Integrating Echo Cancellation Software with DSP Hardware for Mobile Switching Center

Juha Winter

Nokia Networks

Supervisor: Professor Timo O. Korhonen

Instructor: Teemu Himanen, M.Sc. (CS)

NOKIA

Agenda

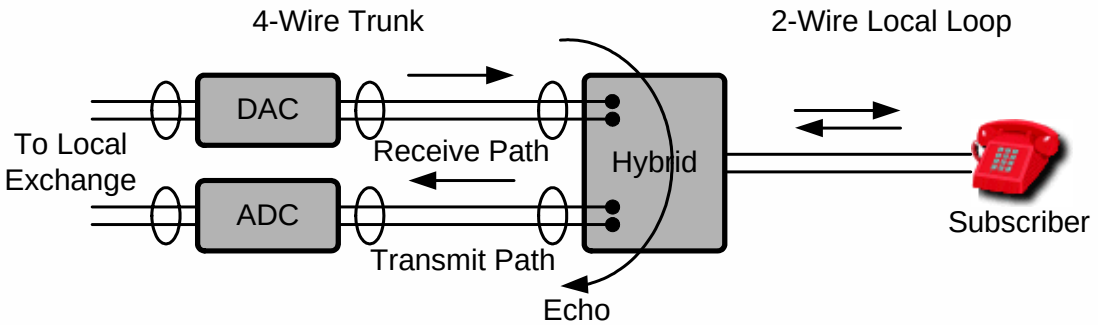
- Background
- Echo in Telephone Networks
- Delay and Echo Tolerance
- Echo Control in GSM/UMTS Networks
- Motivation
- Problem Description and Objectives
- Echo Cancellers
- Echo Cancellation Algorithms
- Echo Canceller Hardware & Software
- Development of the Serial Port Driver
- Conclusions
- Future Work

Background

- In wireless telephony, subjective voice quality is one of the most important factors in assessing the overall service quality of a network operator
- The extensive signal processing in today's digital cellular networks causes much longer end-to-end delays than are common in wireline telephone networks
- Although the increased delays as such do not degrade voice quality, they aggravate problems that would be otherwise unnoticeable
  - One such problem is **echo**
- Echo cancellation is a technique to reduce the echo to an acceptable level
  - Required in calls between a digital cellular network and the Public Switched Telephone Network (PSTN)

Echo in Telephone Networks

- Two fundamental types of echo: electrical and acoustic
- Electrical echo
  - Prevalent in connections involving the PSTN
  - Generated electrically due to discontinuities in the electrical characteristics (impedance) of the network transmission path
  - The main source is a transformer called the **hybrid**



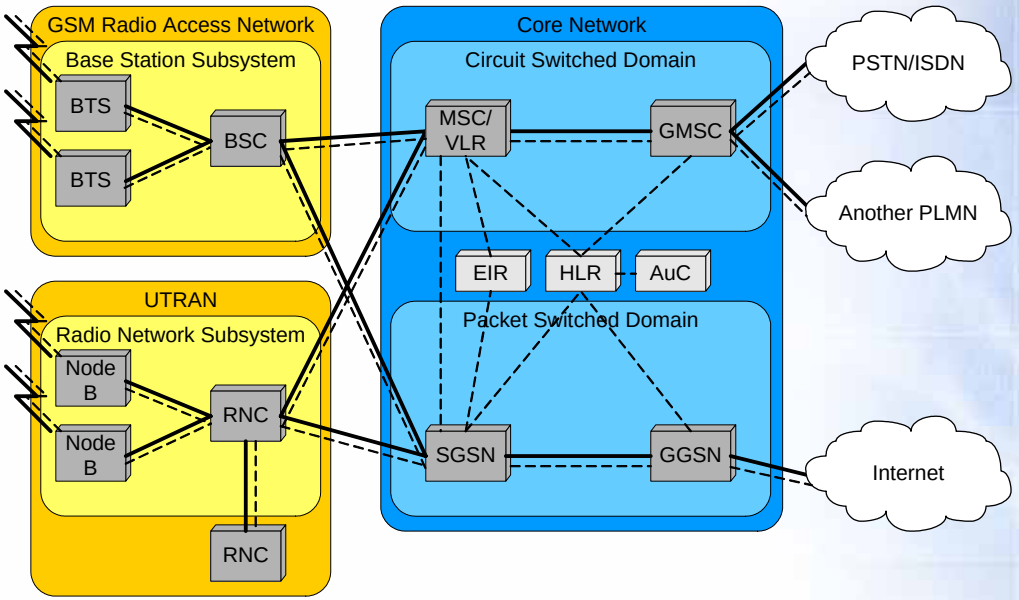
  - Most of the signal from the 4-wire line passes through the hybrid to the 2-wire line, but a fraction of it is reflected back because of the impedance difference between the line and the hybrid
- Acoustic echo
  - Caused by acoustic coupling problems between a telephone's loudspeaker and microphone

Delay and Echo Tolerance

- For echo to be noticeable, there must be some delay between the original signal and the echo
  - Round-trip delay less than ~10 ms ⇒ echo is unnoticeable
  - Round-trip delay more than ~30 ms ⇒ echo is annoying to most callers
  - Round-trip delay more than 50 ms ⇒ echo is annoying to almost all callers; use of an echo canceller in the circuit is **mandatory** [ITU-T Recommendation G.131]
- The longer the delay and the greater the loudness level of the echo, the more annoying the effect is to the talker
- Echoes were once a common annoyance in long-distance calls because the round-trip delays were long and echo control was nonexistent or limited at the least
- Digital wireless networks introduce various delays due to multiplexing, packetization, compression, transcoding, etc.
  - In GSM networks, the round-trip delay may be up to 180 ms

Echo Control in GSM/UMTS Networks

- Due to the various delays introduced, echo cancellers must be deployed in Public Land Mobile Networks (PLMNs)
- In GSM and UMTS Release 99 networks, echo cancellation is performed in the **Gateway Mobile Switching Center** (GMSC)
- In UMTS Release 4 and later releases, the Media Gateway (MGW) takes over the responsibility for echo cancellation



BSC = Base Station Controller
BTS = Base Transceiver Station
RNC = Radio Network Controller
UTRAN = UMTS Terrestrial Radio Access Network
GGSN = Gateway GPRS Support Node
GMSC = Gateway MSC
MSC = Mobile Switching Center
SGSN = Serving GPRS Support Node
VLR = Visitor Location Register

Motivation

- The global mobile subscriber base continues to grow rapidly
  ⇒ a lot more echo cancellation capacity will be needed in the future than ever before
- Echo canceller vendors are developing new high-density solutions to minimize the cost, space, and power consumption required per voice channel
- State-of-the-art **Digital Signal Processors** (DSPs) provide the needed processing power
  - Multiple processor cores per chip (**multicore** architectures)
  - Internal peripherals such as high-capacity, multichannel serial ports
- Multicore DSPs are considerably more complex than their single-core equivalents and are thus harder to program and debug

Problem Description and Objectives

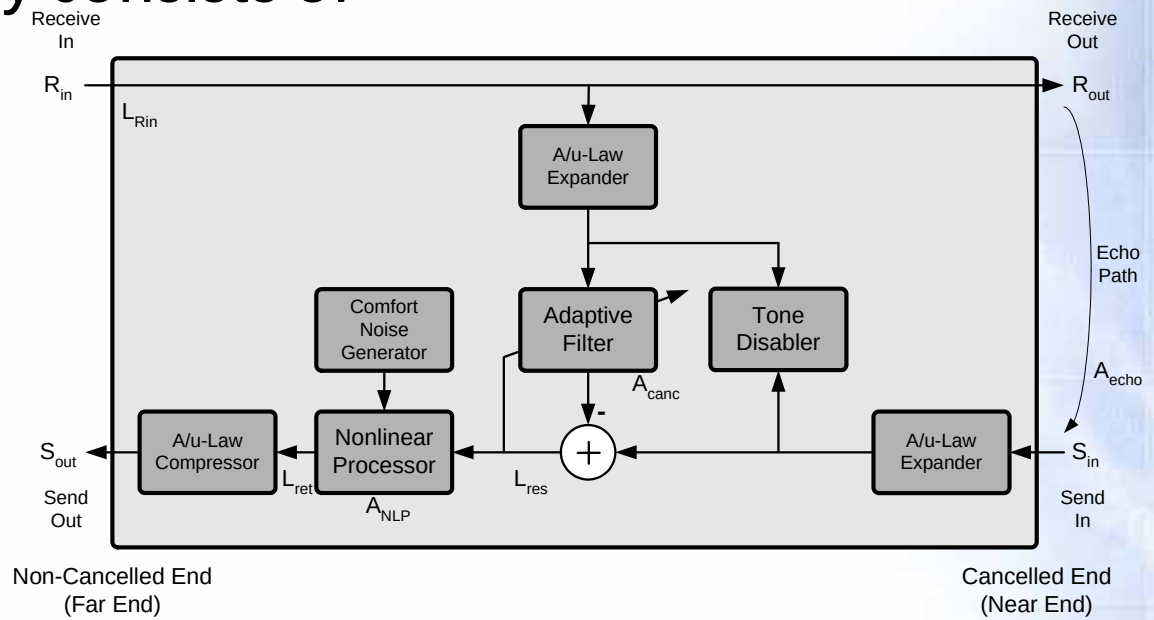
- Given an existing implementation of an echo cancellation algorithm and some other related software modules, how can the echo canceller software be integrated with new multicore DSP hardware?
- Nokia Networks' old echo canceller plug-in unit for the MSC
  - Based on Texas Instruments' TMS320C54x single-core DSP
  - Reliable solution but modest channel capacity
- The new echo canceller plug-in unit
  - Based on Texas Instruments' **TNETV3010** multicore DSP
    - Six TMS320C55x cores per chip clocked at 300 MHz
    - Two 512-channel buffered serial ports integrated on-chip
  - Industry-leading channel density
- No existing software for the TNETV3010 $\Rightarrow$ new low-level **device drivers** must be written from scratch

Echo Cancellers

- Network echo cancellers are
  - Voice-operated devices placed in the 4-wire portion of a circuit
  - Used for reducing the echo caused primarily by the hybrid
- The circuit between the echo canceller and the hybrid is called the **echo path**, a term used to describe the signal path of the echo
- An echo canceller can be treated as a “black box” with two inputs and two outputs
  - The side of the device facing the echo path is called the **cancelled end** or near end
  - the opposite side is called the **non-cancelled end** or far end
- The delay from the receive-out (R_{out}) port to the send-in (S_{in}) port of the echo canceller introduced in the echo path due to transmission facilities is called the **echo path delay**
- The maximum echo path delay for which an echo canceller is designed to operate is called the **echo path capacity**

Echo Cancellers, continued

- An echo canceller typically consists of
  - an adaptive filter
  - a nonlinear processor
  - a comfort noise generator
  - a tone disabler
  - an A- or μ-law compander



- The adaptive filter maintains a model of the impulse response of the echo path using the residual echo signal as feedback for adjusting its coefficients
- Using the R_{in} signal as a reference, the model is used to calculate an estimate of the echo which is then subtracted from the echo-carrying S_{in} signal

Echo Cancellation Algorithms

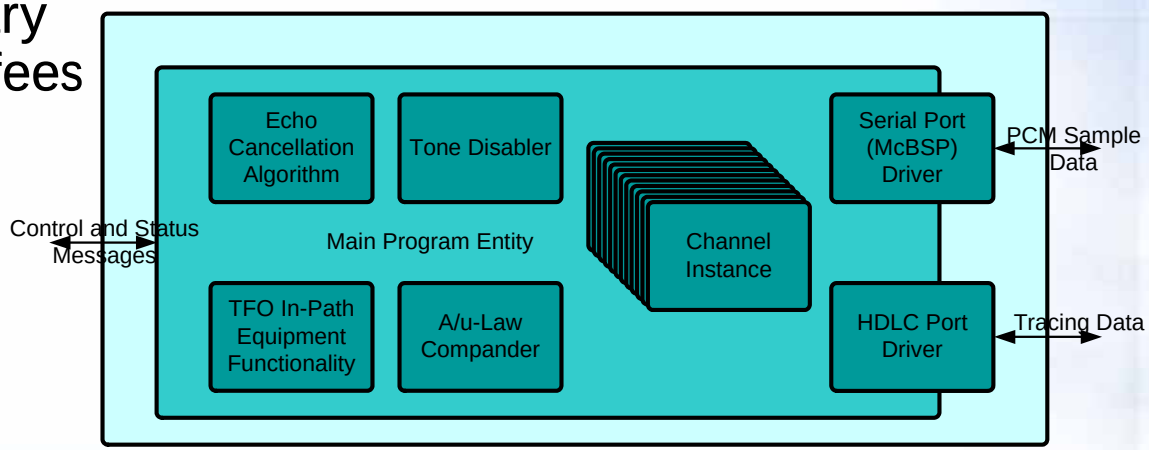
- Least Mean Square (LMS) algorithm
  - Utilized by most echo cancellers in their adaptive filters
  - Has numerous variants (normalized, sliding-window, etc.)
  - Starts with some initial value for the coefficient vector and then recursively updates it along the direction of the negative gradient of the squared error, i.e., the direction where the squared error reduces most rapidly
  - Fairly easy to implement in DSP software
  - Requires modest computational resources (complexity is O(N), where N is the number of taps in the filter)
  - Does not rely on the statistical properties of the input signal
  - Fairly good stability properties and robustness
- Recursive Least Squares (RLS) algorithm
  - Uses the information contained in all the past input samples to estimate the statistics (autocorrelation) of the input signal
  - Faster convergence speed and smaller final estimation error compared to the LMS algorithm
  - Computationally demanding (complexity is O(N^2))

Echo Canceller Hardware

- Echo canceller plug-in unit
  - Hardware board with connectors on its side
  - Is installed in a cartridge which, in turn, is installed in a cabinet of a network element (i.e., Nokia's MSC)
  - Incorporates 4 or 8 (depending on variant) TI TNETV3010 DSPs controlled by a host processor (AMD Am186ER microcontroller)
  - Contains additional logic circuitry including Field Programmable Gate Arrays (FPGAs) to implement PCM multiplexing and Ethernet connectivity (HDLC↔Ethernet frame conversion)
  - Provides external memory for the DSPs and the host processor
- External interfaces
  - Several 8.092 Mbit/s PCM lines for voice traffic
  - Additional PCM lines for control purposes
  - Interfaces for service terminal use, debugging, and tracing (RS-232, JTAG boundary scan, 10/100BaseT Ethernet)

Echo Canceller Software

- Echo canceller DSP software
  - Completely new software modules
    - TNETV3010 serial port driver (and other low-level device drivers)
    - Main program
  - Previously implemented, reusable software modules
    - Echo cancellation algorithm (LMS-based) implementation
    - Tone disabler
    - Tandem Free Operation (TFO) in-path equipment functionality
    - A/μ-law companding routines
  - No Real-Time Operating System (RTOS)
    - Not absolutely necessary
    - Would require license fees

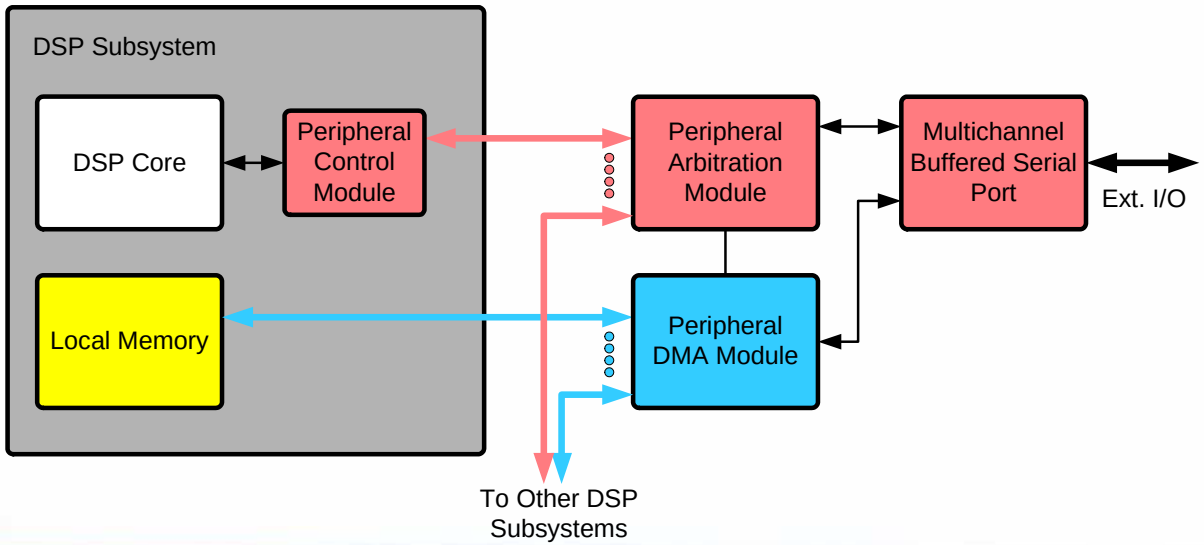


Development of the Serial Port Driver

- Requirements for the driver
  - Must provide an **Application Programming Interface** (API) in the form of data structure definitions and routines implemented in C
  - Must be able to configure the two serial ports of a TNETV3010 device independently
  - Must be able to individually enable/disable the channels of the serial ports (512 receive/transmit channels per serial port)
- Driver architecture
  - A mini-driver built on top of the Hardware Abstraction Layer (HAL)
  - The HAL is implemented in TI's Chip Support Library (CSL) which provides the basic data types, routines, and macro definitions for accessing the device-specific features of the DSP
- API
  - Initialization and configuration functions
  - Channel control functions
  - Other functions (e.g., enable/disable the digital loopback mode)

Development of the Serial Port Driver, continued

- Implementation issues
  - In the TNETV3010, the serial ports are **shared peripherals**
    - Access to the shared peripherals is arbitrated between the DSP cores by using a **peripheral arbitration module**
    - The peripheral registers cannot be accessed directly by the cores, but rather by using a **peripheral control module** with proxy registers mapped into the I/O address space of the core
      ⇒ programming the peripherals is much more challenging than with single-core DSPs



Conclusions

- Hardware design and performance issues aside, the multicore architecture had considerable implications on the software development
- One of the most complex issues is accessing the shared on-chip peripherals
  - Multicore DSPs incorporate cumbersome mechanisms such as peripheral control modules with proxy registers to access the peripherals
  - In the worst case, such proxy mechanisms may even be multi-level

Future Work

- Theoretical or empirical analysis of the performance of adaptive filtering algorithms used in echo cancellation was outside the scope of the thesis
- A lot of room for further study
  - Implementing the various algorithms in DSP software
  - Optimizing the algorithms for performance
  - Conducting measurements of convergence time, final error, and computational load
- Algorithms other than LMS may also prove to be feasible
  - As the processing power of DSPs continues to increase rapidly, it may well be that algorithms which have traditionally been considered too expensive in terms of computational resources gain ground in the future
- The developed serial port driver has good potential for reuse in future projects which utilize the same DSP
  - The implementation is not tied to other parts of the echo canceller software or any operating system

Questions?

Thank you!