SIDEREAL TECHNOLOGY DUAL SERVO TELESCOPE CONTROLLER
MODEL DSCT-1
USER INSTRUCTION MANUAL
http://www.siderealtechnology.com/

Last revised May, 2006. Describes Firmware Version 1.9,
Later and Earlier Versions of Hardware

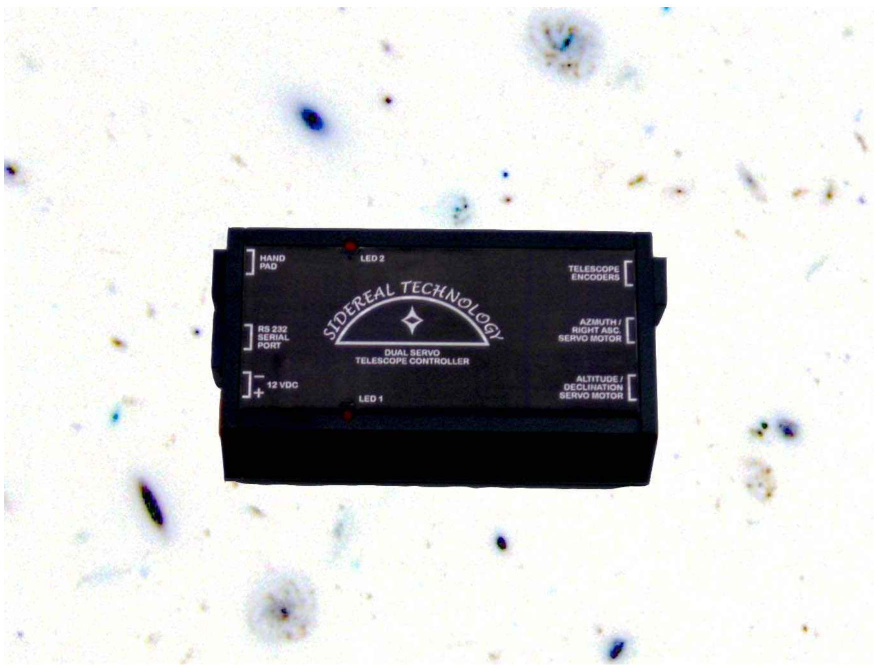


Table of Contents

GENERAL DESCRIPTION:

The Sidereal Technology Dual Servo Telescope Controller is economical, very small, and packed with many features. This telescope control system has been installed on scopes as small as 3 inches, to as large as 41 inches! It connects to 2 servo motors with integral encoders, and 2 telescope encoders. It will track and/or guide your telescope without any computer connected, and provide tracking, guiding, GoTo, etc when connected to another computer, an Argo Navis™, or a Palm handheld computer.

FEATURES:

There are Six ways to use the SiTech Controller on your telescope:

1. Stand alone. In this mode, the controller will track your Alt/Az telescope using the DragNTrack or SlewNTrack modes. It will also track your Equatorial telescope or Equatorial Platform using the Equatorial mode. Many users prefer the stand alone mode, for truly no hassle, telescope tracking. This mode is also the most economical.
2. Use the Argo Navis™ from Wildcard Innovations, for complete and accurate Goto and Tracking of your telescope. This is a really cool way to go if you don't want the hassle of using a PC at the telescope. Using the SiTech unit along with the Argo Navis™ has several advantages over other systems which use the Argo Navis™. If optional telescope encoders are used, they are connected to the SiTech unit instead of the Argo Navis™ (the Argo Navis™ encoder port is un-used), and the serial port of the SiTech unit is connected to the Argo Navis™ serial port. Some of the advantages are that the encoders draw power from the more powerful Servo Controller Batteries, so the Argo Navis batteries last much longer, and also, the fact that the telescope simply doesn't need external encoders to work with the Argo Navis™! The user has the option of using external encoders, or simply use the motor location instead (if you have a system that doen't slip), the best of both worlds!
3. Use the provided FREE ASCOM driver from Sidereal Technology. This requires a Laptop or Desktop PC, running Windows 98 through Windows XP. The advantage of using the ASCOM driver, is almost any planetarium program that supports ASCOM will accurately control your Alt/Az or Equatorial telescope. No special Virtual Serial Port drivers or NULL modem cables to otherwise unused serial ports. This ASCOM driver is very powerful, and supports almost every ASCOM Version 2.1 function. The Argo Navis is not required!
4. Use Mel Bartels SCOPE II Windows/Mac/UNIX software (Java Based), observatory quality, incredibly accurate GOTO and Tracking. Mel has spent many hours providing almost every possible feature in SCOPE II. If you want better than 1 arc minute pointing, sky to sky, or want to do serious photography, Mels SCOPE II is the way to go.
5. Use Mel Bartels Windows software, ScopeIII. We are excited about this, and wait with anticipation what Mel will have put together in this new software.
6. Use a Palm Pilot, a Blue Tooth Serial module and Astromist Software by Cyrille Thieullet. We are also excited about this option, as Cyrille has created a brilliant piece of software with many featurers. Please check out http://www.astromist.com for more details. Also, soon to come is control using JavaME, which can be found on many Cell phones!

Finally, a servo telescope control that is affordable! You can purchase a bare bones servo controller with motors and a handpad for much less than you would imagine!

Directly controls two servo motors with the motor position control loops running at nearly 2000 times a second, with high speed shaft encoders for position feedback on each motor.

In addition to the two encoders mounted on the motors, the SiTech unit can connect to two additional high speed encoders for the telescopes altitude and azimuth (or right ascension and declination). No need to add an additional "box" to connect shaft encoders.

Open Collector RS232 Serial port for connecting 2 or more controllers on the same RS232 bus. Other controller axis's can be used as field de-rotators and/or focusers (Available using PC control only).

Connects to a wired handpad or optional radio handpad for slewing/panning/guiding.

Signals from the wired or radio handpad work immediately, no delay as with some other systems, because the same micro-controller that is controlling the servo motors, “listens” to the handpad signals. No communication protocol delay. The Radio handpad protocol, is a lean and mean protocol, taking only the time to transfer 16 bits at a quick bit rate. There is absolutely no percieved delay between when you press a button on the handpad (radio or wired) and when the servo motors move.

Can automatically track your telescope in the Equatorial mode, the DragNTrack mode, or the SlewNTrack mode without having a personal computer connected.

When running in the computerless modes mentioned in the above paragraph, you may connect a computer running just about any planetarium program, and the servo controller will emulate the Tangent Digital Setting Circle protocol. This allows just about any planetarium program to provide the “crosshair” or other indicator, of where the telescope is pointed.

Can control a dual axis tracking platform, with auto stop and rewind, and dual axis correction without having a personal computer connected.

Servo Motors have a FASTEST mode of operation, where if the motors can't quite keep up with where it is supposed to be, but it is moving in the proper direction, it will not trip on position error which is the industry standard, instead, position error is held at the point of the position error limit. This feature makes it possible to run your motors near their limit of speed, and you can rest assured, it will not trip on position error if your battery voltage falls a bit, or the wind pushes against the scope.

Local search: New for version 1.7 of the firmware. Upon a quick combination keypad request, your telescope will search for an object, in a circular pattern. This is the most efficient search pattern, as there is no overlap between successive search patterns. This is a true circle, no matter where in the sky you are looking. No in-efficent elipses at different altitudes. You have two choices of radius, reverse, forward and pause control, and speed of the search can be changed while looking through the eyepiece!

Internal Backlash: New for version 1.7 of the firmware. Backlash is internal, and instantaneously applied. Each axis has a backlash variable associatted with it, and an adjustable backlash area speed.

Equal Area Pan/Guide Speeds: New for version 1.7 of the firmware. The Azimuth/RA Pan and Guide speed is automatically adjusted for different Altitude/Declinations. As you near the Zenith (or Pole), the Pan and Guide speeds are automatically increased so the same sky area is covered in Azimuth/RA as Altitude/Dec. This is a really cool feature, and if you experience it, you won't want to go back!!!

Small size, 2.5" by 4.25" by 1".

Power Efficient. Will last several nights on one 7 amp hour Gel Cell on most telescopes.

Built in flash memory for saving parameters.

Firmware in the controller is field upgradeable to later versions of the firmware, no need to send it back to the factory for additional features.

Windows Configuration program is supplied free with the unit or is available online for free downloading.

COMPARISON BETWEEN STEPPER SYSTEM AND SERVO SYSTEM:
A servo system has many advantages over a stepper system.

1. Wider dynamic range. This means a servo system will track your telescope nearly perfectly and still obtain faster slew speeds than a comparably geared stepper system.
2. Angular accuracy of the motor while tracking is much better. There are many errors in a stepper system when being micro stepped. Generally speaking, the micro steps are not nearly as accurate as the resolution of the micro step. There are many errors such as mechanical winding differences, magnetic hysterisis, torque error, and several others that can be partially compensated for in software with a stepper system, but never will be as good as a servo system, which will almost all the time be within 1 encoder tick of it's desired location at all times (while tracking).
3. More torque. Small servo motors will control telescopes as large as 41 inches!
4. Much lower current during tracking. Batteries will last much longer.
5. No resonant frequency's to battle.
6. Motors will never “miss” steps. Many times stepper motors will simply stop instead of turning during slewing. The computer controlling the stepper motors will loose its position information when this happens.
7. A real time operating system is in the Servo Controller itself, thus making a real time operating system unnecessary for the controlling computer (some stepper systems have this too). This means the same computer can operate your telescope with a planetarium software interface while still tracking, perform imaging, autoguiding the telescope, controling a dome, etc, all at the same time using the same computer. This can be a significant cost savings of a servo system over a stepper system, evening the cost savings of a stepper over a servo system.
8. Much smaller size (in the case of the SiTech controller anyway).
9. If only tracking is desired, no external computer is necessary (SiTech feature).
10. If no external computer is desired, but GoTo's are, the SiTech Controller can be controlled by an Argo Navis™ or a Palm computer.

The stepper system has only one advantage over the servo system that we can think of: Cost, although a servo system can be less than four or five hundred dollars, inluding controller, surplus motors, and software if you do most of the work yourself. Although cost is an advantage of the Mel Bartels DOS based Stepper system, Mels system is the only stepper system we know of that is less than the SiTech servo system, and then if you want to do imaging, planetarium software control (while still tracking) or other simultaneous computer operations, you'll still have to use a 2^{nd} computer.

Some books or articles may mention Servo Lag as an advantage of a stepper system over a servo system. This isn't an advantage when compared with the SiTech Servo system. When tracking, the servo motors will almost all the time be within **ONE** motor encoder tick and occasionally, two motor encoder ticks of its desired position. This is usually between $1/10^{th}$ to 1/4 of an arc second (depending on your gear ratio).

Despite the many advantages of a servo system over a stepper system, many successful stepper systems have been created, which work extremely well.

SPECIFICATIONS:
Input Voltage: 11.5 -28 volts D.C.
Input Current: 4 amps max
Maximum Motor Current: 2.4 amps each, total of 4 amps max.

RS232 Protocol:
1. The RS232 port has a diode and pulldown resistor built in, for open collector communication. This allows one RS232 port to control up to 3 SiTech Dual Servo Controllers.
2. Extended Pic Servo Emulation (see http://www.jrkerr.com/psdata.pdf for protocol documentation).
3. Very simple, extended Ascii Protocol.
4. Custom Binary protocol for very fast and efficient (low communication overhead) control of your telescope.

2 Each, Motor Encoder Resolution: 32 bit (+/- 2 billion encoder ticks)
2 Each, Scope Encoder Resolution: 32 bit (+/- 2 billion encoder ticks)

Temperature Rating:
Storage: 0 deg's F to 160 deg's F
Operation 0 deg's F to 120 deg's F

Motor Encoder maximum read frequency: Up to 248 KHz. (May need external encoder pull up resistors for operation near maximum frequency, depending on the motor encoders installed).

Telescope Encoder maximum read frequency: Up to 25 KHz.

DISCLAIMER:
Sidereal Technology provides 1 year warrantee period for replacement or repairing cost of the controller only. If the telescope knocks you off the ladder, the eyepiece gives you a black eye, or the telescope dumps the mirror, or any other disaster, Sidereal Technology or Dan Gray will not be responsible!

WARRANTEE:
The Sidereal Technology unit is warranteed for a period of 1 year from the time of delivery. If the controller has been damaged by lightning, faulty wiring, moisture, or other misuse, the warrantee is void. If the controller develops problems beyond the above mention reasons, Sidereal Technology will repair or replace the controller at Sidereal Technology's discretion, if within 1 year of delivery.

TECHNICAL SUPPORT:
If you are using ScopeII and having issues, you need to contact Mel Bartels for Technical Support. Mel is extremely patient and will work with you until you are up and working, if necessary, he will provide telephone support.

If you are having problems with the controller, radio handpad, or the ASCOM driver, or are having problems with the Argo Navis controlling your telescope, Sidereal Technology will help you. We are available by email anytime. You will receive a response, generally within 8-10 hours, sometimes within minutes. You can also call Dan Gray anytime before 10:00 PM Pacific Time. His phone number is 503-887-3701.

grayarea@siderealtechnology.com

MOUNTING THE UNIT:
A piece of Velcro with sticky back is a good way. Mount so both ends of the controller are available for the connections. If exposed, do not mount vertically, otherwise dew or rain may enter the controller.

POWER:
The controller has a power connector/terminal block. Please provide a regulated supply or battery, 4 amp minimum (if full torque is needed) 12 to 24 volt D.C. Supply. A 7 amp hour Gel Cell battery is the best portable way to power the unit. This should last 2 or 3 observing nights before recharging is necessary, if you have a low friction, balanced telescope.

Please observe polarity. The internal protection diode is not wired in series with the power lead as is typical, because the extra .6 volts is crucial to operation of the servo when running at 12 volts. It is wired in parallel, after the fuse. If you hook the power up backwards, you will surely blow the fuse, and possibly damage other components. Please don't be the second person to test this!!!!

LED OPERATION:
There are two LED's on the controller, labeled LED 1 and LED 2. These LED's can keep you informed about what is happening with the controller.

Normally, LED 1 indicates the status of the Altitude/Declination motor, and LED 2 indicates the status of the Azimuth/Right Ascention motor. If the LED has a steady 1 second period blink, then that motor has been forced to manual by a command from the serial port, or by sensing motor runaway. Motor runaway is caused by 3 possible scenarios, Faulty motor encoder or motor encoder wiring, phasing backwards between motor polarity and motor encoder pulses, or motor can't run as fast as it is being told to run. To return to automatic mode, the controller must be powered down, then powered up again, or the command to return to Auto must be given via a command on the serial port. If you are using version 1.6 (or later) of the SiTech Servo firmware, you can also press the top left and the top right buttons on the handpad simultaneously, and both motors will be returned to automatic.

If both LED's are on very faintly, but steadily, there is a serious problem with the CPU, and the unit will probably need to be returned to our factory for repairs. In firmware versions 1.3 or earlier, there was a bug in the software that allowed the flash ROM to be erased under very rare conditions. When this happened, the controller had to be returned to the factory for re-programming. If you have firmware 1.3 or earlier, please upgrade to the latest firmware version. You can download this version from our website.

In the DragNTrack and SlewNTrack modes, LED 2 toggles at every new track command. This happens whenever a handpad button is released, or approximately every 8 seconds if no buttons are released (See section on DragNTrack).

When initializing the DragNTrack or SlewNTrack systems, you must hold the appropriate key down for more than 4 seconds (See section on DragNTrack). When the controller DragNTrack mode is being initializes, LED 1 will flash fast until key let up.

If the controller is in the standalone Equatorial mode, LED 1 flashes at the same rate as the crab pulsar. This is too fast to see it as a flash, your persistence of vision makes it look like it's on steady, a little fainter than normal. If you shake your head as you're looking at it, you will then see the flash! You may wonder why we did this, it's purpose is for testing an interrupter you may fabricate to see the crab pulsar central star fade in and out. See section on the Equatorial mode for more details.

Pressing the TopRight key on the handpad will stop and start tracking unless you are using ScopeII or the ASCOM driver. When the controller has stopped tracking, both LED's will blink off every few seconds.

If the battery voltage falls below 11 volts, both LED's will flash fast.

SERIAL PORT CONNECTION:
The serial port is an 8 position modular jack. Please do not confuse this connector with the motor connectors. It is adjacent to the power connector/terminal block, and on the opposite end of the

controller from the motor connectors. Even if you don't intend to use a computer to control your telescope, you will need a serial cable to configure the unit.

Pin 3 of a laptop or PC is the transmit signal and pin 3 of the controller is the receive signal, so the cable can be wired, pin 2 to pin 2, pin 3 to pin 3, and pin 5 to pin 5. It is a good idea to jumper pins 1,4 and 6 together, and jumper pins 7 and 8 together (at the 9 pin female connector) in case special software uses the handshaking on the PC serial port. See appendix D for a wiring diagram.

If you would like to connect two or more Dual Servo Controllers on the same RS232 communication bus, simply make up a cable as described above, but parallel another 8 position modular jack in parallel with the existing jack. This will require some soldering. This cable can also be ordered from Sidereal Technology (or it's representatives). Also you will have to “give” the second controller another address. This is described later in the manual.

Note: Mel Bartels Scope II software or Sidereal Technology's ASCOM driver and ServoConfig software do not use hardware handshaking, but if you create special software for the controller, you may have to install the jumpers in the 9 pin female connector listed above.

SETTING UP THE MOTORS:

Motor wiring:

Normally the motors will come pre-wired, ready for installation. If wiring is required, see the drawing at the end of this document for wiring connections.

The motor encoders are connected using the same 8 position connector as is used for the motor. For best noise immunity, the motors should be connected using flat cable, and terminated at the controller end with 8 position modular connector. CAT5 cable will work, but the encoders can pick up electro-magnetic interference from the motor leads if the cables are too long.

If you have 24 volt or 36 volt motors, and only have 12 volts available, don't worry, it will still work fine, but the top speed of the motor may not be as fast as you like.

If you wire your own motors, be very careful with the wiring, as if you mix up the motor leads with the encoder leads, you can destroy the Servo Controller. This is not covered by warantee.

Motor Configuration:

Once the motors have been wired, double check all wiring, including the power polarity. Double check that both motors are free to spin at full RPM in either direction, because it may be that the motor wiring moves the motor in the opposite direction that the encoder is expecting. If this happens, the motor will "run away" at full speed, until the controller determines that the motor is running the wrong way, and shuts the motor off. There will be no ramping, it will instantaneously be at it's fastest RPM for the supply voltage.

Now plug in just the altitude motor, and then power up the unit. This is where you will find if you are a lucky guy/gal or not. If the motor takes off in either direction at full speed and then stops, don't buy any lottery tickets today. Your 50% chance worked against you. You have 3 options. 1. At the motor, swap the motor leads. 2. At the motor encoder, swap the A and B encoder phases. 3. Use the supplied ServoConfig software, and change the motor direction, or the encoder direction (note: if you purchased your motors from Mel Bartels or Sidereal Technology, they will be pre-wired and tested).

If the motor didn't take off, you should see that it is stopped, and that it's difficult (or impossible) to turn by hand. If this is so, the motor is now being controlled by the servo controller. The motor may be vibrating noisily, or moving back and forth quickly. If this is so, the PID parameters need to be "tuned". See section on "Tuning Your Motors".   Note: This is not very likely, the default settings work on most motors.

Repeat the above process for the azimuth motor.

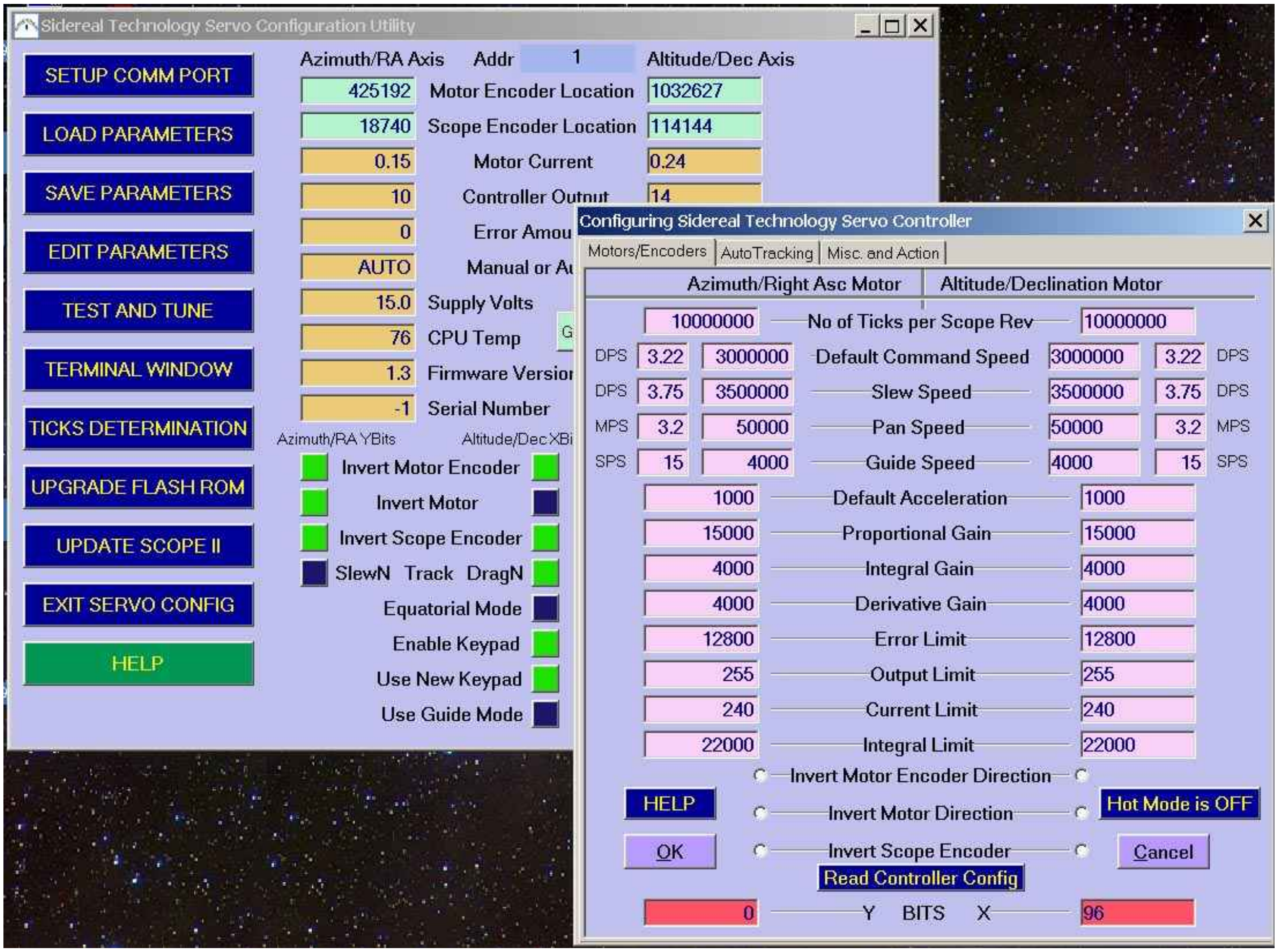
ADJUSTING PARAMETERS:
There are several ways to configure the controller. If you intend to only use Mel Bartels SCOPEII, you will not have to configure the controller, other than the motor directions in relation to the encoder directions. If you are using the SiTech ASCOM driver, you will have to configure the motor tuning parameters (if necessary), and the slew and pan speeds, the encoder and motor Ticks Per Rev parameters, and the backlash and local search parameters.

Even if you intend to only use ScopeII, we feel it's still important to configure your controller completely (including all Motor and Scope encoder ticks per revolution) and get it working in the stand alone AutoTracking mode properly before you configure either ScopeII or the SiTech Ascom driver. This will insure that any problems you have will be on the software side (probably configuration) instead of the controller, and will help in troubleshooting any problems you have. This has the added benefit that your scope will track on applying power, all by itself, before running any software (The controller is configured to drop out of the AutoTracking mode whenever it's connected to a computer or an Argo Navis™, thus removing any chance of the controller trying to track on its own when receiving tracking or GoTo commands from another source).

If you have an equatorial mount, the stand alone AutoTracking mode will be the Equatorial mode. If you have an Alt/Az scope with scope encoders, you will be using the DragNTrack mode (even if you don't have clutches). If you have an Alt/Az scope and you don't have scope encoders, you will need to use the SlewNTrack mode.

You have 3 options to configure the controller.

1. Use the ServoConfig software, provided by Sidereal Technology. This is provided on the supplied CD Rom, and is also available at http://www.siderealtechnology.com/ This is probably the easiest application to use, however it only works on MS Windows machines. Here is some screen shots of the ServoConfig software:



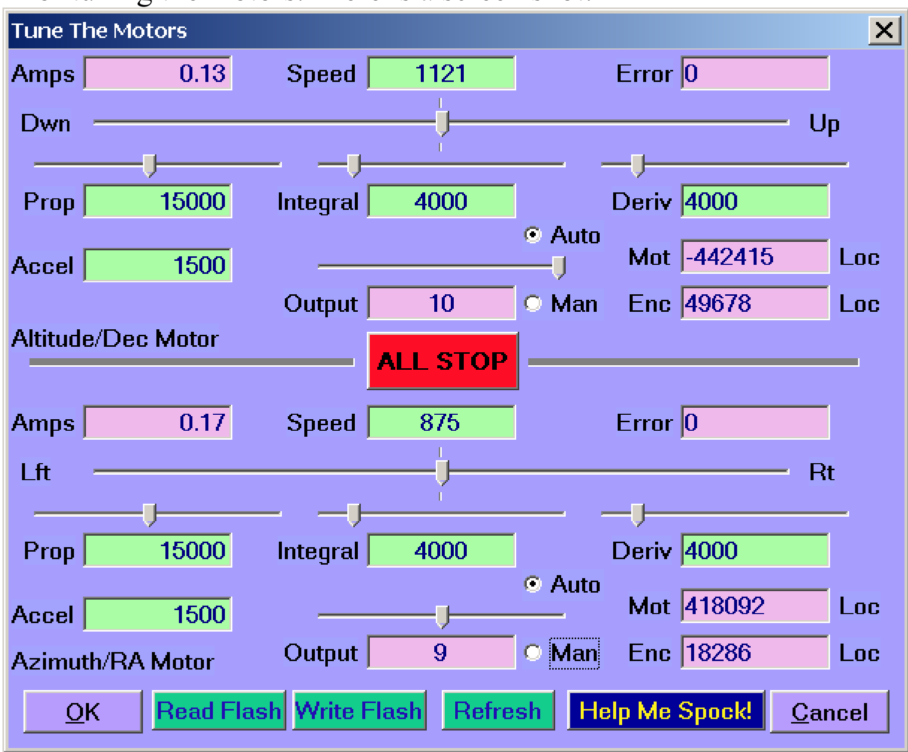
2. Use Mel Bartels SCOPEII using the SiTech Configuration option.. There is a batch file provided on the SCOPEII disk, called SiTech.bat. To use Mels software, you must have the Java Runtime installed. This software will work on any Java enabled machine.

3. Another option would be to use any windows terminal program and configure it yourself, by typing in the appropriate commands and values (see appendix B). This is the least user friendly mode, and is not recommended except for extreme nerds (like myself!).

IMPORTANT NOTE ABOUT SERVOCONFIG OPERATION:
On the CDRom you will have a copy of ServoConfig configuration software, or if not, it is available for downloading at http://www.siderealtechnology.com . If you are using version 0.9 (a pre-release version) **you MUST click on "Read Controller Config" after clicking on "Edit Parameters".** If you don't, you will be editing default values, not what is currently in the servo controller. Onced you've made the changes you want, **you MUST click on the "Misc and Action" tab and "Send Configuration to Controller"**. Many operation descriptions in this document assume that you will be doing this, and won't remind you again. Don't forget this if you are using version 0.9. If you are using version 1.0 or later, this will be done with prompts for you. At any rate, be sure values you are

editing, and the changes you make are from the controller, and go back to the controller.

TUNING YOUR MOTORS:
Hopefully (and very likely), the default motor tuning values will operate perfectly without any adjustment. If you feel you can improve on the factory defaults, or are using motors that require different tuning, the best way to tune your motors is use the ServoConfig software, which provides an interactive screen for tuning the motors. Here is a screen shot:



Step 1. Turn the integral all the way to zero.

Step 2. Turn the proportional down to zero.

Step 3, Adjust the Derivative. Typical proper values will be in the 5000 range. The motor should turn sluggishly but freely now, since only the derivative is working. Adjust the derivative higher and higher until the motor starts humming or vibrating. Adjust it back down until it doesn't vibrate, but "resists" you when you move the motor by hand. If the motor suddenly turns freely, you may find that the controller went to “manual”. You can change it back to “auto” by clicking on the “Auto” radio button.

Step 4, Adjust the Proportional Band. As soon as it is over a few hundred, the motor will "want" to return to it's desired position. Keep adjusting this up until the motor becomes unstable. Now back it down until the motor is stable. Typical values are about 15000.

Step 5, Adjust the Integral. To adjust the integral, start increasing the number from about 1000, until the motor becomes unstable. Then back it off until the motor is completely stable. Typical values are about 3000.

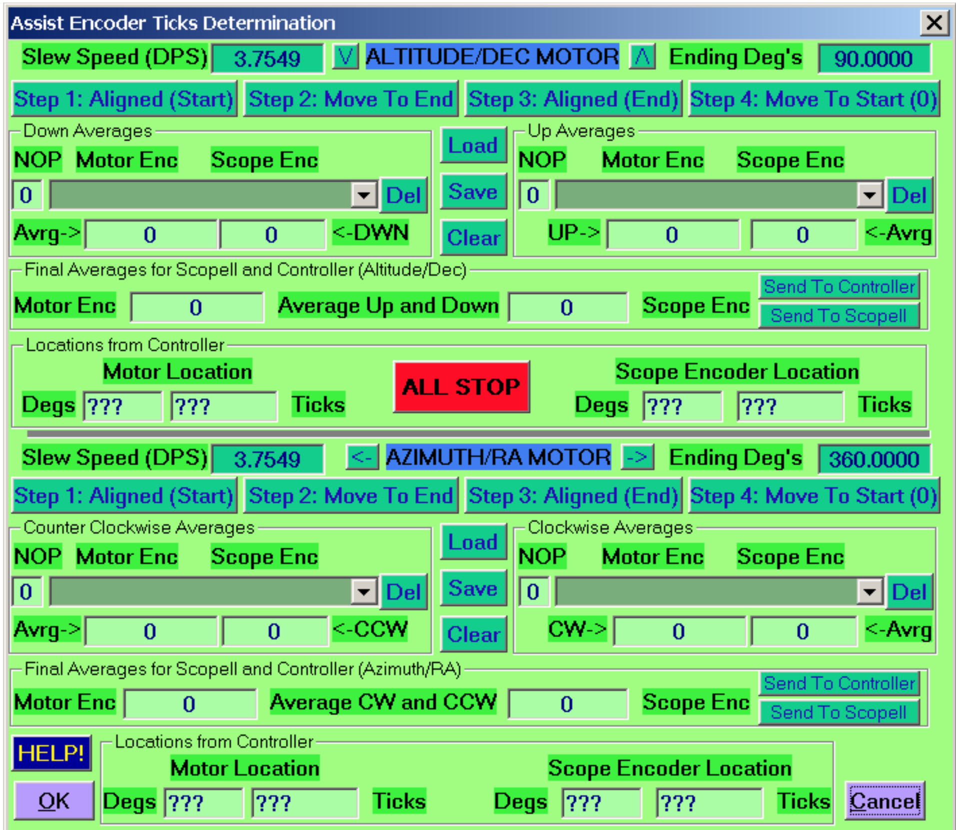
Step 6, Run the motor at several speeds, and be sure it operates smoothly at all speeds. If it doesn't, start by working backwards through the steps, first lower the integral, then lower the proportional. Play around until you're happy with the response of the motor. Pay special attention to the "Error" text box. While moving at tracking speeds, the error should be -1, 0, or 1, with an occasional -2 or 2. If this fluctuates more than this, maybe try increasing the proportional gain.

If the two motors are the same type, the same numbers should work in both motors.

DETERMINING AND SETTING YOUR NUMBER OF ENCODER TICKS:
One of the most important parameters to set in the controller and in ScopeII, are the number of ticks per revolution of the scopes axis. This is done by finding the encoder resolution of your motor, multiplying by 4, then multiplying by the gear ratio of your telescope. Here is an example:
You have a servo motor, it has a 500 count encoder on the motor. The motor has a built in gearbox with a ratio of 10:1. The motor is connected to a 28 tooth spur gear which drives a 320 tooth gear. The shaft on the 320 tooth gear has a 1 inch roller on it, which operates a disk which is 24 inches in diameter.

First we find the effective encoder ticks, which is 500 X 4 or 2000. We multiply this by 10, because of the built in gearbox on the motor. We now have 20,000 effective encoder ticks for each revolution of the servo motor output shaft. Now the first gear ratio is 11.42857 (320 / 28). Now we have 228,571effective encoder ticks for each revolution of the 320 tooth gear (and roller). Since we have a 24 inch disk and a 1 inch roller, we multiply the 228,571 times 24 or 5,485,714 effective encoder ticks for one revolution of the telescope axis. This is a very desireable number for most telescopes, as it is about 4 encoder ticks for each arc second of telescope motion. Values between 4 and 10 million are considered good, with acceptable slew speeds.

If you're using a large gear or wormgear on the final telescope axis, you can rely on the math for the gear ratio. If you're using a roller drive, sometimes the best measurements of the roller/disk diameters in the world aren't going to get you the proper number of encoder ticks per scope revolution. We've spent so much time at telescopes commanding the scope to move to the zenith, then the horizon, (and also complete revolutions in the Azimuth) that we finally got tired of it, and wrote software which does this for you. One of the options of the ServoConfig software is a button labeled "Ticks Determination". Please read the associated ServoConfig Document, or the online helps in the application to assist you. Here is a screen shot:



Don't let all those text boxes and buttons intimidate you, basically, it's pretty easy to use, just click Step 1, Step2, Step3, Step4, then end on Step1. Repeat as many times as you wish for statistical reasons. The software will calculate the average for you, and you have the option to save the averaged numbers to the controller.

Once you have determined the number of Motor and Telescope effective encoder ticks, these values must be entered in ScopeII, Mels new Windows Software, and the Servo Controller. To enter these numbers in the Servo Controller, use the ServoConfig software.

CHANGING THE ACCELERATION VALUES:

You may find it desireable to change the Ramp or Acceleration value. You can do this from the "Edit Parameters" menu from the ServoConfig software. Please read the associated online help files or the ServoConfig document for instructions.

Lower numbers provide very slow ramps and higher numbers provide faster ramps. Careful attention to the mechanics of your particular telescope can help you decide what works the best. The nice thing about a slower ramp is you can use the handpad and slew to your object, and pulse your handpad switches when you get to where your going. It's kind of like having infinately variable slew speeds available to you.

We've found from experience that values of 1000 to 1500 will work well. Please stay below values of about 3500.

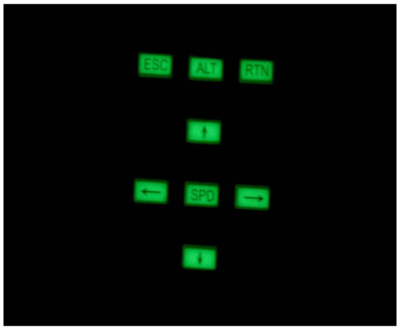
HAND PAD:
The controller is compatible with the old style of BBAstroDesign's hand pad. If you have a handpad of this type, be sure to put the 3 position jumper near the hand pad connector toward the micro processor (towards center of the board). If you are using a new handpad type (there is no switch on a new type, there are only push buttons), you need to be sure the jumper is toward the edge of the board. One other item if you use the old style handpad, you will have to put negative numbers in the pan and slew speeds for proper operation. If you are using the later board with the round motor connectors, then you can't use an old handpad.

The new handpad design allows the movement of both axis's at once, and also allows upgrading to the wireless handpad. The Slew/Pan switch on the old handpad is replaced with a push button. The controller turns the push-button into a "switch" internally, which is used to change between slewing or panning. There are certain functions in the Drag and Track and in the Platform mode that are not available if you don't have a new handpad type.

There is a radio handpad available as well. This is a really cool option, it eliminates 1 more cable to get tangled. This handpad is described on our website, and another document is available for download. Here are some of the features of the radio handpad:

1. **Connects to the Sidereal Technology Dual Servo Controller** for wireless remote control of your telescope. Eliminates the wired Handpad for tangle free operation.
2. **Can be used on Mel Bartels Stepper System:** The receiver has a mode which emulates the original Mel Bartels handpad, so can be used with the Mel Bartels Stepper system with only the application of a jumper and a ground wire on the reciever.
3. **Addressable:** (easily changeable with a keyboard sequence). This allows many users with many telescopes on the same telescope field.
4. **Low Duty Cycle Transmit Time:** Special micro controller routine only transmits about $1/20^{th}$ of the time, even when a button is continually pressed. There are no transmissions when no buttons are pressed. This allows multiple use from several identical handpads and telescopes without interruption of any one user.
5. **Instant, imperceivable delay** between a button press and action of the telescope. This is a real plus when centering an object at pan speeds, there will be no overshoot, as the scope starts and stops movement at exactly the same time as you press and release the key (perception anyway).
6. **Built in Micro-Processor Based astronomers LED flashlight.** There are 4 super bright LEDs which work in tandem. Th brightness is adjustable from very dim to very bright in 16 linear perceived steps. The brightness is “remembered” across a power down.
7. **Keypad lock feature:** When in the locked mode, only the 4 direction keys work. This is a great (nearly mandatory) feature for public star parties where the handpad is given to the general public for panning the moon at high power (it also helps us more experienced observers at 3:00 AM!!!!).
8. **Glow in the dark keypad:** A lot of research went into finding the perfect glow in the dark base and the percentage used. This glow in the dark is not too bright for even the most

2 Second time exposure in a dark room



hardcore dark sky phenatics! Approval was given by both Chuck Dethloff and Howard Banich (experienced dark sky observers from the Portland Oregon area). In addition to this, the glow lasts all night long, making it easy to find on the observing table in the darkest night. Although the labeling on the buttons may not be discernible, the shape (and thus the location of specific keys) are easily discerned.

9. **Keypad:** Reliable, custom manufactured Silicone One Piece Push Button pad, which press against gold plated contacts.
10. **Industry standard, autoguider port** is built in to the receiver. This port can be used to autoguide a telescope, with or without the ASCOM driver or Mel Bartels scope control software.
11. **Auto Off** for both the flashlight (15 minutes) and the transmitter (30 minutes).
12. **Long Battery Life** (Years, if buttons are not inadvertently left pressed). It uses easily replaceable standard 9 volt alkaline batteries.
13. **Wide keypad spacing** for gloved operation.
14. **Transmitter can be used with a direct connection** to the telescope controller (in case of interference or if the battery runs down (SiTech Controller Only).
15. **Handy necklace** is included.

TELESCOPE ENCODER WIRING AND SETUP:

Study the connection drawing at the end of this document, and you will see that you need to make a Y cable for the telescope encoders (if not purchased and supplied with the system). A 6 position modular jack is used on one end of a flat cable. The cable must be fabricated so it branches off to both telescope encoders. The 5 volts and ground must appear at both encoders. Once you have your encoders wired properly, connect your computer to the controller with the serial cable again, run the ServoConfig software. On the main screen of the software, you should see the encoder values change as the telescope is moved. The altitude numbers should increase on raising the altitude (or declination) and the azimuth should increase as the telescope is moved clockwise (birds eye view, or from Celestial Pole if equatorial mount). If the numbers move backwards, you can change the direction by swapping the A and B encoder signals, or use the ServoConfig software and invert the telescope encoder(s).

The resolution of the telescope encoders can be as much as 500,000 before missing encoder ticks is a problem.

SETTING UP SLEW AND PAN RATES:

Now you must set up your telescope slew and pan rates. This is accomplished using the ServoConfig software, and the "Edit Parameters" Button. You can enter them in raw numbers, or in Degrees per

Second. Be sure to set up the telescope and motor encoder ticks per revolution before entering these numbers as degrees per second. Be sure the raw numbers stay below about 6,000,000 or maximum 6,500,000.

Now if your motors are working properly, you can slew or pan using the 4 direction switches. If you are using the old handpad, the switch will change you from the slew or pan mode. If you are using the new hand paddle, pressing the center button for about ¼ of a second will switch between the slew and pan modes. If using SiTech controller firmware version 1.6 or later and are connected to the radio handpad reciever, the SPD switch on the handpad will toggle between Slew, Pan and Guide modes. In this case, the mode is indicated on the reciever.   If you are using the radio handpad, and you don't plan to engage in the art of guided astrophotography, you can use the guide rate as a slower pan rate.   This allows the selection of 3 different speeds using the radio handpad. If you are going to perform guided astrophotography, the guide speed should be adjusted for the best performance guiding.

If you are using an old hand paddle, you can't use all of the possible direction modes together, (including altitude and azimuth together). If you are using a new hand pad, or the radio handpad, you may use all modes of directions, including altitude and azimuth (or RA/Dec) at the same time.

SETTING UP THE BACKLASH:

The servo controller version 1.7 and later has a provision for backlash. The backlash feature can be a very useful feature, but also, it can make things worse if not used properly. If your scope is out of balance then the backlash may be taken out mechanically. If you were to put backlash in for this case, the backlash compensation would actually hurt!

Mels ScopeII also has this feature. It is extremely important that you don't have backlash enabled in both ScopeII and the servo controller. ScopeII backlash is fairly slow to respond, so we think it would be better to use the controller backlash, as the response is immediate. Also, if the telescope is being controlled by the ASCOM driver, the Argo Navis®, or any other device backlash MUST be enabled and adjusted in the controller in order to compensate for your mechanical backlash.

You must use ServoConfig to set up the backlash amounts. ServoConfig also has a useful utility for setting your backlash.

When the motor is in the backlash area, it means the telescope isn't being moved when the motor moves. We call this the No-Mans land. The Motor doesn't "like" to be in No-Mans land, and will move out of it in the proper direction as necessary. The speed in which the motor moves out of the No-Mans land is labeled Backlash Speed. This speed is added to the commanded speed, so it will always move out of this area.

Using ServoConfig, you set up the Backlash amount, and Backlash Speed.
For a slow backlash speed, consider about 30,000 raw counts. For a quick backlash speed, consider a value of about 500,000.   You need to find the right speed that quickly takes up the backlash, but doesn't jar the telescope when the backlash is taken up. The speed in in Minutes per second is an average of the two axis's, and may be different for each axis, depending on your motor encoder ticks per revolution.

SETTING UP AND USING THE LOCAL SEARCH:

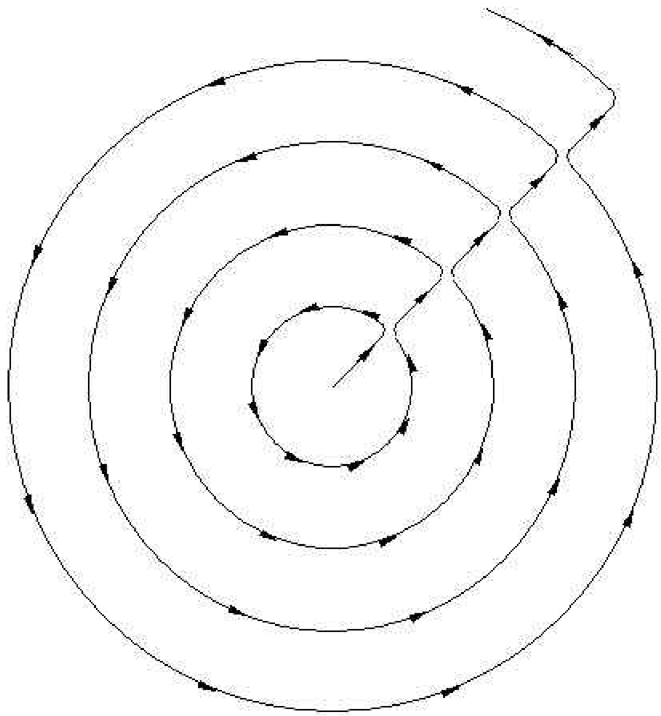
You must use ServoConfig to set up the local search radius and speed. See the ServoConfig document for directions to set this up.

As pointed out by Ed Harvey, a Spiral search isn't the most efficient search algorithm, because if you don't want to miss any sky, you will have a lot of overlapping sky. A more efficient algorithm is to move the scope 1 radius out, then do a circle, move another radius out, and do another circle, etc. This is how the local search routine works on the SiTech servo controller. You must have proper data in the controllers Motor Encoder Ticks per rev for the calibrated speed and distances to work properly.

A good starting point is to figure the field of view of your most frequently used high power eyepiece, and enter this number. When you use this eyepiece, initiate with the TOP LEFT / DOWN key. When using a lower power eyepiece, initiate the search with the TOP LEFT / UP key.

Note: It's not really a true circle, but a 16 sided polygon, which looks like a true circle at the eyepiece.

Here is a picture of the search routine:



You use the Local Search feature in the following manner:

1. Set up the Local Search Distance (radius) and speed (Degrees per second). Make sure the motor encoder ticks per revolution are set up properly too.

2. While aimed at the area of sky you believe the object to be is, hold down the top left key of the keypad, and then press the UP key. You can also initiate it for a higher power eyepiece which halves the circle radius, by holding down the Top Left key, and pressing the DOWN key.

3. To increase the search speed, hold down the Top Left key and press the UP key.

4. To decrease the search speed, hold down the Top Left key and press the DOWN key.

5. To reverse directions and start working inwards, press the DOWN key. You can revert to the normal direction by pressing the UP key. This can be done at any time during the search.

6. To PAUSE the search, press the speed key. To resume, press the UP, DOWN, or the speed key again.

7. To exit the Local Search mode, press the LEFT or the RIGHT direction key.

DRAG AND TRACK MODE:

The DragNTrack mode is used if you would like to push your telescope by hand, or slew with the handpad, but don't need super-accurate tracking, or goto capabilities, but would like computerless accurate tracking, the DragNTrack or SlewNTrack mode may be a nice feature to use. If you have a telescope of aperture 16 inches or more, it may be more cost effective, and there are many other advantages to use the dual servo controller and the Drag and Track mode instead of a tracking platform.

To use the drag and track mode, your Alt/Az telescope must have the following hardware (in addition to the servo controller):
1. A pair of servo motors to drive the telescope in altitude and azimuth.
2. Clutches on both altitude and azimuth connected between the motors and the telescope, so you can "drag" the telescope anywhere in the sky, without damaging the servo motors. (Note: you can use the DragNTrack mode, even if you have no clutches, but do have telescope encoders. You will have the dis-advantage of only slewing with the handpad, instead of pushing the telescope by hand).
3. A pair of encoders mounted to your telescopes' altitude and azimuth axis's.
4. A handpad.
5. A 12 -24 volt power supply.

A computer is not required for operation, only to configure your servo controller.

Setting up the Drag and Track mode:

For the drag and track mode to work properly, the controller must "know" several things.
1. The number of **motor** encoder ticks for a complete revolution of the telescope in both the altitude and the azimuth
2. The number of t**elescope** encoder ticks for a complete revolution of the telescope in both the altitude and the azimuth
3. How fast the slew setting should be
4. How fast the pan speed should be
5. (Optional) Latitude of the observing session

When you calculate the encoder ticks, it's important to understand the resolution of your encoders. Generally speaking, the effective number of ticks is 4 times the resolution of the encoder. This document uses the effective encoder ticks unless otherwise stated. If your encoders have 2048 as part of the part number, the encoder wheel has 2,048 lines on it. The effective encoder ticks will be 4 times this, or 8,192.

Using the ServoConfig software, and the “Edit Parameters” button, be sure all of the motor and encoder ticks per revolution are set properly (in effective encoder ticks). Also, be sure that the Right button on the handpad moves the scope clockwise, and the Up button on the handpad moves the telescope up. If not, change them using the features of ServoConfig. Now select the “Auto Tracking” tab. Now click on the “DragNTrack” radio button.

It would be good, but not absolutely necessary to enter the latitude of your observing site, which is also on the “Auto Tracking” tab.

For the DragNTrack to work properly, it's important to note that the UP handpad key needs to move the telescope up in altitude, and the RIGHT key needs move the telescope azimuth in a clockwise direction. It would be easy to put a negative number into the slew and pan rates to obtain these results, and the controller would respond with the opposite direction, but the trigonometry would still be wrong.

It's also important to note that when the telescope is moving clockwise, the azimuth telescope encoder should be increasing, and when the telescope is moving up, the altitude telescope encoder should be increasing. If these move backwards, then the direction needs to be changed using the ServoConfig software, or change the wiring of the encoders.

Note: If using an old BBAstroDesign handpad (the kind with a switch in the middle instead of a push button) and Servo firmware version 1.5 or earlier, the slew and pan speeds **MUST** be a negative number, but the direction of motion must still be the same as listed above (unless using servo version 1.6).

Operation of Drag and Track mode:

If you want the scope to start tracking without initialization, point the telescope at the celestial pole, and then turn the power on the controller. At this point, if you've already set up and saved your latitude, the system will be initialized. It will immediately start tracking. No further initialization is required (while pointing at the celestial pole, the scope won't appear to track because the celestial pole doesn't move!).

If your latitude isn't set up and saved to flash ROM, you will have to initialize at the scope zenith and the celestial pole. To initialize the Altitude, move the scope (by hand or with the handpad) up to the scope zenith. It could be convenient to have a mechanical stop mounted on your scope, so it is easy to find this location. Press and hold the **top right** (RTN) key for more than 4 seconds. At this point, the controller adjusts your altitude so it reads 90 deg's. (Important Note: there was a bug in the controller flash ROM in versions 1.5 and earlier, and this number was calculated wrong. If you have firmware version 1.5 or earlier, please upgrade to version 1.6 (or later) or enter the latitude using ServoConfig).

To initialize the azimuth, move the scope to the celestial pole. Now press and hold the **top left** key (ESC) for more than 4 seconds. At this point, the controller initializes the azimuth, and if you haven't initialized the altitude, it initializes the altitude to your saved latitude, otherwise it initializes the latitude.

If you have initialized both the altitude and azimuth, you may save the new latitude value by pressing and holding both top keys for more than 10 seconds. This saves the latitude to the flash ROM, so next time you can simply initialize the scope on the celestial pole, and press the left hand key for more than 4 seconds.

To perform photography, or for very fine centering, it may be desirable to enter the guide mode. This is done by holding down the top right key, then pressing the right key. Now the speed changes by adding or subtracting the guide rate from the current tracking rates. To revert back to the normal mode, hold down the top right key, and press the right key again. This feature doesn't work with the old BBAstroDesign handpads with a center switch. If you are using servo version 1.6 and you have a radio handpad reciever, you will toggle through slew/pan/guide modes using the SPD button.

You may slew the telescope with the handpad, or move the telescope by hand, and when done slewing or moving, the telescope will immediately start tracking again, at the new drive rates based on the telescopes altitude, azimuth and the latitude. To stop tracking, press the top right key. To start tracking again, press it again. When the tracking is stopped, the LED's will blink off briefly every few seconds (version 1.6 of firmware and later).

Review of Initialization options:
It may be confusing, because there are several ways to initialize the telescope. The user has three choices.

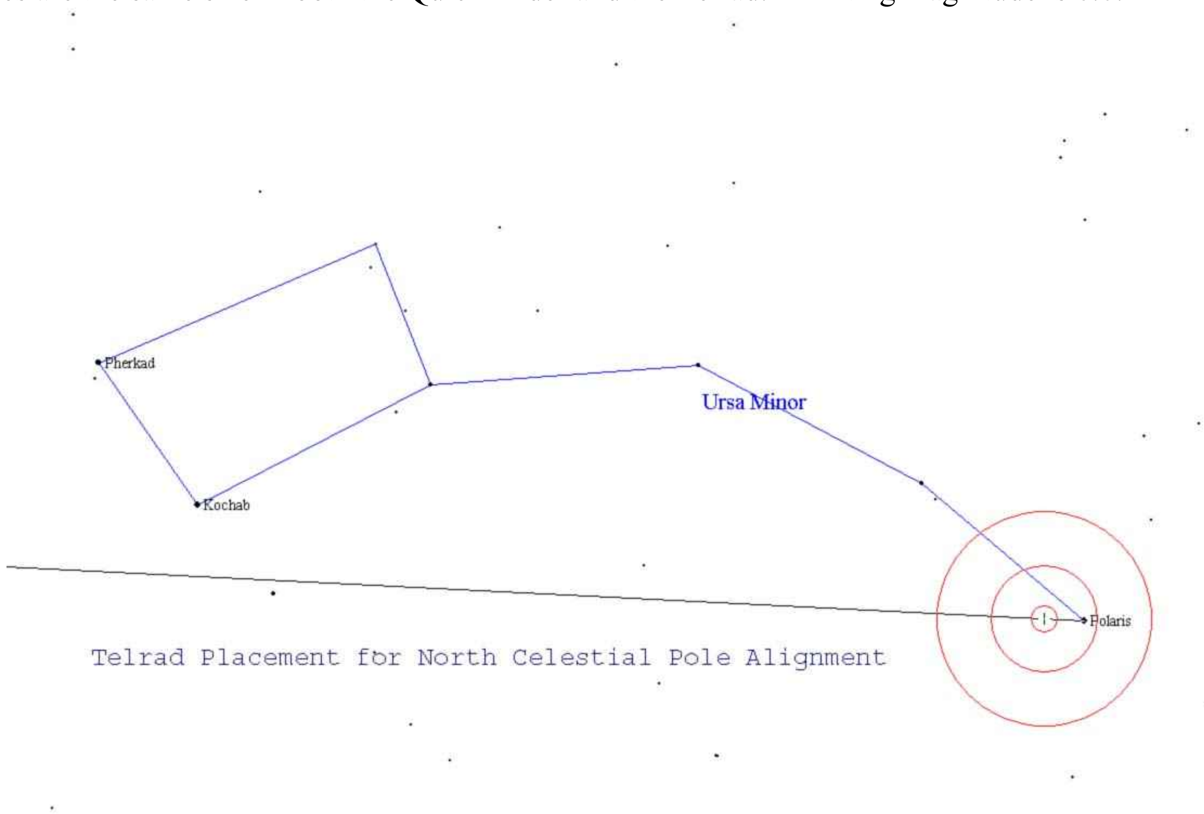
Choice 1:
Step 1. Before turning on power to the controller, position the telescope to the celestial pole. Turn on power.
The controller initializes the azimuth, and initializes the altitude to the same value as the last saved latitude. Using Choice 1 means you don't even have to use the hand paddle, although you must have previously saved your latitude.

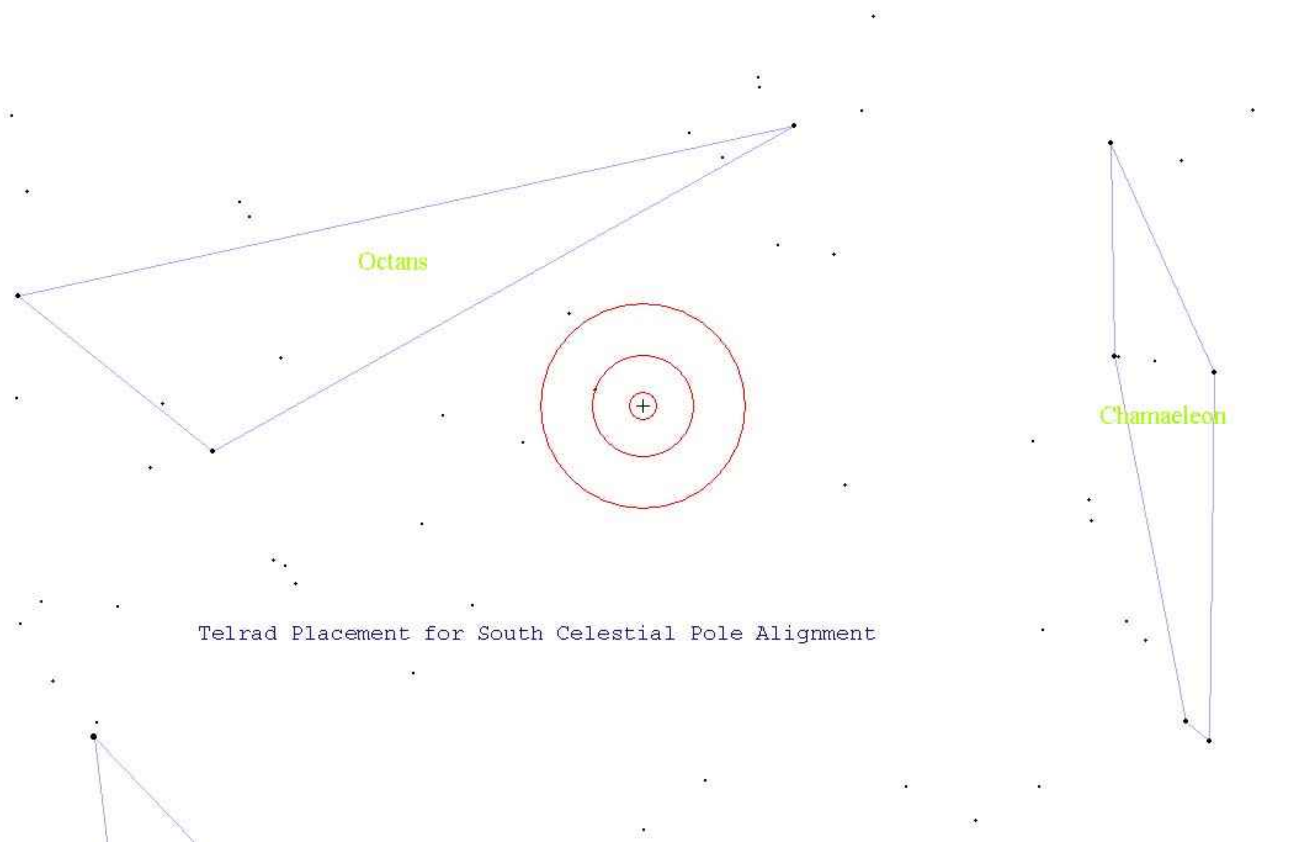
Choice 2:
Step 1. Move the scope to the telescope zenith and hold down the right hand bottom button for 4 seconds
The controller initializes the altitude position at 90 deg's.
Step 2. Move scope to celestial pole, and hold down the left hand bottom button for 4 seconds
The controller initializes the azimuth, and initializes the latitude. (Note, must have version 1.6 of firmware for this to work properly).

Choice 3:
Step 1. Move scope to celestial pole, and hold down the left hand bottom button for 4 seconds
The controller initializes the azimuth, and initializes the altitude at the same value as the last saved latitude.

Before a user can use choice 1 or choice 3, the latitude needs to be set up and saved to flash ROM.
If a user used choice 2, they may want to save the latitude, so next time they don't have to initialize at the zenith. This is done by holding down both the left and the right hand bottom buttons for more than 10 seconds. This action will save the latitude into the flash ROM, thereby making it possible that next time the telescope is set up (assuming the base is reasonably level, and you are near the same latitude), the user will not have to use choice 2, but only choice 1 or choice 3.

Here are some screen shots from Earth Centered Universe which show placement for celestial pole alignment. The QuickFinder has the two middle circles but not the outer circle. The two middle circles are the same size in both the QuickFinder and the Telrad. Limiting magnitude is 6.0.





While the controller is in the DragNTrack mode, the serial port can be connected to a laptop running almost any planetarium software. Set your software to connect to a "Tangent" encoder interface box, and you will be able to surf the sky using the planetarium software! When setting up your planetarium software, set up both encoder resolutions to 18000. Also, do not use the option that your telescope is on a tracking platform. Think of it like a telescope without a platform where you have this little guy pushing it all the time, exactly keeping up with the stars. That's what your planetarium program will "think"!

Also, set the baud rate of your planetarium software to 19200, no parity, 8 data bits, and 1 stop bit. If your software doesn't support 19200 baud rate, contact us, there is a way to set the controller to 9600 baud if you have firmware version 1.5 or later.

It's a good idea to set up the DragNTrack mode even if controlling with a computer or the Argo Navis™. When connected to a computer or Argo Navis™, the controller automatically leaves the DragNTrack, SlewNTrack or Equatorial modes, but it doesn't save the mode to the flash ROM. If you do this, you will have tracking (if initialized on the celestial pole) even before you connect the Argo Navis™ or the computer.

If tracking before connecting a computer or Argo Navis™ is not desired, there is no need to initialize the controller on the celestial pole.

SLEW AND TRACK MODE:

This mode is similar to the drag and track mode, but is used if you don't have shaft encoders on the altitude and azimuth of the telescope.
To use the Slew and Track mode, your Alt/Az telescope must have the following features:
1. A pair of servo motors to drive the telescope in altitude and azimuth.
2. A handpad.
3. A 12 -24 volt power supply.

This mode uses the servo motor encoders as the telescope position. You must use the handpad to slew the telescope. You must select the SlewNTrack mode from the “Auto Tracking” tab of the “Edit Parameters” of the ServoConfig software.

You must also set up the number of telescope encoder ticks to be the same number as the number of motor encoder ticks using the ServoConfig software. This is done automatically for you if you use the ServoConfig software, and you click on SlewNTrack mode.

Important Note: If your telescope has encoders for altitude and azimuth, but doesn't have clutches, your best bet would be to use the DragNTrack mode, but slew with the handpad instead of “dragging” the scope.

If you don't have external encoders, and you plan to use the SlewNTrack mode, you must set your controller up for SlewNTrack so the controller will ignore the telescope encoders if using the Argo Navis™. If tracking before connecting a computer or Argo Navis™ is not desired, there is no need to initialize the controller on the celestial pole.

EQUATORIAL MODE:

The Equatorial mode is provided for equatorial platforms, or equatorial mounts. When configured properly, the Right Ascension motor will start tracking when powered up, at the rate specified.

Setting Up the Equatorial Mode:
Using the ServoConfig software, click on “Edit Parameters”, then “Auto Tracking”. Click on the Equatorial mode radio button. Set up the Equatorial rate. If you know the number of effective encoder ticks for one revolution of the Right Ascension axis, enter that number on the green text box to the right. After pressing the Enter key, the proper Equatorial rate will be calculated and will appear in the “equatorial rate” text box.

You can adjust the tracking rate up and down by pressing a certain sequence on the handpad. The amount it is adjusted per handpad press is in the text box labeled “Equatorial Up/Down Adjust”.

If you have a platform, set the “Tracking Platform Goal” to the number of motor encoder ticks where you want the platform to stop. The SiTech controller will stop at this point, and then you can then perform a re-wind.

If you have an equatorial telescope, set this platform goal to 10,000,000. This will make it impossible to stop at the platform goal, it will continually track forever (if you don't stop it).

If the motor tracks the wrong way, you can put a negative number in the "Tracking Platform Goal".

Operation of the Equatorial or tracking platform mode:
If using an equatorial platform, before powering up the controller, be sure the platform is fully rewound. This will be it's home position.
Upon power up, or after a rewind, the Right Ascension motor will turn at the configured speed, until it reaches the configured goal. To rewind, press both top switches. This is possible at anytime. The servo motor will return to the same location it was when power was turned on; at it's programmed slew speed rate.
To Cancel rewind, press any direction key.

If using an equatorial telescope, the above paragraph will not be applicable, as the platform goal will be set for 10,000,000 or greater.

To stop tracking, press the top right key. To start tracking again, press it again. When the tracking is stopped, the LED's will blink off briefly every few seconds (version 1.6 of firmware and later).
Only the new handpad will allow fine tuning the tracking speed.
To raise the tracking speed, press the top left key, and while holding it down (think of it as an ALT key) press the up direction.
To lower the tracking speed, press the top left key, and while holding it down (think of it as an ALT key) press the down direction.
The up/down adjustment will not work if you have set up the local search feature (version 1.7 or later of the firmware). To enable the UP/Down adjustments, you will have to enter a value of zero in the local search distance and speed parameters.

To perform photography, it may be necessary to enter the guide mode. This is done by holding down the top right key, then pressing the right key. Now the speed changes by adding or subtracting the guide rate from the current rates. To revert back to the normal mode, hold down the top right key, and press the right key again. This feature only works with the new handpad. This will be un-necessary if you are using a radio handpad reciever, as in this case, the SPD switch on the transmitter will toggle between Slew/Pan/Guide modes.

Challenging Idea: If you have or have access to, a very large aperture telescope, 28 inches or larger, you may be able to see the crab pulsar blink. Here's how. Obtain a servo motor with encoder that is able to run at least 1,800 RPM, you could connect this motor to a servo controller, and put an interupter on the shaft of the motor (a circle with a pie shape cut out of it), balance the interupter, and hold this assembly between the secondary mirror and the eyepiece, while running at somewhat less than 1800 RPM. You may be able to see the pulsar wink in and out! To test this out, you may use one of the LED's. It winks at the same frequency of the crab pulsar while in the equatorial mode! You can configure the equatorial mode to automatically run this motor at the proper speed.

If you do this, please let us know, we would really like to hear about it!

OPERATION WITH ARGO NAVIS™:
With Version 1.6 of the firmware and later, it is possible to use the Sidereal Technology Servo Controller in the GoTo and Tracking mode, using an Argo Navis™ Digital Telescope Computer. The Argo Navis™ needs to have version 1.1.6 or later installed in its firmware. You can download and upgrade either or both units by visiting the appropriate website.

The telescope encoder information comes from the Sidereal Technology Servo Controller instead of the encoders themselves. There is only one cable connected to the Argo Navis™, and it is the Serial Cable. See Appendix D for the cable diagram. If you don't have telescope encoders, not to worry, because you can select the SlewNTrack mode of operation, and now the encoder information will come from the servo motors, not the telescope encoders. Also, if you do use telescope encoders, they will be powered from the Servo Controller power, thus the Argo Navis™ batteries will last much longer.

Setting up the Argo Navis™:
Several items need to be set up on the Argo Navis™ before operation. Go to the Setup menu, then select the Serial Port then Serial 1. Make the Startup command be sitech. Also, set the baud rate to 19200.

Go to Setup and find the Altitude and Azimuth encoder resolutions. Change these to +18000. Set up other parameters as required for your telescope, such as location, etc. Be sure to read the Argo Navis™ manual. Note, the encoder resolution parameters in the Argo Navis™ do need to be set for +18000, regardless of the actual encoder resolution. This is extremely important. The Servo Controller always scales the encoder ticks for 18000, even if you have other resolution encoders.

Setting up the SiTech controller:
Now you need to set up some things in the Sidereal Technology Servo Controller.
Using the ServoConfig software, click on the "Edit Parameters" button, then on the "Misc and Action" tab. You must have version 1.0 or later of the ServoConfig software. Check the box labeled "Use Argo Navis™"

If you don't have telescope encoders, set the Sidereal Technology Servo Controller in the "SlewNTrack" mode. If you do have telescope encoders, put the Sidereal Technology Servo Controller in the "DragNTrack" mode. Please read the section on DragNTrack and SlewNTrack for setting in this mode. Be sure all of the parameters are set properly for proper SlewNTrack or DragNTrack operation. Be sure the telescope tracks reasonably accurately in the auto track mode.

Setting your SiTech to be in the DragNTrack or SlewNTrack modes has several advantages. In addition to proving that all the telescope encoder resolutions and directions are programmed properly, it has the advantage that if you want to simply track without the Argo Navis™ on a particular evening, you can initialize on the Celestial Pole, and then you can have tracking without the Argo Navis™, or if you connect the Argo Navis™, you can have more accurate tracking and also GOTO's.

Important: If you don't have external encoders, you MUST select the "Ignore Encoders" option, so the controller will ignore the external encoders.

Operation of the Argo Navis™:
OK, now connect the Sidereal Technology Servo Controller to the Argo Navis™ using a serial cable fabricated as shown in appendix D. As soon as you press both top buttons on the handpad (ESC and RTN) at the same time, the communication with the Argo Navis™ will begin, and the DragNTrack or SlewNTrack modes will be disabled. To check that the communication is working properly, go to the Encoder menu item in the Argo Navis™, and make sure the angles change with telescope movement.

You are now ready to initialize the Argo Navis™. Please follow instructions in the Argo Navis™ manual for initializing the Argo Navis™.

Now the telescope should start tracking properly when the Argo Navis™ is initialized. For GOTO's, simply select an object using the Argo Navis™, then after confirming that the object is above the horizon, press the handpad ESC and RTN buttons at the same time (Top Left and Top Right buttons). The scope should now move to the object.

To stop a slew, press any direction key on the handpad. To restart the slew, press the ESC and RTN (Top Left and Top Right) buttons again.

To stop tracking, press the top right key. To start tracking again, press it again. When the tracking is stopped, the LED's will blink off briefly every few seconds (version 1.6 of firmware and later).

Some useful info about the Argo Navis™ mode. The SiTech controller does two slews each time you slew to another object. The first slew gets it in the area, then the final slew takes it to the exact position. The reason it does this, is because on long slews, the earth will turn, and thus the location will have changed after the 1st long slew.

TROUBLESHOOTING:

Motor operation:

Problem: When power is applied to the SiTech controller and a handpad direction button is pressed, the motor slowly increases, faster and faster until it's full speed.
Solution: The motor encoder is not supplying feedback. Check cabling, connectors, and encoder You can swap the altitude and azimuth motors for testing. See if the problem follows the motor, or if the problem remains. If the problem remains, there may be an issue with the controller. If it follows the motor, it is definitely a wiring or encoder problem.

Problem: Motor runs away as soon as power is applied or commanded to move. After running fast for a short time, it stops, and the light flashes.
Solution: The motor encoder relationship to the motor polarity is wrong. See section on "Setting Up the Motors".

Problem: When commanded to move, the motor starts ramping up, then stops, and the LED flashes.
Solution: It could be that you are using higher voltage motors with only 12 volts applied to the controller. Either slow down the speed, or use a higher voltage on the power to the controller.

Problem: Nothing works at all, and the two LED's are on very faintly.
Solution: There is a major problem with the CPU. If you were using version 1.3 or earlier, there was a bug that allowed a remote chance of the flash memory to be erased. You will have to send the controller back to the factory for a re-program.

Problem: Motor gets abnormally hot:
Solution: Check the motor current. You can do this using the ServoConfig software. The motor current appears on the main screen. If more than 0.5 amp while the scope is moving slowly, you may have more friction on the gear train, or maybe your scope is out of balance. Sometimes worm gears add a lot of friction to a drive system. Be sure the thrust clearance on the worm is not too tight, and be sure the worm pressure against the wormgear is not too much. Be sure lubricant is applied properly.

Problem: Upon applying power, one of the motors starts slewing, all by itself. Nothing will stop it.
Solution: Unplug the handpad and see if it still happens. If it stops moving, you've a faulty handpad. If it keeps on moving, it may be there is a problem with the digital input for the handpad. Please contact us.

Handpad Operation Troubleshooting:

Problem: I'm using an old style handpad (the one with a center switch) and the scope only moves in two directions instead of 4.
Solution: Check that the internal jumper is set properly for the old handpad.

Problem: Some of the switches don't work properly.
Solution: Use the ServoConfig software. At the main screen, look at the virtual handpad. Pressing each handpad button should make the appropriate virtual button highlight. If not, check the handpad

for problems. If the virtual handpad works properly, check the slew and pan rates.

DragNTrack mode:
Problem: I initialize on the zenith, then the celestial pole, but it doesn't track properly.
Solution: Maybe the firmware version of the SiTech controller is version 1.5 or lower. If so, please upgrade to version 1.6 or later. There was a bug in the calculate latitude routine in versions 1.5 or earlier.

Problem: I live in the southern hemisphere, and the tracking is the wrong direction. I have initialized on the Zenith, then the south celestial pole.
Solution: The SiTech controller has no way of knowing which hemisphere it's in when initializing on the Zenith or the Altitude. You've got to set the Latitude to a negative number (using ServoConfig) before it will work properly. Once a negative, the initialization process will always keep it a negative number.

Problem: I know the Latitude is correct, but it doesn't track properly.
Solutions: 1. If you've a clutched telescope, check for clutch slippage on tracking. Sometimes the clutches will work on a slew, but will slip during tracking.
2. Double check the encoder and motor ticks per scope revolution for altitude and azimuth.
3. Be sure the scope encoder and motor encoder polarity is set up properly. You can do this at the main screen of the ServoConfig program. While you watch the scope encoder and motor encoder location for the altitude, raise the scope. Both the Scope encoder and the Motor encoder should increase. Check also the azimuth. They should both increase when the scope moves from north to east.

Problem: It tracks fairly ok, but I think it should track better.
Solutions: This could be caused by several factors, the most common one is celestial pole mis-alignment. Use the maps provided in this document, and star hop to the celestial pole.

It could be that the latitude isn't set up properly.

It could be that your mount is not level in the north/south direction.

It could be that your drive system or encoders are slipping.

It could also be caused by the wrong values in the scope encoder configuration (or Motor Encoder if using the SlewNTrack mode). To check this, use ServoConfig and from the main screen, keep clicking on the “Get Angles” button. Start at the horizon then move the scope to the zenith. *It should have moved 90 deg's. Do the same for the azimuth. Moving you scope clockwise will give you an increase, starting at North, should be 0, then to the East as 90, South, is 180, West is 270, etc.

Also check the motor encoder ticks per rev. If these are wrong, the motors will operate at the wrong rates. Keep in mind that the tracking math is proven, and it works great. When things are set up properly, and you've a good mount with no slippage, the scope should track with only 3 or 4 arc minutes of drift per hour. Howard Banich can track an object at 450 power for an hour in his 28 inch scope!

**Important Note:* If you are using ServoConfig version 0.9, the Azimuth reads backwards from above description when clicking on Get Angles!!!

Problem: The telescope doesn't track at all sometimes.
Solutions: 1. Press the top right key to turn tracking on or off.
2. Be sure it really is in the DragNTrack mode. LED 2 should turn on and off every 8 seconds or so. If it does this, it really is in the DragNTrack (or SlewNTrack) mode. If it doesn't do this, the scope is not in the DragNTrack or SlewNTrack mode.

Problem: I'm using an old style handpad (the one with a center switch) and the DragNTrack doesn't work properly.
Solution: If you are using an old style handpad, and you are using Servo version 1.5 or earlier, you must enter negative values for the Slew and Pan rates. After you check this, be sure the scope moves from north to east when pressing the right button, and be sure the scope moves up when pressing the up button.

SlewNTrack mode: Same as DragNTrack mode.

Equatorial Mode:
Problem: I have an equatorial telescope. The telescope tracks fine for a while, then stops.
Solution: Using ServoConfig, increase the Platform Goal to a number greater than 9,999,999 or less than –9,999,999.

Problem: The equatorial mode works, but the direction is the wrong way.
Solution: Using ServoConfig, change the Platform Goal from positive to negative (or vice versa).

Problem: The equatorial mode works, but the stars drift in Right Ascension.
Solution: Check for slippage if a roller drive. Also check the Equatorial rate using ServoConfig.

Argo Navis™ mode:
Problem: The Argo Navis™ mode doesn't work at all. I pressed both top left and top right buttons simultaneously. The encoder values don't change on the Argo Navis™ when I move the scope.
Solutions: Controller firmware must be at least version 1.6. Argo Navis™ firmware should be at least 1.1.6 Check the serial cable. Check that the SiTech controller is configured for Argo Navis™. Check that the Argo Navis™ Com Port is set up to run the SiTech command at startup. Check that the Argo Navis™ serial port is set for 19200 baud.

Problem: I don't have encoders on my scope, and I'm trying to use the Argo Navis™, but the encoder locations on the Argo Navis™ don't change when I move the scope.
Solution: Be sure the SiTech controller is in the SlewNTrack mode.

Problem: I can do a normal Argo Navis™ Initialization, but the tracking and GoTo's are inaccurate.
Solutions: Set up your telescope in the DragNTrack (or SlewNTrack if you don't have scope encoders) mode. Be sure to get that working well. If the DragNTrack (or SlewNTrack) doesn't work properly, neither will the Argo Navis™ mode. Be sure the Argo Navis™ is set up properly. Check the encoders on the Argo Navis™. They should change at the same angles as you move the telescope. If not, check the Argo Navis™ encoder resolution. It should be +18,000. Check the motor and scope encoder values in the SiTech configuration. If necessary, use ServoConfig and Ticks Determination to find the encoder values. Remember, the math in the Servo controller has been proven, time and time again, it is a mechanical or a configuration problem.

Problem: When I do a slew, the telescope moves close to the object, then moves again about the same distance in azimuth, or altitude.
Solution: Sounds like one of your scope encoders is not working properly. Check the scope encoders and/or wiring. This is easiest using the ServoConfig main screen.

Guide Mode
Problem: I have an equatorial mount, and I'm writing my own software to control the scope. The declination doesn't guide properly.
Solution: The controller adds or subtracts the guide rate to the current rate. It is possible if your declination motor is at it's destination, the guide mode gets mixed up. Try giving the motor a command to move "far far away", with a rate of ZERO. This should fix the problem. This is not an issue with Servo version 1.6 or later.

Problem: The guiding doesn't seem to work properly.
Solution: Connect up the autoguider and try the test mode on your guiding software. Run the ServoConfig software. Be sure each of the virtual handpad keys are highlighted when the guider is guiding in a particular direction. Be sure that the SiTech controller is in the Guide mode.

Problem: It doesn't seem to guide at all, but it is panning or slewing when guide commands are sent.
Solution: Be sure it's in the guide mode. This is a software issue if using the ASCOM software or ScopeII. If in DragNTrack or Equatorial mode, hold down the top right key, and press the right key to toggle the guide mode. If you are using the Radio Handpad Reciever, and are using the guiding input provided, this shouldn't be a problem, because the controller will automatically change to the guide mode when a guiding signal is present (if using Servo Version 1.6 or later).

FUTURE PLANS:
We are planning the following:
1. Add PEC to the SiTech ASCOM driver
2. Create a simple GoTo/Tracking/Database to a JavaME program, so the scope can be controlled by a cell phone.

APPENDIX A - AGENCY APPROVALS:

FCC:
This equipment has been tested and found to comply with the limits for a class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:
* Reorient or relocate the receiving antenna.
* Increase the separation between the equipment and receiver.
* Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
* Consult the dealer or an experienced radio/TV technician for help.

The user is cautioned that changes and modifications made to the equipment without the approval of manufacturer could void the user's authority to operate this equipment.

C-Tick
Manufacturers Name: Sidereal Technology
Manufacturers Address: 6040 N. Cutter Circle #302
Portland, OR 97217 (USA)

Australia Representative Name:
Wildcard Innovations
ACA Suppliers Code: N11511

Declares that the products
Dual Servo Telescope Controller, Model DSTC 1
Handpad Transmitter, Model HT1
and Handpad Receiver, Model HR 1
conforms to the following standards:
EMC: AS/NZS 3548 Class B Complies
AS/NZS 61000-4-3 Complies, Criterion A
These units are not for mains connections.

Industry Canada:
United Sates Company number: 5942A
Canadian Representative: Nova Astronomics
Canadian Representative Company number: 5930A

APPENDIX B: UNDERSTANDING AND USING THE X_BITS AND Y_BITS:
This section is written for protocol developers only. You will not have to understand this section if you use the ServoConfig software.

The controller has two bytes that are used to change various options, or read the digital inputs from the handpad reciever. One byte is dedicated to the X axis (altitude), the other is dedicated to the Y axis (azimuth). Some of the bits in the X bits are common to both motors.

A byte is a group of 8 bits. The first bit, bit 0, has a "weight" of 1. The second bit, bit 1, has a "weight" of two. The 3rd bit, bit 2, has a weight of 4, and so on until the eighth bit, bit 7, has a "weight" of 128.

Bit 0: Weight of 1: Altitude Motor Encoder Direction
Bit 1: Weight of 2: Altitude Motor polarity
Bit 2: Weight of 4: Altitude Scope Encoder Direction
Bit 3: Weight of 8: A '1' here, puts the controller in the Drag and Track or Slew and Track Mode (**X Only**)
Bit 3: Weight of 8: A '1' here, puts the controller in the Slew and Track Mode **(Y Only)**
Bit 4: Weight of 16: A '1' here, puts the controller in the Platform mode (X Only). Digital Input from the Radio Handpad Reciever if you are using Servo Version 1.6 (Y Only).
Bit 5: Weight of 32: A '1' here, enables the handpad to control the motors. You must also set up the Slew and Pan rates (X Only). Digital Input from the Radio Handpad Reciever if you are using Servo Version 1.6 (Y Only).
Bit 6: Weight of 64 A '1' here, tells the controller you're using a "new" handpad, one with only push buttons, no switch (X Only). Digital Input from the Radio Handpad Reciever if you are using Servo Version 1.6 (Y Only).
Bit 7: Weight of 128: A '1' here, tells the controller it's in the "guide" mode (X Only). In the guide mode, when a direction button is pushed, the guide value is added or subtracted from the current speed. You must also set up the Guide rates. Digital Input from the Radio Handpad Reciever if you are using Servo Version 1.6 (Y Only).

To figure out what number to program into the X or Y bits, figure out the modes you want, then add up all the "weights" of the bits that are a '1', and that is the number you use.

As an example, lets say you want to use the drag and track mode, you're using a new handpad, you're altitude motor needs to be reversed and the altitude telescope encoder needs to be reversed.

Bit 1 = 2, reverse the motor polarity
Bit 2 = 4, reverse the scope encoder
Bit 3 = 8, Drag and Track mode
Bit 5 = 32, Enable the handpad
Bit 6 = 64, Using a new handpad.

This adds up to 110. Issue the command "XB110<E>". Now read back the value, "XB<E>", the controller responds with "B110". Now save this to flash by issuing the command "XW<E>".

Substitute 'Y' for 'X' for the azimuth motor.

Here is a description of the bits:
Bit 0: If this bit is a '0', the motor encoder is incremented normally. If this bit is a '1', then it is reversed.
Bit 1: If this bit is a '0', the motor moves one direction for a given number. If it is a 1, it moves the opposite direction.
This being said about bit 0 and bit 1, it is important to understand the implications of changing these bits. Lets say that the controller has just been powered up. In this state, before you've pressed a handpad button, or given the motor a command to move, the controller is trying very hard to keep the motor at position zero. If the position moves to 1, 2, or 3, the controller responds by putting an output to the motor to move it the opposite direction, and it will move back, 3, 2, 1, 0. If the motor starts moving in the other direction, -1, -2, -3, the controller responds by putting an output to the motor to move it back to zero, the opposite polarity as in the first example. Now try to think what would happen if the position moved to 1, 2, or 3, but the controller responded by moving the motor to even higher numbers. This would create positive feedback, and the motor will "run away" out of control. More Error would create more wrong voltage, which would create more error, etc, until maximum output voltage is reached.

Bit 0 and bit 1 of X_Bits and Y_Bits give you control over which direction the motor turns, and which way the encoder counts.

Bit 0 and bit 1 rules:

1. If the motor runs away, you can invert either one of the bits, bit zero or bit 1. The motor will now run properly.
2. If the motor runs properly, but in the wrong direction, you must invert both bits.

Bit 2: If this bit is a '0', the telescope encoder is incremented normally. If this bit is a '1', then it is reversed.

APPENDIX C: COMMUNICATION PROTOCOL:
You will not need to read or understand this section unless you intend to write your own telescope driver.

List of ASCII commands (all get terminated with an <E> (CR)). Be sure to use upper case.
For the Azimuth or Right Ascension servo, use 'Y' instead of 'X' The # symbol means there is a number required.
The X servo is the Altitude or Declination. The Y servo is the Azimuth or Right Ascension.
If you change the Address of the controller (“AD1”, “AD3”, or “AD5”), the controller will respond differently. If the address is 3, then all beginning X's should be T's, and beginning Y's should be U's. If you've changed your address to Address 5, then use V and W for X and Y.

To change the address of a controller from the default (1) to (2), send the byte sequence: “AD2<E>” (Make sure any other controllers are not connected to the serial port first). If you want to make this permanent, you will have to write to the flash ROM. This would normally be the “XW” command, but since the controller address is 2 now, you would issue the command “TW”.

If the controller is Address 1, and if you send an <E> with nothing before it, the controller responds with a lot of useful information as follows:
X# Y# XZ# YZ# XC# YC# V# T# X(A or M) Y(A or M) K#

Where:
X# is the location of the Altitude/Dec Axis in MOTOR encoder ticks
Y# is the location of the Azimuth/RA Axis in MOTOR encoder ticks
XZ# is the location of the Altitude/Dec Axis in SCOPE encoder ticks
YZ# is the location of the Azimuth/RA Axis in SCOPE encoder ticks
XC# is the Altitude/Dec motor current * 100.0
YC# is the Azimuth/RA motor current * 100.0
V# is the controller power supply voltage * 10.0
T# is the controller CPU temperature (Deg's F)
XA (or XM) is the Altitude/Dec Motor status, if XA, it's in Auto (normal) if XM, it's in manual.
YA (or YM) is the Azimuth/RA Motor status, if YA, it's in Auto (normal) if YM, it's in manual.
K# is the handpad Status. (Version 1.5 and later).

The handpad number is an 8 bit number where each bit represents a push button.
The handpad bits are as follows:
Bit 0 Left
Bit 1 Right
Bit 2 Up
Bit 3 Down
Bit 4 Speed (this is a toggle, a 1 is Pan mode and a 0 is slew mode).
Bit 5 Top Right key
Bit 6 Top Left key
Bit 7 Changed key.
The Changed key (bit 7) is used to tell if someone has pressed a handpad button since the last speed

command is received. This feature is useful so a new tracking coordinate can be obtained, and the new tracking setpoint will track from the new position. This bit is automatically cleared if a new speed command is received.

About the Velocity:
Here are some "C++" functions that may help you calculate velocities:

```
double DegsPerSec2MotorSpeed(double dps, double TicksPerRev)
{
	return Round(TicksPerRev * dps * 0.09321272116971);
}
double MotorSpeed2DegsPerSec(double Speed, double TicksPerRev)
{
	if (TicksPerRev == 0.0) TicksPerRev = 1.0; // so we don't ever divide by zero
	return Speed / TicksPerRev * 10.7281494140625;
}
public static double TicksPerRev2SiderealRate(double TicksPerRev)
{
	return Round(TicksPerRev * 0.000389468194);
}
```

To really understand the speed relationship between the speed value and RPM, please have a look at the PicServo document, available at: http://www.jrkerr.com/psdata.pdf

ASCII COMMANDS:

SB# Set Baud Rate (1 = 9600, 2 = 19200) Changes all controllers on serial bus. (1.5 or later)
X# Move Servo (-2147483648 to +2147483647) You can tag a speed command at the end.
Example: **X**-2345**S**1000000<E>
X Returns the X position of the servo
XF# Forces the X position to be equal to the number (-2147483648 to +2147483647) (This stops the controller if moving)
XS# Velocity of X Servo (0-2147483647)
XS Returns the X velocity of the servo (what it would be if it was currently moving at fully accelerated speed, this may not be what it acually is at the moment).
XR# Ramping speed or Acceleration of X servo (0-3900)
XR Returns the Ramp speed.
XP# Proportional Band of X servo (0-32767)
XP Returns the Proportional band
XI# X Integral (0-32767)
XI Returns the Integral

XD# X Derivative (0-32767)
XD Returns the Integral
XE# Maximum position error limit before servo turns off (0-32767)

XEL Returns position error limit.
XE   Returns the position error of the servo
XO# X Output limit (0-255)
XO   Returns the PWM output of the servo (0-255)
XC# X Current Limit (0-240 = 0-2.40 amps)
XC   Returns the X motor current * 100 (240 MAX)
XM# X to manual mode, the number is the PWM value, -255 to +255.
XA   X to Auto mode.
XN   X Normal Stop   (ramps down, then stops. Automatically clears when new position is given)
XNT Just like XN, except when it slows down enough, it starts tracking in the DragNTrack or SlewNTrack mode.
XG   X Emergency stop   (stops immediately)
XL# Set Integral Limit (0-24000)
XL   Returns Integral Limit
XB# Number sets the servo bits like direction, etc. (0-255)
XB   Returns current X Bits Value.
XZ# Forces the scope encoder position to be equal to the number (-2147483648 to +2147483647)
XZ   Returns the scope encoder position

The following commands don't have a 'Y' command, only X. They affect both servos
XK   Returns the handpad info in Decimal.
XH   Returns the temperature of the CPU chip (in deg's F)
XV   Returns the firmware version * 10.
XJ   Returns the motor power supply voltage * 10 (please divide returned number by 10)
XQ   Resets the servo system (both)
XU   Programs factory defaults into the flash ROM.
XW   Writes the configuration of both the x and y parameters from working RAM to the flash ROM.
XT   Reads the configuration from the Flash Rom into the working RAM.
FC   Flash Configure, Send 128 more bytes, then 2 bytes of checksum (Simple Addition of all data bytes sent), and it will write the new data to configuration. It also reads the new data into the working RAM.

SC   Send Configuration, The controller Sends 128 bytes, followed by two bytes of checksum. This data is what is in the flash, not necessarily what is in the working RAM.

All of the following XX extended commands have a corresponding read command
as an example, if you type "XXL<E>" it responds with the latitude.

XXL# Store the latitude to the controller (4500 = 45 deg's north, -4500 = 45 deg's south)
XXZ# Store the azimuth encoder ticks per full circle.
XXT# Store the altitude encoder ticks per full circle.
XXU# Store the number of encoder ticks for the Altitude Motor Encoder to the controller
XXV# Store the number of encoder ticks for the Azimuth Motor Encoder to the controller
XXA# Stores the Altitude Slew Rate to the controller

XXB# Stores the Azimuth Slew Rate to the controller
XXC# Stores the Altitude Pan Rate to the controller
XXD# Stores the Azimuth Pan Rate to the controller
XXE# Stores the Platform tracking rate to the controller
XXF# Stores the Platform up/down adjuster to the controller
XXG# Stores the Platform Goal to the controller
XXH# Stores the Altitude Guide Rate to the controller
XXI# Stores the Azimuth Guide Rate to the controller
XXJ# Stores the PicServo Timeout Value to the controller (Seconds).
XXQ# Turns off or on the digital outputs of the Radio Handpad Reciever
XXN# A zero turns off the Argo Navis™ mode. A one turns it on.
When the controller receives a 'Q' as the first character in a line, it responds emulating the Tangent "Q" protocol. The encoder resolution is always 18000 (the controller uses the configured values for encoder ticks per revolution for the Scope Encoders, and converts it to 18000).
If you are using servo firmware version 1.6 or later, the following commands have been added:
XXK# Local Search Radius in arc Minutes
XXM# Local Search Speed in arc seconds per second.
XXO# Altitude Backlash
XXP# Azimuth Backlash

These are not acted upon until version firmware 1.7.

New commands in version 1.7:
XXW# Backlash Motor Speed (Speed while in the backlash area) Should be 200,000 to 500,000.

XXX# Altitude in deg's times 100. The altitude should be sent to the controller by the host computer (ScopeII, the Ascom driver, etc) after initialization. Then the Local searches will be round, and the Pan and Guide speeds will be linear in relation to the sky (version 1.7 or later).

XXY# Sets the Azimuth/RA in the controller. Not useful until version 1.8.

0AAh Puts the servo into the PicServo Emulation mode (no CR required). See http://www.jrkerr.com/psdata.pdf for a description. There are some items different, all motion control is the same. The differences are descibed below.

The following two ASCII commands with Binary data are for reading and controlling the servo controller in a very efficient manner. They have been added at version 1.5 of the controller firmware.
XXS<E>
The controller responds with four long ints of position, then other bytes for keypad status, X and Y Bits, and other bits as follows:
Byte 0-3 Current Altitude/Dec Motor Position (least significant byte first)
Byte 4-7 Current Azimuth/RA Motor Position.
Byte 8-11 Current Altitude/Dec Scope Encoder Position
Byte 12-15 Current Azimuth/RA Scope Encoder Position.

Byte 16 is Keypad Status (see the top of this section for a description of bits)
Byte 17 is XBits
Byte 18 is YBits
Byte 19 is Extra bits

Bit 0 - Set if Altitude/Dec motor is stopped
Bit 1 – Set if Altitude/Dec motor is in manual
Bit 2 - future
Bit 3 – future
Bit 4 - Set if Azimuth/RA motor is stopped
Bit 5 – Set if Azimuth/RA motor is in manual
Bit 6 - future
Bit 7 – future

Byte 20 is a 1 byte checksum of the previous 19 bytes (simple addition).

XXR<E>
The controller now waits for an additional 20 bytes from a host as follows:
Byte 0-3 New Altitude/Dec Motor Position Goal (least significant byte first)
Byte 4-7 New Azimuth/RA Motor Position Goal.
Byte 8-11 New Altitude/Dec Motor Speed
Byte 12-15 New Azimuth/RA Motor Speed
Byte 16 Various bits

Bit 0 – if 0, the following new XBit and YBit values are ignored. If '1', the new XBit and YBit values are used.

Bit 1 -7 future

Byte 17 New XBits value (must have previous bit 0 set)
Byte 18 New YBits value (must have previous bit 0 set)
Byte 19 Checksum of all above data bits (simple 8 bit addition)

If you use the above binary command to control the servo controller, be sure to have a restart communication if the command gets out of sync by a communication fault. If you receive a bad checksum, you should resync communication by sending a few of XXS commands.

The flash ram values are loaded from the flash Rom to Ram on reset.
Both the X and Y parameters are stored to the flash ROM.

Use X and Y for module address 1.
If the module address is 3, use 'T' and 'U', for address 5, it's 'V' and 'W'.

Description of the bits for the XBits and YBits:
0 if 1, the motor encoder is incremented the other direction
1 if 1, the motor polarity is reversed
2 if 1, the azimuth (or altitude) encoder is reversed
3 if 1, (x only) we're in the computerless drag and track mode
3 if 1, (y only) we're in the computerless slew and track mode (no clutches, must use handpad to

slew)
(must be in drag and track too)
4 if 1, (x only) we're in the tracking platform mode
5 if 1, (x only) we enable the hand paddle
6 if 1, (x only) hand paddle is compatible with New Hand Paddle (allows slewing in two directions, and guiding)
7 if 1, (x only) we're in the guide mode. The pan rate is added or subtracted from
the current tracking rate.

Note about the Guide Mode: If PC software puts the controller in the guide mode, and you are controlling an Equatorial mount, you must give the declination motor a destination, even if you want it to be stopped. If you want the declination axis to be normally stopped, give the declination a far far away goal, and set the speed to zero. This is because the controller adds or subtracts from the current speed and direction, and the controller gets confused if the axis is at its destination. This is changed inf servo version 1.6, and this is automatically done for you in the controller.
The Y_Bits bits 4-7 are the digital input status for the 4 digital inputs from the radio handpad reciever. This is only available if you're using servo version 1.6 and also if you have a handpad reciever connected.

PicServo Mode Differences:
If you intend to write a telescope driver for the SiTech controller, we recommend you use the new binary protocol mentioned above, as it is simpler and faster. If you have exsisting software for the PicServo chips, you can use the PicServo protocol, but there are a few differences as described here:

The controller address must be set beforehand, and saved to flash ROM. The PicServo address commands are ignored with the SiTech controller. When the controller is set for address 1, the altitude motor will be addressed with address 1, and the azimuth with address 2. You can set address using the ASCII command AD1, AD3, or AD5. You then save the address to flash ROM using the following ASCII commands. “XW” for address 1, “TW” for address2, and “VW” for address 3. This must be done with only 1 controller connected at a time. You can also set the address of the controller using the ServoConfig software.

When the Define Status specifies the Home position for return, the “HOME” position returnes the location of the telescope encoder location, instead of the HOME position.

When the Define Status specifies the A/D to be returned, it returns the motor current times 10 instead. Bit 6 of the Define Status must be clear.

When you set bit 6 of the Define Status, the motor voltage will be returned in the A/D location if it is an odd address for the controller.

When you set bit 6 of the Define Status, the CPU temperature will be returned in the A/D location if it is an even address for the controller.

The ReadVersion bit of the Define Status returns the handpad status instead of the version. The handpad bits are as follows:

The handpad number is an 8 bit number where each bit represents a push button.
Bit 0 Left
Bit 1 Right
Bit 2 Up
Bit 3 Down
Bit 4 Speed (this is a toggle, a 1 is Pan mode and a 0 is slew mode).
Bit 5 Top Right key
Bit 6 Top Left key
Bit 7 Changed key.
The Changed key (bit 7) is used to tell if someone has pressed a keypad button since the last speed command is received. This feature is useful so a new tracking coordinate can be obtained, and the new tracking setpoint will track from the new position. This bit is automatically cleared if a new speed command is received.

Several more functions have been added to the PicServo protocol. They are accessed using the PicServo NOP function with various lengths specified.

Length of 0 (send a 0Eh. This is the same as the PicServo NOP function)

Length of 1 (send a 1Eh for the PicServo function, then another control byte).
If Control byte = 4, the controller reads the configuration from flash ROM
If Control byte = 8, the controller writes the current configuration to flash ROM
If Control byte = 16 the controller writes the factory defaults to the flash ROM
If Control byte = 32 the controller reverts to the ASCII mode.

Length of 2 (send a 2Eh for the PicServo function, the control byte, then another data byte.
If Control byte = 2 the controller will update the Xbits (or Ybits if even address) with the data byte.

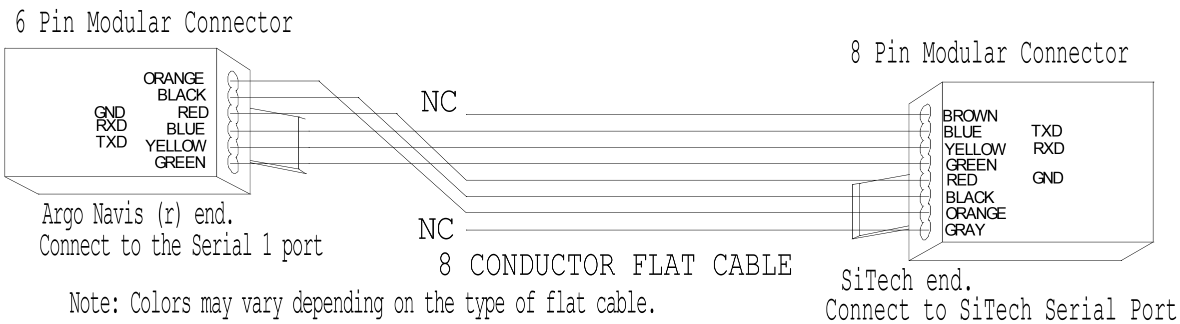
Length of 5 (send a 5Eh for the PicServo function, a control byte, then four data bytes.
If Control byte = 1 the controller will update the Altitude (Azimuth if even address) location with the next 4 bytes.
If Control byte = 64 the controller will update the Altitude (Azimuth if even address) guide rate with the next 4 bytes.

Length of 9 (send a 9Eh for the PicServo function, a control byte, then eight data bytes.
If Control byte = 32 the controller will load the next two long ints (8 data bytes) into the Slew rate and the Pan rate. For odd address it is the altitude, for even addresses it is the azimuth.

If the controller hasn't received a PicServo command for more than the PicServo timeout period, the controller will stop any motion, and revert to the ASCII mode. The PicServo timout is set with the XXJ command, and the units are Seconds. Set to zero to disable PicServo timout.

APPENDIX D: CABLE DIAGRAMS:

Diagram of communication cable for the Argo Navis™ This is for the early controller version:



Later Controller Version (4 x 4 connector at SiTech end):

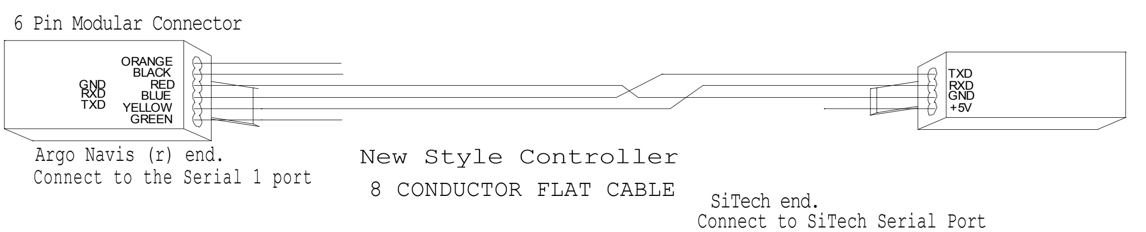
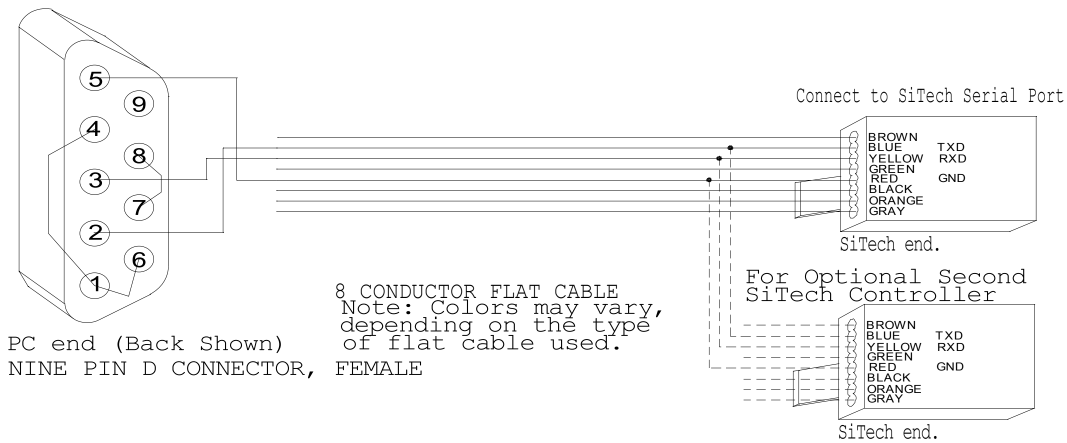
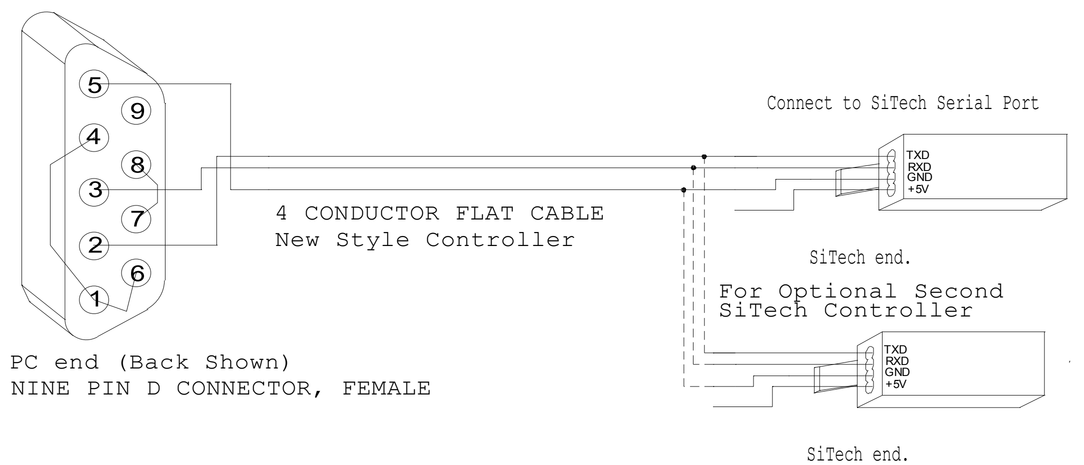


Diagram of communication cable for connecting to a Personal Computer:
Early Controller Version (8 x 8 connector at SiTech end):

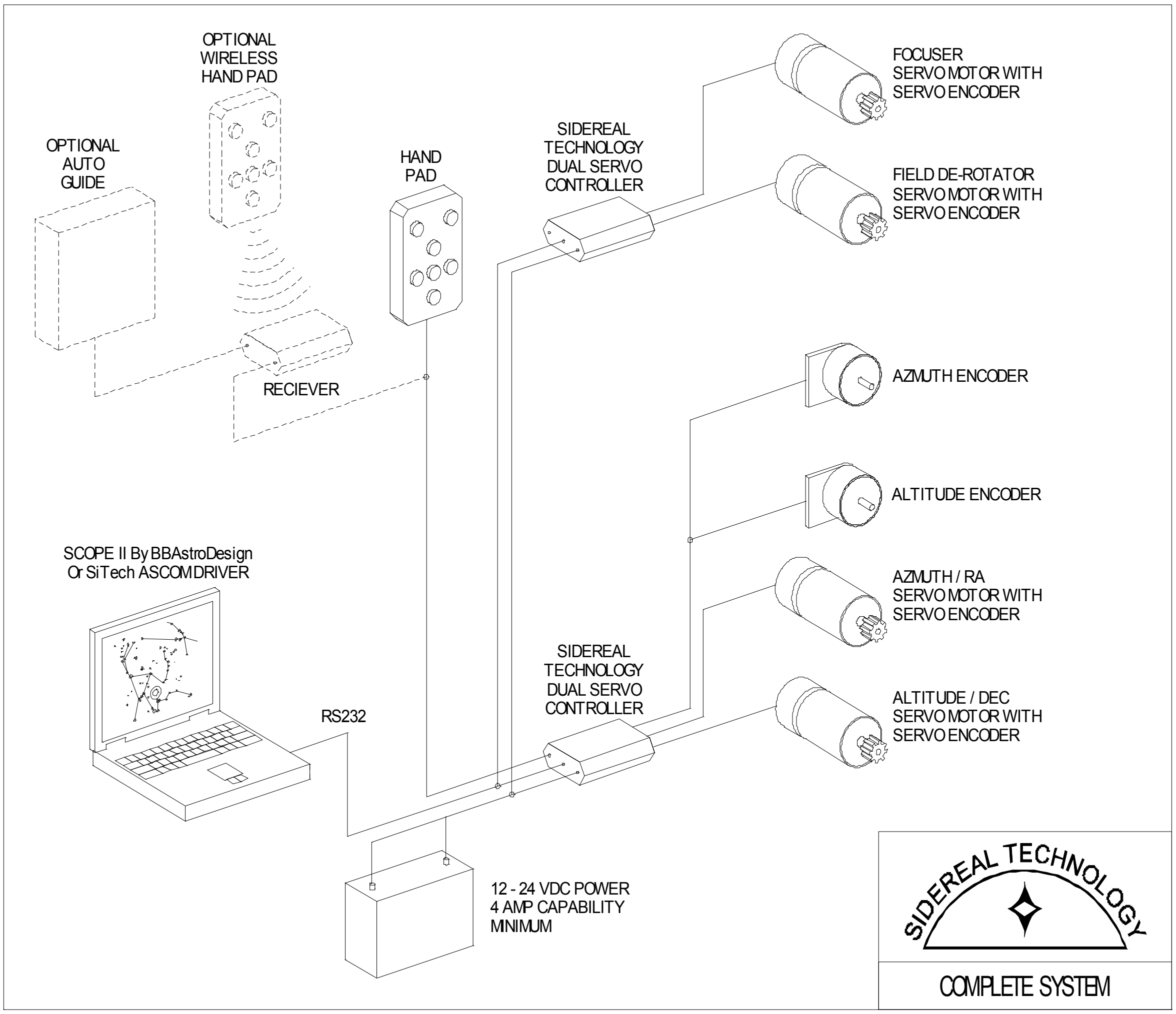


Later Controller Version (4 x 4 connector at SiTech end):

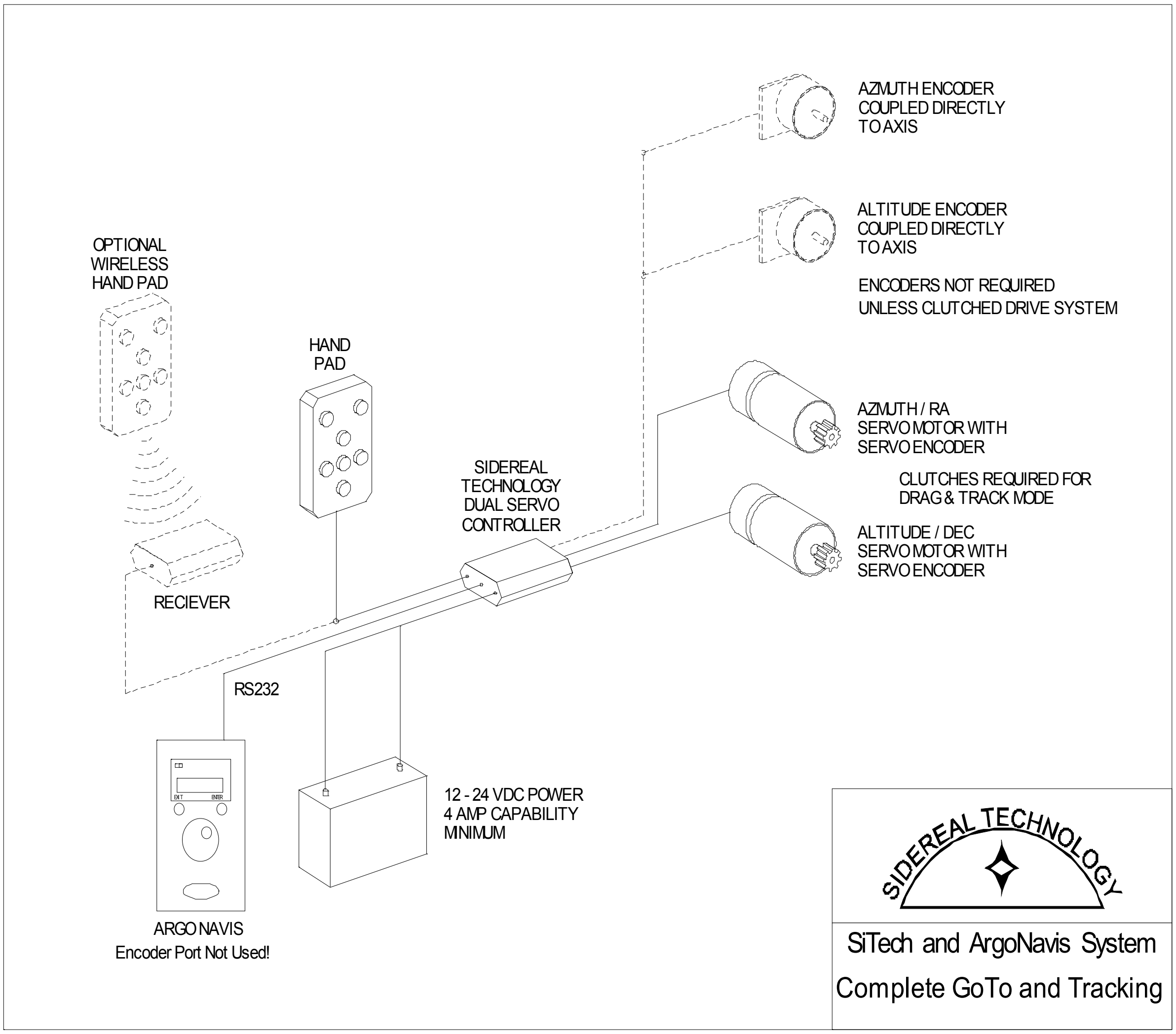


APPENDIX E: BLOCK DIAGRAMS OF SOME POSSIBLE CONFIGURATIONS:

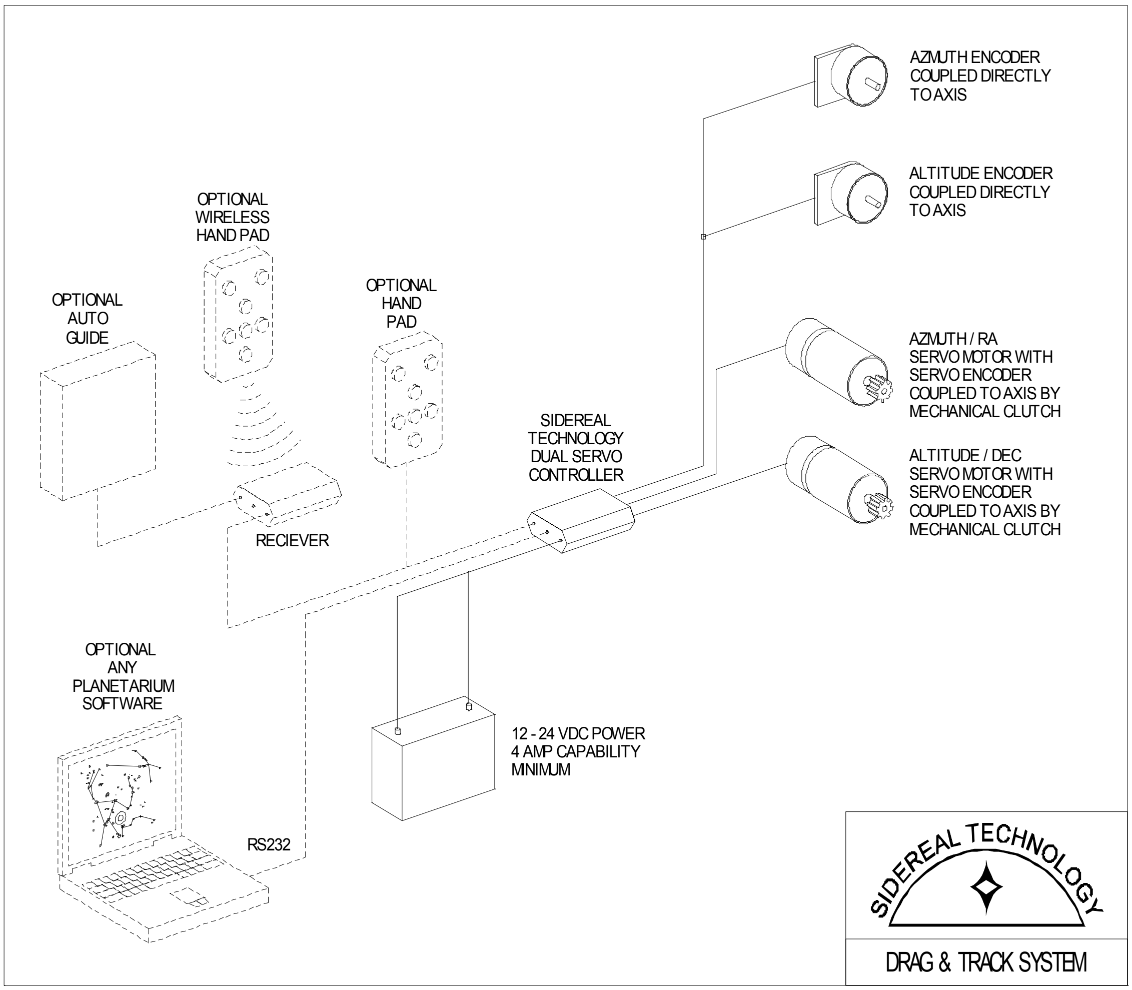
Complete System:



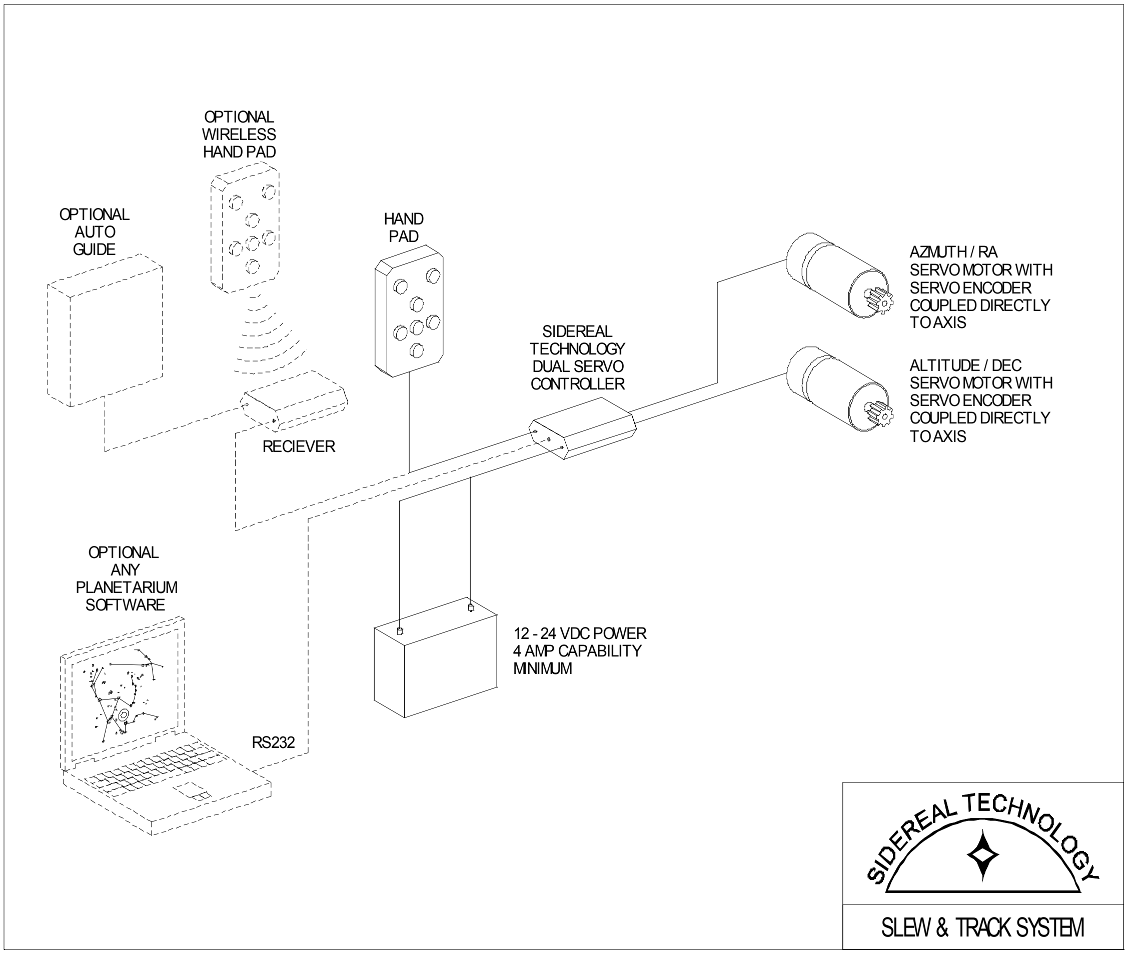
System using Argo Navis™ for full GoTo and Tracking:



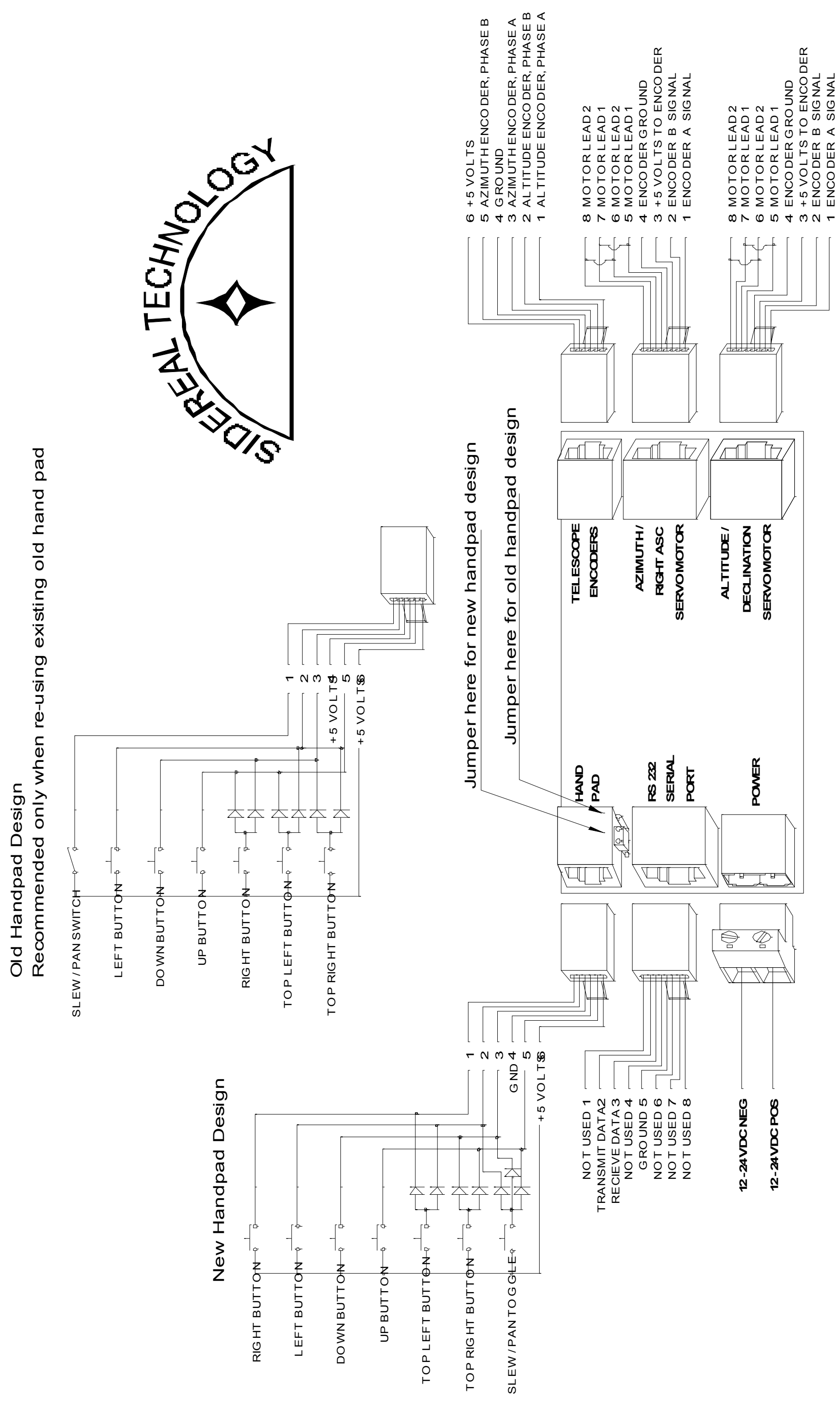
DragNTrack System:



SlewNTrack System:



APPENDIX F: WIRING DIAGRAM FOR EARLIER CONTROLLER



APPENDIX G, WIRING DIAGRAM FOR LATER CONTROLLER