TinyTrak Owner's Manual

Version 1.0.2

Overview

TinyTrak is a small, inexpensive radio controller designed to receive and broadcast position reports from a GPS receiver. It removes the need for a full TNC (terminal node controller) in an APRS tracker. When combined with an NMEA-0183 compatible GPS receiver, and a radio transmitter, TinyTrak will key the radio at user-defined intervals, and transmit the GPS's current position.

Acknowledgments

TinyTrak was developed with the help of many individuals who deserve recognition. Randy Holland (KO6KC), Dave Lee (W6ZL), Ken Mirabella (KM6YH), Marty Mitchell (N6ZAV), Luc Bodson (ON9AAV), Walter Crauwels (ON4BCB), Vesa Kauppinen (OH4EA), Barbaros Asuroglu (TA2CBA), Klaus Hirschelmann (DJ7OO), Mårten Persson (SM7SYX), Sylvain Mercier (VE2SIL), James Gutshall (N7VHF), Anders Richardsson (SM7WGG), Allan Gibbs (G7GFU), James Gorr (N3TOY), David Inkster (ZL2BLI), Ron Graham (VK4BRG), Robert Stessel (K1WXY), Jacob Tennant (KB8QIR), Hans Meijer (PA1PG), Mike Palmer (K8LG), Alfons Wittoeck (ON4AWT), Wes Johnston (KD4RDB), Peter Mulder (PE1IEE), David Andersen (K0RX), Ernie Howard (W8EH), Bert van Dalen (PA0DAL), Ed Newman (VK4JEN), Curt Mills (WE7U), Brian DeYoung (KE4HOR), Harry Bloomfield (M1BYT), Ulrich Stolz (DJ9XB), Johan Hansson (SM0TSC), Fred Reimers (KF9GX) of FAR Circuits, and Lara Garrabrant (KD6AYO). Thanks go to all these people for helping make TinyTrak all that it is.

Construction

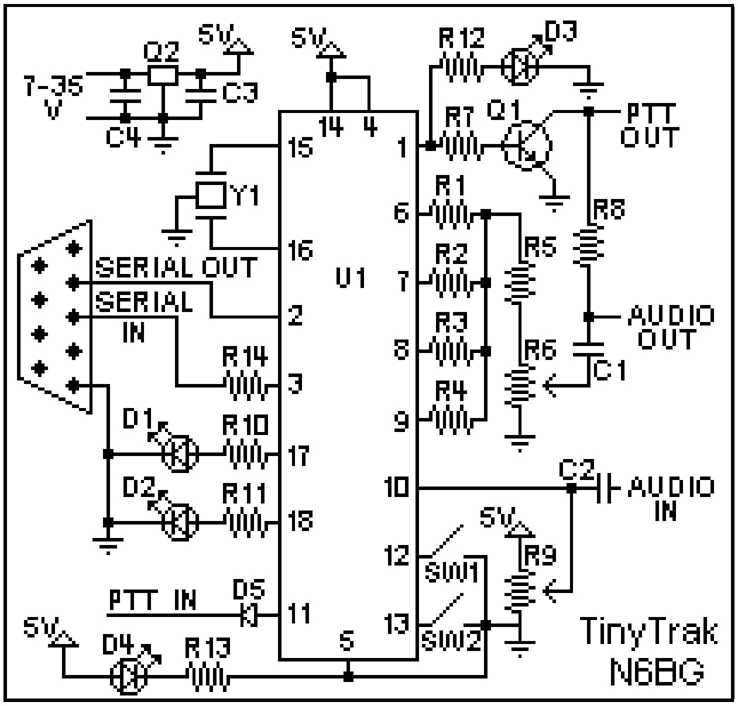
FAR Circuits PCB

A printed circuit board is available for TinyTrak from FAR Circuits under the name “Garrabrant TinyTrak Board”. The price is $4.00 each plus $1.50 shipping per 4 boards. Visa and MasterCard accepted with $3.00 service charge. To contact FAR Circuits, e-mail farcir@ais.net or browse to http://www.cl.ais.net/farcir/ . Voice and Fax number is (847) 836-9148 and the address is FAR Circuits 18N640 Field Ct. Dundee, Il. 60118.

Parts List

Part	**Description**	**Digi-Key Part Number**
U1	PIC 16F84-10/P microcontroller (programmed with TinyTrak Firmware)	PIC16F84-10/P
Y1	10 Mhz ceramic resonator	X906
Q1	2N2222A NPN transistor	PN2222ADICT
Q2	78L05 +5V voltage regulator	NJM78L05A
R1	8.2K ohm resistor (gry-red-red)	8.2KQBK
R2	3.9K ohm resistor (org-wht-red)	3.9KQBK
R3	2K ohm resistor (red-blk-red)	2.0KQBK
R4,R10,R11,R12,R13	1K ohm resistor (brn-blk-red)	1.0KQBK
R5	220K ohm resistor (red-red-yel)	220KQBK
R6,R9	10K trimmer potentiometer (103)	D1AA14
R7,R14	10K ohm resistor (brn-blk-org)	10KQBK
R8	2.2K ohm resistor (red-red-red)	2.2KQBK
C1,C2,C3,C4	0.1 uf capacitor (104)	1210PHCT
D1	Yellow T1-3/4 LED	67-1111
D2	Green T1-3/4 LED	67-1108
D3,D4	Red T1-3/4 LED	67-1102
D5	1N4148 diode	1N4148DICT
J1	DB9 female connector	209F
	18 pin dip socket	A9318

Schematic



Programming the chip

The TinyTrak kit is shipped with a programmed microcontroller, so programming the chip is not necessary if you have this kit. If you do not already have a programmed TinyTrak microcontroller, you will need to burn the firmware into a blank chip. You will need a copy of the TinyTrak firmware (TINYTRAK.HEX), a Microchip PIC16F84-10/P microcontroller, and the necessary hardware and software to program the firmware into the chip. The latest firmware can be found at http://www.byonics.com/tinytrak. The PIC chip need not be blank, since it is an EEPROM, and can be erased during programming. There are several options for programming equipment, including a PicStart Plus and Microchip's MPLAB software, a LidiPipo programmer or TAPR PIC-E and PicProg or PIX software, or a Tait-sytle programmer and corresponding software. Follow the specific instructions included with the programmer to program the chip.

Firmware Revision Notes

Version 1.0 was the first official release of TinyTrak.

Assembly Instructions

TinyTrak is a fairly simple construction project that can usually be built in less than an hour. It can be built on perf-board, or using wire wrap equipment, but it will be easier to build on the FAR Circuits PCB. These instructions are meant to aid PCB users. You will need a low wattage pencil-type soldering iron with a small tip, some thin solder, a pair of diagonal cutters, and a pair of needle nose pliers. The microcontroller (U1) is static sensitive, so use standard precautions. For each item, insert on the component side (blue silk-screen side), then turn the board over and solder the leads to the pads on the trace side. Be sure to only solder the correct pad, and do not let any solder touch any other pad or trace. Trim any excess leads with diagonal cutters after soldering each component. The following checklist will be useful to insure all components are properly assembled.

Required Components

- ✓ Install the 18-pin IC socket for U1. Be sure to align the notched end of the socket with the marked end on the silk screen. Do not insert the chip into the socket at this time.
- ✓ Install ceramic resonator Y1, which provides the clock oscillation for the chip. Direction does not matter.
- ✓ Install resistors R1 – R5 which create the 4–bit digital to analog resistor ladder. For each resistor, bend one of the leads 180 degrees at the bulb of the resistor so that both leads are parallel and 0.1 inch apart. Install vertically on the board. Polarity does not matter. Bend the leads apart once inserted to hold in place. See parts list for color coding and identification.
- ✓ Install potentiometer R6, which adjusts audio output level.
- ✓ Install capacitor C1, which smoothes and de-biases the audio output. Polarity does not matter. Leads may need to be bent to fit holes.
- ✓ Install resistors R7 and R14. Use instructions for resistors above.
- ✓ Install transistor Q1, which provides Push-To-Talk (PTT) for the radio transmitter. Be sure to orient flat side to match the long flat side shown on the silk-screen pattern (labeled QT on some boards).

Optional Components

The following component is used to interface to a computer for configuring, and to a GPS for operations. It is expected that most users will install this component, but if serial interfacing is desired via some other method, the three holes near J1's traces can be used instead,

- ✓ Install female DB-9 connector J1. Wedge the board in between the two rows of solder cups, with the 5 pin side aligned with the 5 pads on the PCB. Solder all five cups to the pads.

The following 3 components provide the circuit with 5 regulated volts when provided with 7-35 volts input. It is expected that most users will install these components, but if the board will be supplied with 5V directly, they will not be needed.

- ✓ Install voltage regulator Q2. Be sure to orient flat side to match the long flat side shown on the silk-screen pattern.
- ✓ Install bypass capacitors C3 & C4. Use instructions for capacitors above.

The following component provides current keying through the mic line, as needed by some HTs. This option is usually not needed for Kenwood radios, or mobile radios. If this option is used, you should not wire PTT OUT to the radio.

- ✓ Install resistor R8. Use instructions for resistors above.

The following 2 components are needed to read audio input level for carrier detection. This should help prevent unwanted transmissions over other existing transmissions. It is not used to decode incoming packet data, just detect its presence. If installed, carrier detect LED (D1) should also be installed. If these components are installed, be sure to adjust correctly, or all transmissions may be disabled.

- ✓ Install potentiometer R9.
- ✓ Install capacitor C2. Use instructions for capacitors above.

The following component is used for reading the microphone Push-To-Talk state. This can allow for position transmission on a voice channel after the microphone in un-keyed, as used with the TAPR MIC-E and PIC-E devices.

- ✓ Install glass diode D5. The black strip should be furthest from the PIC.

The following components are used to show TinyTrak operation state via LEDs. For each one, install both the LED and corresponding resistor. Be sure to align the flat side of the LED with the stripe on the silk-screen. Colors are only recommendations, and can be changed.

- ✓ Install Carrier Detect LED D1 (yellow) and resistor R10. This LED will light when audio is detected from the radio receiver. R9 and C2 are also required.
- ✓ Install GPS State LED D2 (green) and resistor R11. This LED will light constant when receiving good (locked) GPS data, and flash when receiving bad (unlocked) GPS data. It will be off when no GPS data is received.
- ✓ Install PTT LED D3 (red) and resistor R12. This LED will light when the radio is keyed via PTT.
- ✓ Install Power Display LED D4 (red) and resistor R13. This LED will light when device is powered.

Assembly Completion

After all components have been installed, inspect the solder side of the board for poor or cold solder joints. All pads should be shiny and smooth. Inspect for any undesired solder bridges. If the board looks ready,

- ✓ Insert programmed microprocessor U1. The rows of pins may need to be bent slightly. Be sure to align the notch on the chip with the notch on the socket, as well as the notch on the silk-screen (nearest connector J1). An improperly inserted chip may be permanently damaged.

Interfacing

There are a few interface points for the TinyTrak PCB which must be connected before operation. Some of them are optional

Power

If Q2, C3, & C4 were installed on the board, power can be applied via the Vin + and – points. The voltage regulator can handle 7-35 volts. At 12 volts, current draw is 6.6ma + 3ma for each LED lit.
If Q2, C3, & C4 were NOT installed on the board, 5V power can be applied via the +5 and GND points near the LEDs. Only 5V can be applied at this point.

Serial

TinyTrak must be connected to a computer for configuration of call sign and other operating parameters, and then connected to a GPS to receive position data. J1 pin 2 is used to transfer serial data from the TinyTrak to the computer. J1 pin 3 is used to transfer serial data from the computer or GPS to the TinyTrak. J1 pin 5 is serial ground. If female DB-9 connector J1 was installed on the board, the TinyTrak can be plugged directly into a computer 9-pin serial port, either with or without a serial extension cable. Due to the voltage levels used, some laptop computers may not be able to communicate with the TinyTrak. If the GPS to be used normally can plug directly into a computer's serial port, both a gender-changer (male-to-male) AND a null-modem adapter will be needed to interface the GPS to TinyTrak. If the GPS does not connect directly into a computer's serial port, an interface will need to be built. The GPS should have a male DB-9 with GPS serial data out wired to pin 3, and ground to pin 5. GPS serial input is not required. If J1 was not installed on the board, interface serial via the three holes near J1.

Transmitter

To transmit the positional packet beacon, a transmitter must be connected to TinyTrak via the AUD OUT and optionally PTT OUT points. If the transmitter transmits (PTT) when the Microphone input is grounded (most handheld (HT) radios, except the Kenwood brand), resistor R8 must be installed, but PTT OUT will not need to be connected to the transmitter. For all other transmitters, PTT OUT will be needed. Connect AUD OUT, Ground, and PTT OUT if needed to the transmitter microphone / PTT input. Refer to the transmitter's manual for more information, and look for a section on installing a terminal-node controller (TNC) for packet operation.

Receiver

If R9 & C2 carrier detect components were installed on the board to prevent transmissions over other stations, TinyTrak must be interfaced to a radio receiver. Connect the receiver's audio out (earphone) jack to the AUD IN and Ground points near R10.

Mic Input

When sending position beacons after unkeying on a voice channel is desired, the radio microphone input should be interfaced to this point. The line should be grounded when PTT is active, and floating at other times.

Switch 1

This switch input will select the alternate timing settings when grounded. It should be left floating, or at 5V at other times. This could be useful to change transmission rates when a car's engine is not running.

Switch 2

This switch input is not currently used.

Operations

Configuring User Options

To set user options, such as call signs and transmit rates, use the TinyTrakConfig.EXE. First, connect TinyTrak to a computer serial port. Launch TinyTrakConfig.EXE, and select the connected serial port. Click “Check Version” to confirm the TINYTRAKCONFIG program can configure the TinyTrak options. If a firmware version number is reported, the software and circuit are communicating. If not, re-check all previous steps. Click Read Configuration to upload the current configuration, edit all fields for desired options, and click Write Configuration to download options to the chip. Reading the Configuration is always performed twice and compared in the program to insure correct data. Writing the configuration is always followed by a read and compared to insure correct data. If either process fails, it will be retried. A dialog will notify the user when the read or write is complete, and if it was successful. Detailed information about each configuration parameter is included below.

Callsign

This is the identification of the transmitting station. It can each be an amateur radio call sign such as KD6BCH, or a tactical call such as SHUTLE. No more than 6 characters (excluding SSID) may be used. If a tactical call is used, an amateur radio call sign should be included in the beacon for FCC identification requirement compliance. As with normal packet radio, an optional SSID between 1 and 15 can be included, such as KD6BCH-9 or SHUTLE-15.

Path

This optional path will allow the transmission to be repeated by digipeaters. It should be entered in the form or call signs or aliases with optional SSIDs, separated by commas, such as RELAY,WIDE,WIDE . The number of call signs in the path is limited by the length of the beacon message.

Timing

The timing parameters affect when and how the transmissions occur. There are two sets of timing parameters, primary and secondary. Primary parameters are used for normal transmissions. Secondary parameters are used for manual, after-voice transmissions (MIC-E style), and for the alternate transmissions triggered with switch 1.

KeyUp Delay (TXD)

This sets the delay in milliseconds after the transmitter is keyed, until the data begins. During the key-up delay, the AX.25 flag byte (0x7E) is sent. It is similar to the TXD setting in most TNCs. A value of 100 ms would be equal to 1/10 second.

Transmit Every

This setting controls how often, in seconds, a position transmission will occur, rounded to 10 seconds. Valid range is between 10 seconds and 2550 seconds (42.5 minutes).

Quiet Time

This setting controls the delay in seconds that must occur after the receiver squelches, before a transmission will occur. It can keep transmissions from occurring on a busy channel. It requires the carrier detect components to be included (R9, C2, D1, R10)

Calibration

This setting is included to help compensate for inaccuracies in the ceramic resonator. By adjusting the value, packet rates can be sped up or slowed down. A value or 63 represents no correction. Experiment with different values to see which has the best results on a receiving TNC.

Beacon

TinyTrak can send a text beacon message after periodic location transmissions. The following two fields control the beaconing.

Message

This setting sets the beacon text. The length of the beacon message is limited by the number of digipeaters in the path.

Beacon Every

This setting controls how often a beacon is sent, in units of normal position transmissions. A setting of 1 would send the beacon with every position transmission, a setting of 2 would beacon with every other position.

APRS/MIC-E

The following four settings set special APRS settings which were introduced by the Tucson Amateur Packet Radio (TAPR) Mic-Encoded (MIC-E). For additional information about these settings, refer to TAPR's web page at www.tapr.org.

Message

This setting selects one of the 8 pre-assigned MIC-E messages. Caution should be used when selecting this setting, as beaconing with the "Emergency!" setting will alert most APRS receiving users to your emergency.

Path

This setting selects of the 16 pre-assigned MIC-E paths. In order to use the path entered above, this should be set to "Conventional".

Symbol

This setting sets the symbol most APRS programs with display when this beacon's position is received. A symbol setting of '>' will display a car, 'k' will display a truck, 'v' will display a van. Other symbol characters can be found in the APRS documentation.

Alternate Table

This setting will modify the meaning of the Symbol selected above to reference icons from the APRS Alternate Symbol Table. For normal use, it should not be checked. Refer to APRS documentation for more information on this setting.

Programming

This section controls communication with the TinyTrak, and will allow the setting selected above to be configured into the TinyTrak.

Com Port

This setting selects which communication port the TinyTrak is connected to during configuration. Currently only the first four serial ports are supported.

Check Version

This button will get the current firmware version from the connected TinyTrak. This is a useful, simple test to confirm the computer and TinyTrak can communicate.

Write Configuration

This button will download all configuration settings entered above into the TinyTrak. The settings are stored in non-volatile memory, so they will persist when power is removed fro TinyTrak.

Read Configuration

This button will upload all configuration settings from the connected TinyTrak into the setting fields above. This step is useful to change a single setting on the TinyTrak configuration.

TinyTrak Adjustment

There are only a few adjustments required for proper operations of TinyTrak. The transmit audio level should be adjusted at R6 for proper deviation. You can listen on a separate receiver, and start the resistor at maximum drive. When transmitting, lower the drive until there is a noticeable change in the receiver. It may help to temporarily set a long transmit delay (TXD) during this step in order to have more time to adjust the level. Overdriving the transmitter is a common cause of failure to decode. The other adjustment point is R11, which is used to adjust the carrier detect base level. This should be adjusted such that Carrier Detect LED is usually off, but turns on when the radio opens squelch. Radio receiver volume should be set to maximum, and the radio can NOT have a constantly open squelch.

Testing

After setting the desired configuration options, TinyTrak should be connected to a radio, GPS, and power supply, and will be ready for use. When power is applied, the carrier detect and valid LED should flash three times to show proper firmware operation. If a radio is connected, it should transmit periodically and a packet burst should be heard on a receiving radio. When the receiver’s squelch is opened, the carrier detect LED should light, and all transmissions will be delayed. If a radio receiver and TNC is available, packets should be able to be monitored. The data is not sent in human readable form, but instead in compressed MIC-E form. It should look something like this:

N6BG-9>S8PRPY,RELAY,WIDE:'2+"!r,j/]"4K}

This compressed form contains position, speed, and bearing. More information about this format can be found in the APRS documentation.

Hints, Tricks, Notes, & Troubleshooting

- The TinyTrak uses inverted TTL levels (0 & 5 volts) for serial communication, rather than true RS-232 levels (-12 & +12 volts). This may cause communication problems with some GPS receivers and computers, especially laptops. A RS-232 level converter, such as a MAX232, may be used, but the TTL levels will need to be un-inverted, with transistors or a TTL inverter chip.
- Currently, the only NMEA-0183 GPS string recognized by TinyTrak is $GPRMC. Any GPS to be connected to TinyTrak must be configured to output this string, if possible. GPS data must be sent at 4800 baud, N81.
- If the TinyTrakConfig.EXE program cannot communicate with the TinyTrak, it may be possible to test the serial link by running a terminal program, such as HyperTerm, configuring for 4800 baud, N81, and sending ESC followed by 'V'. These two keys should cause TinyTrak to respond back with the firmware version number. If serial communications are still failing, monitor the PIC pin 3 with a meter or an oscilloscope to check for incoming serial data.
- TinyTrak was designed for hand held radios. Some mobile radio require more audio drive than TinyTrak puts out. If audio levels are too low, even with the R6 pot set to maximum, consider replacing the 220K R5 with a 100K resistor. This should allow for about double the audio range.
- The current draw for TinyTrak, fed with 12 volts, is approximately 6.6 ma + 3 ma for each LED lit. If desired, the 4 1K LED resistors can be replaced with 10K resistors, which will cause the LED to only draw about 0.5 ma each, or the LEDs and resistors can be left of switched off completely.
- TinyTrak is configured as a DCE with a female DB-9 connector to allow simple connection to a PC for programming. If a DTE configuration with a male DB-9 is desired, this can be accomplished by placing a DB-9 male connector of the PCB edge, but shifted one pin over, so the old DB-9 female pin 2 pad mates with the new DB-9 male pin 3, the old pin 3 mates with the new pin 2, and the new DB-9 male pin 5 is not mated with any pad, but is instead just past the old pin 1. Then, jumper the new DB-9 pin 5 to ground. This configuration should allow simple connection to a DCE GPS, but will require a gender changer and NULL modem to connect to and configure with a PC.
- If TinyTrak keys a transmitter, sends a packet burst, but then fails to un-key the transmitter, it may be due to local RF energy. Lowering power output, or moving the transmitting antenna further away from the TinyTrak can resolve this. If it doesn’t, place a 0.1uf capacitor across the base and emitter of PTT transistor Q1 (ground and center).
- On powerup, LEDs D1 and D2 should flash twice, then the radio should send a test transmission, if carrier is not detected, and so LED D3 should light for about a second. LED D4 should always be lit. If the LEDs light this way, it will confirm that the LEDs are wired properly, and that the firmware is correct and running.
- If the radio keys up upon connection to TinyTrak, even before power is applied to TinyTrak, it could be that the radio does not key via current through the mic line (most mobile and Kenwood radios), and resistor R8 should be removed.
- If higher current is desired from the TinyTrak’s voltage regulator in order to provide 5 volts to a GPS, the supplied 100ma 78L05 can be replaced with a 1 amp 7805. A heat sink may also be required.
- TinyTrak’s carrier detects circuit detects all audio energy, not just packet audio tones. Therefore, it will not transmit over received voice audio. Be sure NOT to run an open squelch, as this will prevent all transmissions. The receiver should be set for maximum volume.
- If carrier detect, MIC-E style PTT input, SW1, or SW2 are not desired, the components normally connected to these pins can be left out. Unconnected microcontroller pins will float to the default correct behavior.
- For the 4-bit digital to analog resistor ladder, it has been determined that neither high precision resistors (1%) nor perfect powers of 2 resistors (1K, 2K, 4K, 8K) are required.
- TinyTrak can be mounted in a case with a GPS, with a single DB-9 jack for PC configuring. To do this, wire the GPS serial output through a 10K resistor before connecting to the solder side of pin 3 of TinyTrak’s DB-9 connector. This way, with no PC connected, the GPS serial data will feed into the TinyTrak, but when the PC is connected, the PC will over-power the GPS during configuration.