FodTrack

La Parfaite définition d'un Projet Radio Amateur! Peu Coûteux, Simple et ça marche!

Les systèmes automatisés de controle d`antenne satellite sont tres dispendieux souvent entre 600 a 1000$ US et ca n`inclus pas les rotors! De plus, à ceci il faut ajouter les Logiciels qui peuvent être aussi TRES DISPENDIEUX!

Manfred (XQ2FOD) a créé il y a quelques années un système très simple pouvant être adapté à presque n`importe quel rotor. Il s`occupe de suivre le satellite avec vos antennes, d`ajuster le radio pour l`effet DOPLER et est même compatible avec un GPS! Et c`est un FreeWare (Gratuit!)

Encore mieux, l`interface rotor du FODTrack (environ 30$) est maintenant compatible avec le nouveau Station Program de AMSAT.

Pour Download FODTrack Clicker ICI

Mais vous m`avez assez entendu! Voici le manuel complet du FODTrack! Amusez-vous bien! (en Anglais pour l`instant. Si quelqu`un veut aider avec une traduction, contactez-moi ve2dx@amsat.org)

FODTRACK Satellite Tracking System

Version 2.4

By XQ2FOD

User Manual

Introduction:

FodTrack is a simple, straightforward program intended to control an azimuth elevation rotator like the Yaesu-Kenpro 5400/5600, and a transceiver, for any kind of satellite orbits. Best efforts have been made to provide state of the art tracking accuracy, within the limitations imposed by commonly available orbital data.

FodTrack runs in foreground, continuously displays what it is doing, uses no special system resources, and is so simple to use that you should not have any trouble trying to figure out what is going wrong when the silly satellites seem to go in another direction than your antenna...

The rotator can be connected to any parallel port using an interface built according to the schematic provided with the program. LPT2 is a good candidate, as it is often present and seldom used. It can also be controlled via any serial port, using the Yaesu GS-232 or the RC-2800 interface. Nonstandard ports are welcome, just you need to know the address. No IRQ is needed.

The radio can be connected to any serial port. Just the same as for the rotator port, only the address needs to be known to be able to use any port.

The radio can be combined with converters for any satellite band from 145 MHz to 24 GHz.

FodTrack supports an NMEA talker device like a GPS receiver, for automatic setting of the time and location. It can be connected to any serial port at any address. It does need an IRQ line. All possible IRQs are supported (2, 3, 4, 5, 7, 9, 10, 11, 12, 15).

FodTrack is best suited for those setups where a single PC runs a multitasking environment, like DESQview, with the satellite software in one window, a BBS in another, and your favorite game in the third. Just open a fourth window and put FodTrack there, it will be happy. It works with WISP under Windows too, but it has not been specially written as a Windows-based program.

You can also use an independent computer for FodTrack. But you cannot run it in background. If you want background operation, there are other programs available for you.

FodTrack implements antenna flipping, so the satellites will not run against your rotator's end stop. Rotator stop position can be south or north.

The FodTrack rotator interface cannot drive the rotator against its stop position. The limits are set in hardware, so even if your computer goes crazy, locks up, or whatever, your rotator will not be damaged by being driven against a mechanical end stop.

It can run under manual or automatic control. In automatic mode, a skeduler like WA2N's SatSked can provide the commands. Two modes for automatization are provided: In the preferred one, FodTrack runs continuously and is controlled via a command file; in the alternative one, FodTrack is called for a specific satellite, and aborts after one pass.

You can use FodTrack to control only your rotator, only your radio, or both. Control for Yaesu FT736, Kenwood TS790 and several Icom radios is implemented. IK3NWV tested it on a TS790, while IK0XBQ made the Icom test with an IC820.

Thanks to them for the beta testing. On the FT736 I tested it, and I'm using it permanently with that radio.

Probably it will run with other Kenwood radios, and perhaps also with other Yaesu radios, but this has not yet been tested.

Copyright:

FodTrack is free for noncommercial use. If you want to reward me somehow, write a piece of useful software and put it in the public domain!

Disclaimer:

FodTrack is provided without any guarantee that it will really do anything of all the nice stuff this document says. But please, if you find a bug, tell me, so I can fix it for the next version.

Setting it up:

This is VERY easy. Copy the FODTRACK files to a directory of your choice. You can also maintain the kepfile there, but if you prefer you can use a kepfile at some other place.

Edit the FODTRACK.CFG file to reflect conditions at your station. The file explains itself. If you don't understand the use of some parameter, leave it at its default value.

Edit the FODTRACK.FRC file according to the satellites and frequencies you want to use. Do not include those birds for which you don't want automatic transceiver control. It's good enough to put the nominal frequencies into this file. Later you can fine-tune the frequencies from inside the program.

If you want to track a satellite on several different frequencies, you can define several data blocks for that sat, differentiating them by a tilde character (~) and any designator you like, after the sat name. For example, you could define AO-16 with the normal mode-J frequencies, plus AO-16~S with the frequency of the S-band beacon. In both cases the program will use the keps for AO-16.

You can define a "pseudo-satellite" in this file, called PARK. This will send your Icom or Kenwood radios to the parking frequencies and modes specified there. Yaesu radios return to the frequency they were on before the start of the pass, regardless of any PARK data specified here, if you leave the SAT switch in OFF position.

If you will run the program from somewhere else, you need a path to the FodTrack directory.

The kepfile must be in 2-line format, with the satellite name appearing above each 2-line block. Title lines are no problem. They are not needed, but they do not disturb.

Running it:

Execute FODTRACK.EXE. The program will come up, read its configuration file, and then it will read the command file (FODTRACK.CMD). The default command file says "NONE", so the program will stay idling. That's a good time to look around the screen:

You will see a status display, which says what the program is doing. It can be idling, waiting for a satellite, tracking it, calculating AOS and LOS while guessing if it is convenient to flip the antenna over, calibrating the rotator, or accepting your incremental tuning input.

There is a nice clock, ticking away your valuable seconds; also there is a line telling you that the satellite selection is automatic, and several windows without any data in them. They will come to life when you select a satellite.

There's also a small reminder for the commands you have available. Please note that the commands in the box are only available while in manual mode. During AOS-LOS calculation and GPS reading the keyboard is dead.

Now let's play a bit: Type the letter m to get the program into manual mode, then to start tracking a satellite. The program will ask you which satellite you want. Enter its name exactly as it appears in the kepfile, otherwise FodTrack will be very unhappy with you. For example, type KO-23.

The program reads the kepfile, and tells you the age of KO-23's keps in the proper window. It will also tell you if you can get good tracking precision with those keps, or if you should get new ones.

Then it calculates AOS and LOS times (for the novice: AOS means Acquisition Of Signal and LOS is Loss Of Signal) If you haven't disabled flipping, FodTrack also looks into its crystal ball, to see if the bird will run against your rotator stop on the next pass. If so, it will tell you after a while that the antenna will be "flipped" over. If not, it will be on the "normal" side. If you chose to park your antennas in the configuration file, then the "Flipped" or "Normal" display will be overrun by the word "Parked".

The AOS, LOS and flipping calculation takes only a few seconds if you have a coprocessor, but it can take nearly a minute on slow non-coprocessor machines. For geosynchronous sats this calculation is not done, because there may be no AOS or LOS at all!

After flipping determination is complete, the program will start tracking the bird. Every second, if the computer is fast enough, the position is updated. If the AOS time is more than 2 minutes away, no data is sent to the rotator nor to the radio.

When the great moment arrives, two minutes before AOS, the program will start sending target position data to your rotator, indicating so in the rotator status display. The two-minute allowance assures that the rotator has enough time left to point at the satellite before it comes over the horizon. At this time, the program will also start controlling your radio, and showing the Doppler-corrected frequencies. If the FODTRACK.FRC file does not contain data for the selected satellite, then the program will not access the radio, and will display Doppler correction in PPM (parts per million). This is useful as a help for manual tuning of satellites.

You may notice that the last figure of the frequency does not change at the same time on your radio display and on the FodTrack indication. This happens because most radios truncate the frequency, while FodTrack rounds it off. So, 435175.478 KHz would be displayed as 435175.5 by FodTrack, while most radios would display it as 435175.4.

If the rotator is not flipped, azimuth and elevation on the rotator are the real ones. If the program had to flip the antennas, azimuth is 180 degrees shifted and elevation starts backwards from 180 degrees.

If you selected a stepsize of zero degrees in the configuration file, then the program will send the rotator position to the interface at a rate of up to once every second. In this case, the rotator will move in fine steps, their size being given by the dead-band in the interface and the stepsize of the D/A converter (256 steps). If you selected a bigger stepsize, FodTrack will freeze the rotator until the position error is half as big as your selected stepsize.

Then it will move the rotator a full step in the proper direction. If your antennas have fairly broad lobes, you can use this feature to reduce wear and noise. The recommended stepsize is about half of your antenna beamwidth.

When the satellite goes below the horizon, the rotator output is frozen, the program stops controlling the radio, and calculates flipping for the next pass, then starts waiting for the bird to come up again.

If you selected parking, the antennas will then be parked. Otherwise, they will stay in the position the satellite left them, until the next pass is about to start.

If you have an Icom or Kenwood radio, and you specified parking frequencies and modes in the FODTRACK.FRC file, then your radio will be parked too. Yaesu radios will return to whatever frequency they were on before the start of the pass, if you leave the SAT switch in OFF position.

At any time except during AOS-LOS calculation or GPS reading, you can stop tracking a bird using the s command, or quit using q. If you press a, the program will go back into automatic mode, reading the command file and doing whatever it tells.

Tuning:

While you are tracking a satellite in manual mode, you can enter the manual incremental tuning mode. FodTrack will use the bottom bar on the screen to display the theoretical frequencies at the satellite, the tuning mode selected, and a list of available commands.

You can use the cursor keys to increment and decrement the frequencies. The up/down keys change the frequencies in large steps, while the left/right keys do the fine tuning. The size of these steps is definable in the FODTRACK.CFG file.

There are four tuning modes: In RX mode, the receiving frequency is tuned. In TX, it's the transmission frequency (sounds obvious, doesn't it...?). In DIRECT mode both frequencies are tuned in step, which is useful for operating linear transponders that don't invert the passband. Finally, INVERSE mode tunes the RX and TX frequencies in opposing sense, for inverting transponder operation like that on AO-10 and FO-20.

When entering tuning mode, it defaults to RX tuning. The SAVE command saves the modified frequencies into the FODTRACK.FRC file. The EXIT command exits the tuning mode.

During manual incremental tuning the orbital calculation and everything else continues. The theoretical frequencies are changed as you touch the proper keys. Every second, the Doppler shift affecting these frequencies is calculated, and while the sat is above the horizon, the corrected frequencies are sent to the radio. So although the theoretical sat frequencies can be tuned quickly, the frequencies in the radio are updated only once a second. This is done because most radios don't support a much quicker update rate through their control ports.

When you are first setting up FodTrack, you will need to fine tune the frequencies, specially the RX frequencies of the PSK sats. You can tune them until the data is being properly decoded and the signal is in the center of the modem's working range, then SAVE them. You should not need to retouch this tuning again for considerable time. Do this adjustment while using reasonably fresh keps, the PC clock accurate to the second, and preferably near the start or the end of pass, which is when the Doppler shift variation speed is minimal.

This same kind of tuning can be done for the FSK sats, using the FM discriminator center meter, but tuning is far less critical on FSK.

The manual incremental tuning mode is also extremely useful on the analog satellites. You can tune the RX frequency to some clear spot, then tune your TX to the corresponding uplink frequency, then fine-tune it while transmitting and receiving your own echo. After doing this, you can save the frequencies, and from then on always use the INVERSE or DIRECT tuning modes.

Reagardless of Doppler shift and position inside the sat's passband, your uplink and downlink will now always agree! This allows for completely hands-free operation, even on fast-drifting sats like FO-20... provided the station you are talking too is also keeping his frequency on the sat constant!

Radio specifics:

The Kenwood TS790 is notorious for an unwelcome "feature": It will shortly mute the receiver everytime it is commanded to update the frequency! So every frequency update causes one or several packets to be lost. To reduce the impact of this, FodTrack can be configured to send a frequency update only when the frequency error is more than a specified value. But the only real solution for this problem is to modify the radio, eliminating this muting.

The Yaesu FT-736 also has its quirks: It is impossible to switch bands in sat mode via the CAT control, unless you have an optional band module (or a dummy module), and even in that case the procedure is a bit tricky. If you have at least one band module or dummy, you can tell FodTrack about this good new in the CFG file. FodTrack will then take care of the bandchanging.

If you don't have such a band module or dummy, then if the radio is in mode B, FodTrack cannot put it into mode J. A simple workaround is to manually set mode J into SAT VFO A, and mode B into SAT VFO B. If you want to track a mode B sat, you then just have to press the VFO B button before FodTrack switches the CAT on, and press the VFO A button for mode J passes.

Fortunately this is not too bad, as presently all the satellites that require full automatic tracking are mode J.

There are some special situations in which both TX and RX are on the same band. The MIR station is one example of this. The FT-736 cannot set TX and RX to the same band in SAT mode. FodTrack gets around this problem by setting the radio in normal (nonSAT) mode, tuning RX directly, and using the programmable offset to tune the TX frequency. No special setup is required, just program the correct frequencies in the FODTRACK.FRC file. Please be aware that in this operating configuration the FT-736 cannot use separate modes for TX and RX, but
this is usually not required anyway.

Using converters:

When using converters for the bands not covered by your radio, there are several aspects to be considered. The most common converters ADD a fixed frequency to your radio's output, in order to reach the operating frequency.

In this case, all that FodTrack must do is calculating the Doppler shift for the operating frequency, then subtract the converter's local oscillator frequency and send the result to the radio.

On the other hand, some converters use a higher oscillator frequency and SUBTRACT your radio's frequency to reach the destination. In this case, the tracking on the radio must be inverted, and also the MODE (if it is SSB) must be inverted! FodTrack takes care of these issues, so you must simply do the following to use converters:

In the .CFG file, for each band you will operate through a converter, you must specify the converter's local oscillator frequency. After this, you can set up the .FRC file with the real frequencies and modes. As simple as that! FodTrack will display the real frequencies during tracking and tuning, while sending the converted ones to the radio.

Easy, isn't it?

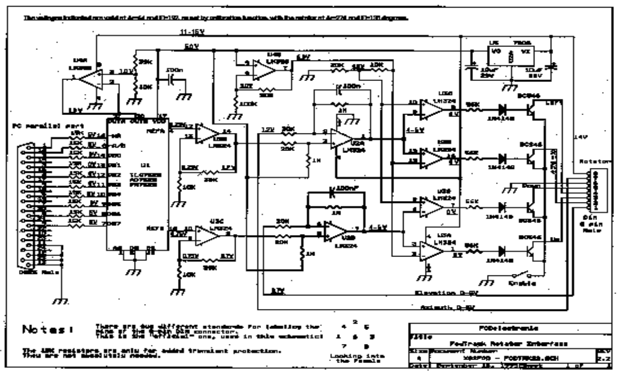
I hope this scheme is flexible enough to accomodate most (or hopefully all) needs. If you have some suggestion, don't hesitate to contact me.

The FodTrack Rotator Interface:

The program generates target position data as 8 bits for each azimuth and elevation, using one of the printer control lines (pin 14) to tell if azimuth or elevation is meant. 0 is az, 1 is el.

The schematic diagram of the interface is provided as a PCX file. This format was chosen because in this particular case it gave the smallest file size among all common formats, when compressed with PKZIP. The image is about 1700 x 1100 pixels; this seems to be the best compromise between file size and quality.

Please refer to the schematic in ROTORINT.PCX for this explanation:



A dual D/A converter (TLC7528 or similiar) is directly connected to the parallel port. The port's strobe output is wired to the D/A's chip select input, pin 14 to the output select pin, and the 8 data bits to inputs 0 to 7.

The D/A converter is used in a reversed fashion, with the reference applied to the outputs, and the output taken from the reference terminals. This allows using a low-cost current-mode DAC in voltage-mode.

The analog outputs are compared to the rotator position output using differential amplifiers, the output of this comparison beeing fed to two window comparators driving the motors. The value of the (presently) 1M resistors defines the rotator's dead range, so you can adjust it by modifying these values. Lower resistance values cause the dead range to be bigger, reducing wear and tear, but worsening pointing accuracy. If you have stability problems
(oscillations), lower these resistances.

A switch is provided which allows disabling the FodTrack interface. When this switch is ON, the interface will not allow you to move the rotator with the pushbuttons on the controller. It will always hold the rotator to the position stored in the interface. If you move the switch to the OFF position, you will have full manual control over your rotator, while the interface still listens to the PC and updates the D/A's internal registers.

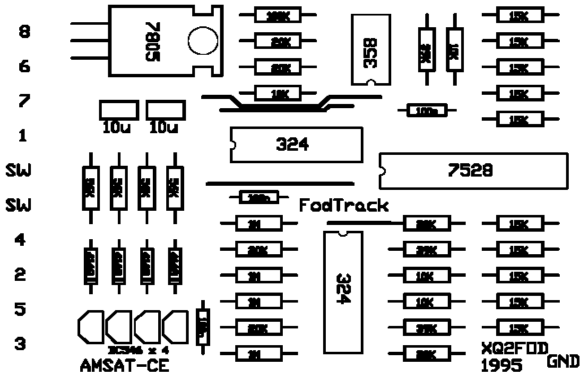
As you can see, the loop is closed outside the PC, in hardware. This reduces the processing load for FodTrack, and eliminates the possibility that a computer crash may drive your rotator against its end stop.

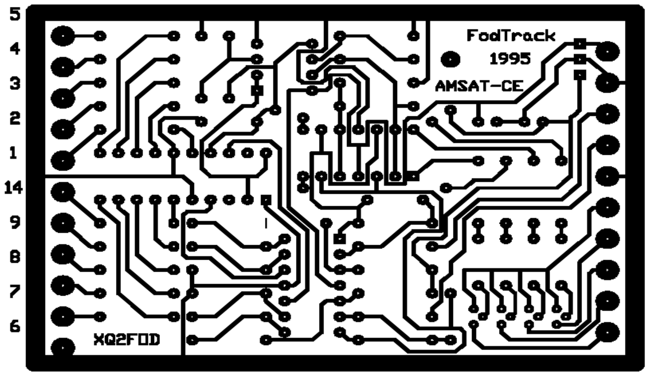
The program has no means to check if a rotator interface is connected, so you can start it up without one.

The interface can be mounted in a small box. It is connected to the rotator controller's 8-pin DIN connector, from which also the power is taken.

Please note that there are two standards in common use for the pinout of this connector. The schematic diagram includes a picture of the proper pinout.

On popular demand I'm now including the printed circuit board design for the FodTrack interface. This is exactly the same design used for the interfaces built and sold by AMSAT-CE. The design is provided in two PCX files: One for the copper side and one for the component overlay. Just print them out and make your board, or have it made by some specialized company. Be careful, because some traces are very close together. You need accurate etching to avoid shorts between those traces.





If you have any trouble getting it running, the voltages indicated on the schematic may help you find the problem. But be aware that most of these voltages are valid ONLY while the rotator is being properly controlled by the interface, at the positions noted on the top of the schematic.

The program provides a calibration function which allows you to adjust your rotator pots for correct ranging.

This sequence is valid for my interface circuits, and a Yaesu 5400 or 5600 rotator:

- Assemble the interface, set everything up, load the program.
- Calibrate the rotator's full scale adjustments according to its manual.
- Type c to get into the calibration routine.
- Enter 255 for both azimuth and elevation.
- Calibrate the rotator's output voltage adjustments to such positions that the rotator stays just before its end stops (or limit switches) in both axes.
- Enter 0 for both azimuth and elevation.
- Check that the rotator turns around completely in both axes, without running against the limits. If it does run against a limit, something is wrong...
- Play around entering values of your choice, and make sure the rotator reacts correctly (compare the meters to what FodTrack says).

************** NOTES*************

-This same circuit can run on any voltage from 9 to 15Vdc

- Rotor control boxes can be modified with small relays to enregise the proper switch during rotation.

****

Using the GS-232 or the RC-2800:

Instead of the FodTrack rotator interface, you may use the Yaesu GS-232 or the RC-2800. This allows to use FodTrack with ready made hardware for those who don't want to build things. But of course the cost is much higher!

To use one of these interfaces, you simply set up the configuration file accordingly (details are commented in that file), and connect the interface to the selected serial port. FodTrack uses only the Wxxx yyy command of the GS-

232 and the A and E commands for the RC-2800. It does not expect any answer, so you can use a very simple 2-line connection!

Radio Interface:

You can use the original RS232 interfaces, but these are costly. You can as well use a simple level converter mounted inside a plug. Several firms offer them at a much lower cost than the original ones, but it is still much cheaper to build your own...

You really don't need a fancy MAX232 here. A simple CMOS chip, powered from 5 Volts, will do the job. Put a 100K resistor between the RS232 output and the chip's input; that resistor together with the chip's clamping diodes will make a very nice RS232 to TTL converter!

You can power the CMOS chip from a 78L05 regulator, fed through a few diodes and a capacitor from the RS232 handshaking lines. FodTrack will "switch them on" for you! You can put all these parts into an RS232 connector (even a 9-pin shell is big enough, if you are careful...), to get a very nice and cheap control interface.

Some radios use TRUE polarity, others need the RS232 signal inverted. Check the docs of the radio, and use one or two gates of your 74HC04 accordingly.

Remember that for FodTrack you don't need data FROM the radio TO the PC, but if you want this for some other program, just put the TTL signal into the PC's RS232 input. I have never seen any port that does NOT work that way!

I'm providing the schematics of the adapters I use in my station. RS232Y.PCX is the version for Yaesu, which I use on my FT736, while RS232K.PCX is for Kenwood radios. I use that one for my TS450, and I hope it will run also on the TS790, but I have not tested this.

I'm also providing a schematic for an interface which perhaps could get the Guiness record for simplicity: just three components! This circuit does not provide a "reply" connection for the radio, and it can be used only with radios that have an input with pull-up, and use inverted signals. It works perfectly well with FodTrack and the Yaesu FT736R, and will probably also work with Icom radios (using another plug, of course...). This circuit is in RS232SIM.PCX.

Ready made interfaces:

For those who don't want to assemble their own interfaces, AMSAT-CE can provide the circuits in ROTORINT.PCX and RS232SIM.PCX assembled and tested, ready for "plug and play". Any funds collected go to our CESAR-1 project, a 9600 baud Pacsat. If you are interested in this deal, write to Carlos Godoy, CE2HI, via any of the following ways:

Mail: AMSAT-CE
P.O.BOX 803
VINA DEL MAR

CHILE

e-mail: ce2hi@entelchile.net

Fax: +56 32 88 4073

Cost, for the combination of the two interfaces, including shipping via registered airmail, is about US$ 110. Unfortunately AMSAT-CE cannot charge credit cards. The two interfaces can be provided separately.

GPS support:

If you have a GPS receiver or another device that can output NMEA-0183 datagrams, you can connect it to a serial port of your PC and use FodTrack to keep the PC's clock accurately set. Optionally you may enable a function that gets the location from the GPS. This should be very attractive for maritime mobile stations and for anyone who moves his station around a lot!

But even if you operate a typical fixed station, the time setting feature alone may warrant the purchase of a GPS receiver for many of you.

At this stage the GPS support is still experimental. I need input from you to implement more features.

What the GPS feature in FodTrack does right now is this:

- Whenever the program is sent to a satellite, or if you press the "G" key, FodTrack will read the specified serial port and wait for up to 6 seconds for the arrival of two consecutive datagrams of the same type bearing different time stamps. It will then assume that the latter time stamp is reasonably fresh, and will set the PC clock to this time plus a small offset which is there to compensate for the time interval between the instant the GPS fixed the position, and the moment the datagram actually arrived at the PC. You can specify this offset in the CFG file. The proper value depends on your specific GPS receiver, but will typically be about 1 to 3 seconds.

- If the position feature is enabled, then FodTrack will also set the new geographic coordinates, and it will write the datagram to a file called FODTRACK.GPS, in order to allow other programs to make use of this data.

- During the first and last minute of each day the GPS access is inhibited. I did this because the NMEA datagrams sent by most low-cost GPS receivers do not provide the date. So, by inhibiting GPS timesetting near the date switch I hope to avoid setting today's date with tomorrow's time!

I would very much like to use an NMEA datagram that contains valid UTC time and date, not just the position fix timestamp, but it seems that many low-cost GPS receivers don't support such datagrams.

FodTrack presently understands the GLL, GGA and RMC datagrams. This should provide compatibility with almost any GPS receiver. If your's doesn't give any of these, please tell me which one it produces, so I can add support for it.

I tested the GPS function using my Magellan Trailblazer, and I have heard of several people using it with Garmins.

Automation:

There are two different ways for automatic tracking of multiple satellites.

The preferred mode is this:

Configure your skeduling program in such a way that it writes a FODTRACK.CMD file into the FodTrack directory, containing the name of the satellite to be tracked, exactly as it appears in the kep file. This CMD file should be written two or three minutes before the start of the pass, to give FodTrack enough time to do the crystal ball business and preset the antennas. The easiest way to write these command files is simply to copy them from somewhere else into the FodTrack directory, using a COPY command inside the a batch file you run at the start of a pass. For example, you can have a FODTRACK.CMD file in your KO-23 directory, containing the text "KO-23" (without the quotes, of course...). Into the BAT file you run at the start, you put the command COPY C:\SAT\KO-23\FODTRACK.CMD C:\SAT\FODTRACK\FODTRACK.CMD
After the pass, the skeduling system should write a CMD file saying NONE, so FodTrack stops.

Now you load FodTrack inside your multitasker. When your satellite skeduler decides to run a pass, it calls the BAT file, which copies the proper FODTRACK.CMD into the FODTRACK directory. When FodTrack reads the file within the next second, it starts tracking the specified sat. After the pass, the CMD file contents is changed to "NONE", and FodTrack goes to idle, waiting for the next pass.

FodTrack checks the CMD file continuously, so it is mandatory to have a disk caching program like SMARTDRIVE, or some hardware cache. Otherwise your hard disk will complain. But anyway, nobody should be working without a cache these days!

The alternative mode for automatization is simpler:

It consists just in configuring FodTrack for automatic exit after a pass (in the CFG file), and calling it before each pass specifying the satellite on the command line. For example, you use the command FODTRACK KO-23.

FodTrack will start up, wait for the specified sat, track it until the end of the pass, and then quit. The command file is not read in this mode.

The disadvantage of this mode is that when FodTrack starts for a pass, it doesn't know where the rotator is. This eliminates the possibility of selecting flipping according to the status from the last pass, which is useful to reduce rotator wear. In the preferred mode, with flipping and without parking, this kind of rotator wear reduction works.

If you switch to manual mode after starting FodTrack with a sat on the command line, it will then work in the normal way. If you then switch into automatic mode, it starts reading the command file. But if you switch directly from command line mode into automatic mode, the program will abort, because it considers the pass finished.

If you use WISP, you can tell the event scheduler in GSC to do the command file business via batch files, as described above. There is also a program available, called FOD-INIT and written by CN8HB, which does the glueing job between WISP and FodTrack. I'm not using WISP myself (don't like Windoze), so I cannot give detailed instructions on this setup. For any question about FOD-INIT, refer directly to its author. He can be contacted on the sats.

In case of trouble:

If you have any trouble with FodTrack, tell me, so I can fix the bugs for the next version. Also I will do my best to act upon any suggestion for improvement, if reasonable and possible to implement.

To contact me, send a message on KO-23, KO-25, UO-22, AO-16 or LO-19. You can also try through the terrestrial packet net, if my program shook your satellite antennas to pieces. The address is XQ2FOD@XQ2FOD.SER.CHL.SA.

On Internet, you can reach me at mmornhin@eso.org. Be patient, sometimes I'm away for several weeks.

Is there anyone around who likes to use snailmail? If so, use this address:

Manfred Mornhinweg
Radio Club Coquimbo
Casilla 381
Coquimbo
Chile

A last word:

I'm not a professional programmer, so you are allowed to smile about my program and me...

Retour à ma page Principale!