is 'i DoIt main'.

Radio Link

The chip our boards are based on incorporates an RF transceiver, but it's not a full duplex system... each end can only be transmitting or receiving at any given moment. So we had to decide how to manage the link.

By design, TeleMetrum firmware listens for an RF connection when it's in "idle mode" (turned on while the rocket is horizontal), which allows us to use the RF link to configure the rocket, do things like ejection tests, and extract data after a flight without having to crack open the airframe. However, when the board is in "flight mode" (turned on when the rocket is vertical) the TeleMetrum only transmits and doesn't listen at all. That's because we want to put ultimate priority on event detection and getting telemetry out of the rocket and out over the RF link in case the rocket crashes and we aren't able to extract data later...

We don't use a 'normal packet radio' mode because they're just too inefficient. The GFSK modulation we use is just FSK with the baseband pulses passed through a Gaussian filter before they go into the modulator to limit the transmitted bandwidth. When combined with the hardware forward error correction support in the cc1111 chip, this allows us to have a very robust 38.4 kilobit data link with only 10 milliwatts of transmit power, a whip antenna in the rocket, and a hand-held Yagi on the ground. We've had flights to above 21k feet AGL with good reception, and calculations suggest we should be good to well over 40k feet AGL with a 5-element yagi on the ground. We hope to fly boards to higher altitudes soon, and would of course appreciate customer feedback on performance in higher altitude flights!

Configurable Parameters

Configuring a TeleMetrum board for flight is very simple. Because we have both acceleration and pressure sensors, there is no need to set a "mach delay", for example. The few configurable parameters can all be set using a simple terminal program over the USB port or RF link via TeleDongle.

Radio Channel

Our firmware supports 10 channels. The default channel 0 corresponds to a center frequency of 434.550 Mhz, and channels are spaced every 100 khz. Thus, channel 1 is 434.650 Mhz, and channel 9 is 435.550 Mhz. At any given launch, we highly recommend coordinating who will use each channel and when to avoid interference. And of course, both TeleMetrum and TeleDongle must be configured to the same channel to successfully communicate with each other.

To set the radio channel, use the 'c r' command, like 'c r 3' to set channel 3. As with all 'c' sub-commands, follow this with a 'c w' to write the change to the parameter block in the on-board DataFlash chip on your TeleMetrum board if you want the change to stay in place across reboots.

Apogee Delay

Apogee delay is the number of seconds after TeleMetrum detects flight apogee that the drogue charge should be fired. In most cases, this should be left at the default of 0. However, if you are flying redundant electronics such as for an L3 certification, you may wish to set one of your altimeters to a positive delay so that both primary and backup pyrotechnic charges do not fire simultaneously.

To set the apogee delay, use the 'c d' command. As with all 'c' sub-commands, follow this with a 'c w' to write the change to the parameter block in the on-board DataFlash chip.

Please note that the TeleMetrum apogee detection algorithm always fires a fraction of a second *after* apogee. If you are also flying an altimeter like the PerfectFlite MAWD, which only supports selecting 0 or 1 seconds of apogee delay, you may wish to set the MAWD to 0 seconds delay and set the TeleMetrum to fire your backup 2 or 3 seconds later to avoid any chance of both charges firing simultaneously. We've flown several airframes this way quite happily, including Keith's successful L3 cert.

Main Deployment Altitude

By default, TeleMetrum will fire the main deployment charge at an elevation of 250 meters (about 820 feet) above ground. We think this is a good elevation for most airframes, but feel free to change this to suit. In particular, if you are flying two altimeters, you may wish to set the deployment elevation for the backup altimeter to be something lower than the primary so that both pyrotechnic charges don't fire simultaneously.

To set the main deployment altitude, use the 'c m' command. As with all 'c' sub-commands, follow this with a 'c w' to write the change to the parameter block in the on-board DataFlash chip.

Calibration

There are only two calibrations required for a TeleMetrum board, and only one for TeleDongle.

Radio Frequency

The radio frequency is synthesized from a clock based on the 48 Mhz crystal on the board. The actual frequency of this oscillator must be measured to generate a calibration constant. While our GFSK modulation bandwidth is wide enough to allow boards to communicate even when their oscillators are not on exactly the same frequency, performance is best when they are closely matched. Radio frequency calibration requires a calibrated frequency counter. Fortunately, once set, the variation in frequency due to aging and temperature changes is small enough that re-calibration by customers should generally not be required.

To calibrate the radio frequency, connect the UHF antenna port to a frequency counter, set the board to channel 0, and use the 'C' command to generate a CW carrier. Wait for the transmitter temperature to stabilize and the frequency to settle down. Then, divide 434.550 Mhz by the measured frequency and multiply by the current radio cal value show in the 'c s' command. For an unprogrammed board, the default value is 1186611. Take the resulting integer and program it using the 'c f' command. Testing with the 'C' command again should show a carrier within a few tens of Hertz of the intended frequency. As with all 'c' sub-commands, follow this with a 'c w' to write the change to the parameter block in the on-board DataFlash chip.

Accelerometer

The accelerometer we use has its own 5 volt power supply and the output must be passed through a resistive voltage divider to match the input of our 3.3 volt ADC. This means that unlike the barometric sensor, the output of the acceleration sensor is not ratiometric to the ADC converter, and calibration is required. We also support the use of any of several accelerometers from a Freescale family that includes at least +/- 40g, 50g, 100g, and 200g parts. Using gravity, a simple 2-point calibration yields acceptable results capturing both the different sensitivities and ranges of the different accelerometer parts and any variation in power supply voltages or resistor values in the divider network.

To calibrate the acceleration sensor, use the 'c a 0' command. You will be prompted to orient the board vertically with the UHF antenna up and press a key, then to orient the board vertically with the UHF antenna down and press a key. As with all 'c' sub-commands, follow this with a 'c w' to write the change to the parameter block in the on-board DataFlash chip.

The +1g and -1g calibration points are included in each telemetry frame and are part of the header extracted by ao-dumplog after flight. Note that we always store and return raw ADC samples for each sensor... nothing is permanently "lost" or "damaged" if the calibration is poor.

In the unlikely event an accel cal that goes badly, it is possible that TeleMetrum may always come up in 'pad mode' and as such not be listening to either the USB or radio interfaces. If that happens, there is a special hook in the firmware to force the board back in to 'idle mode' so you can re-do the cal. To use this hook, you just need to ground the SPI clock pin at power-on. This pin is available as pin 2 on the 8-pin companion connector, and pin 1 is ground. So either carefully install a fine-gauge wire jumper between the two pins closest to the index hole end of the 8-pin connector, or plug in the programming cable to the 8-pin connector and use a small screwdriver or similar to short the two pins closest to the index post on the 4-pin end of the programming cable, and power up the board. It should come up in 'idle mode' (two beeps).

Updating Device Firmware

The big conceptual thing to realize is that you have to use a TeleDongle as a programmer to update a TeleMetrum, and vice versa. Due to limited memory resources in the cc1111, we don't support programming either unit directly over USB.

You may wish to begin by ensuring you have current firmware images. These are distributed as part of the AltOS software bundle that also includes the AltosUI ground station program. Newer ground station versions typically work fine with older firmware versions, so you don't need to update your devices just to try out new software features. You can always download the most recent version from http://www.altusmetrum.org/AltOS/.

We recommend updating TeleMetrum first, before updating TeleDongle.

Updating TeleMetrum Firmware

1. Find the 'programming cable' that you got as part of the starter kit, that has a red 8-pin MicroMaTch connector on one end and a red 4-pin MicroMaTch connector on the other end.

2. Take the 2 screws out of the TeleDongle case to get access to the circuit board.

3. Plug the 8-pin end of the programming cable to the matching connector on the TeleDongle, and the 4-pin end to the matching connector on the TeleMetrum. Note that each MicroMaTch connector has an alignment pin that goes through a hole in the PC board when you have the cable oriented correctly.

4. Attach a battery to the TeleMetrum board.

5. Plug the TeleDongle into your computer's USB port, and power up the TeleMetrum.

6. Run AltosUI, and select 'Flash Image' from the File menu.

7. Pick the TeleDongle device from the list, identifying it as the programming device.

8. Select the image you want put on the TeleMetrum, which should have a name in the form telemetrum-v1.0-0.7.1.ihx. It should be visible in the default directory, if not you may have to poke around your system to find it.

9. Make sure the configuration parameters are reasonable looking. If the serial number and/or RF configuration values aren't right, you'll need to change them.

10. Hit the 'OK' button and the software should proceed to flash the TeleMetrum with new firmware, showing a progress bar.

11. Confirm that the TeleMetrum board seems to have updated ok, which you can do by plugging in to it over USB and using a terminal program to connect to the board and issue the 'v' command to check the version, etc.

12. If something goes wrong, give it another try.

Updating TeleDongle Firmware

Updating TeleDongle's firmware is just like updating TeleMetrum firmware, but you switch which board is the programmer and which is the programming target.

1. Find the 'programming cable' that you got as part of the starter kit, that has a red 8-pin MicroMaTch connector on one end and a red 4-pin MicroMaTch connector on the other end.

2. Find the USB cable that you got as part of the starter kit, and plug the "mini" end in to the mating connector on TeleMetrum.

3. Take the 2 screws out of the TeleDongle case to get access to the circuit board.

4. Plug the 8-pin end of the programming cable to the (latching) matching connector on the TeleMetrum, and the 4-pin end to the matching connector on the TeleDongle. Note that each MicroMaTch connector has an alignment pin that goes through a hole in the PC board when you have the cable oriented correctly.

5. Attach a battery to the TeleMetrum board.

6. Plug both TeleMetrum and TeleDongle into your computer's USB ports, and power up the TeleMetrum.

7. Run AltosUI, and select 'Flash Image' from the File menu.

8. Pick the TeleMetrum device from the list, identifying it as the programming device.

9. Select the image you want put on the TeleDongle, which should have a name in the form teledongle-v0.2-0.7.1.ihx. It should be visible in the default directory, if not you may have to poke around your system to find it.

10. Make sure the configuration parameters are reasonable looking. If the serial number and/or RF configuration values aren't right, you'll need to change them. The TeleDongle serial number is on the "bottom" of the circuit board, and can usually be read through the translucent blue plastic case without needing to remove the board from the case.

11. Hit the 'OK' button and the software should proceed to flash the TeleDongle with new firmware, showing a progress bar.

12. Confirm that the TeleDongle board seems to have updated ok, which you can do by plugging in to it over USB and using a terminal program to connect to the board and issue the 'v' command to check the version, etc. Once you're happy, remove the programming cable and put the cover back on the TeleDongle.

13. If something goes wrong, give it another try.

Be careful removing the programming cable from the locking 8-pin connector on TeleMetrum. You'll need a fingernail or perhaps a thin screwdriver or knife blade to gently pry the locking ears out slightly to extract the connector. We used a locking connector on TeleMetrum to help ensure that the cabling to companion boards used in a rocket don't ever come loose accidentally in flight.

Chapter 7. AltosUI

The AltosUI program provides a graphical user interface for interacting with the Altus Metrum product family, including TeleMetrum and TeleDongle. AltosUI can monitor telemetry data, configure TeleMetrum and TeleDongle devices and many other tasks. The primary interface window provides a selection of buttons, one for each major activity in the system. This manual is split into chapters, each of which documents one of the tasks provided from the top-level toolbar.

Packet Command Mode

Controlling TeleMetrum Over The Radio Link

One of the unique features of the Altos Metrum environment is the ability to create a two way command link between TeleDongle and TeleMetrum using the digital radio transceivers built into each device. This allows you to interact with TeleMetrum from afar, as if it were directly connected to the computer.

Any operation which can be performed with TeleMetrum can either be done with TeleMetrum directly connected to the computer via the USB cable, or through the packet link. Simply select the appropriate TeleDongle device when the list of devices is presented and AltosUI will use packet command mode.

One oddity in the current interface is how AltosUI selects the channel for packet mode communications. Instead of providing an interface to specifically configure the channel, it uses whatever channel was most recently selected for the target TeleDongle device in Monitor Flight mode. If you haven't ever used that mode with the TeleDongle in question, select the Monitor Flight button from the top level UI, pick the appropriate TeleDongle device. Once the flight monitoring window is open, select the desired channel and then close it down again. All Packet Command Mode operations will now use that channel.

• Save Flight Data—Recover flight data from the rocket without opening it up.

• Configure TeleMetrum—Reset apogee delays or main deploy heights to respond to changing launch conditions. You can also 'reboot' the TeleMetrum device. Use this to remotely enable the flight computer by turning TeleMetrum on while horizontal, then once the airframe is oriented for launch, you can reboot TeleMetrum and have it restart in pad mode without having to climb the scary ladder.

• Fire Igniters—Test your deployment charges without snaking wires out through holes in the airframe. Simply assembly the rocket as if for flight with the apogee and main charges loaded, then remotely command TeleMetrum to fire the igniters.

Packet command mode uses the same RF channels as telemetry mode. Configure the desired TeleDongle channel using the flight monitor window channel selector and then close that window before performing the desired operation.

TeleMetrum only enables packet command mode in 'idle' mode, so make sure you have TeleMetrum lying horizontally when you turn it on. Otherwise, TeleMetrum will start in 'pad' mode ready for flight and will not be listening for command packets from TeleDongle.

When packet command mode is enabled, you can monitor the link by watching the lights on the TeleDongle and TeleMetrum devices. The red LED will flash each time TeleDongle or TeleMetrum transmit a packet while the green LED will light up on TeleDongle while it is waiting to receive a packet from TeleMetrum.

Monitor Flight

Receive, Record and Display Telemetry Data

Selecting this item brings up a dialog box listing all of the connected TeleDongle devices. When you choose one of these, AltosUI will create a window to display telemetry data as received by the selected TeleDongle device.

All telemetry data received are automatically recorded in suitable log files. The name of the files includes the current date and rocket serial and flight numbers.

The radio channel being monitored by the TeleDongle device is displayed at the top of the window. You can configure the channel by clicking on the channel box and selecting the desired channel. AltosUI remembers the last channel selected for each TeleDongle and selects that automatically the next time you use that device.

Below the TeleDongle channel selector, the window contains a few significant pieces of information about the TeleMetrum providing the telemetry data stream:

- The TeleMetrum callsign

- The TeleMetrum serial number

- The flight number. Each TeleMetrum remembers how many times it has flown.

- The rocket flight state. Each flight passes through several states including Pad, Boost, Fast, Coast, Drogue, Main and Landed.

- The Received Signal Strength Indicator value. This lets you know how strong a signal TeleDongle is receiving. The radio inside TeleDongle operates down to about -99dBm; weaker signals may not be receiveable. The packet link uses error correction and detection techniques which prevent incorrect data from being reported.

Finally, the largest portion of the window contains a set of tabs, each of which contain some information about the rocket. They're arranged in 'flight order' so that as the flight progresses, the selected tab automatically switches to display data relevant to the current state of the flight. You can select other tabs at any time. The final 'table' tab contains all of the telemetry data in one place.

Launch Pad

The 'Launch Pad' tab shows information used to decide when the rocket is ready for flight. The first elements include red/green indicators, if any of these is red, you'll want to evaluate whether the rocket is ready to launch:

- Battery Voltage. This indicates whether the LiPo battery powering the TeleMetrum has sufficient charge to last for the duration of the flight. A value of more than 3.7V is required for a 'GO' status.

- Apogee Igniter Voltage. This indicates whether the apogee igniter has continuity. If the igniter has a low resistance, then the voltage measured here will be close to the LiPo battery voltage. A value greater than 3.2V is required for a 'GO' status.

- Main Igniter Voltage. This indicates whether the main igniter has continuity. If the igniter has a low resistance, then the voltage measured here will be close to the LiPo battery voltage. A value greater than 3.2V is required for a 'GO' status.

- GPS Locked. This indicates whether the GPS receiver is currently able to compute position information. GPS requires at least 4 satellites to compute an accurate position.

- GPS Ready. This indicates whether GPS has reported at least 10 consecutive positions without losing lock. This ensures that the GPS receiver has reliable reception from the satellites.

The LaunchPad tab also shows the computed launch pad position and altitude, averaging many reported positions to improve the accuracy of the fix.

Ascent

This tab is shown during Boost, Fast and Coast phases. The information displayed here helps monitor the rocket as it heads towards apogee.

The height, speed and acceleration are shown along with the maxium values for each of them. This allows you to quickly answer the most commonly asked questions you'll hear during flight.

The current latitude and longitude reported by the GPS are also shown. Note that under high acceleration, these values may not get updated as the GPS receiver loses position fix. Once the rocket starts coasting, the receiver should start reporting position again.

Finally, the current igniter voltages are reported as in the Launch Pad tab. This can help diagnose deployment failures caused by wiring which comes loose under high acceleration.

Descent

Once the rocket has reached apogee and (we hope) activated the apogee charge, attention switches to tracking the rocket on the way back to the ground, and for dual-deploy flights, waiting for the main charge to fire.

To monitor whether the apogee charge operated correctly, the current descent rate is reported along with the current height. Good descent rates generally range from 15-30m/s.

To help locate the rocket in the sky, use the elevation and bearing information to figure out where to look. Elevation is in degrees above the horizon. Bearing is reported in degrees relative to true north. Range can help figure out how big the rocket will appear. Note that all of these values are relative to the pad location. If the elevation is near 90°, the rocket is over the pad, not over you.

Finally, the igniter voltages are reported in this tab as well, both to monitor the main charge as well as to see what the status of the apogee charge is.

Landed

Once the rocket is on the ground, attention switches to recovery. While the radio signal is generally lost once the rocket is on the ground, the last reported GPS position is generally within a short distance of the actual landing location.

The last reported GPS position is reported both by latitude and longitude as well as a bearing and distance from the launch pad. The distance should give you a good idea of whether you'll want to walk or hitch a ride. Take the reported latitude and longitude and enter them into your handheld GPS unit and have that compute a track to the landing location.

Finally, the maximum height, speed and acceleration reported during the flight are displayed for your admiring observers.

Site Map

When the rocket gets a GPS fix, the Site Map tab will map the rocket's position to make it easier for you to locate the rocket, both while it is in the air, and when it has landed. The rocket's state is indicated by colour: white for pad, red for boost, pink for fast, yellow for coast, light blue for drogue, dark blue for main, and black for landed.

The map's scale is approximately 3m (10ft) per pixel. The map can be dragged using the left mouse button. The map will attempt to keep the rocket roughly centred while data is being received.

Images are fetched automatically via the Google Maps Static API, and are cached for reuse. If map images cannot be downloaded, the rocket's path will be traced on a dark grey background instead.

Save Flight Data

TeleMetrum records flight data to its internal flash memory. This data is recorded at a much higher rate than the telemetry system can handle, and is not subject to radio drop-outs. As such, it provides a more complete and precise record of the flight. The 'Save Flight Data' button allows you to read the flash memory and write it to disk.

Clicking on the 'Save Flight Data' button brings up a list of connected TeleMetrum and TeleDongle devices. If you select a TeleMetrum device, the flight data will be downloaded from that device directly. If you select a TeleDongle device, flight data will be downloaded from a TeleMetrum device connected via the packet command link to the specified TeleDongle. See the chapter on Packet Command Mode for more information about this.

The filename for the data is computed automatically from the recorded flight date, TeleMetrum serial number and flight number information.

Replay Flight

Select this button and you are prompted to select a flight record file, either a .telem file recording telemetry data or a .eeprom file containing flight data saved from the TeleMetrum flash memory.

Once a flight record is selected, the flight monitor interface is displayed and the flight is re-enacted in real time. Check the Monitor Flight chapter above to learn how this window operates.

Graph Data

Select this button and you are prompted to select a flight record file, either a .telem file recording telemetry data or a .eeprom file containing flight data saved from the TeleMetrum flash memory.

Once a flight record is selected, the acceleration (blue), velocity (green) and altitude (red) of the flight are plotted and displayed, measured in metric units.

The graph can be zoomed into a particular area by clicking and dragging down and to the right. Once zoomed, the graph can be reset by clicking and dragging up and to the left. Holding down control and clicking and dragging allows the graph to be panned. The right mouse button causes a popup menu to be displayed, giving you the option save or print the plot.

Note that telemetry files will generally produce poor graphs due to the lower sampling rate and missed telemetry packets, and will also often have significant amounts of data received while the rocket was waiting on the pad. Use saved flight data for graphing where possible.

Export Data

This tool takes the raw data files and makes them available for external analysis. When you select this button, you are prompted to select a flight data file (either .eeprom or .telem will do, remember that .eeprom files contain higher resolution and more continuous data). Next, a second dialog appears which is used to select where to write the resulting file. It has a selector to choose between CSV and KML file formats.

Comma Separated Value Format

This is a text file containing the data in a form suitable for import into a spreadsheet or other external data analysis tool. The first few lines of the file contain the version and configuration information from the TeleMetrum device, then there is a single header line which labels all of the fields. All of these lines start with a '#' character which most tools can be configured to skip over.

The remaining lines of the file contain the data, with each field separated by a comma and at least one space. All of the sensor values are converted to standard units, with the barometric data reported in both pressure, altitude and height above pad units.

Keyhole Markup Language (for Google Earth)

This is the format used by Googleearth to provide an overlay within that application. With this, you can use Googleearth to see the whole flight path in 3D.

Configure TeleMetrum

Select this button and then select either a TeleMetrum or TeleDongle Device from the list provided. Selecting a TeleDongle device will use Packet Comamnd Mode to configure remote TeleMetrum device. Learn how to use this in the Packet Command Mode chapter.

The first few lines of the dialog provide information about the connected TeleMetrum device, including the product name, software version and hardware serial number. Below that are the individual configuration entries.

At the bottom of the dialog, there are four buttons:

- Save. This writes any changes to the TeleMetrum configuration parameter block in flash memory. If you don't press this button, any changes you make will be lost.

- Reset. This resets the dialog to the most recently saved values, erasing any changes you have made.

- Reboot. This reboots the TeleMetrum device. Use this to switch from idle to pad mode by rebooting once the rocket is oriented for flight.

- Close. This closes the dialog. Any unsaved changes will be lost.

The rest of the dialog contains the parameters to be configured.

Main Deploy Altitude

This sets the altitude (above the recorded pad altitude) at which the 'main' igniter will fire. The drop-down menu shows some common values, but you can edit the text directly and choose whatever you like. If the apogee charge fires below this altitude, then the main charge will fire two seconds after the apogee charge fires.

Apogee Delay

When flying redundant electronics, it's often important to ensure that multiple apogee charges don't fire at precisely the same time as that can overpressurize the apogee deployment bay and cause a structural failure of the airframe. The Apogee Delay parameter tells the flight computer to fire the apogee charge a certain number of seconds after apogee has been detected.

Radio Channel

This configures which of the 10 radio channels to use for both telemetry and packet command mode. Note that if you set this value via packet command mode, you will have to reconfigure the TeleDongle channel before you will be able to use packet command mode again.

Radio Calibration

The radios in every Altus Metrum device are calibrated at the factory to ensure that they transmit and receive on the specified frequency for each channel. You can adjust that calibration by changing this value. To change the TeleDongle's calibration, you must reprogram the unit completely.

Callsign

This sets the callsign included in each telemetry packet. Set this as needed to conform to your local radio regulations.

Configure AltosUI

This button presents a dialog so that you can configure the AltosUI global settings.

Voice Settings

AltosUI provides voice annoucements during flight so that you can keep your eyes on the sky and still get information about the current flight status. However, sometimes you don't want to hear them.

- Enable—turns all voice announcements on and off

- Test Voice—Plays a short message allowing you to verify that the audio systme is working and the volume settings are reasonable

Log Directory

AltosUI logs all telemetry data and saves all TeleMetrum flash data to this directory. This directory is also used as the staring point when selecting data files for display or export.

Click on the directory name to bring up a directory choosing dialog, select a new directory and click 'Select Directory' to change where AltosUI reads and writes data files.

Callsign

This value is used in command packet mode and is transmitted in each packet sent from TeleDongle and received from TeleMetrum. It is not used in telemetry mode as that transmits packets only from TeleMetrum to TeleDongle. Configure this with the AltosUI operators callsign as needed to comply with your local radio regulations.

Flash Image

This reprograms any Altus Metrum device by using a TeleMetrum or TeleDongle as a programming dongle. Please read the directions for connecting the programming cable in the main TeleMetrum manual before reading these instructions.

Once you have the programmer and target devices connected, push the 'Flash Image' button. That will present a dialog box listing all of the connected devices. Carefully select the programmer device, not the device to be programmed.

Next, select the image to flash to the device. These are named with the product name and firmware version. The file selector will start in the directory containing the firmware included with the AltosUI package. Navigate to the directory containing the desired firmware if it isn't there.

Next, a small dialog containing the device serial number and RF calibration values should appear. If these values are incorrect (possibly due to a corrupted image in the device), enter the correct values here.

Finally, a dialog containing a progress bar will follow the programming process.

When programming is complete, the target device will reboot. Note that if the target device is connected via USB, you will have to unplug it and then plug it back in for the USB connection to reset so that you can communicate with the device again.

Fire Igniter

This activates the igniter circuits in TeleMetrum to help test recovery systems deployment. Because this command can operate over the Packet Command Link, you can prepare the rocket as for flight and then test the recovery system without needing to snake wires inside the airframe.

Selecting the 'Fire Igniter' button brings up the usual device selection dialog. Pick the desired TeleDongle or TeleMetrum device. This brings up another window which shows the current continutity test status for both apogee and main charges.

Next, select the desired igniter to fire. This will enable the 'Arm' button.

Select the 'Arm' button. This enables the 'Fire' button. The word 'Arm' is replaced by a countdown timer indicating that you have 10 seconds to press the 'Fire' button or the system will deactivate, at which point you start over again at selecting the desired igniter.

Chapter 8. Using Altus Metrum Products

Being Legal

First off, in the US, you need an amateur radio license [http://www.altusmetrum.org/Radio/] or other authorization to legally operate the radio transmitters that are part of our products.

In the Rocket

In the rocket itself, you just need a TeleMetrum [http://www.altusmetrum.org/TeleMetrum/] board and a LiPo rechargeable battery. An 860mAh battery weighs less than a 9V alkaline battery, and will run a TeleMetrum [http://www.altusmetrum.org/TeleMetrum/] for hours.

By default, we ship TeleMetrum with a simple wire antenna. If your electronics bay or the airframe it resides within is made of carbon fiber, which is opaque to RF signals, you may choose to have an SMA connector installed so that you can run a coaxial cable to an antenna mounted elsewhere in the rocket.

On the Ground

To receive the data stream from the rocket, you need an antenna and short feedline connected to one of our TeleDongle [http://www.altusmetrum.org/TeleDongle/] units. The TeleDongle in turn plugs directly into the USB port on a notebook computer. Because TeleDongle looks like a simple serial port, your computer does not require special device drivers... just plug it in.

The GUI tool, AltosUI, is written in Java and runs across Linux, Mac OS and Windows. There's also a suite of C tools for Linux which can perform most of the same tasks.

After the flight, you can use the RF link to extract the more detailed data logged in the rocket, or you can use a mini USB cable to plug into the TeleMetrum board directly. Pulling out the data without having to open up the rocket is pretty cool! A USB cable is also how you charge the LiPo battery, so you'll want one of those anyway... the same cable used by lots of digital cameras and other modern electronic stuff will work fine.

If your rocket lands out of sight, you may enjoy having a hand-held GPS receiver, so that you can put in a waypoint for the last reported rocket position before touch-down. This makes looking for your rocket a lot like Geo-Cacheing... just go to the waypoint and look around starting from there.

You may also enjoy having a ham radio "HT" that covers the 70cm band... you can use that with your antenna to direction-find the rocket on the ground the same way you can use a Walston or Beeline tracker. This can be handy if the rocket is hiding in sage brush or a tree, or if the last GPS position doesn't get you close enough because the rocket dropped into a canyon, or the wind is blowing it across a dry lake bed, or something like that... Keith and Bdale both currently own and use the Yaesu VX-7R at launches.

So, to recap, on the ground the hardware you'll need includes:

1. an antenna and feedline

2. a TeleDongle

3. a notebook computer

4. optionally, a handheld GPS receiver

5. optionally, an HT or receiver covering 435 Mhz

The best hand-held commercial directional antennas we've found for radio direction finding rockets are from Arrow Antennas. [http://www.arrowantennas.com/] The 440-3 and 440-5 are both good choices for finding a TeleMetrum-equipped rocket when used with a suitable 70cm HT.

Data Analysis

Our software makes it easy to log the data from each flight, both the telemetry received over the RF link during the flight itself, and the more complete data log recorded in the DataFlash memory on the TeleMetrum board. Once this data is on your computer, our postflight tools make it easy to quickly get to the numbers everyone wants, like apogee altitude, max acceleration, and max velocity. You can also generate and view a standard set of plots showing the altitude, acceleration, and velocity of the rocket during flight. And you can even export a data file useable with Google Maps and Google Earth for visualizing the flight path in two or three dimensions!

Our ultimate goal is to emit a set of files for each flight that can be published as a web page per flight, or just viewed on your local disk with a web browser.

Future Plans

In the future, we intend to offer "companion boards" for the rocket that will plug in to TeleMetrum to collect additional data, provide more pyro channels, and so forth. A reference design for a companion board will be documented soon, and will be compatible with open source Arduino programming tools.

We are also working on the design of a hand-held ground terminal that will allow monitoring the rocket's status, collecting data during flight, and logging data after flight without the need for a notebook computer on the flight line. Particularly since it is so difficult to read most notebook screens in direct sunlight, we think this will be a great thing to have.

Because all of our work is open, both the hardware designs and the software, if you have some great idea for an addition to the current Altus Metrum family, feel free to dive in and help! Or let us know what you'd like to see that we aren't already working on, and maybe we'll get excited about it too...