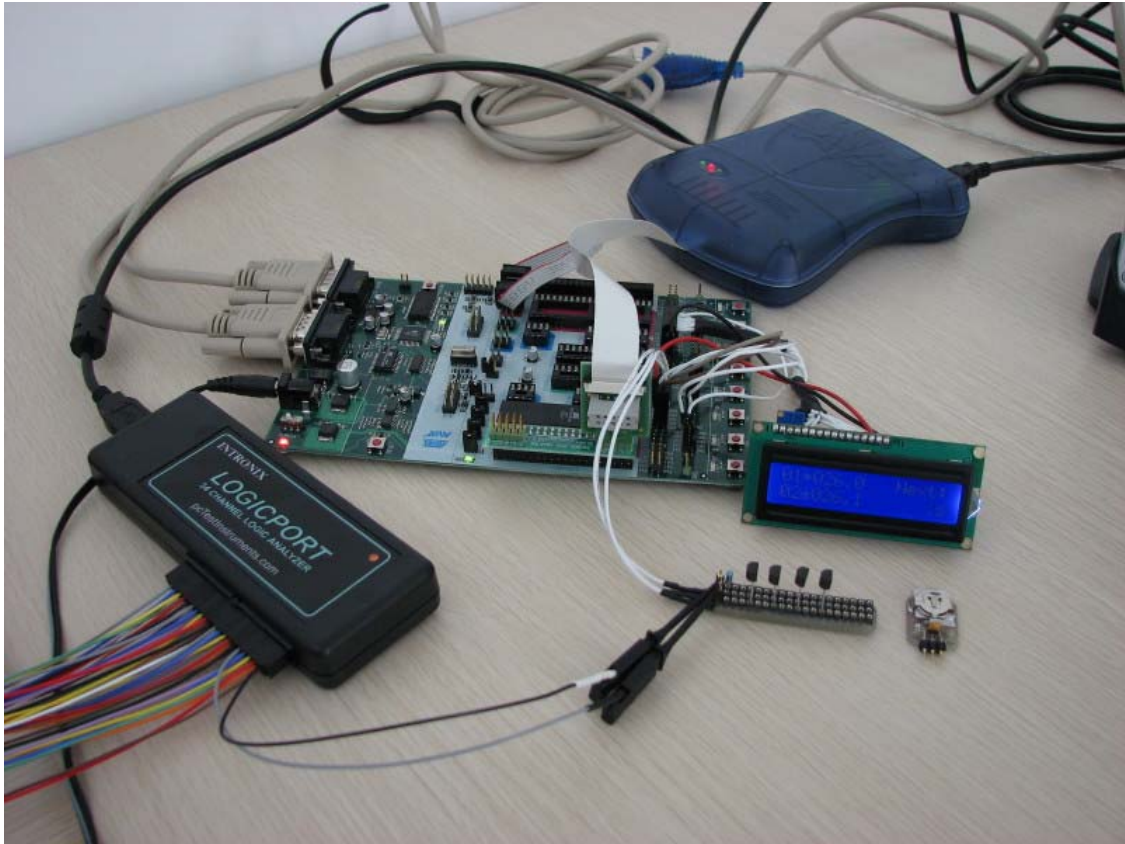


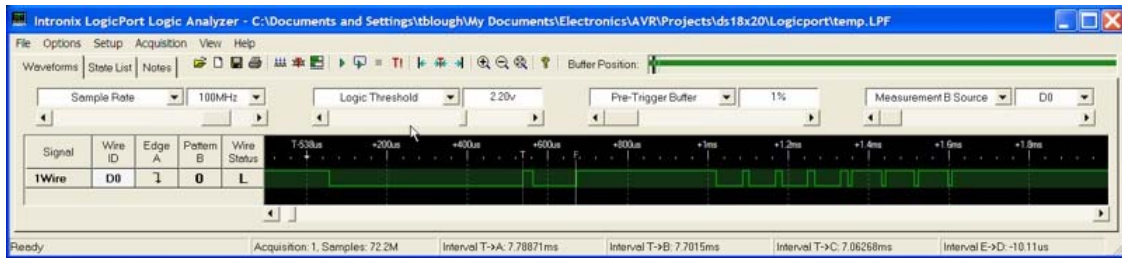
This is the test setup:



As you can see, I've gotten quite a few new toys to play with, so this is just as much for me as it is for you ;-). The blue box at the top is an in-circuit emulator. That allows me to step line by line through my code so I can find the bugs. The circuit board in the middle is an Atmel development board. It has common things like LEDs, switches, serial ports, and power regulators already on it so you can just concentrate writing your program. The LCD will display info about the temperature logging and allow you to set the logging interval (5 seconds to 9999 seconds), change the backlight brightness, change the order the temperatures are displayed on the screen, and display raw sensor data.

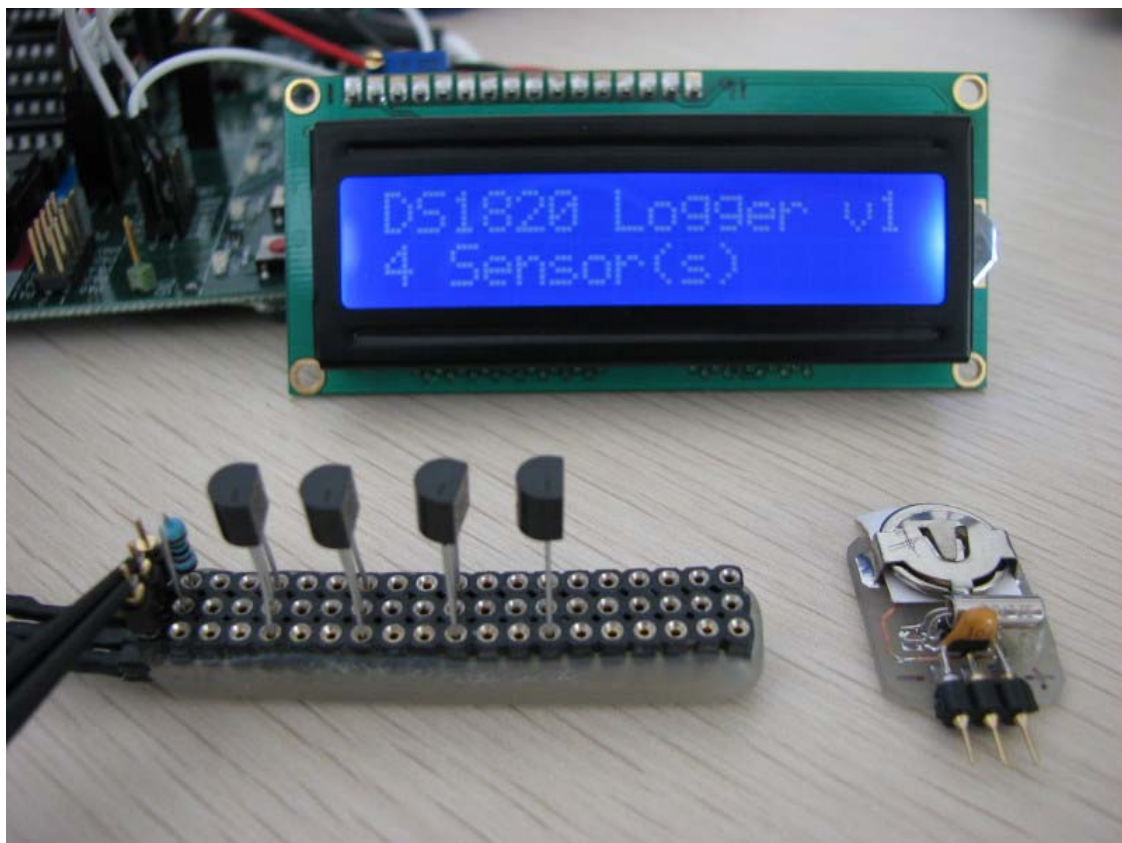
The little strip with the DS18B20s in it is just a 1-Wire bus I made to make it easy to test varying number of sensors. The little board in the lower right is a DS2415 real time clock and it's supporting circuitry (crystal, backup battery, and filtering capacitors, and header to fit in my bus). That will allow timestamping the serial output.

Finally the Logicport box is a Logic Analyzer. It allows me to see the data on the 1-Wire bus (I could also use it to view all of the microprocessor chip data as well). It generates an image like this:



What you actually see in the squiggly lines, it the attention signal, followed by four 0s and four 1s. That is the 1-wire command to search the 1-wire bus for sensors. The width of the little blips is very critical and is done by turning the output pin on, delay for a specific number of microseconds, and then turning the port back off. This is all done in code on the microprocessor. If you need a 6 microsecond pulse, you just can't program a 6us delay because the instructions to turn the port on and off and the delay instruction itself take a certain amount of time. That's why you need an logic analyzer or oscilloscope to measure the exact width of the pulse and then adjust the programmed delay accordingly. It's been a blast learning about all of this!

So, here is the first screen:



It shows there were 4 sensors identified on the bus. The serial output (there's a photo at the end of the email) actually shows a bunch more info about the sensors. Here is also a

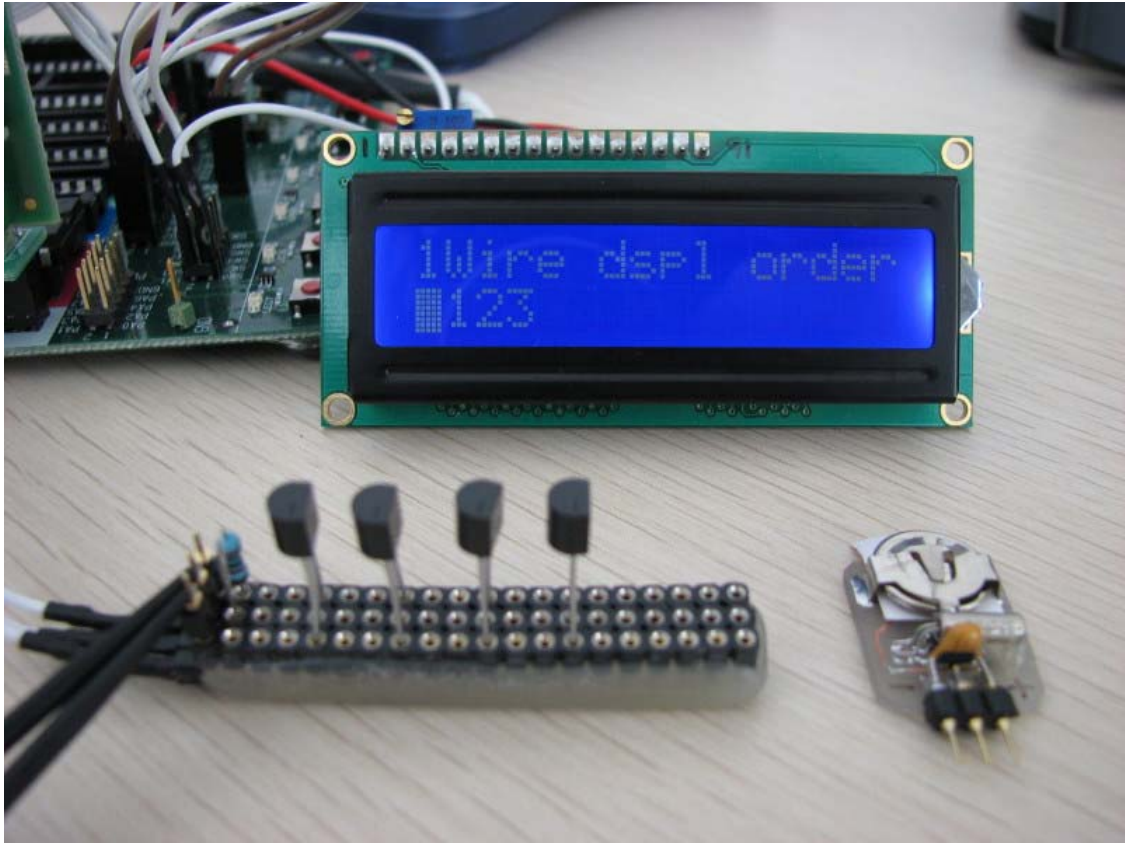
better view of the clock. You can barely see the DS2415 6-pin chip under the round silver crystal and the yellow capacitor. That was my first board, I've since changed it to a little SMD capacitor.

The first screen pauses for a couple of seconds and then goes to the start screen:

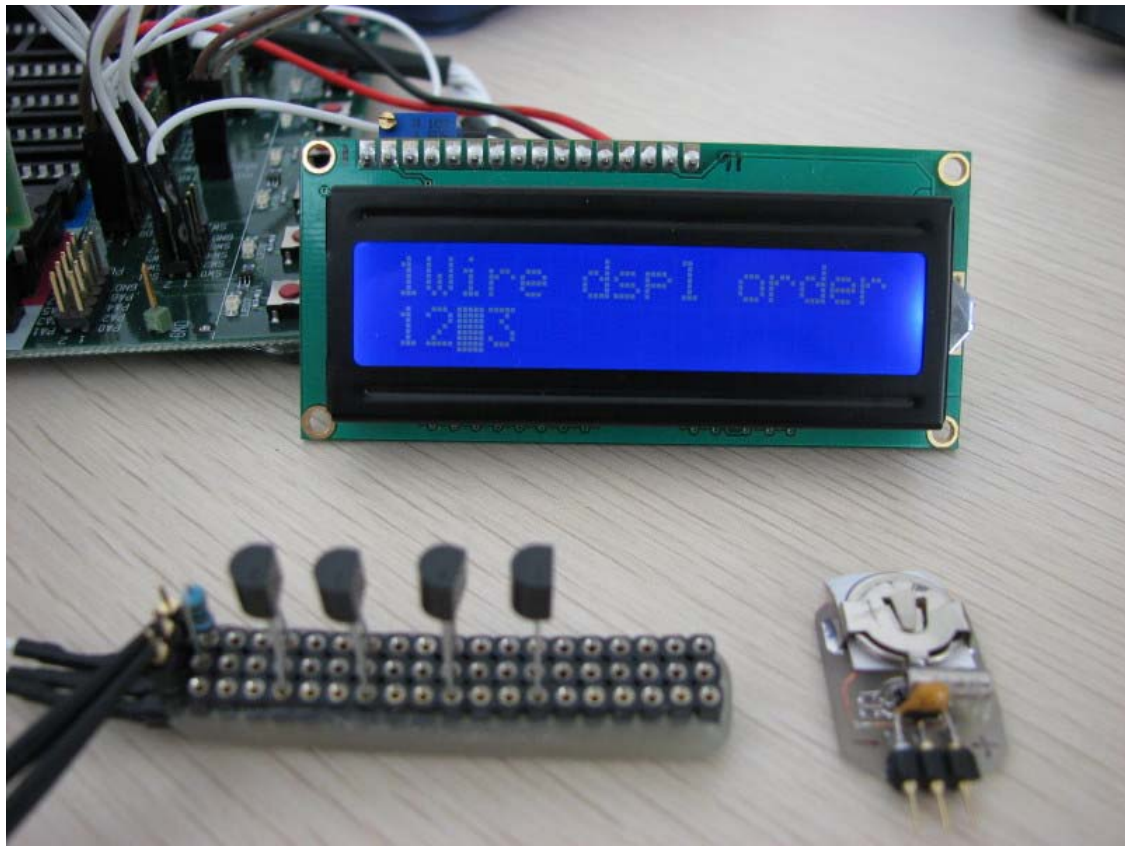


There are three pushbuttons; MODE, UP, and DOWN. Pressing up or down when in this screen starts the logging process. Pressing mode switches to the display order screen:

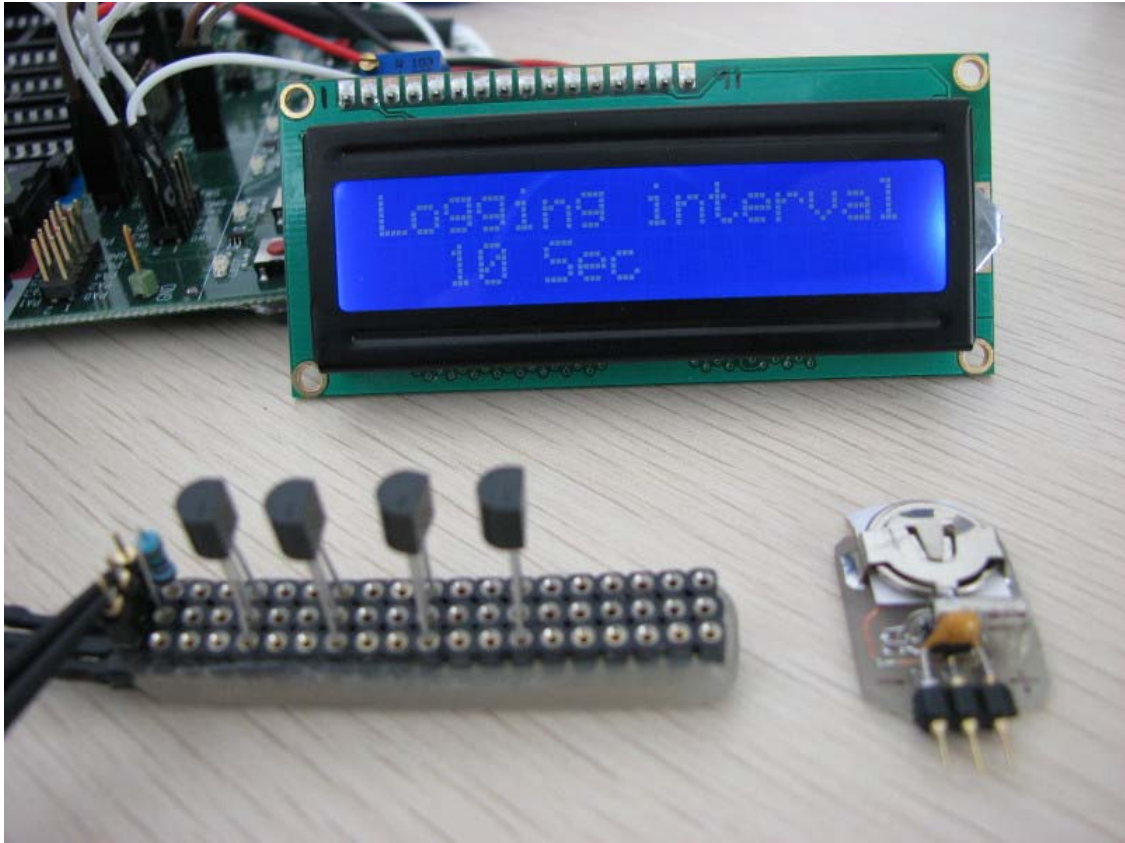




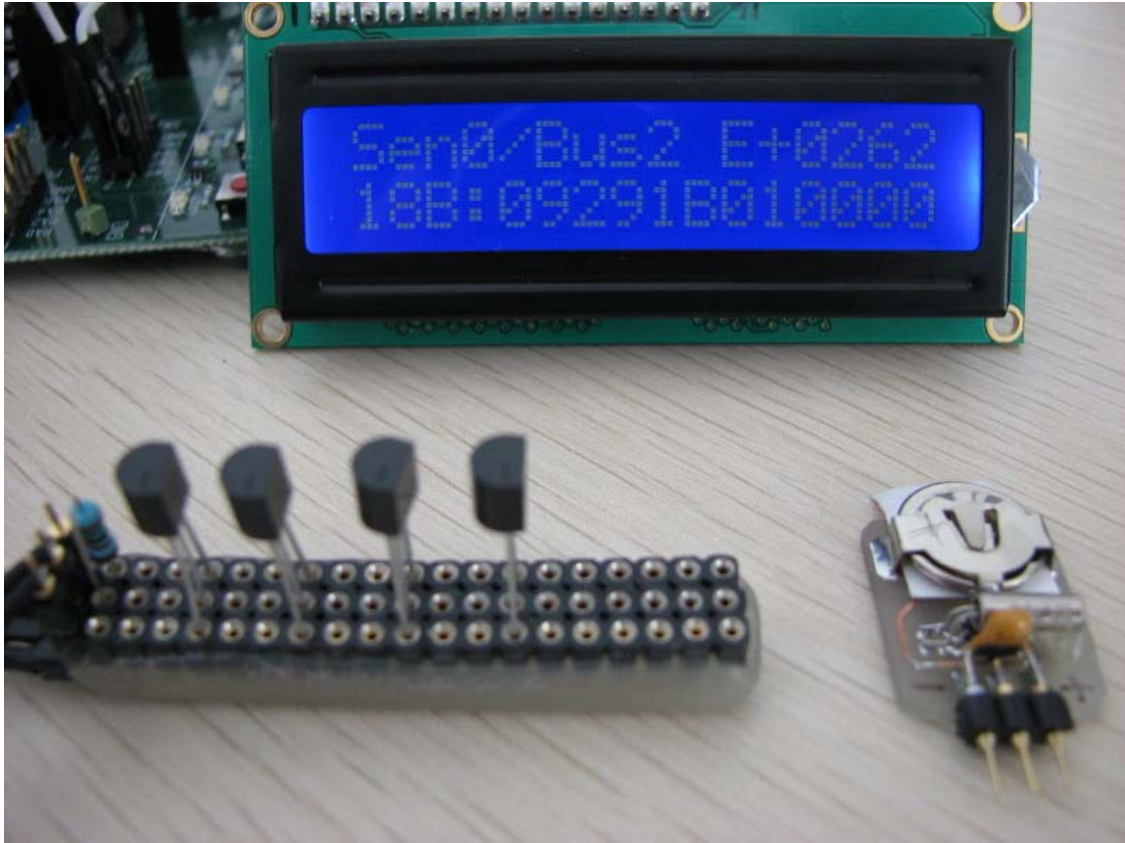
The search algorithm for the 1-Wire bus returns the sensors in their serial number order, not necessarily the order they are located on the bus, or even the order you would like to view them in. This screen allows you to select a sensor number (the white rectangle is actually blinking over the number 0). Sensor 0 is selected in this case to using the up and down buttons will move sensor 0 in the display order like this:



Pressing both up and down at the same time changes the sensor number you are moving. Pressing mode, changes to the interval screen:



This is where you adjust the logging interval from 5 to 9999 seconds. 2 hours and 4 minutes between readings should be enough, right? Up and down adjust the interval (the longer you hold one, the faster the interval changes), and the mode button changes to the sensor info screens:



For sensor info, you see the sensor number (the is the number that goes with the temperature on the serial output and logging display), the bus number (the order it was found during searching), either (E)xternal or (P)arasite powered, and the current temperature without the decimal point. The second lines shows the type of sensor (18B, or 18S), and the unique serial number. Pressing up or down scrolls though the sensors on the bus. Pressing mode changes to the LCD brightness screen:





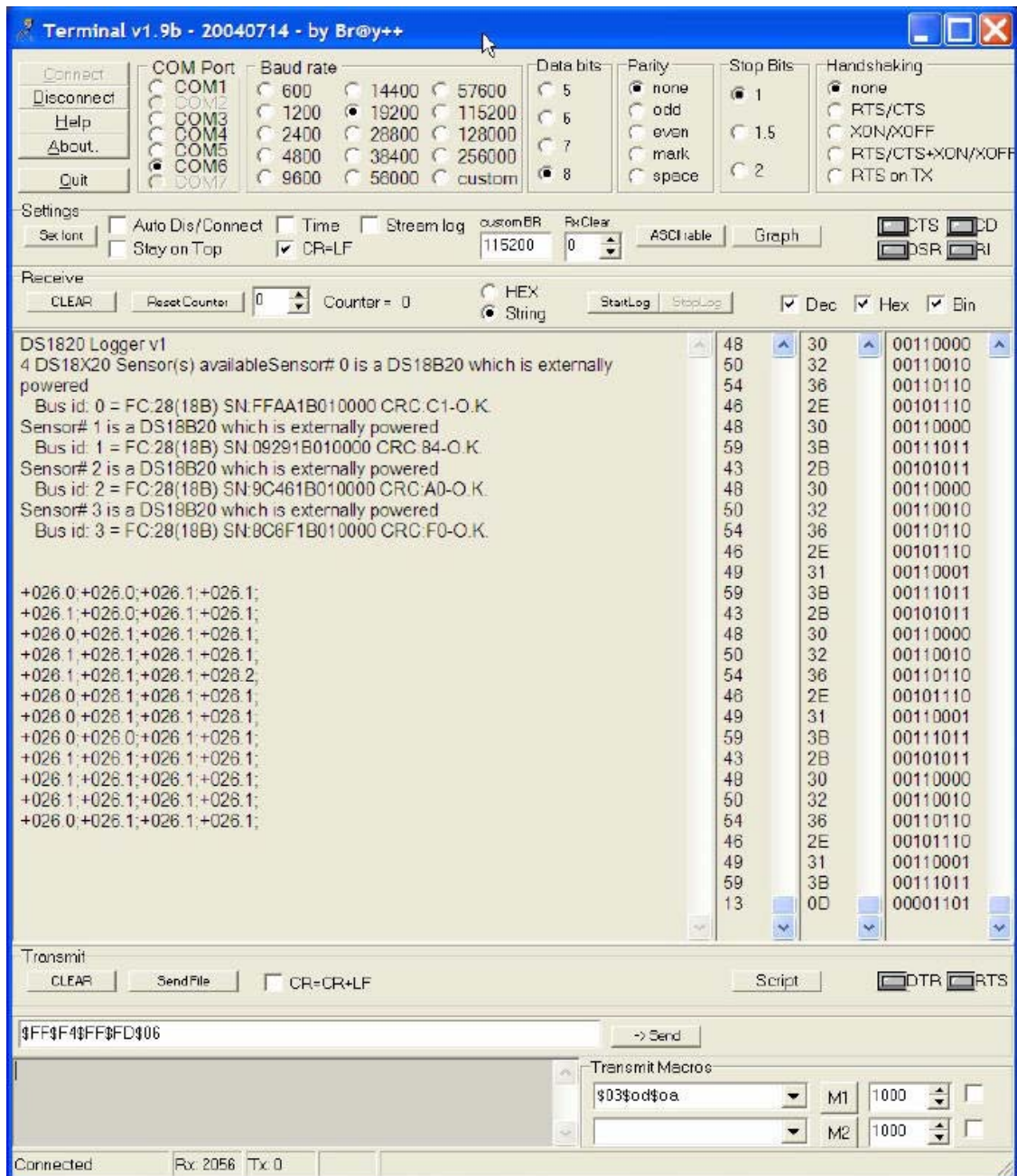
You can adjust the LCD backlight brightness between 0 (off) and 255 (full on) using the up and down keys. Pressing mode changes back to the start logging page. Pressing either the up or down button when on the start logging page, begins the logging:





Here you see the temperature for the first two sensors (in the display order), and a countdown when the next temperature reading will occur. Pressing up or down will change the display page to show the next pair of sensors temps on the display. In order to stop logging, you have to press both up and down at the same time. This was to prevent accidentally ending a logging session if you happened to bump a button. When you stop logging, you go back to the start logging screen.

This is the info that comes out the serial port:



There is some sensor info at the beginning then the lines of sensor temperatures. Each line will contain one temperature for each sensor on the 1-Wire bus. When I get the real time clock working, the each line will start with the data and time of the measurement. You should be able to capture this data with a terminal program (like HyperTerminal), or when I get this finished I'll rework my graphing program to read the data directly.

So, that's what I have so far. I still need to write the code to talk to the RTC then design a circuit board to hold it all. Do you want me to make it a through-hole board, or surface

mount? There really won't be very many components. Just the processor chip, the voltage regulator, a diode, a power jack and 3 capacitors in the power supply portion, a three pin header for the 1-Wire bus, a 9-pin serial connector, a 16 pin header for the LCD, three little switches, and 4 or 5 components for the serial interface. The LCD module will plug in right on top of the board.

I'm not getting a lot of time to work on it, but I'll try to have it all working before I leave for the US in early November for a week. Right now, I don't need anything from the electronics mart. I found DS18B20s here, just not DS18S20s. I might have you pick up some of the blue LCDs if I can't find them here - I haven't looked for them yet.

I'd love to get the books. We went to Shanghai last week and the only English book store we found was pitifully small. I got two there and I plan on stocking up when I'm in the US next month.

Talk to you soon,

Tom