



DSP-Based Software Radio

Background

A client was developing a software radio for base station use. They had done all the system calculations to determine how the signal path should operate, and what kind of signal processing they required. They had excellent in-house expertise for the RF design, but did not have sufficient in-house experience to do the digital design.

System Overview

The Software Radio required two boards, a Digital board with the signal acquisition and data processing hardware, plus a client-designed board with the RF Front-end, IF stages, and RF Power Amplifier. Bolton Engineering designed the Digital board around an Analog Devices [Blackfin®](#) processor running Linux. The system included 64Mbytes of SDRAM, 64Mbits Serial Flash, a 10/100-Base T Ethernet interface, a Compact Flash Slot and a moderate-sized Programmable Logic Device (PLD). The system used the Blackfin PPI to transfer data from the high speed A/D converter and to the high speed D/A converter. A system supervisory A/D monitored the various supply voltages as well as the system temperature.

In a Software Radio, the clock generator and clock distribution network are two of the most important driving factors in determining system performance. Bolton Engineering designed and simulated the low-phase noise clock distribution network at the design phase to ensure the best possible performance. The layout was carefully segmented and planed to separate the various design sections. After layout, Bolton Engineering used Hyperlynx to simulate and verify the system clock tree. The clock subsystem – and the A/D and D/A subsystems – were tested to meet the system requirements shortly after delivery, without requiring subsequent changes.

Project Scope

Bolton Engineering wrote the specification, designed the schematics, designed the 10-layer Digital circuit board, worked with the client to define the interface to the RF board, brought up UBOOT and Linux, wrote diagnostic and driver software in 'C', debugged the system, and delivered ten working prototypes. The client developed and maintained the PLD code.