Guide to the N3ZI DDS Computer Control interface.

The DDS board includes a simple low cost serial interface with RS232 levels, no interface electronics is necessary for control from most computers serial port. Control via a USB port is also possible by purchasing a simple USB to RS232 converter cable. These are available on Ebay for a few dollars.

Make a cable from your PC's serial port There is a 3 pin connector near the upper right hand corner of the PCB. Pin1 is ground, Pin2 is DDS Output data, Pin3 is DDS input data.
Pin 1 (Ground) goes to Computer DB9 connector pin 5 Ground
Pin 2 (DDS TxData) goes to Computer DB9 connector pin 2 Computer Rx Data
Pin 3 (DDS RxData) goes to Computer DB9 connector pin 3 Computer Tx Data

You should set your serial port to: 4800 baud, 8bits, no parity, 2 stop bits, no flow control. However there is an autobaud detection routine, that will lock up to anything from 1200 baud to 9600baud. You can manually set it to higher rates, but the autobaud routine won't detect them. Type a bunch of "U"s until you see the prompt. It will save that baud rate for future use.

The common Windows HyperTerminal program works fine. Under linix Minicom works. Many others will work too. If you are using a USB to RS-232 converter, then you have to figure out which serial port it is, on Windows it's usually just the next com port. On linix the device is /dev/ttyUSB0, for a regular serial port it's /dev/ttyS0

On power up you will see this greeting: N3ZI DDS V1.0
If you send anything, you will get this response: 'ON' to activate
Type ON to activate serial communications.

Command	Action
ON	Turns on Comm's
<ENT>	Disp Frequency and Mem #
Q	QSY, change operating frequency
I	Change IF
M	Incr DDS Mem#
F	Turn off USART
Advanced Commands	
R	Readout EEPROM contents.
W	Edit EEPROM <space> increment to next location <ent> exit "=" enter new value (Zero must be entered with decimal point)
D	Debug mode
L	Load parameters from EEPROM
S	Save all to EEPROM

Note that the Serial control and normal front panel controls operate at the same time.
So if the tuning dial is rotated, or memory # is changed, the change will be sent out the RS-232 port. But some RS-232 command will delay acceptance of front panel commands. For example it takes a few seconds to dump out the EEPROM contents, if the dial is turned while this is happening it will be ignored. Similarly while you are in the EEPROM editor, front panel controls are locked out. If you primarily use the front panel controls, you should turn off the serial comm's using the F command, when they are not in use. With the comm's on tuning will lag a bit, more so in debug mode.

Advanced commands: In general changes to EEPROM will not actually be used until you reboot, or use the "L" command. Double check your changes by using the "R" command before you reboot or use L. Note: use of this is for experienced users only. Parameters are not range checked, and bad values can cause malfunction.

Loc	Default	Description	Normal Range	Units	Description
0	3,054,198.96	Indicates Valid EEPROM	Do not change	None	Do not change
1	1.74	Baud Rate	6.99 to 0.10 (1200 to 57600Baud)		=(8400/BAUD)-0.01
2	562,949.95	timebase factor	562,949.95 to 281,474.97		=2^48/10/Mosc(in Hz) (in the Jan 2009 version, this was a delta parameter, default 0.0, +/- 327.00)
3	0.00	Saved Memory Number	0.00 to 0.11	Integer (ignore decimal point)	Saved Memory Number
4	0.06	Default Step	0.01 to 1.00	10Hz	There are 6 step sizes, Fine (10Hz), default, 1k, 10k, 100k, 1M. Use some thing from 10Hz to 100Hz for default.
5	0.90	Rotary encoder queue threshold	0.0 to 1.01	10Hz	Rotary encoder queue threshold. Rather complex, but refers to what step setting turns off queuing for encoder pulses. Odd values enable quantizing
6	2.50	Main loop delay	0.50 to 25.00	100uSec	How often frequency is actually updated. Tied into encoder use and queuing
7	0.00	Speedup Hold	Binary 0.00 or 0.01	Binary	0.00 Speedup Hold causes speedup to return to default after a few seconds idle.
8	0.20	Rotary encoder Debounce	0.10 to 5.00	100uSsec	Rotary encoder Debounce Higher values cause slower but more predictable control.
9	0.02	Tick Divider	-0.10 to 0.01 to 0.10	+/- Integer	Number of encoder ticks it takes to cause a freq increment. Can be negative which reverses direction.
10	0.14	Cursor Type	0.12 or 0.14 or 0.15	Digital	Cursor Type in fast step modes: none, underline or block.
11	0.04	Speedup Button	0.04, 0.02, 0.01	Digital	Selects button to be used for speedup. 0.04=RENC push, 0.02=SW1, 0.01=SW2 (Memory up/down is reassigned to other 2)
12	1.10	SLCD Delay	01.01 to 99.99	100uSec	Delay time for SLCD xx.yy xx is final delay, yy is initial delay, in tenths of ms. Use long delays if a long cable or system noise causes bad readings. Use short delays for direct connect
13-20	N3ZI DDS		0.32 to 1.27	ASCII	ASCII string for last 8 digits on second line of Character LCD
21-31		Not Used			
32-44		Saved DDS Frequencies for each memory			You can changes values here but it's much easier to use the "QSY command
45-63		Not used			
64-76		Saved IF Frequencies for each memory			You can changes values here but it's much easier to use the "I" command
77-127		Not Used			