DRAFT - Ideal Field System Antenna Interface - DRAFT

Ed Himwich, NVI/GSFC

April 18, 2001

1.0 Introduction

Normally when the Field System is interfaced to an antenna, there is some defined interface for the antenna and the Field System is forced to adapt to it. This is because either the antenna has very limited interface capabilities or the development of the antenna's side of the interface has been completed and no further revisions are possible. However, occasionally we have an opportunity to specify the interface to the antenna. This document describes the interface that the Field System would prefer to use. The details of the interface are very flexible. The description here is intended only to outline the general capabilities that the Field System requires for normal operations. It is not intended to provide detailed plan for what the interface should do. That should be developed in personal discussion with the antenna manufacturer.

Separate sections describe the Physical Connection, Line Protocol, and Transactions. The two most important transaction types, trajectory designate (TD) and position status (PS) are described in some detail in separate sections. The way in which the Field System normally uses the antenna interface is described in the Usage section. The Focus Control section describes a basic approach to focus control. The Alternative Control Method section describes an alternative method for controlling the antenna. Finally, the NORAD Orbital Elements section touches briefly on this subject.

As a point of terminology, the antenna is assumed to be controlled by a computer that is referred to as the antenna *controller*. The interface is assumed to be between this computer and the Field System computer. In the Field System computer, the interface is implemented by the *antcn* program. The antenna interface is a highly station dependent feature. By confining the idiosyncrasies of the interface in separate program, the rest of the Field System is kept station independent. For the remainder of this document, the program name *antcn* will be used to refer to the Field System's side of the interface. This term may actually refer to a set of programs used to implement the interface.

2.0 Physical Connections

It is convenient if the physical connection is an RS-232 serial line. Only four signals should be needed: Frame Ground, Transmit Data, Receive Data, and Signal Ground (on a typical DB-25 connector these would be pins 1, 2, 3, and 7). There should be no hardware handshaking.

It should at least in principle be possible to support GPIB and Ethernet socket connections as well.

3.0 Line Protocol

Any standard line protocol can be used. A good choice is 9600 or higher BAUD, 7 data bits, even parity, 1 stop-bit. BAUD rates of 19200, 38400, and 57600 are usually easy to support In addition 115200 BAUD may be available as well. We try to use the highest possible BAUD rate.

No software handshaking, e.g., X-on/X-off, is used. There should be no echoing of data as it is received. One possible format for exchanging data is to send messages with three parts, in order: (1) the message consisting of printable ASCII characters including a trailing space; (2) a check-sum which is formed by summing all the byte values of the ASCII characters is part (1) above and representing the low order eight bits as a two digit hex value in ASCII; and (3) a terminating character, which might be the (non-printable) ASCII line-feed. Obviously other schemes are possible. Any useful check-sum would be acceptable as long it is documented. A CRC algorithm might be preferable.

A special character called the "synchronization character" is used for initialing the line. The character can be just some printable character that is otherwise not used in normal messages, perhaps '#'. Synchronization is discussed further below. For now the important point is that the controller should always be listening for the synchronization character. Whenever that character arrives, the controller should stop any output it may be generating, flush its input buffer, and wait for the next input character.

4.0 Transactions

All communications between *antcn* and the antenna are organized around the concept of transactions. *antcn* initiates each transaction by sending the controller a *command*. The antenna completes each transaction by returning a *response*. In no case does the controller initiate a tranasction. All of the controller responses are sent in response to an *antcn* command. All commands and responses are in format described above under **Line Protocol**. The response must arrive within a fairly short interval, know as the *time-out,* after the command has been transmitted. The time-out value should reflect the worst case response time of the antenna controller including the actual time required to transmit the characters of the response. This value might be as large as a few tenths of a second. Except for very slow BAUD rates, it should not need to exceed 0.5 seconds in any case and should normally be as small as is practical. If possible, the best scenario would require the controller to meet a per character time-out value, which might be as short as 2-3 times the character transmission rate at the BAUD rate being used.

An important point is that controller should not wait until a requested action is complete to respond. Any time-consuming processing of the command should be deferred until the controller has responded, which in some sense is just an acknowledgment of having received the command. One implication of this that if the controller needs to return data as part of the response, the data (for example position information) must readily so that it only needs to formatted and passed back.

In order to synchronize the communication lines between *antcn* and the controller, the following scheme can be used. When *antcn* first initiates communication it should send the synchronization character and wait until the line from the controller is idle at least two character time-out times. After that *antcn* can start sending commands and receiving responses in normal fashion. In addition if the controller should time-out at any time, then *antcn* should re-initiate the synchronization procedure.

Some distinction between local and remote operation needs to made by the controller. When the controller is in “local” operation, it will not accept commands from the interface, or if accepts them it will not act on them. Command that merely request information should still work, but no commands that attempt to command the antenna should be acted on. The term “local” operation usually indicates that the antenna is being controller by command being input directly into the contreoller perhaps by pushbuttons or a keyboard. In ‘remote” mode the antenna will act on commands from the interface. When in remote the antenna may or may not still accept “local” commands, different implementers may make different decisions.

4.1 Commands

There are two primary commands: trajectory designate (TD) for commanding the antenna and position status (PS) for determining where the antenna is. In addition there might be some special commands for functions such as stowing the antenna, inserting or removing stow pins, moving the antenna to a predetermined service position, setting or releasing the brakes, or any other function that might be useful for a particular antenna. The TD and PS commands are discussed in more detail below.

4.1.1 Trajectory Designate Command

The trajectory designate (TD) command is the primary way that *antcn* commands the antenna to move. A TD command should consist of: (1) a string to identify the command as a TD, perhaps "TD", (2) the command coordinates for the two antenna axes, (3) command rates for the two antenna axes, and (4) the epoch for which the positions and rates refer to. Fields are normally separated by one or more blanks. If a field is omitted it should be replaced with a double comma. Other schemes for delimiting fields are possible.

The epoch, position, and rate in the TD command define a trajectory along which the antenna controller should attempt to move the antenna. The position defines where the antenna should be at the epoch and the rate defines how to extrapolate the position to future or past times. Defining the trajectory in this way frees *antcn* and the antenna controller from having to worry about time-delays in the communications or other processing.

The units of the command coordinates can be any convenient choice. Degrees are recommended, but any other convenient units, e.g., radians or encoder counts, are acceptable. The command position should be specified with at least as much precision as the encoders will support. It should be possible to specify EVERY encoder position in whatever units are used. If the antenna has more than one "wrap" in the azimuth coordinate, a field should be included to indicate which wrap is being commanded. In addition to commanding a particular wrap it should be possible to request the wrap that would result in the shortest motion. If the antenna can plunge in elevation, there should be a way to specify whether the position is "plunged" or not. In this case, commanding an elevation greater than 90E is a convenient way to specify a plunged position. In any event it should be possible to uniquely identify every possible encoder position the antenna can reach including any overlapping regions.

The units of the command rates can be anything convenient; degrees per second are recommended. The command rate should be specified with more precision than the command position. There should be enough additional precision so that the position can be extrapolated for several seconds (or well beyond the anticipated TD command updated interval in any event) by the antenna controller without becoming less precise than the encoders. For example, specifying the rate in degrees per second with one more digit than the position is specified in degrees will allow the rate to be extrapolated without error for about 10 seconds. However, given the way ASCII representation of numbers are decoded in modern software, it should be trivial for the controller to accept any precision provided by *antcn*, up to say 16 digits, and to carry that precision in its internal calculations.

The epoch can be specified in seconds of time. The precision of the time should be sufficient to allow an object moving at sidereal rate to have its position specified to an encoder unit. Typically a one-hundredth of a second is sufficient. Some care must be taken in choosing the origin for the epoch. The simplest approach is to measure time from an arbitrary epoch in the past, although in this case the time value will become rather large after a few years. An alternative is to measure the time from the last UTC midnight. This case is much easier to deal with from a human engineering point-of-view but it also introduces a problem because the time does not change monotonically near midnight. For example, suppose *antcn* forms a TD command just before midnight. However because of processing delays, the antenna controller does not receive the command until after midnight. Does the controller interpret the time it receives as referring (correctly) to the previous day or (incorrectly) to the more natural current day? To remove this ambiguity, a window, of say ±100 seconds can be established around midnight. During the first 100 seconds of a day, if the controller receives a command for an epoch in the last 100 seconds of the day, it should be interpreted as referring to the previous day. Conversely, if in the last 100 seconds of a day, the controller receives a command for an epoch in the first 100 seconds of the day, then it is assumed to refer to the next day. This approach of using seconds of the day and a ±100 second window around midnight is the preferred way to implement the epoch.

If the controller does not know the UTC time, a different approach for specifying the epoch is required. The use of this approach is signaled to the antenna controller by the fact that the epoch is not included in the TD command. In this case the receipt of the end-of-message character (ASCII line-feed) defines the epoch for the command. The implied definition of this time can usually be determined by *antcn* to within two or three 100-ths of second. Even if the controller normally has UTC available, it is useful to implement this method as a back-up in case there is some problem with the time.

The response to the TD command might be either "TD ACK" or "TD NAK". The former response is used if the TD command had no transmission errors, i.e., there were no parity errors, the check-sum was correct, and all the appropriate fields contained valid quantities. Once the command has been accepted it should be treated as the active command immediately, any previous commands should be ignored. The "TD NAK" response is used if an error is detected. In this case, the controller should discard the command and continue using the most recently received correct command. It is useful if the controller can return some indication, perhaps a small ASCII integer to identify why the command was rejected.

In processing the TD command, the antenna controller should protect the antenna from any damage and prevent it from running into any hardware limits. Generally speaking *antcn* will never command the antenna to move into a limit, but problems may occur. There may be an undetected transmission error despite the use of parity and check-sums. Another possible problem might be that the Field System computer could loose communication while the antenna is tracking close to a limit. The extrapolated position may cross into the limit. In any event, the antenna controller is presumably closing the servo-loop for the antenna and is in a much better position to determine based on position and velocity whether the antenna must be stopped or slowed down to avoid running into a limit. It seems that the most sensible approach is for the antenna controller to form the main and final line of protection for the antenna.

If the commanded trajectory of the antenna is outside the limits, this should not cause a NAK response from the controller. That response should be used only for communication errors or incorrectly formed TD commands. Instead, the controller should keep the antenna as close to the commanded trajectory as possible, without going into the limits. The primary reason for doing this is so that a trajectory can be acquired as quickly as possible as it moves out of the limits.

4.1.2 Position Status Command

The position status (PS) command is used by *antcn* to determine where the antenna is. The command would normally contain some string identifying the command as a PS command, perhaps "PS". The antenna controller should respond with: (1) a string to indicate that this is a PS response, perhaps "PS", (2) the antenna’s command position, (3) the current position, and (4) the epoch for which this information applies. In addition if it is available, a flag might be returned to indicate whether the servo-loop is in a high-precision "tracking" mode or a low precision "slewing" mode. Of course any other information that needs to be monitored on a frequent basis could be returned in the PS response as well. It is recommended that the PS command also return a indication of whether the controller is local or remote mode. General status and other information should be retrieved using other commands.

When the end-of-message character for the command is received, the controller should "latch" the information (including the time) to be sent in the response. When the data has been latched the first character of the response should be sent immediately. The remainder of the message could be sent more leisurely as long as it doesn’t exceed either the per character or per transaction time-out (whichever is in use). This latching and quick response with the first character is intended as an aid to *antcn* for determining whether the controller has the correct UTC time. If the controller is not locked to UTC then it serves to define the epoch of the returned data as precisely as possible. It useful to include in the controller documentation information about how old the data might be relative to when it was latched. For example if the data is collected from an internally executing loop in the controller, the period of the loop might define an upper bound on the age of data, although it might be possible to be more precise.

The returned command and actual positions should be transmitted with a precision at least as great as that of the encoders. The returned command position is used by *antcn* to verify that the correct trajectory was received by the antenna controller and was extrapolated to the current epoch correctly. The command and actual positions may each require additional fields to identify unambiguously which azimuth wrap and/or plunge the coordinates refer to. To support medium or high dynamic applications it might be useful to return the command and actual rates as well.

The returned epoch, command positions (and rates if sent), and actual should be consistent, i.e., the epoch should represent the time that the command positions (and rates) was calculated for and the actual were measured at.

5.0 Usage

This section describes the ways that *antcn* would normally control the antenna. The intent is to provide an explanation of how the TD, PS, and other commands would be used. It should also provide some insight into what sort of situations the controller will have to deal with.

There will of course be significant periods of time when there is no communication between *antcn* and the controller. This will usually be because the Field System is not running. When the Field System is started, normally it will enter one of two modes. In one of these modes, *antcn* will monitor the position of the antenna, but not command it. In this mode, *antcn* will send PS commands about once per second. The other mode occurs when the antenna is being actively controlled. In this mode a PS command and a TD command will each be sent about once per second. While the antenna is being controlled, it will usually be supporting one of two activities, normal VLBI operation or pointing calibration runs. Of course either of these activities may be interrupted by periods of position monitoring and/or no communication.

During normal geodetic VLBI operations, a radio source is observed for a few minutes at a time. While the antenna is tracking a source, the successive TD commands will normally define a continuous trajectory with minor rate changes as the source’s motion deviates from linear in azimuth and elevation. As long as the TD commands specify only these minor adjustments, the antenna should track smoothly as it receives and accepts each new command. When an observation completes, a new source will be commanded. At this point, *antcn* will immediately begin sending TD commands for the trajectory of the new source. It is normally left to the antenna controller to handle accelerating the antenna up to slewing speed, slewing, and then decelerating the antenna to join the trajectory defined by the TD commands for the new source.

During pointing calibration measurements, *antcn* controls the antenna in a manner similar to normal VLBI operations. However, while tracking a particular source, the trajectory for the TD commands will be modified several times to move the antenna beam in small steps relative to the nominal trajectory of the source. The Field System measures the detected power from the source at each relative position. Using these measurements the differential between the expected and actual trajectory of the source in antenna coordinates is determined. The steps are typically a fraction of a beamwidth, but sometimes as large as a few beamwidths. The beam size can be calculated from the observing frequency, which for geodesy is usually 2 or 8 GHz. It is desirable for the antenna to make these small steps and settle quickly on the new trajectory. Some balance must be struck in the servo system between the need to track smoothly when the trajectory is not changing and the need to make these small steps quickly.

Occasionally for tests and special applications, it is necessary to point the antenna in a fixed direction. This is no different than tracking a source that moves with a rate of zero.

When the antenna is tracking a source, it should stay within a tenth of a beamwidth, or less, of the commanded trajectory in an root-mean-square (RMS) sense. This is the nominal specification for geodetic VLBI. The combined effects of both the non-repeatability of the antenna's pointing and the errors in the servo system should meet this specification. It is straight-forward for *antcn* to compensate for any repeatable point errors, such as those caused by axis misalignments, biases, and gravitational sag. Virtually any repeatable pointing error can be compensated for.

Occasionally, other commands may be intermingled with the TD and PS commands. These might include various commands to change the state of the controller or commands for more detailed status reports. It is convenient, but not necessary, if all the commands that are available at the controller's console are also available across the interface. This might include commands to: move the antenna to stow position, insert stow pins, remove stow pins (or higher level "stow" and "unstow" commands), point to service position, set special switches, etc.

For some antennas, it might be necessary to compensate for effects that are measured with additional sensors, e.g., tilt meters. The values measured by these sensors can be returned in the PS response or in a separate command response. A separate response may be more appropriate unless the measured effect changes fairly quickly and requires the command angles to be adjusted on a second by second basis. If a separate response is adequate, the value might be queried by *antcn* once when a new source is acquired and then periodically as necessary after that. It may be appropriate to compensate for the tilt meters or other instrumentation in the antenna controller and thereby free *antcn* from worrying about this detail. This is particularly useful if the correction changes rapidly in which case it is best handled internally by the controller.

6.0 Focus Control

Its is desirable for *antcn* to able to control and monitor the focus of the antenna through the interface. This is somewhat simpler than controlling the antenna pointing. It is normally not necessary to specify an epoch or a rate. The focus and/or subreflector control may have more than one axis for which values will need to be commanded. The units that the focus position are measured in can be arbitrary as long as they proportional to the position. Millimeters are a convenient unit.

The focus position status response should return the current position for all the controlled axes. It is also useful if a flag can be returned indicating whether focus motion has stopped, indicating that the focus has reached its command location.

The focus control might be used in one of two ways. In the first, the antenna will be commanded to track a strong source. The focus will then be adjusted in small steps to find the peak response. In the second approach a strong source would also be tracked. In this case though, the focus would be commanded to move continuously from one limit of its motion to the other, and then back again. The received power will be measured as function of read-out position. The response curves for two directions of motion will be compared to estimate the position of the peak response. In order to use the later approach, the focus must move at a fairly uniform rate and slowly enough so that it would spend several seconds traveling between the half power points

The focal position that gives the best response may be measured for several antenna orientations. The resulting focus curve will be used to determine the best average position. Thereafter, except for other focus checks, the focus is not usually adjusted for geodetic VLBI. Other applications including astronomical VLBI may require adjustment of the focus for different antenna orientations to maintain the best response.

7.0 Alternative Control Method

In this section a different style of antenna control than was described in the Transaction and Usage sections is discussed. The approach used in those sections requires *antcn* to send position and rate commands at least every few seconds. Although this requires a high degree of control by *antcn*, it simplifies the antenna interface in some ways. In particular, the algorithms for the reduction of coordinates are under *antcn*'s control. However, in some cases this may not be an appropriate approach. For very large antennas observing at high frequencies, the precision of the tracking may not be sufficient using this approach. Also some controllers may not be able to support the frequent communications that are required. For these, and perhaps other, cases an alternative approach can be used where the “topocentric celestial coordinates of date” for the source are transmitted to the controller. In this case the controller would calculate the command angles for the antenna and track the source itself. In cases requiring extremely high precision it may be necessary to use “mean coordinates” at some standard epoch. However this should be an option in addition to supply coordinates of date.

Generally it is useful to be able command arbitrary offsets in both the antenna's natural coordinate system, typically AZ-EL, and in Hour Angle and Right Ascension. In addition it should be possible to command pointing corrections, although these can also be commanded as offsets in the antenna's natural coordinate system.

In this alternative control method, pointing corrections and offsets can be handled in one of several ways depending on the level of precision required. One alternative is for *antcn* to modify the source's coordinates before sending them to account for the effect of the corrections and offsets. This approach is usually adequate as long as the required precision is not very high. A second approach would be to calculate the corrections in *antcn* and use commands to send the corrections and offsets to the controller. In this case the controller would apply the offsets and corrections as part of its reduction. The third and highest precision approach would be to have the controller calculate the corrections itself using its own model or from a model sent to it by *antcn*. In this case it is still useful to have an independent set of offsets that *antcn* can control for pointing measurements.

In order for this alternative approach to work, obviously it is necessary for the controller to know the UTC time. If the controller doesn't have time, it may be possible to transmit time to it with a precision of a few 10s of milliseconds if that is sufficient. In addition it is necessary to know how the controller will reduce the coordinates. The algorithms used for the reduction should be included in the controller documentation. This should include information on how any corrections are calculated.

The usage of the interface in this approach is not much different than is described in the Usage section. The only major difference is that instead of a command being sent every second, a new command is sent only when either a new source is selected or new offsets are applied. The position of the antenna will still be read back with a PS type command. This might be done every second or less frequently if the interface won't support such a high rate.

If fixed pointing is implemented in this approach it should use a separate command. The fixed direction might either be specified in antenna coordinates or as the celestial coordinates for a hypothetical source at some time.

8.0 NORAD Orbital Elements

No detail is provided in this document on using NORAD (North American Aerospace Defense Command) orbital elements for controlling the antenna. This is not a required feature for normal VLBI usage. However, the ability to use elements would be useful for some exotic VLBI applications as well as for some joint usage of the antenna with satellite tracking applications. It is useful if in addition to the orbital elements, an along track time bias and cross track bias can be specified in a separate command from the one that specifies the elements.