Pocket Scope

A shirt pocket sized 2 channel digital storage oscilloscope with 8 channel logic analyzer, function generator, bode plotter, spectrum analyzer and 2 channel volt meter with simultaneous AC / DC display and data logging functionality.

Note: This is still a proto type and as such I do not have a front membrane for the design yet.

The idea

The idea for Pocket Scope began when I could not find a low cost, (under US$ 200.00), small, portable oscilloscope. I wanted an instrument that I could carry with me almost anywhere, not a 2kg portable scope such as those produced by Fluke and Tektronics, amongst others. I typically do not need 50 million samples per second out in the field; a scope with a 1 Mhz bandwidth and 250k samples per second would be adequate. I wanted an instrument with a price tag low enough that I could carry the device into filthy locations, like a coal mine, without being concerned that the unit would never look the same again or I could leave the device out in the field to log data samples, only to retrieve it at a later date.

I started working on the design and soon realized that the design could fill a void in the market place as a low cost all in one instrument for students and small business. All I had to do was add logic analyzer and function generator capability. I considered calling the design “Pocket Lab” rather than “Pocket Scope” and, perhaps if I could have squeezed in a bench power supply, I would have. Once you have a scope and a function generator together in the same instrument, you can then start to offer unique features like a bode plotter, which will allow students to test and compare “real world” circuits against what they have studied in the lecture rooms and would free the student from out of the institutions engineering labs which have limited capacity.

Design criteria

The Pocket Scope had to be small and low cost. It had to be electrically and mechanically rugged with a good usable battery life. The design had to be compatible with standard scope probes and should pack as much functionality as possible. The Pocket Scope should allow simple, safe updating of the code once deployed in the field.

Microcontroller requirements

The above design criteria place stringent requirements on the microcontroller to be chosen for this project.

The microcontroller must have speed.

The CPU needs to execute arithmetic and DSP type functions at high speed and so will need to incorporate a hardware multiplier and a powerful instruction set. A 16bit CPU will have great advantage over an 8bit CPU in this area. The CPU also needs to move a lot of data samples between memory and peripherals. This is where DMA is extremely useful.
The MSP430 is a 16bit CPU with a powerful instruction set and does include DMA and HW multiplier in many parts.

The microcontroller must have RAM.

A digital storage scope needs to store large buffers of high-speed sampled data. The graphics display necessitates a large chunk of RAM as a working buffer for the display in addition to the large chunk of Flash used by the font bitmaps.
The MSP430F161x parts with 5k or 10k RAM are an excellent fit.

The microcontroller must have secure Flash.

To release a product that can be upgraded once out in the field requires the microcontroller to support self-programmable flash memory technology. Not only must this memory be available, but in order to secure ones product against duplication, this memory must be secure. The MSP430 provides for all this.

The microcontroller must be low power.

High speed processing and low power are usually conflicting requirements, but the MSP430 microcontroller from Texas Instruments marries these two requirements together impressively allowing the Pocket Scope design to use small batteries in a small package and still maintain a useful operating life.

The microcontroller must have high-speed high-resolution peripherals

Again, these requirements are conflicting and again the Texas Instruments MSP430 pulls it off with 12bit 250k samples per second ADCs and 12bit DACs that can be tailored to the speed, power requirements of the application.

The microcontroller must support 32kHz operation

For the Pocket Scope to support time and date stamping of logged data, the microcontroller must support low power modes running, or sleep with a 32kHz

oscillator active and must support fast wakeups so as to minimize wasted power in retrieving the sample and going back to sleep.
The MSP430 again rises to the occasion.

The microcontroller must be available in physically small packaging
The MSP430 is available in 0.5mm pitch leaded and leadless quad flat pack packaging. These packages are small, but still very usable.

When you consider the sheer number of microcontrollers now available from many different manufactures, it is rather startling to realize that taking all the above requirements into account leaves you with exactly one choice, the Texas Instruments MSP430 range of micros, and I have specifically chosen the MSP430F1611, with it's large 10k ram memory footprint.

An important point to consider is the availability of a good C compiler. I was privileged to use the excellent Rowley CrossWorks C compiler for MSP430. The Rowley compiler allows an engineer to squeeze a lot more functionality into the available flash memory than any other C compiler I have tried for MSP430. A C compiler can make or break the platform you have chosen to develop on.

Specifications

Pocket Scope
Weight: 200g
Size: 125mm long, 70mm wide, 35mm high
Screen: 128 x 64 LCD with LED back light
Running time of over 24 hours continues operation
Screen scaling from 2.0mV to 2.0V per pixel with a 1:1 scope probe fitted
RS232 serial port interface for downloading of logged data samples as well as for code updates.
Code updates can be encrypted to the serial number of the device, and so can be emailed to the end client.

2 ch analog inputs
50 Ohm BNC inputs for use with standard oscilloscope probes
1M ohm || 10pf input impedance
1Mhz bandwidth and 250k s/s
Maximum displayable input signal is ±30V with a 1:1 probe fitted, and ±300v with a 10:1 probe fitted.
Maximum survivable long term input voltage with a 1:1 probe fitted is 500v rms

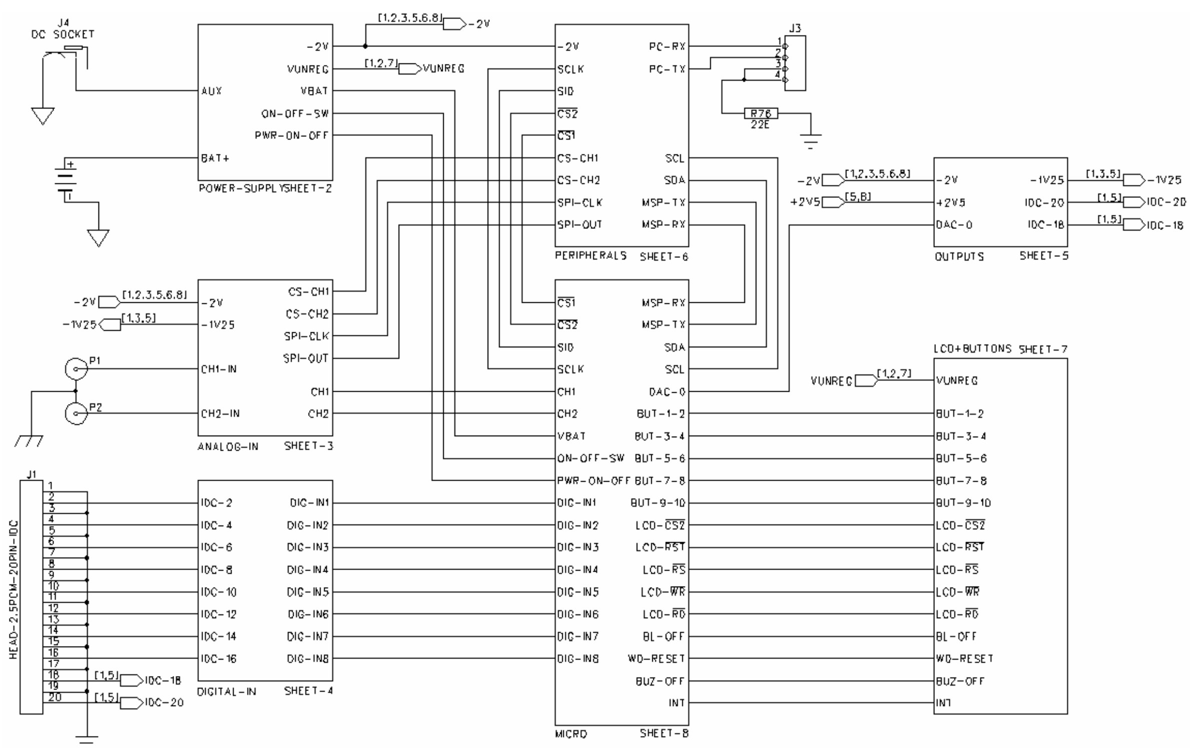
8 ch digital inputs
Greater than 100k ohm input impedance
500k s/s
Short-term survivable input voltage of up to 300v rms

1ch analog output
2k ohm source impedance
Sine, Square and Triangle waveform output
Frequency output of 1.0 Hz to 59 kHz, menu programmable
Voltage output of 0V to 3.3V p/p menu programmable
Short-term survivable input voltage of 30v rms

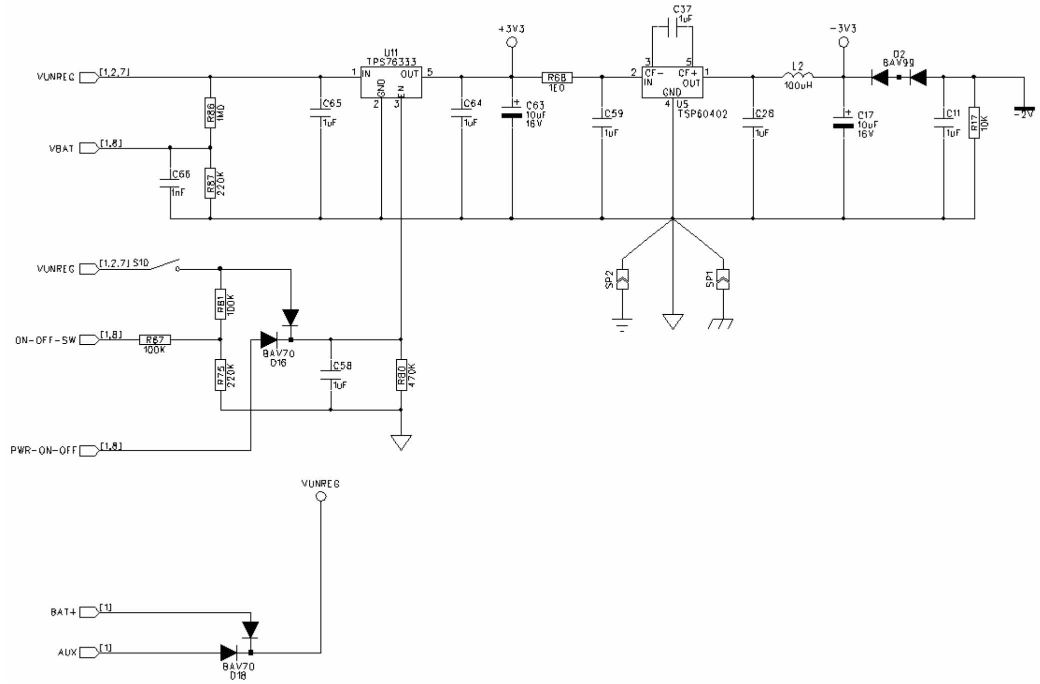
1ch open collector output
220 ohm to ground on resistance
Short-term survivable input voltage of +24v DC

Design

The hardware design can be broken down into a number of functional blocks as presented in the block diagram bellow;
The power supply, the analog input circuitry, the digital input circuitry, the function generator output, the user interface, and the microcontroller at the heart of the design.

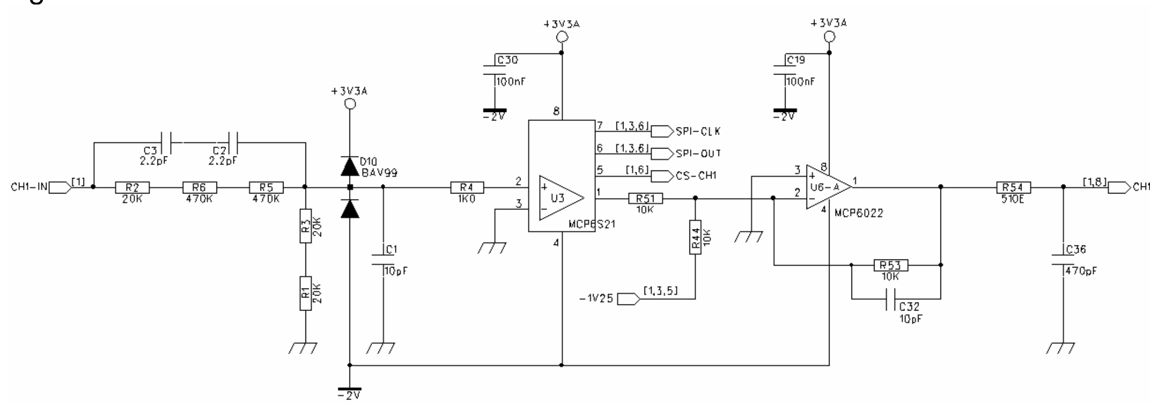


Power supply
Texas Instruments produces excellent analog components in addition to their excellent MSP430 microcontrollers. The power supply consists of a 3v3 linear regulator supplying the digital and analog positive supply rails, and a switched capacitor voltage inverter supplying the analog negative supply rail. The linear regulator’s ENABLE input is used to power the instrument up and switch it off as needed.



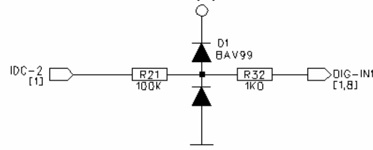
Analog input channels

Two identical analog channels are implemented. Bellow is the schematic of one of these channels showing the input conditioning and protection circuitry followed by a digitally controllable gain stage. A level shifter follows the gain stage as input signals are symmetrical about the GND rail and need to be shifted up above GND in order for the ADC to correctly convert the incoming signals.



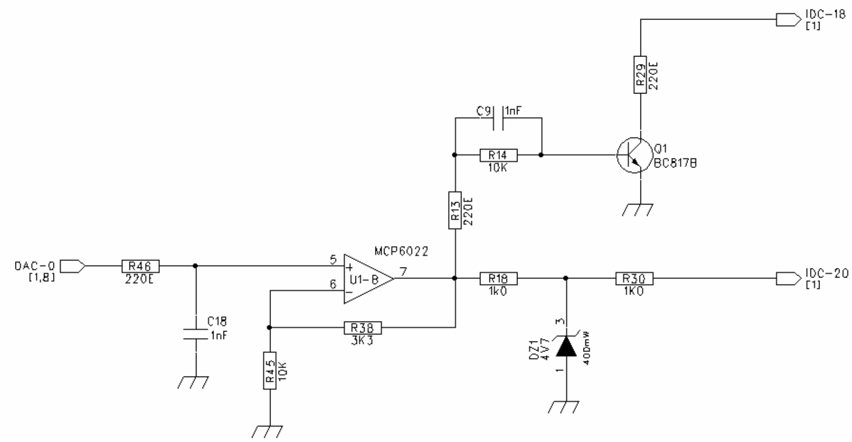
Digital input channels

Eight identical digital channels are implemented. Bellow is the schematic of one of these channels, which essentially is a protection network in front of an MSP430 input pin.



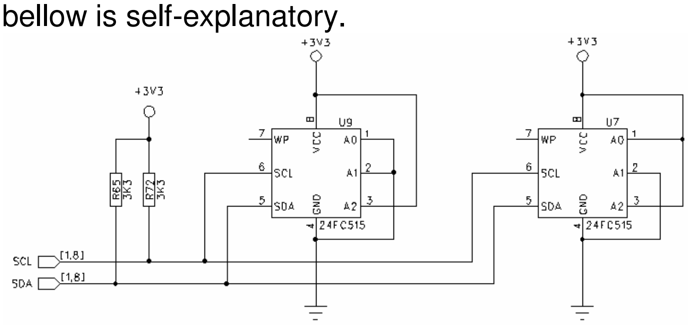
Function generator and open collector output

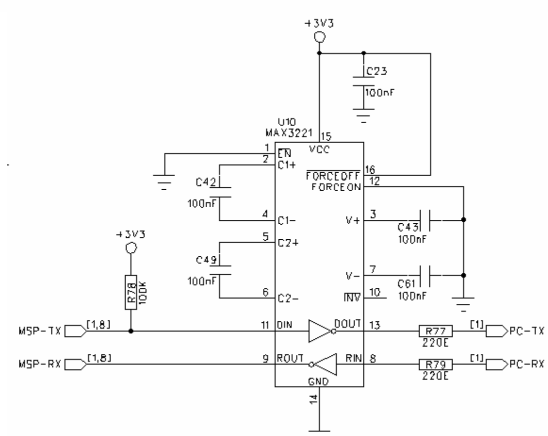
A major portion of the function generator is implemented within the MSP430 processor using waveform tables in flash memory, which are scaled and pushed out through the DAC using a DMA channel, all of which is part of the MSP430 microcontroller. The output from the DAC is then buffered and amplified with the bellow circuit. The output is zener protected against connection to external voltages. Additionally I have made available an open collector output to interface as a clock signal to external circuitry. To keep power consumption of the open collector circuit down, but still retain snappy transistor transitions, I have placed a 10k || 1nF resistor capacitor network in series with the base of the transistor.



Low power RS232 serial port and EEPROM data logging memory

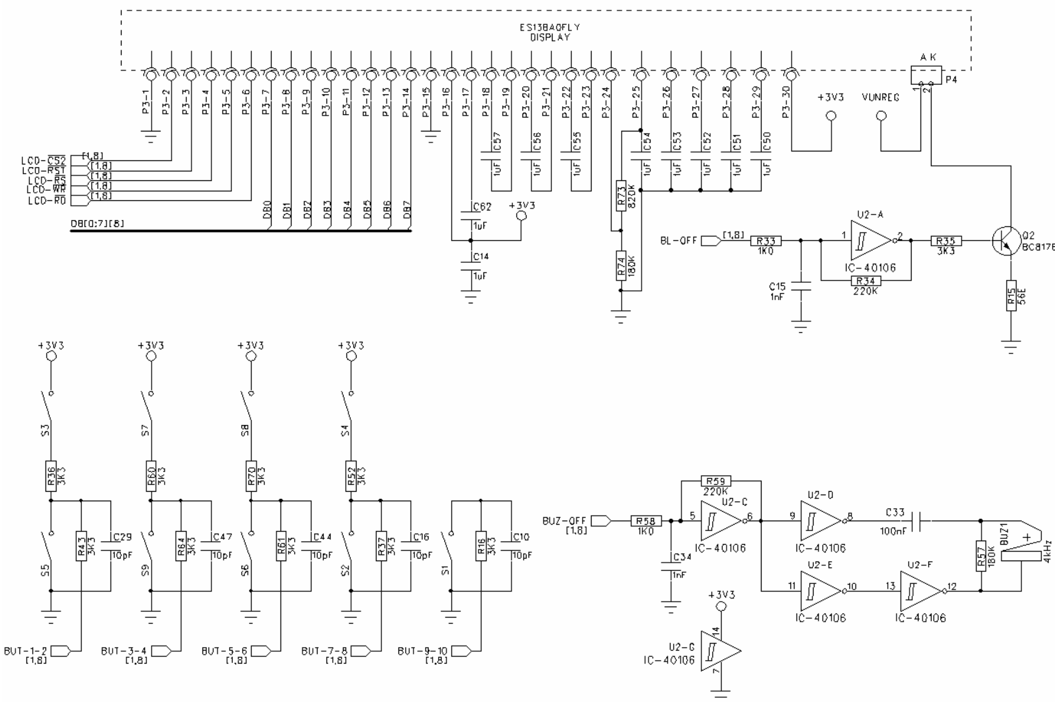
The circuitry bellow is self-explanatory.





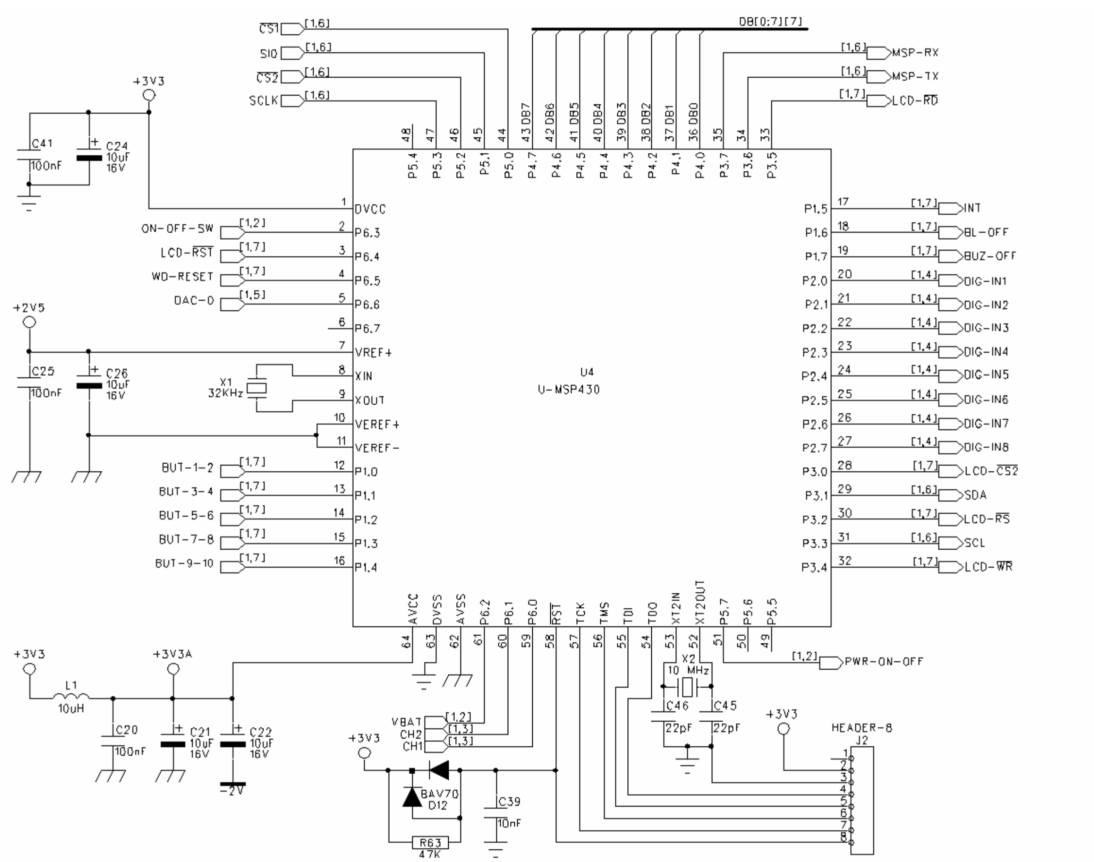
User interface

The bellow circuitry shows the 128 x 64 graphic LCD connections along with the button interface and backlight and buzzer driver circuitry. To free the MSP430's internal timers to be used for high speed ADC and DAC sample triggering, the LED backlight and buzzer are driven with external oscillators created using a low cost hex inverter IC. To save on IO lines, two switches are connected per IO and the IOs are periodically switched to be outputs and alternately driven high and then low, reading back the state of the lines each time. The end result is that an IO now has two bits per IO. (00) indicates that the switch to GND is being pressed, (11) indicates that the switch to Vcc is pressed, (10) indicates no switches are pressed, and the (01) combination is not used. The resistor between the switches guards against accidental pressing of both switches simultaneously.



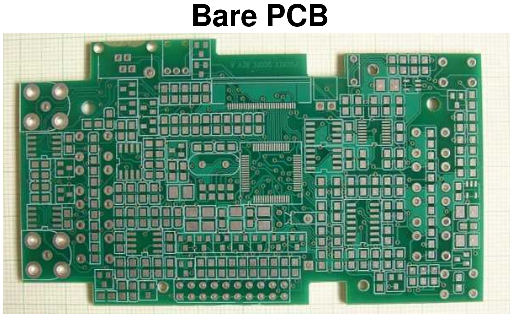
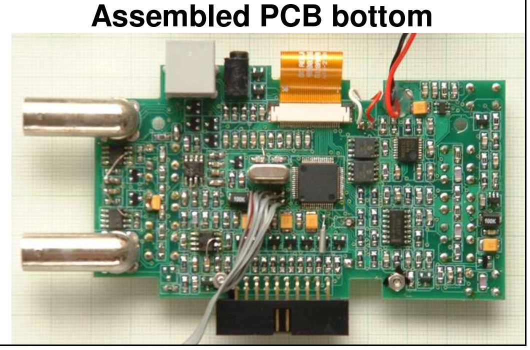
The microcontroller core

As can be seen, the design is rather simple due to the highly flexible and capable MSP430 microcontroller as shown bellow. Special attention was paid to decoupling of the various digital and analog power rails as well as the voltage reference. An instrument with 1mV sensitivity requires generous filtering and decoupling of these lines.

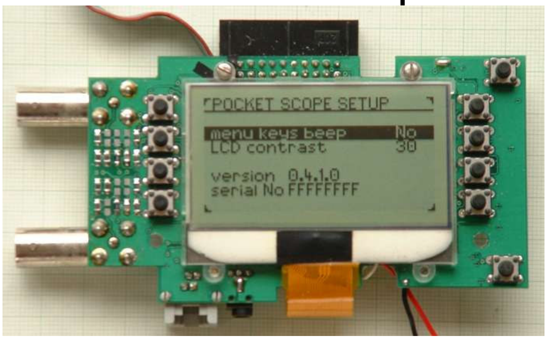


Putting it all together

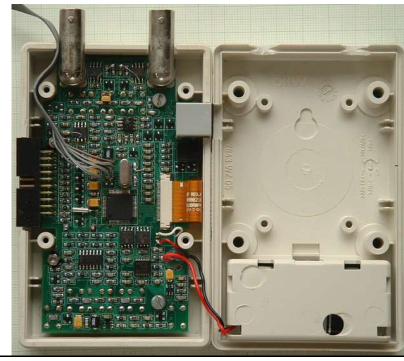
Bellow you will find a series of pictures displaying the Pocket scope in various stages of assembly as well as various modes of operation. The small grey umbilical cord exiting close to the top BNC connector in the bellow pictures is the MSP430 JTAG debugging and programming interface.

Bare PCB	Assembled PCB bottom
	

Assembled PCB top



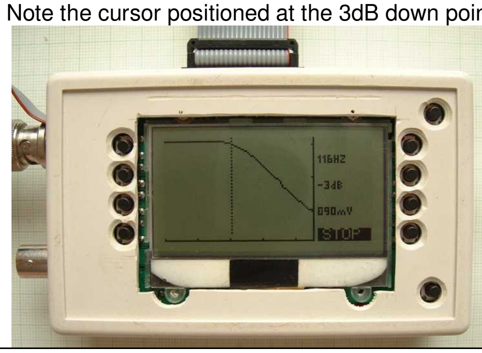
In the box



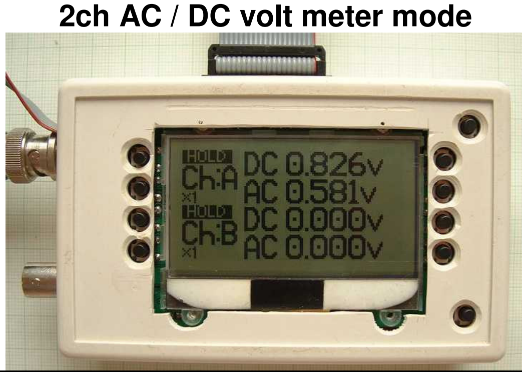
Back light on

Bode plotting mode

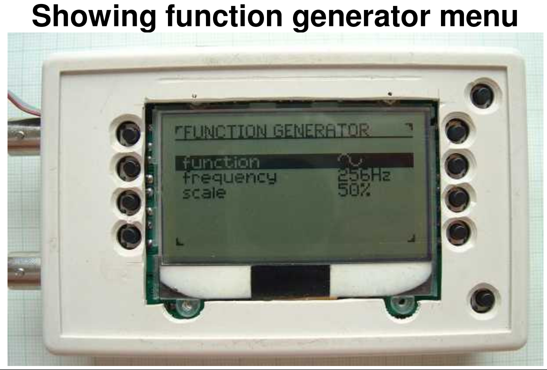
Note the cursor positioned at the 3dB down point



2ch AC / DC volt meter mode

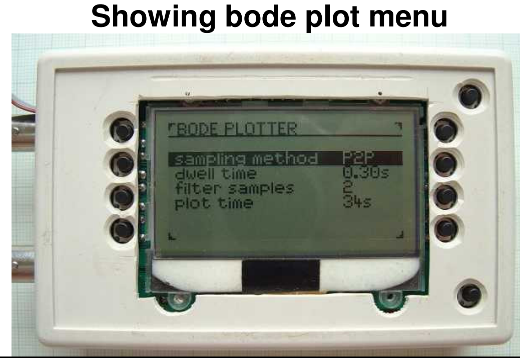


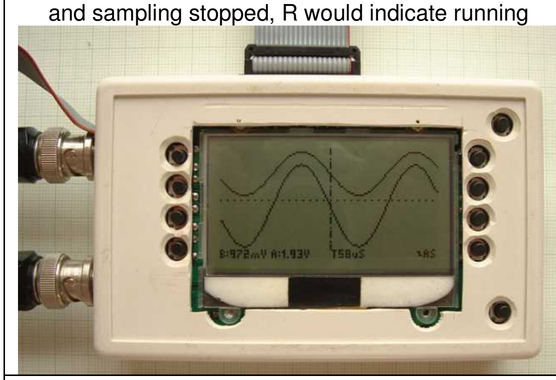
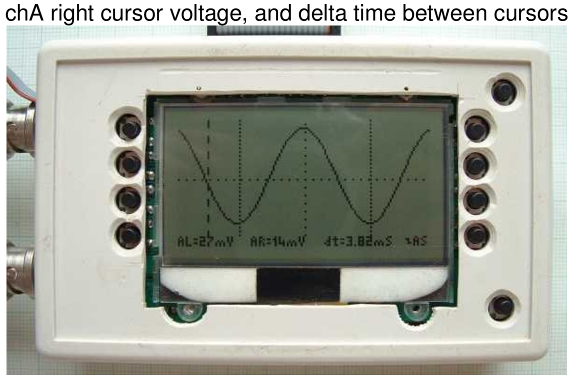
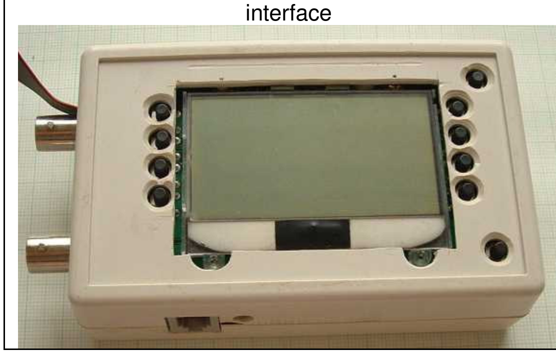
Showing function generator menu



Showing part of scope menu

Showing bode plot menu



Scope in 2 ch mode Info at bottom of LCD is channel A&B scale, sampling time, bottom right indicates falling edge trigger on chA and sampling stopped, R would indicate running 	**Scope with cursors activated** Info at bottom of LCD indicates chA left cursor voltage, chA right cursor voltage, and delta time between cursors 
More Pocket Scope 1 External view showing DC input jack and RS232 interface 	**More Pocket Scope 2** 

Challenges

By far the greatest challenges to me in this design where;

Fitting all of the menu and function controls intuitively onto only 10 user buttons. As an example, the bottom right hand button functions as the on / off button, back light on / off control, and the trigger run / stop button.

Coding a fast efficient trigger detection interrupt service routine. Without the MSP430’s 16 sample ADC buffer and DMA function, this would not have been possible at these high sampling speeds.

Implementation of a bootloader incorporating TEA (Tiny Encryption Algorithm) encryption and XMODEM protocol.

Designing the analog channel hardware with the associated signal conditioning and protection requirements.

Fitting all of it into a small box.

What I would change

The 9v to 3v3 linear regulator should be changed to a more efficient switching buck regulator to more efficiently make use of the energy from within the PP3 9v battery.

The two 32k byte I2C EEPROMs used for code updates as well as data log should be replaced with a single SPI data flash.

The RS232 interface is now dated and should be replaced with a USB interface.

What I would not change

I would not change the magnificent MSP430 microcontroller.

Credit

Conceived, designed, coded, assembled and documented by;
Shaun Parsons, Johannesburg, South Africa

Thank you for considering my design in the Texas Instruments MSP430 design competition October 2006.