

# Leveraging an Open-Source VoIP Application to Speed VoIP Product Development

Scott Kurtz  
Adaptive Digital Technologies



# Agenda

- Overview – What are we trying to accomplish? Leverage Technology
- Brief view of traditional telecom
- Voice-over IP (VoIP)
- The Open Source Business Model
- Introducing G.PAK™: An open source VoIP DSP software application
- Hands-on Examples

Minds in Motion

# Leveraging Technology

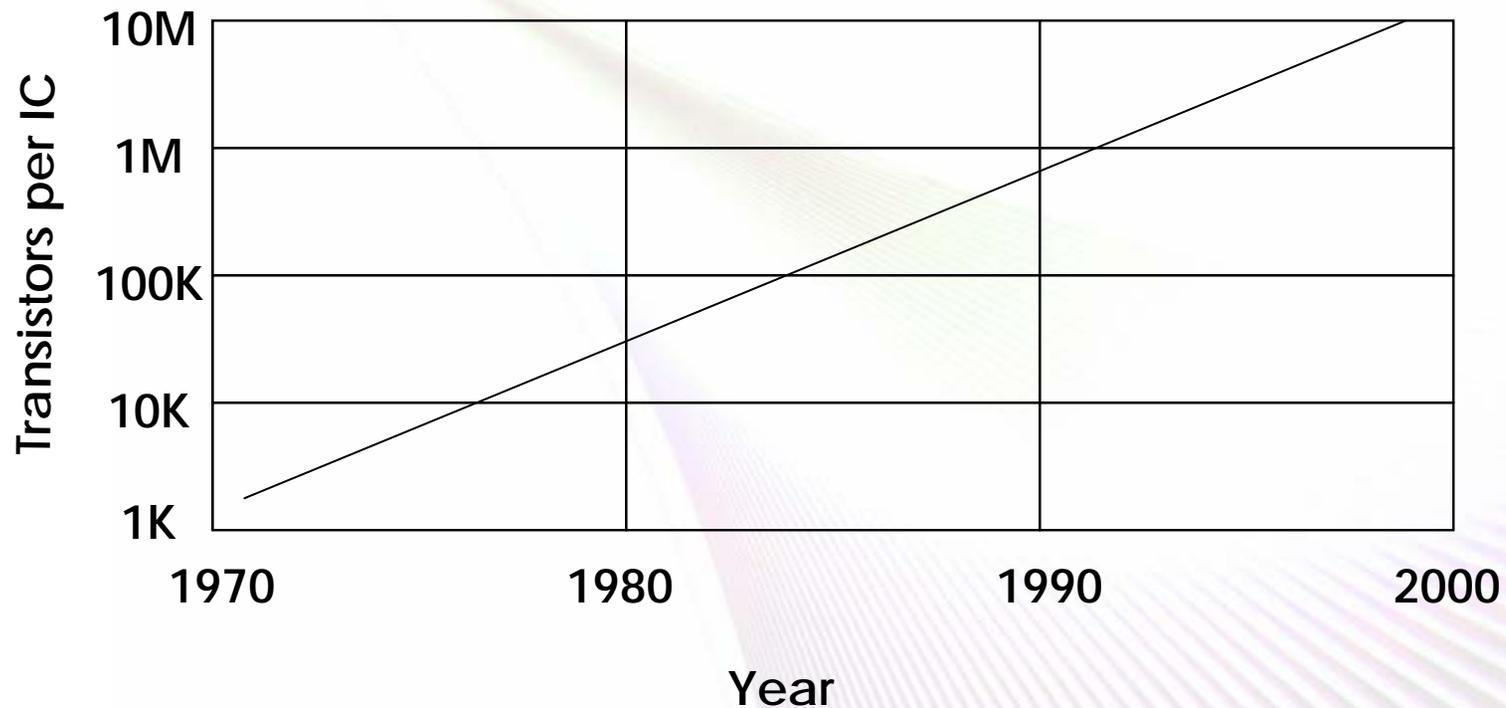
- It is difficult enough to keep up with the pace of complexity in technology, let alone make use of it.
- How do we make use of very complex technology?
  - We stand on the shoulders of those who preceded us.

Minds in Motion

# Overview

Minds in Motion

# Semiconductor Complexity



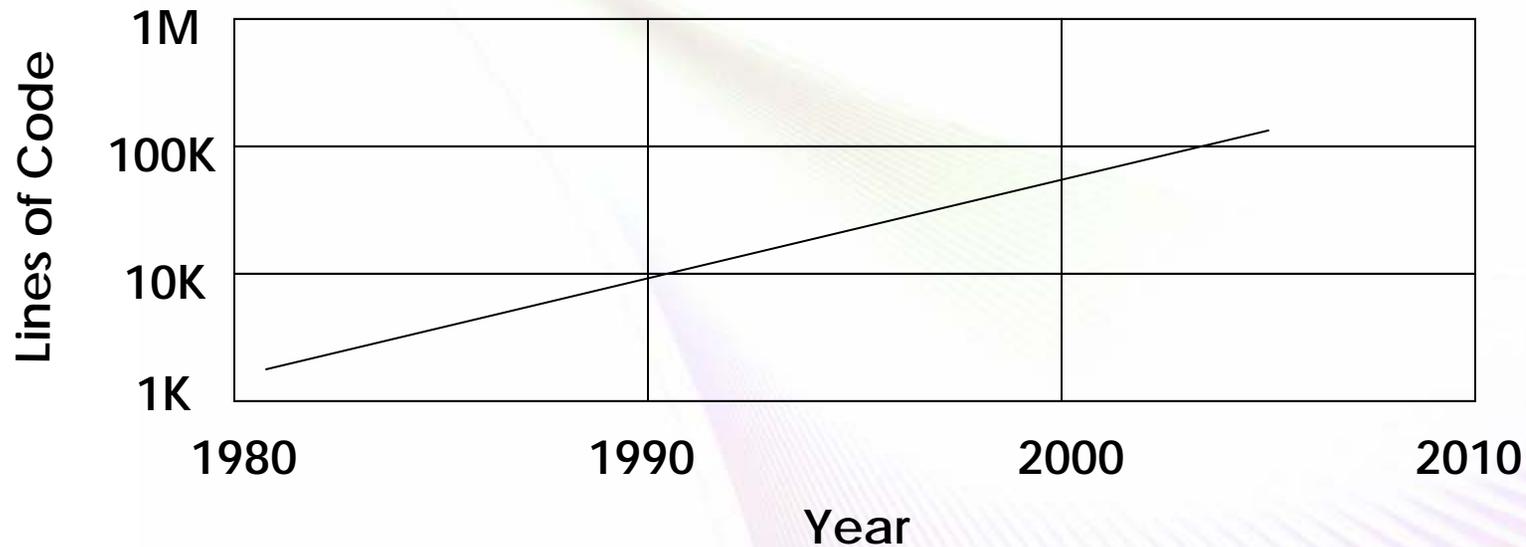
Minds in Motion

# Semiconductors: Managing Complexity

- IC Design
  - 1960s and 1970s: transistors
  - 1970s and 1980s: gates
  - 1990s-Present: VHDL, functional blocks, cores, etc.
- Can you imagine designing a DSP or Pentium chip from transistors?

Minds in Motion

# DSP Application Complexity



Minds in Motion

# DSP Application: Managing Complexity

- DSP Software
  - 1970s: Hardware Implementations and Microcode
  - 1980s: Assembly Language Coding
  - 1990s: High-Level Language (“C”) Coding, Libraries, Operating Systems
  - 2000s: Application-Specific Signal Processors (ASSP)
- Can you imagine writing a complete VoIP DSP application in Microcode or assembly language code?

Minds in Motion

# DSP Applications: Today's Options

- Start From Scratch
- Leverage DSP Libraries
- Leverage Operating System
- Leverage Software Framework
- Leverage ASSPs
- Leverage Open Source Software

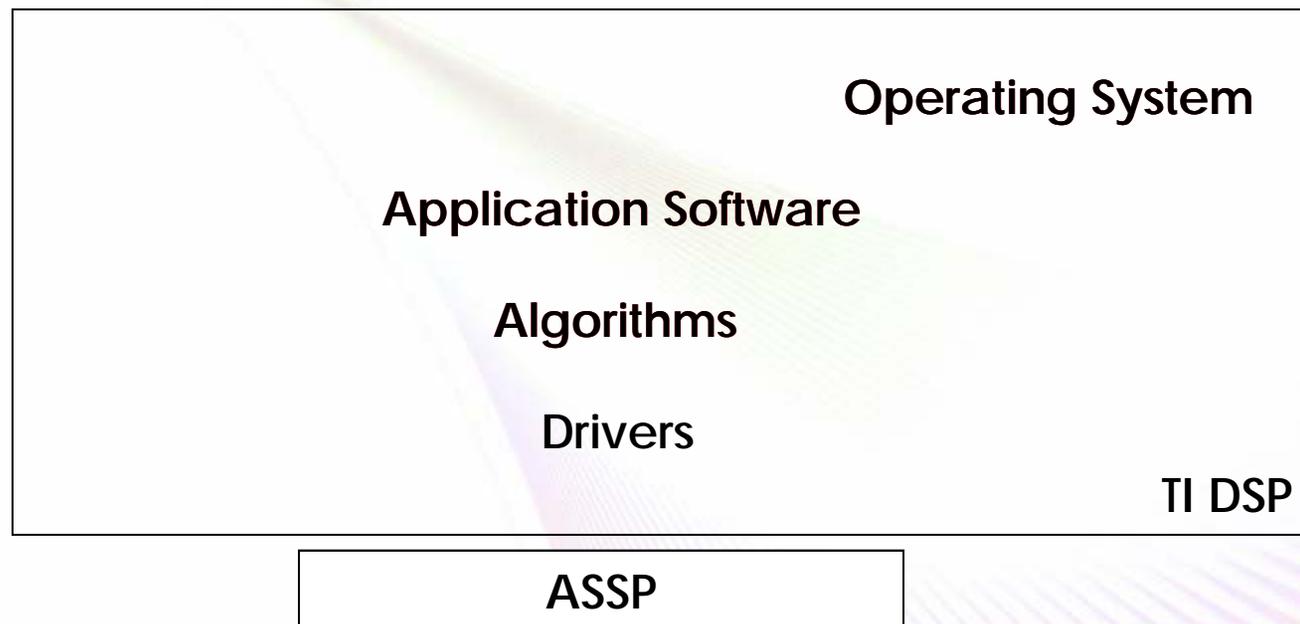
Minds in Motion

# Leveraging Proven Algorithms

- Voice Quality – the bottom line
- Field hardened algorithms
- Voice Quality related algorithms:
  - Echo Cancellation
  - Conferencing
  - Automatic Gain Control (AGC)
  - Packet Loss Concealment (PLC)
  - Discontinuous Transmission (DTX)

Minds in Motion

# DSP Applications: Today's Options



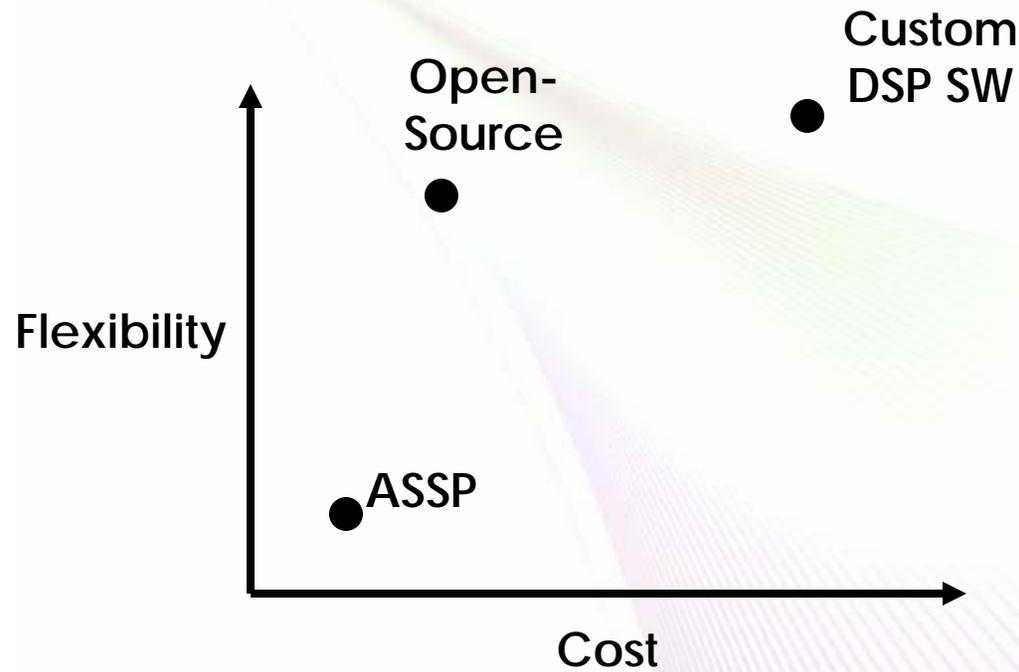
Minds in Motion

# Introducing Open-Source DSP

- Open-Source DSP lies between the ASSP and custom development.
- With open-source, you can
  - Leverage existing software
  - Focus on your application
  - Keep control of your design
  - Hit the market quickly with your product

Minds in Motion

# Cost and Flexibility



What about opportunity cost?

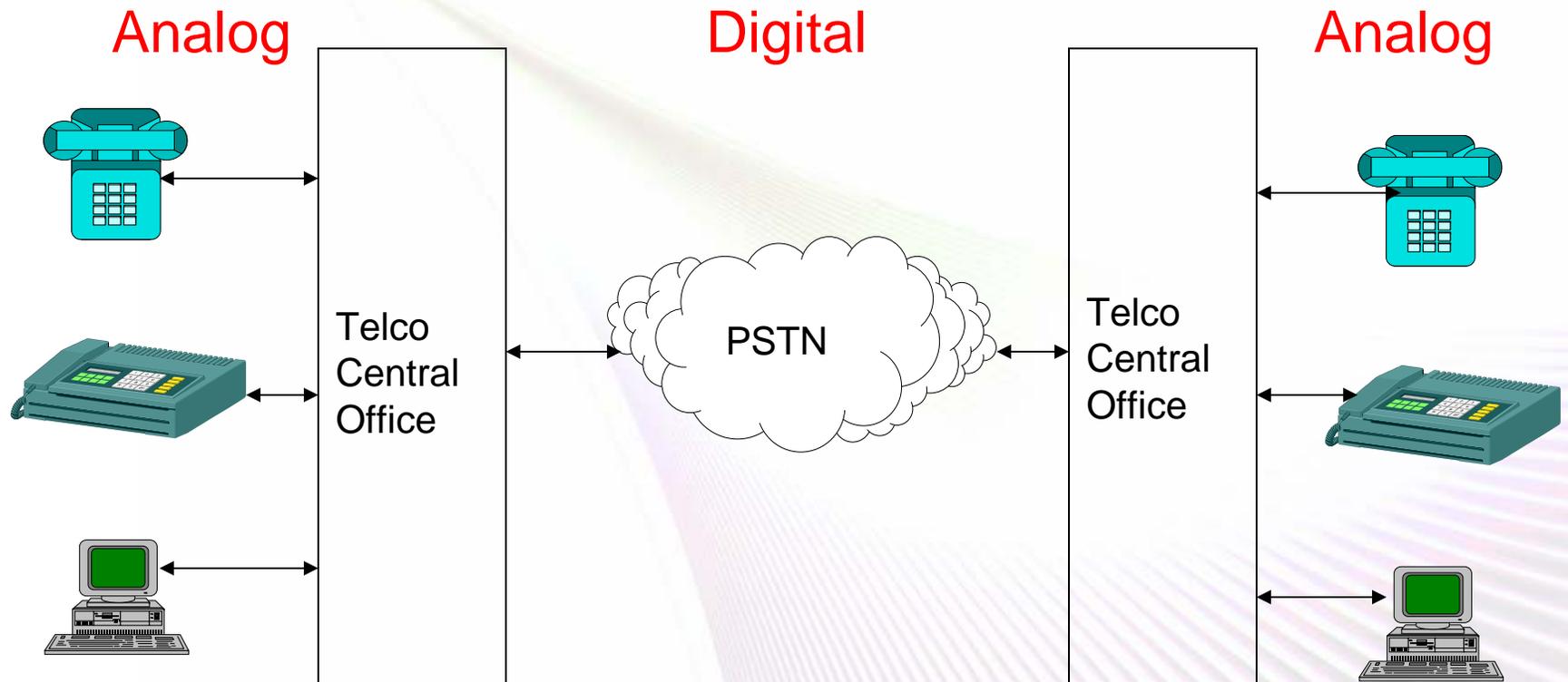
Minds in Motion

# Brief View of Traditional Telecom

Minds in Motion

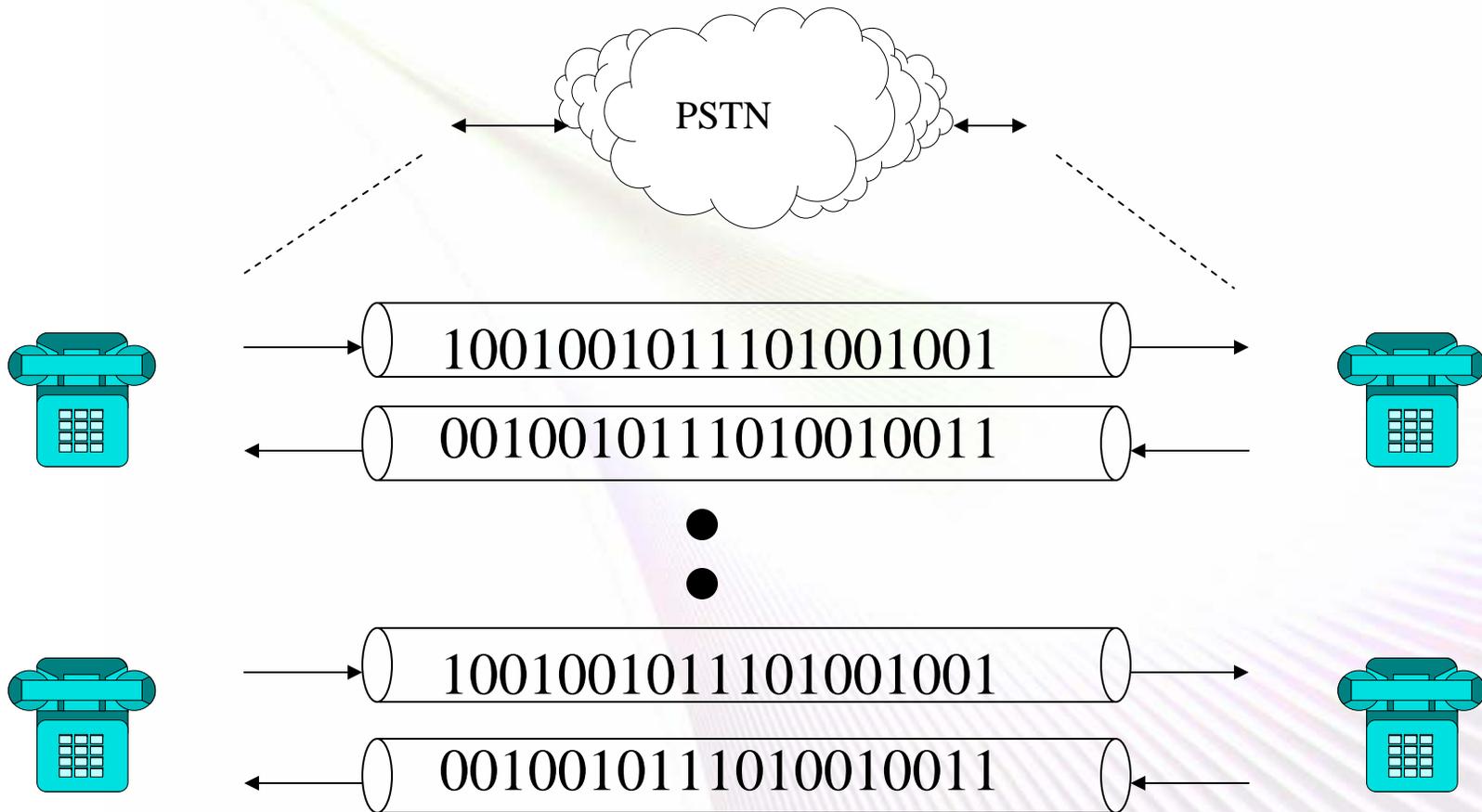
# The PSTN

(Public Switched Telephone Network)



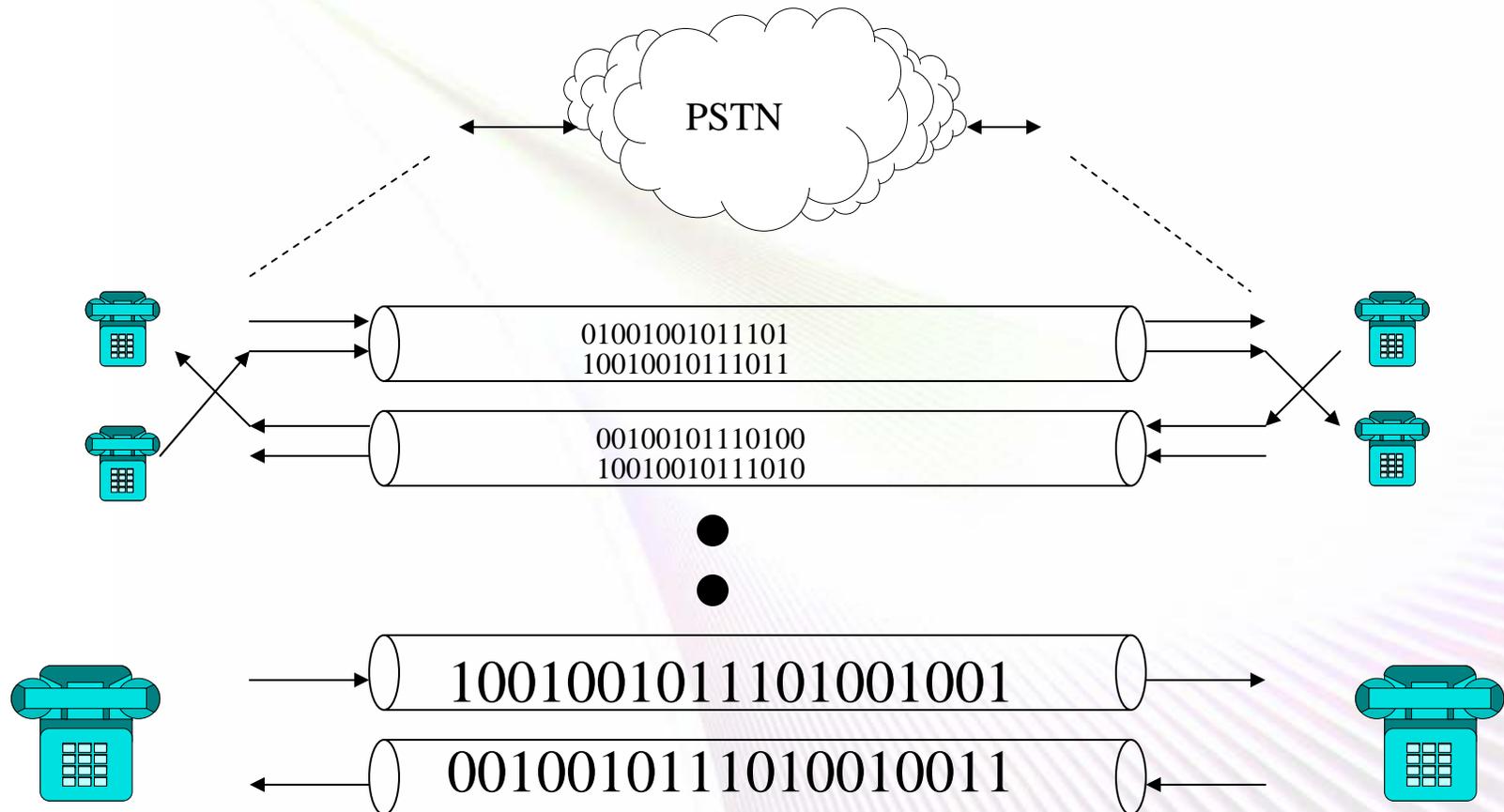
Minds in Motion

# Digital Channels



Minds in Motion

# Compressed Data



Minds in Motion

# A Few Voice Compression Standards

Product	MOS	Rate(s)	MIPS
G.711	4.1	64	0.36
G.726 (Low Mem)	3.9 (32 kbps)	16, 24, 32, 40	11.0
G.726 (Low MIPS)	3.9 (32 kbps)	16, 24, 32, 40	9.8
G.729	3.9	8	21.5
G.729B	3.9	8	22.2
G.729A	3.7	8	11.4
G.729AB	3.7	8	12.2
G.728	3.7	16	35.0
G.723.1	3.9/3.6	6.3/5.3	16.9/16.5
G.722		48, 56, 64	15.6

Minds in Motion

# PSTN Functions

- PSTN Functions
  - Call Setup/Teardown
  - Pass Signaling (next slide)
  - Pass Voice, Fax, Modem Signals

Minds in Motion

# Signaling

- Dual-Tone Multi-Frequency (DTMF) Dialing (or Touch-Tone Dialing)
- Pulse Dialing
- Hook-Switch Signaling
- Ringing
- Hook Flash
- Disconnect
- Many others

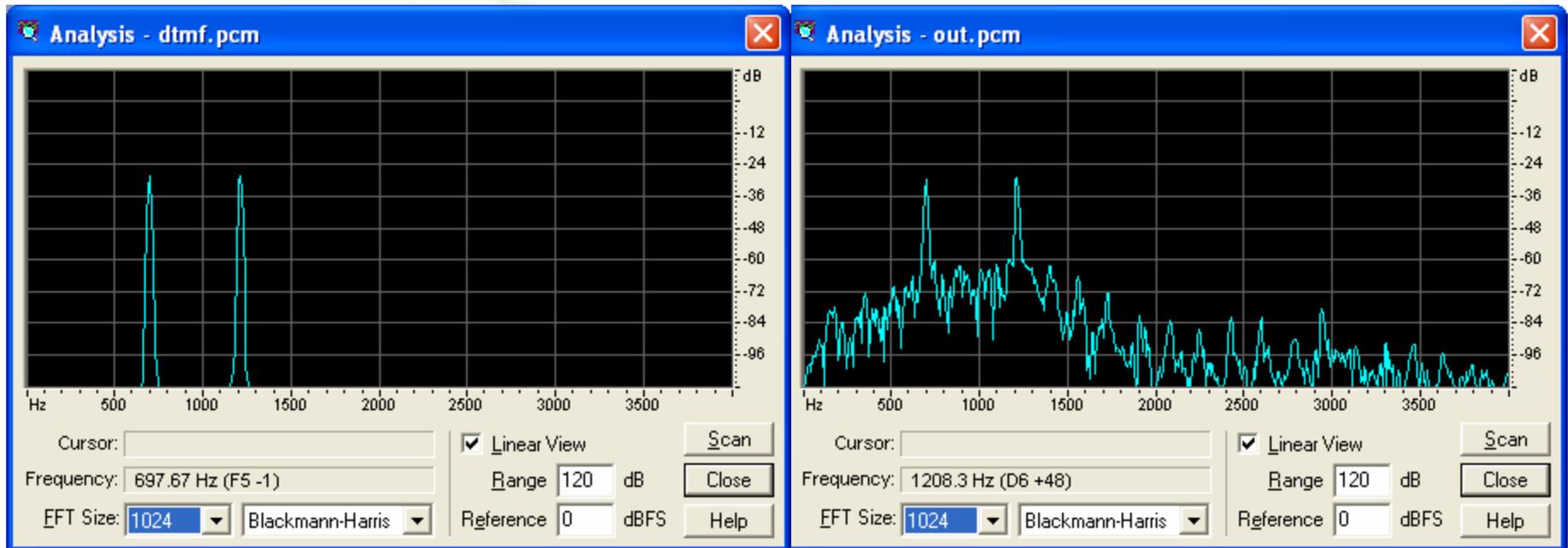
Minds in Motion

# Transition to VoIP

- Signaling – must be captured and sent in packets. Why?
- Tonal Signaling – may need to be captured and sent in packets. (Next Slide)
- Modem Signals – may need to be relayed (T.38 Fax Relay)
- Voice – G.711 PCM or further compression

Minds in Motion

# Need for Tone Relay



FFT Before Speech Compression

FFT After G.729A Compression (8 kbps)

Minds in Motion

# Tone or Fax Relay



PCM = Pulse Code Modulation – 8000 Samples/Sec, 64 kbps

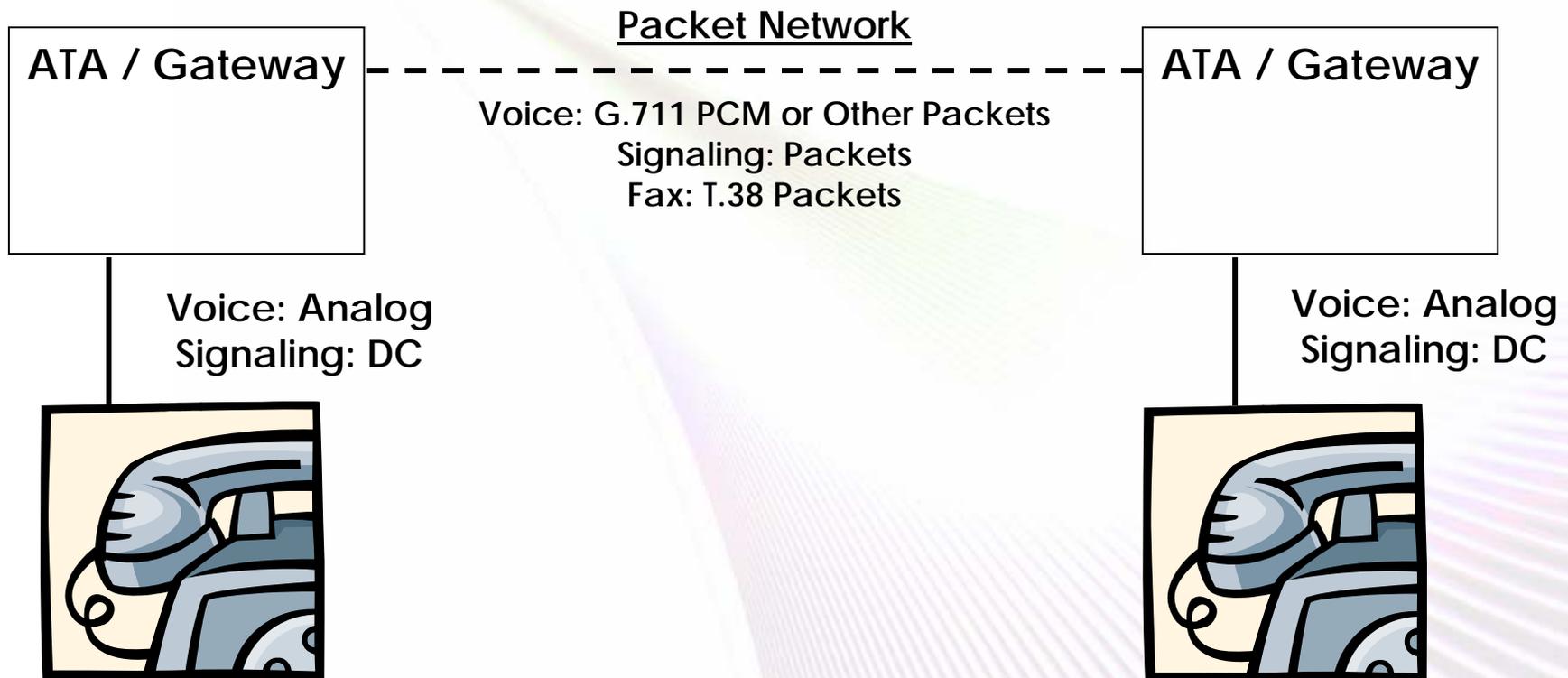
Relay: Tone or Fax

Minds in Motion

# VoIP

Minds in Motion

# VoIP



Minds in Motion

# Voice Over Data Networks

- Why carry voice over data networks?
- PSTN vs. Data Networks
- Integrating voice into data networks

Minds in Motion

# Traditional Voice And Data Networks

## PSTN

- Circuit Switched
- Very Low Delay
- Continuous Data Flow

## Data Network

- Packet Switched
- Higher Delay
- Bursty Data Flow
- Throughput and delay depend upon network activity

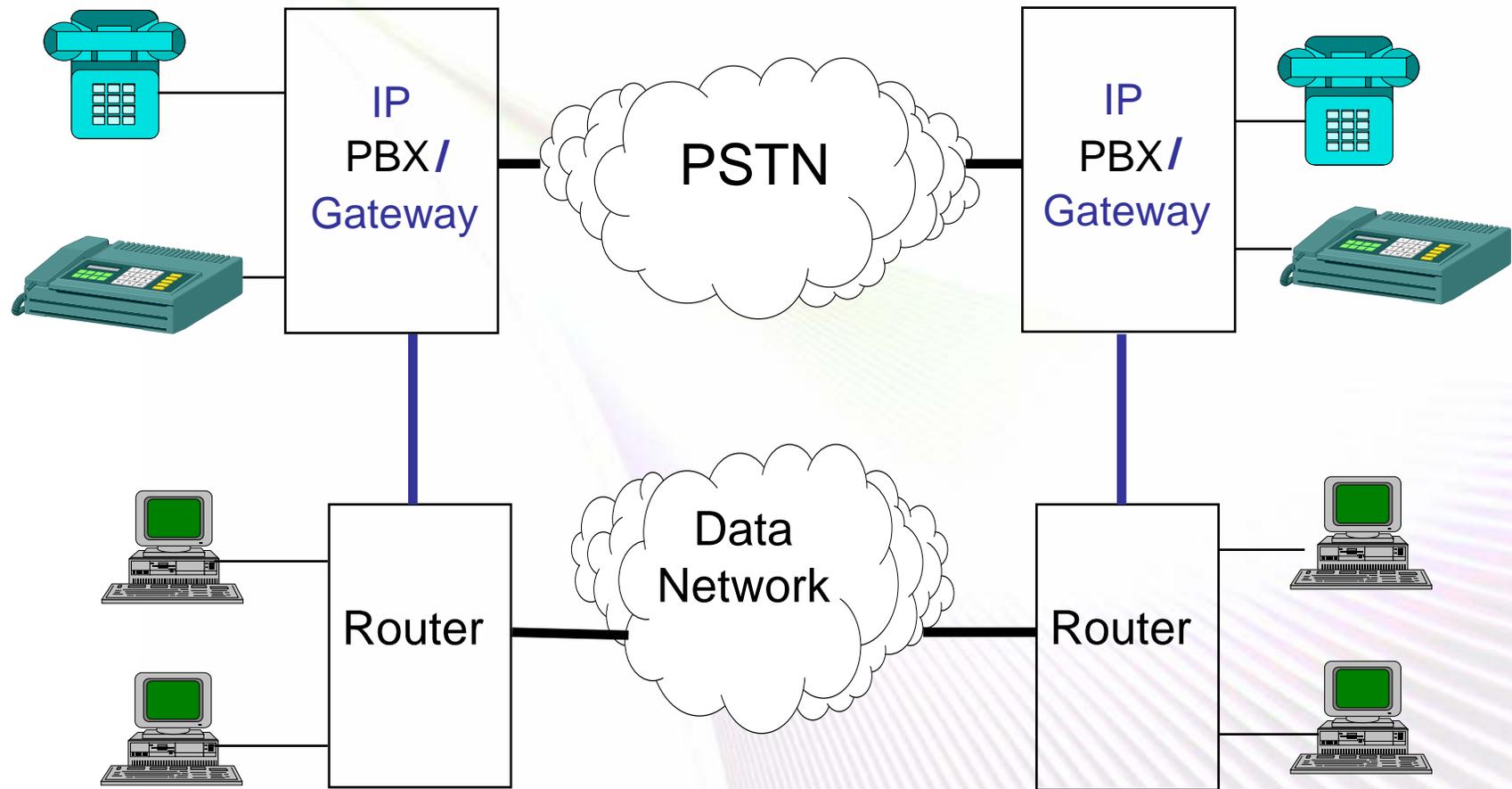
Minds in Motion

# Traditional Voice and Data Networks

- Circuit Switching - A circuit is dedicated to a specific communication link for the duration of the call.
- Packet Switching - Packets of data sent from user A to user B are not necessarily carried on the same physical circuit; allows for sharing of circuits.

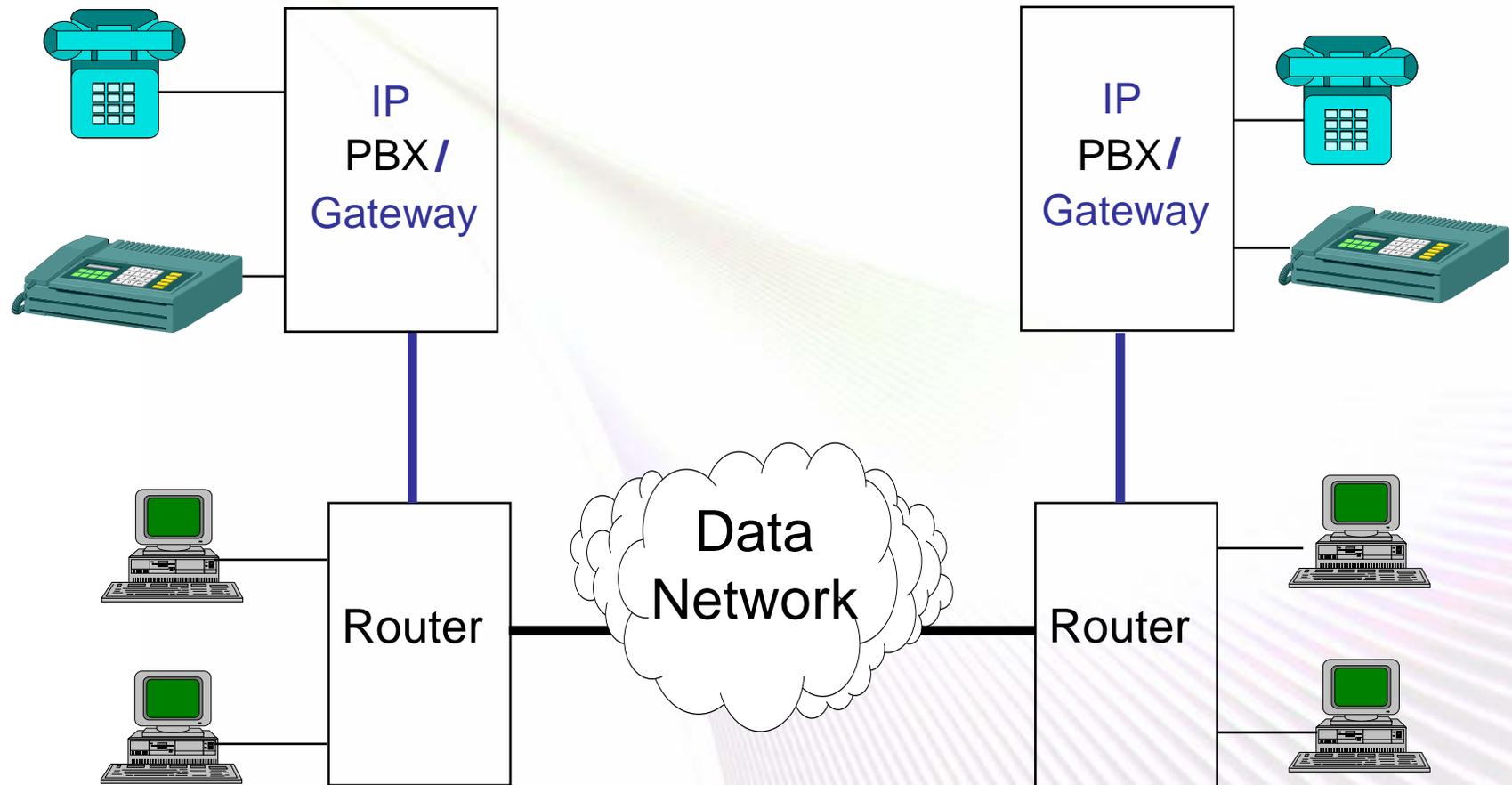
Minds in Motion

# Voice Over Data Network



Minds in Motion

# Voice Over Data Network



Minds in Motion

# Why Carry Voice on Data Networks?

## Advantages

- Share Hardware Costs
- Share Access Costs
- Reduce Long Distance Charges

## Disadvantages

- Network Impairments
- Security?

Minds in Motion

# Issues With VoIP

- Types of errors

- Bad packet received
- Missing packet
- Packet received out of order

- Recover Mechanisms

- ACK/NACK
- Sequence Numbers
- Packet Retransmission
- Timers
- Drop packet on floor

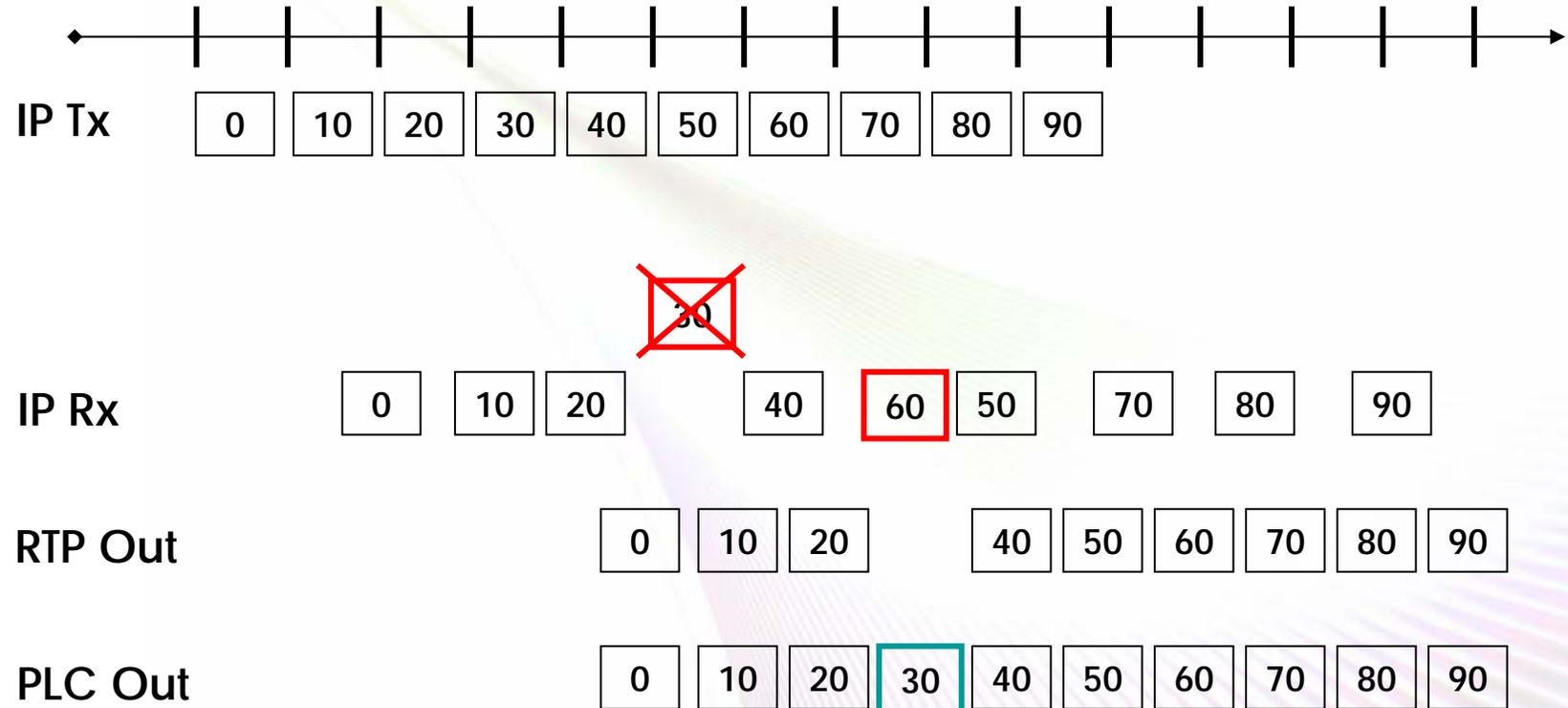
Minds in Motion

# Error Recovery – Why Not?

- By the time an error is detected and recovered, it's too late to play out the voice.
- RTP (Real-time Transport Protocol) **does not** implement error recovery.
- RTP **does** handle re-sequencing out-of-order packets and detecting lost or late packets
- RTP may be used with diversity

Minds in Motion

# Packet Transmission Errors



# Integrating Voice into Data Networks

- Traditionally, voice has been transmitted over circuit switched networks with constant data rate and delay.
- Data networks introduce many undesirable characteristics that need to be addressed:
  - Blocking
  - End-to-end delay
  - Delay jitter
  - Packet Loss
  - Header overhead
  - Throughput

Minds in Motion

# Integrating Voice into Data Networks

- Desired Features for Packet Voice Implementation
  - Compression
  - Silence compression (a.k.a. DTX)
  - Jitter Compensation
  - Packet Re-sequencing
  - Packet Loss Concealment
  - Transmission of Signaling
  - Echo Control

Minds in Motion

# VoIP Summary

- Data networks are not inherently suited to voice transmission. Data networks are packetized and bursty in nature. Voice signals are more continuous in nature.
- Special care must be taken when integrating algorithms to minimize the effects of the network.

Minds in Motion

# The Open Source Business Model

- Open Source does not necessarily mean free software
- GPL (GNU General Public License)
  - The most widely known open-source license.
- Other Open-Source Licenses

- |          |              |
|----------|--------------|
| Academic | Microsoft®   |
| Apache   | MIT          |
| Apple®   | Mozilla®     |
| BSD      | NASA         |
| CPL      | Nokia®       |
| Eclipse™ | Python®      |
| Intel®   | RealNetworks |
| Lucent   | Sun™         |
|          | Many More    |

Minds in Motion

# GPL - The Up-Side for a Vendor

- Market Penetration
- Become the de facto standard
- High Visibility
- Sell additional services
- Sell associated hardware
- License associated software
- Many customers will pay to license software to obtain support and to avoid GPL restrictions



Minds in Motion

# GL - The Down-Side for a Vendor

- Enables competitors
- Exposes intellectual property / trade secrets
- Code Contamination



Minds in Motion

# GPL - The Up-Side for a Vendor

- No need to reinvent the wheel
- Leverage existing code that hopefully already works
- Faster time to market



Minds in Motion

# GPL – The Down-Side for a Developer

- You must open up your source
- Enables competitors
- Exposes intellectual property / trade secrets



Minds in Motion

# GPL - The Up-Side for Users

- Free software is available



Minds in Motion

# GPL - The Down-Side for Users

- Free software comes with no warranty and no support. Proceed at your own risk.
- Free software tends to be light on documentation.



Minds in Motion

# G.PAK Source Licensing

- G.PAK license is not under GPL.  
Why Not?
  - G.PAK framework source code is not redistributable (protects our IP)
  - Derivative works are therefore not required to be redistributed (protects your IP)
- Unpaid license comes without support

Minds in Motion

# G.PAK Source License

- You **Must**:

- Use G.PAK source code in conjunction with Adaptive Digital's algorithm libraries

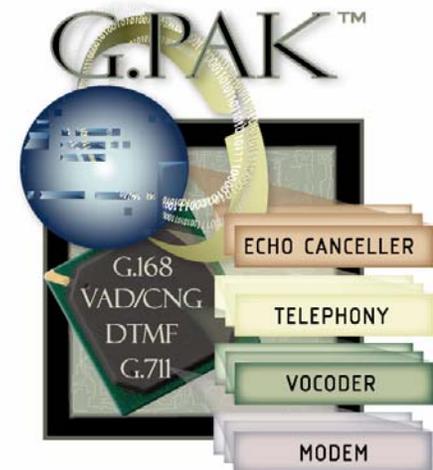
- You May **Not**:

- Redistribute G.PAK Source Code

Minds in Motion

# G.PAK: An Open Source VoIP DSP Framework

- G.PAK Overview
- G.PAK Channel Types
- Interfacing with G.PAK via host
- Host supplied routines
- G.PAK Internals
- Modifying G.PAK with new Vocoders



Minds in Motion

# G.PAK Features

- Most Standard Vocoders
- Multiple Channel Types (PCM-PKT, PCM-PCM...)
- Conferencing
- Echo Cancellation (PCM and Packet)
- Voice Activity Detection/Comfort Noise
- Tone Detection and Generation  
(DTMF, MFR1, MFR2 FORWARD, MF2 BACK, CALL PROGRESS)
- RTP/AAL2 Payload Formats
- Self Contained DSP Code
- Host API routines

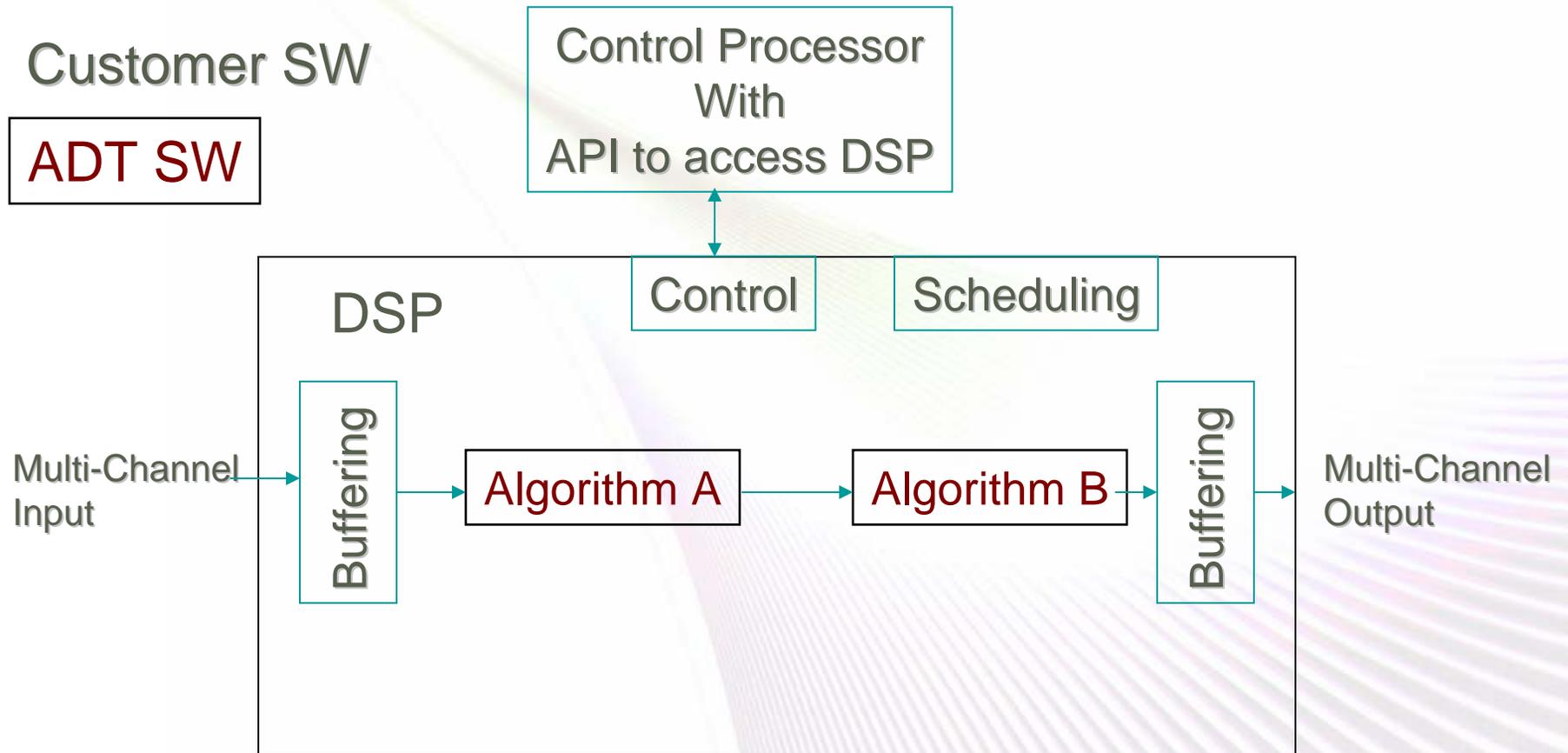
Minds in Motion

# G.PAK - Overview

- Algorithms vs. Turnkey
  - Algorithms – write your own framework  
Turnkey – No DSP Programming
  - Turnkey – Shorter Time To Market  
Lower NRE
  - Turnkey + Open Source – Increases flexibility and design/product control

Minds in Motion

# Algorithms vs. G.PAK

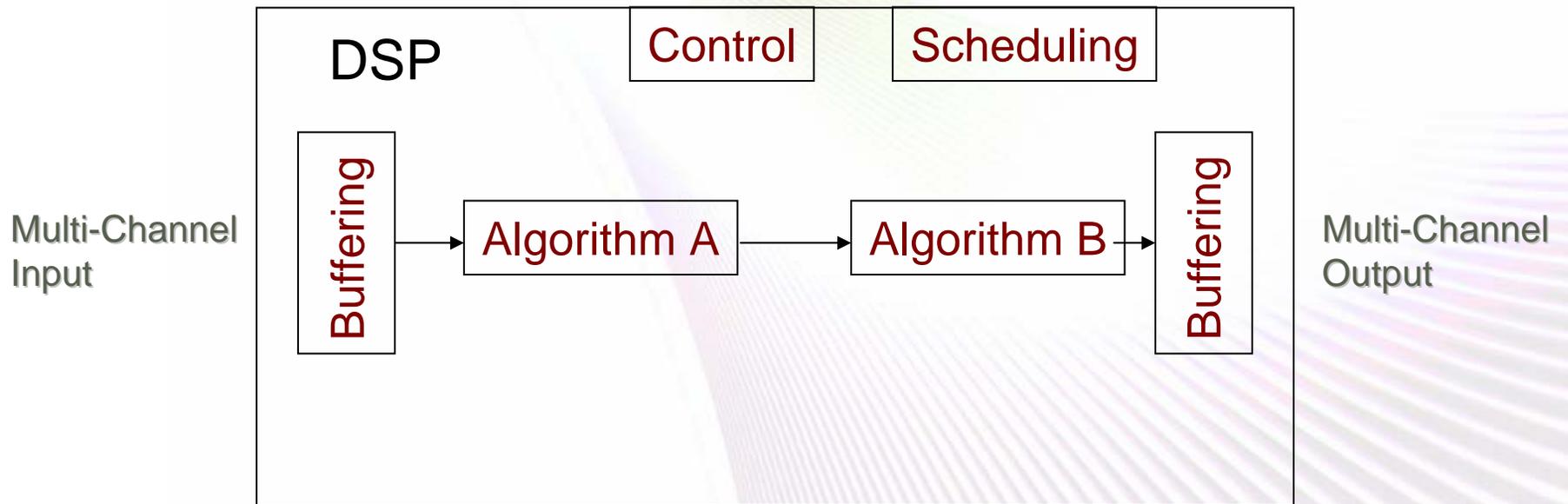


Minds in Motion

# G.PAK vs. Algorithms

Customer SW  
ADT SW

Control Processor  
With  
API to access DSP

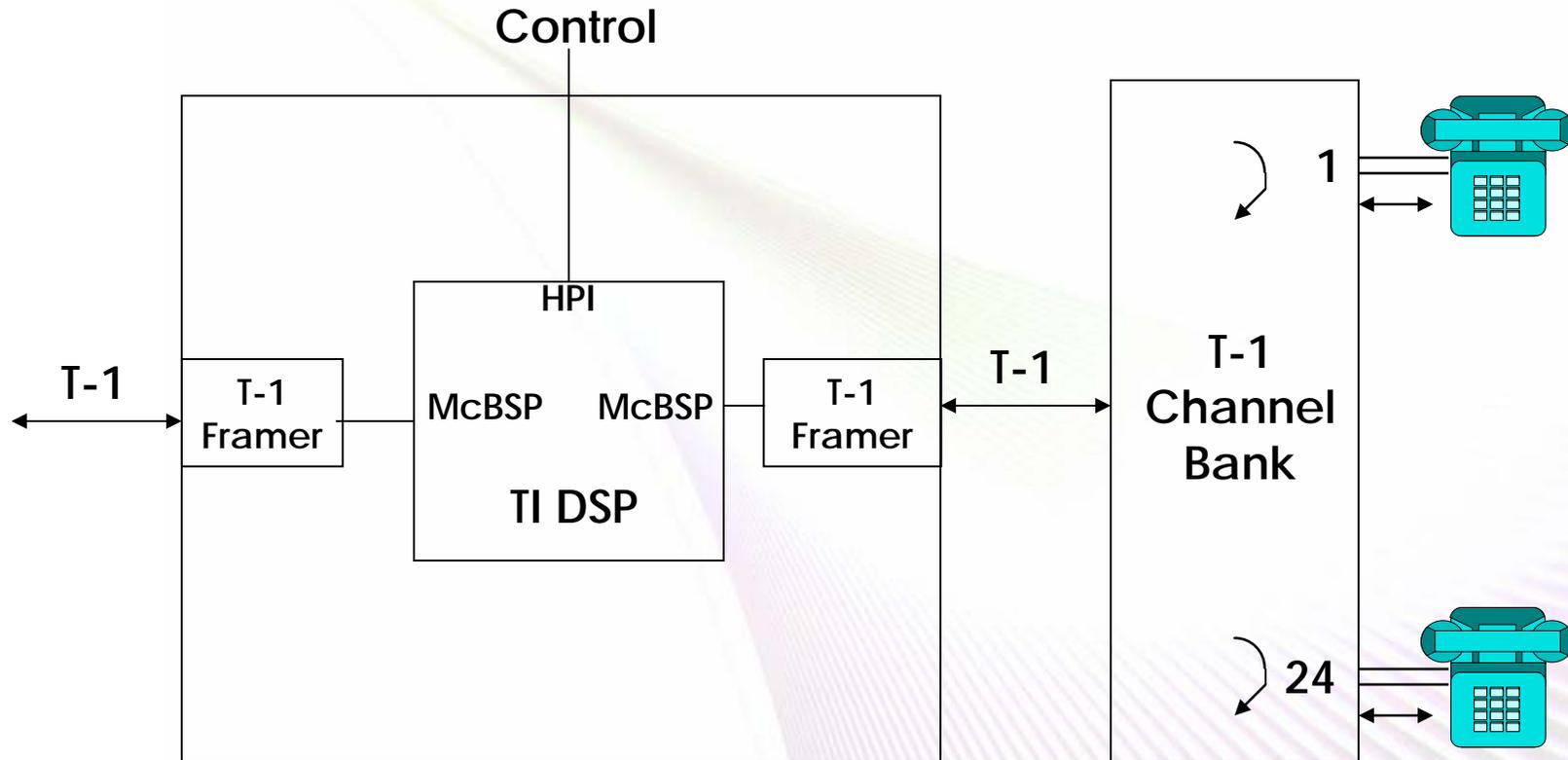


Minds in Motion

# G.PAK Channel Types

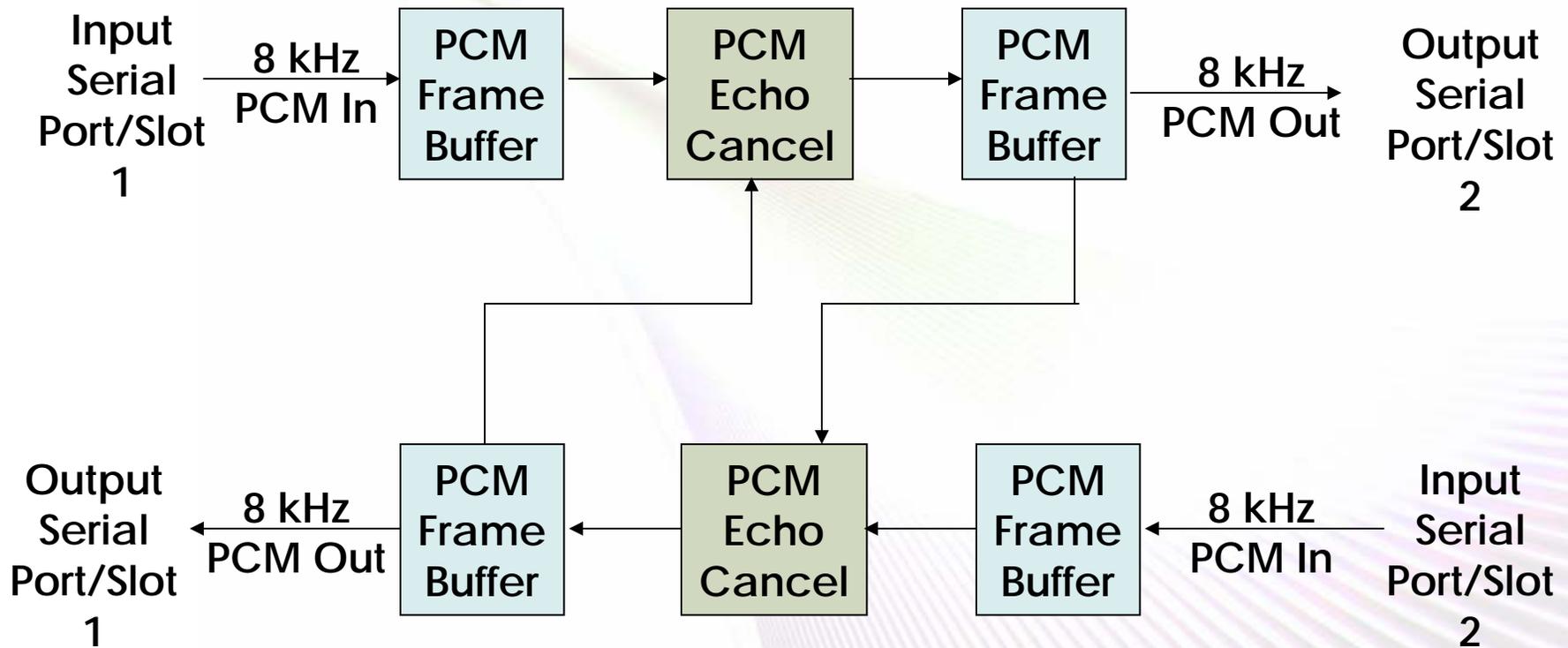
Minds in Motion

# PCM to PCM (TDM to TDM) Channel



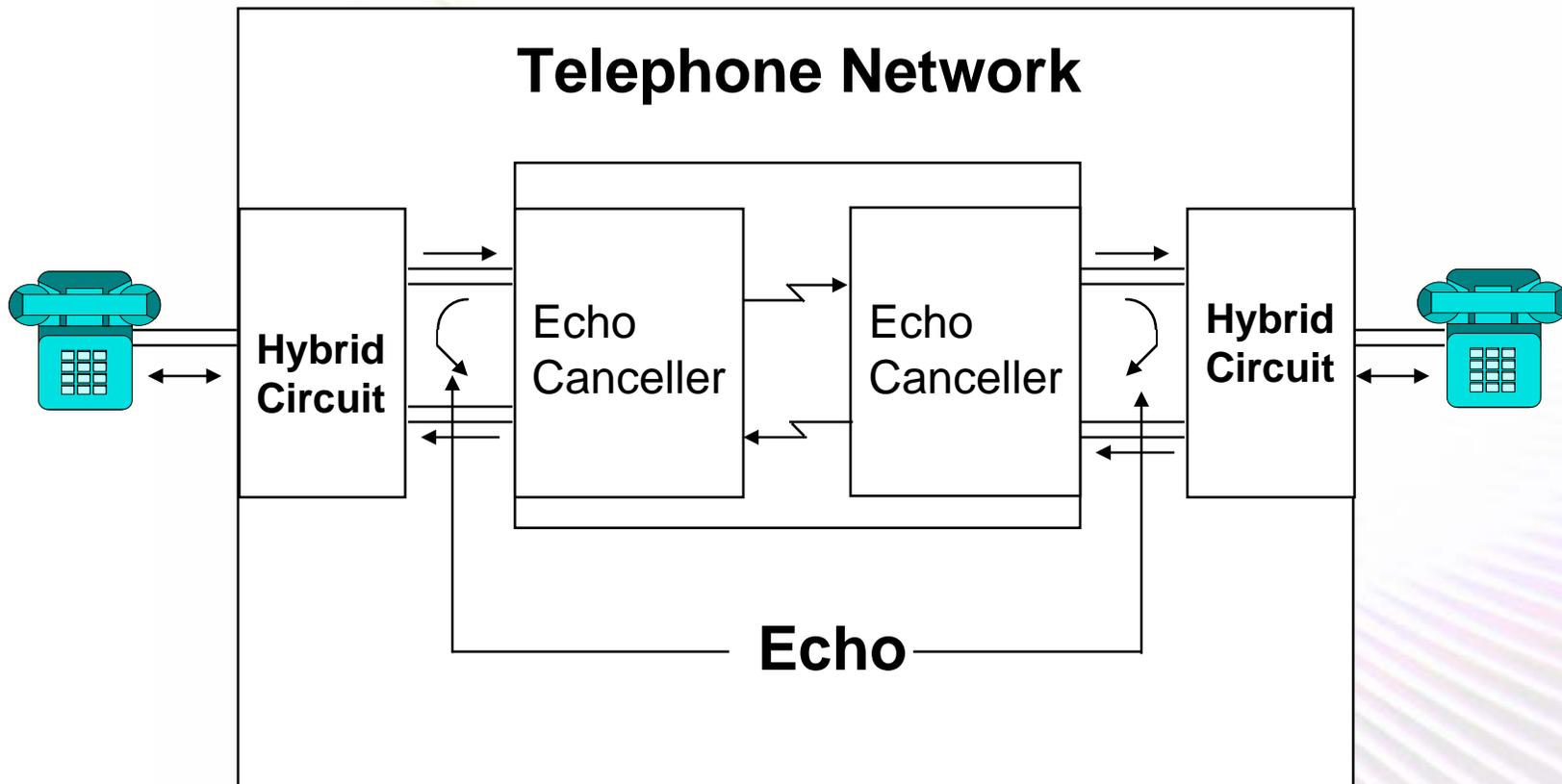
Minds in Motion

# G.PAK: PCM to PCM Channel



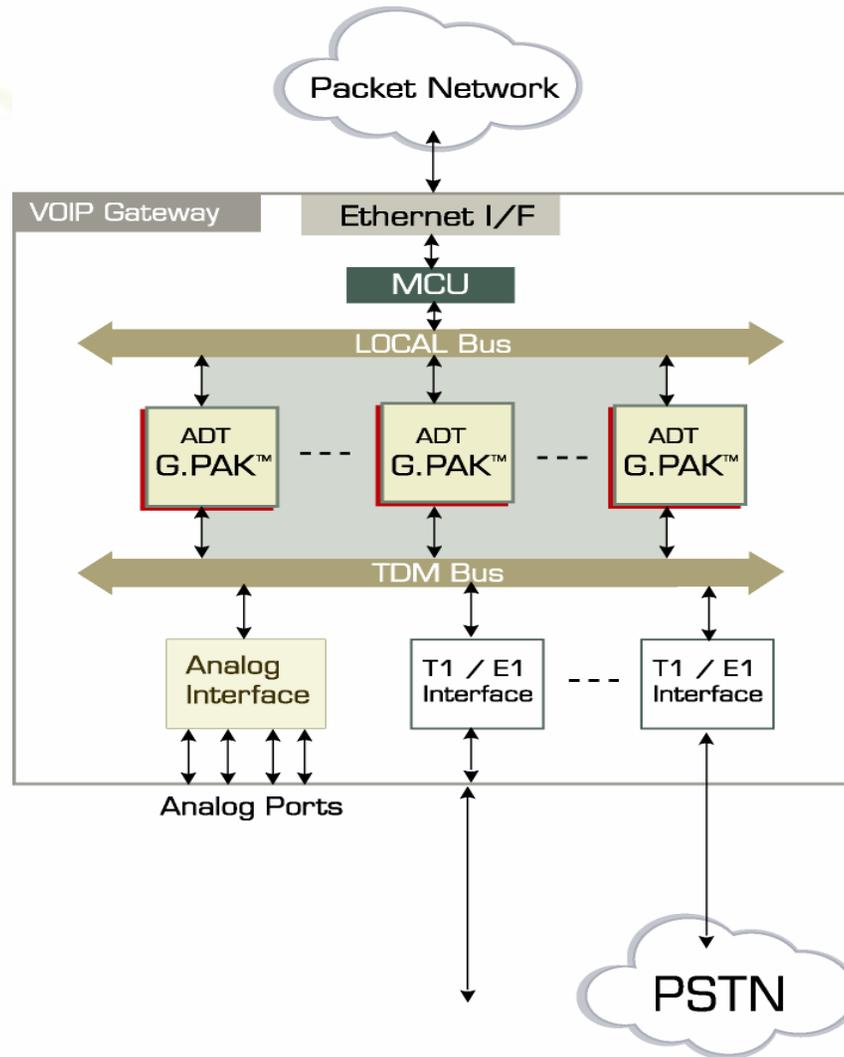
Minds in Motion

# Example: PCM Echo Cancellation



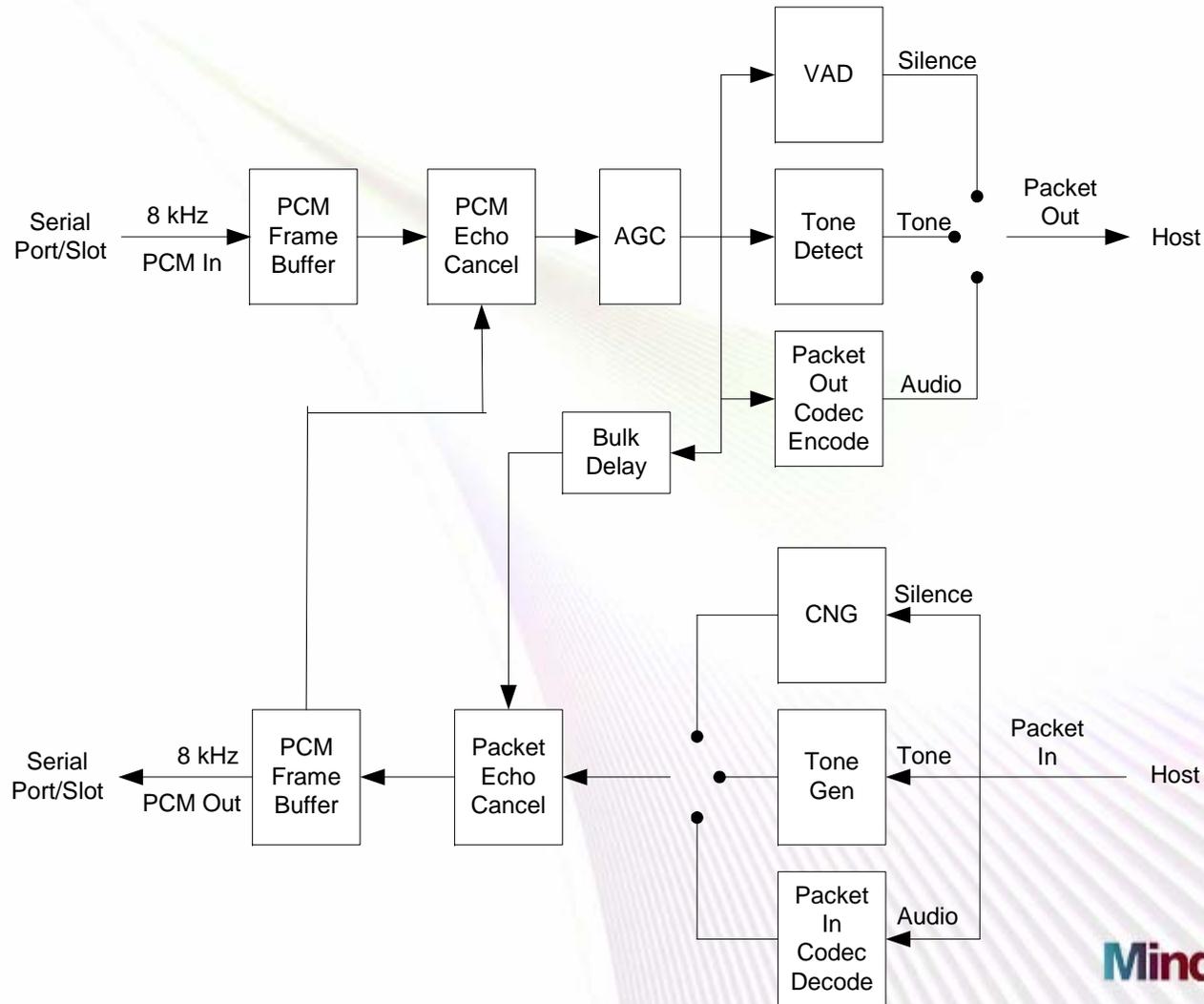
Minds in Motion

# PCM – Packet Channel: VoIP Gateway



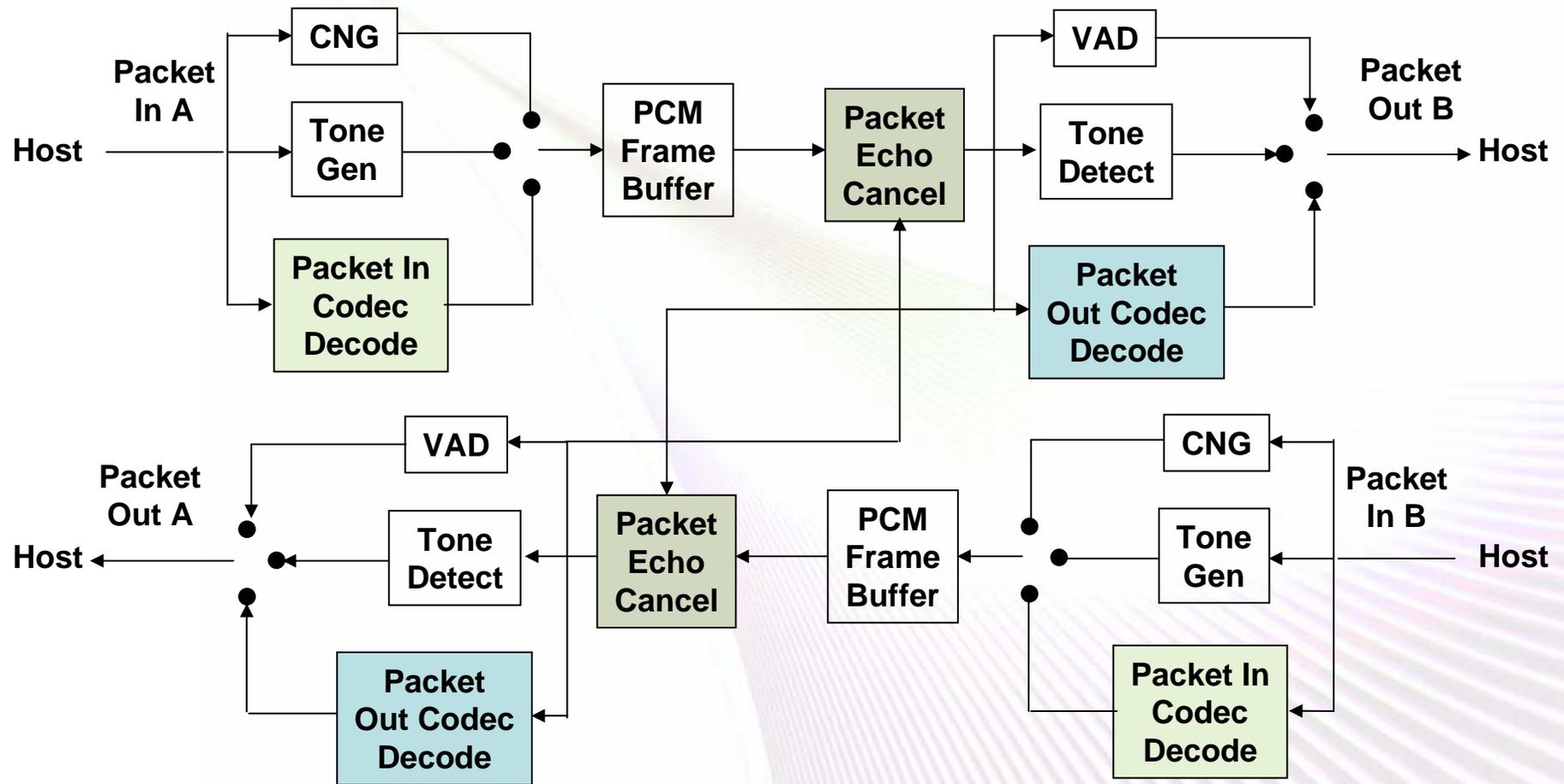
Minds in Motion

# G.PAK: PCM to Packet Channel



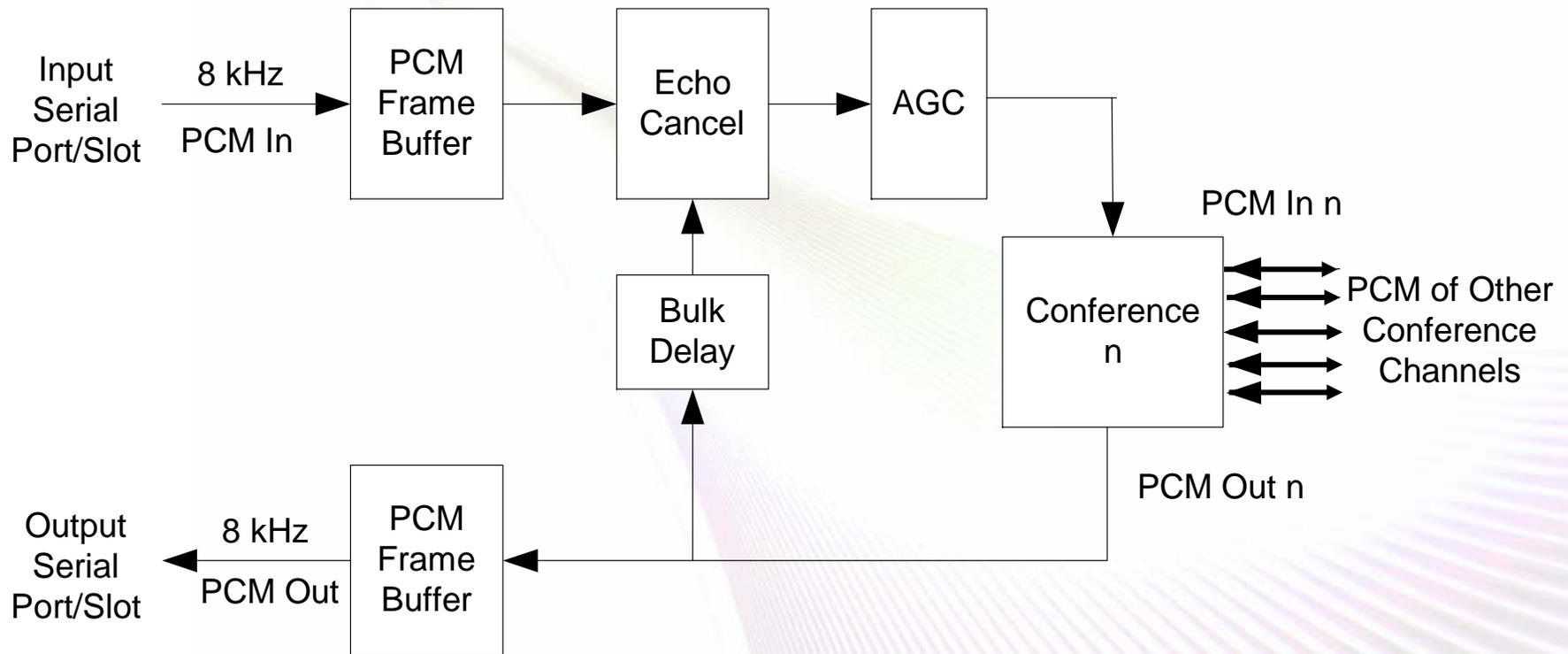
Minds in Motion

# G.PAK: Packet to Packet Channel



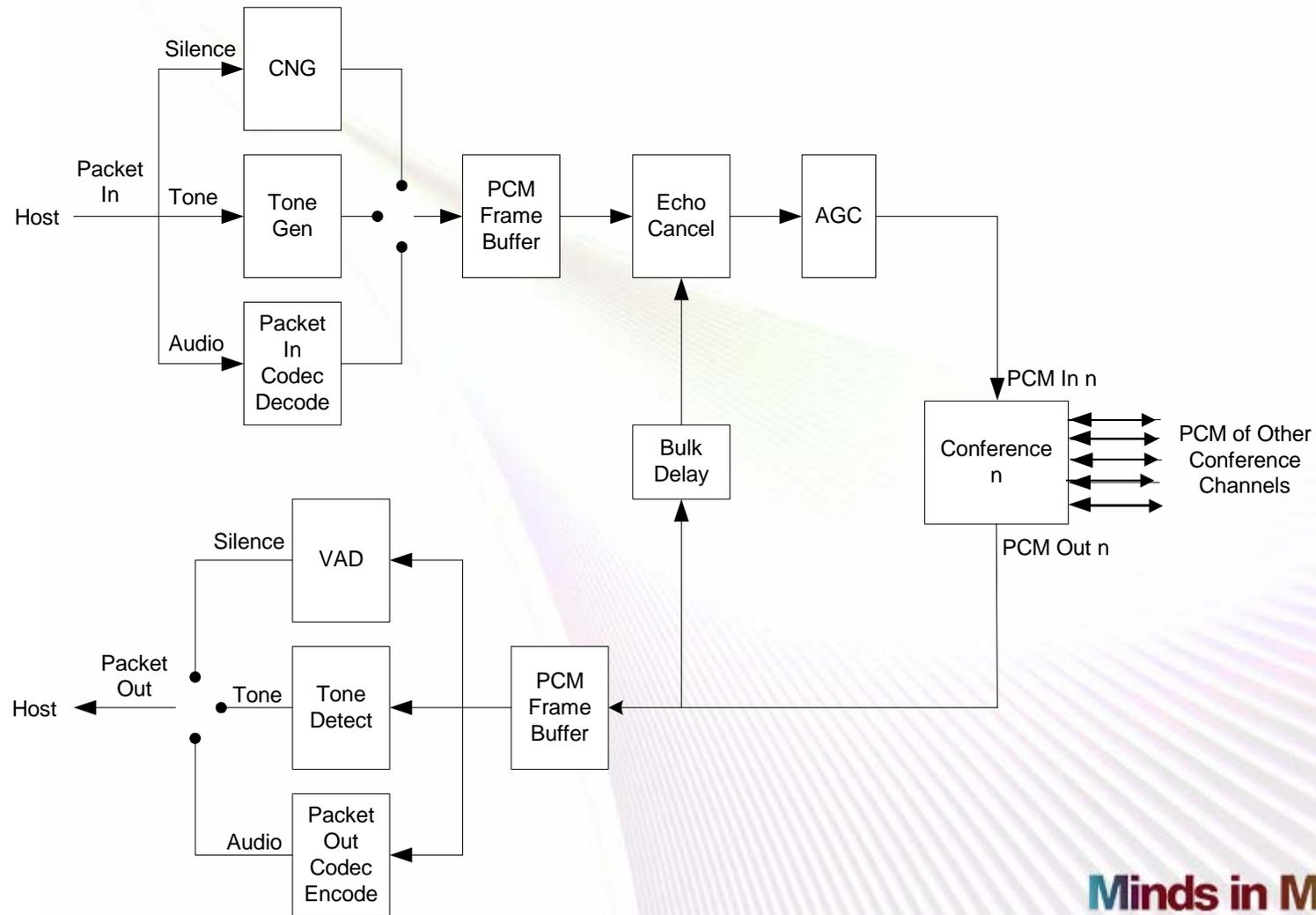
Minds in Motion

# G.PAK: PCM to Conference Channel



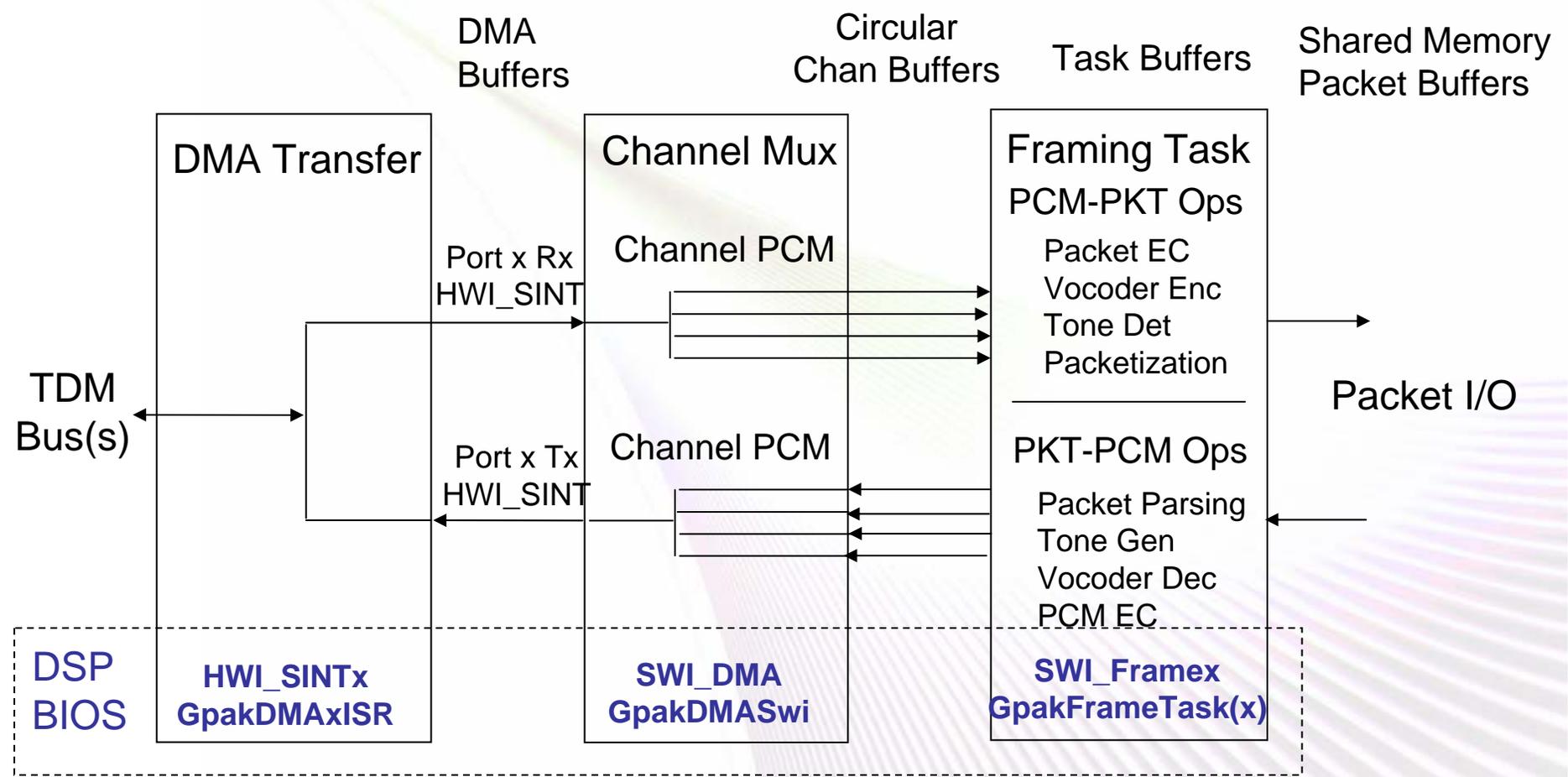
Minds in Motion

# GPAK: Packet to Conference



Minds in Motion

# G.PAK Data Flow



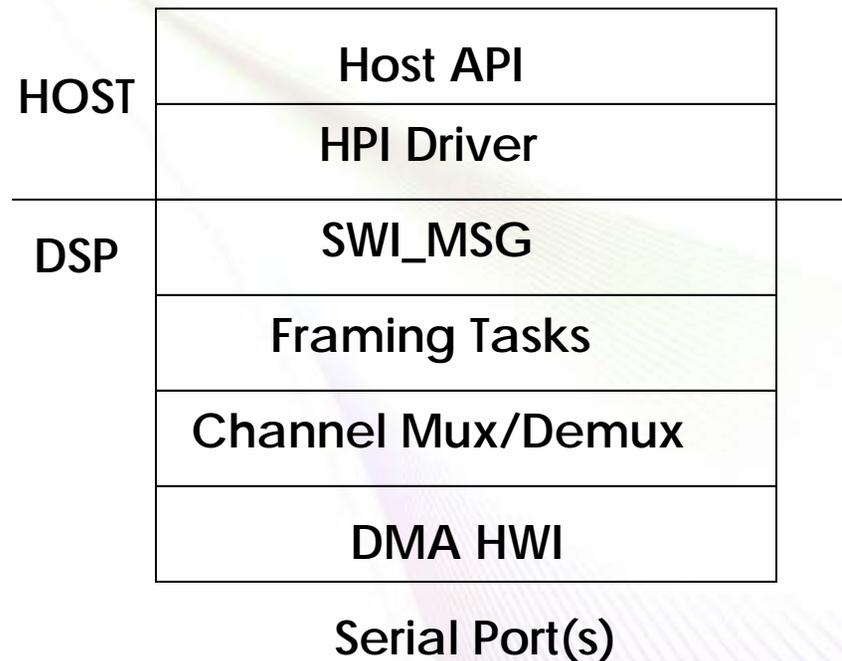
Minds in Motion

# Host Interface

- Configuration Messages
- Payload Transfer
- DSP Download
- CPU Usage

Minds in Motion

# Host Interface: Layers



Minds in Motion

# Configuration Messages

Functionality	G.PAK API Routine
Configure System Parameters	<code>gpakWriteSystemParms</code>
Configure Serial Ports	<code>gpakConfigurePorts</code>
Configure Conference	<code>gpakConfigureConference</code>
Configure Channel	<code>gpakConfigureChannel</code>
Tear Down Channel	<code>gpakTearDownChannel</code>
Read System Configuration	<code>gpakGetSystemConfig</code>
Read System Parameters	<code>gpakReadSystemParms</code>
Read Channel Status	<code>gpakGetChannelStatus</code>

Minds in Motion

# Payload Transfers

Functionality	G.PAK API Routine
Write payload to DSP	<code>gpakSendPayloadToDsp</code>
Read payload from DSP	<code>gpakGetPayloadFromDsp</code>

Minds in Motion

# Other API Calls

Functionality	G.PAK API Routine
Write image to DSP	gpakDownloadDsp
Read frame task statistics	gpakReadCpuUsage

Minds in Motion

# Host Supplied Routines

Routine	Functionality
Reads memory from specified DSP address	gpakReadDspMemory
Writes memory at specified DSP address	gpakWriteDspMemory
Delays at fixed time interval before polling for message responses	gpakHostDelay
Provides host task with exclusive access to DSP's memory	gpakLockAccess
Releases exclusive access to DSP's memory	gpakUnlockAccess
Reads data from 'File' for DSP download	gpakReadFile

Minds in Motion

# Hands-On Exercises

1. Use DSP Resource Wizard to estimate resource requirements and select DSP
2. Use G.PAK Configuration Builder to build DSP image (Load and test)
3. Customize G.PAK Source Code (Load and Test)

Minds in Motion

# Hands On #1: DSP Resource Wizard

1. Use DSP Resource Wizard to size application and select DSP
2. Browse to c:\adaptivedigital\tidc
  - Available at <http://wizard.adaptivedigital.com>
3. Double-click on DSPResourceWiz.htm
4. Select DSP, algorithms, and channel count

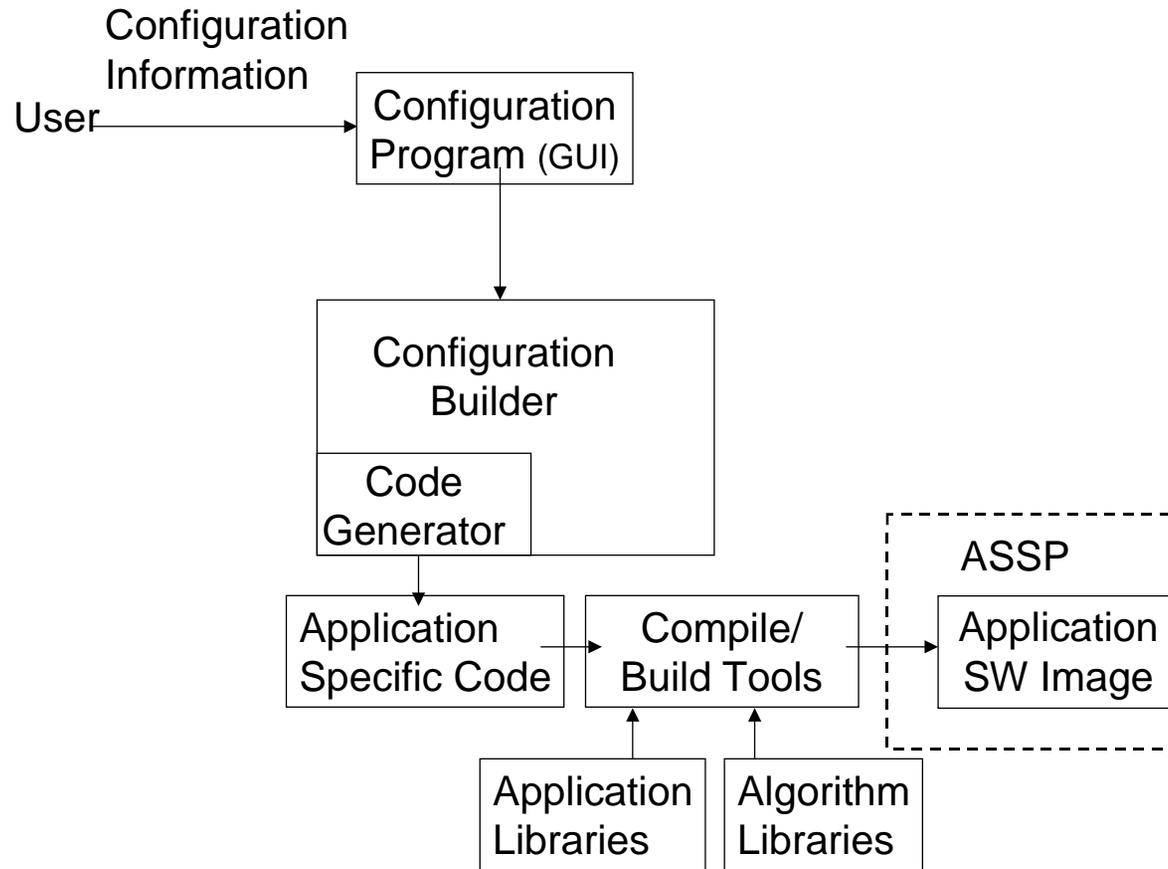
Minds in Motion

## Hands On #2: G.PAK Build

1. Use the G.PAK Configuration Builder to configure and build a DSP image
2. Load the DSP image into the DSK board
3. Use the host interface program to set up a phone call

Minds in Motion

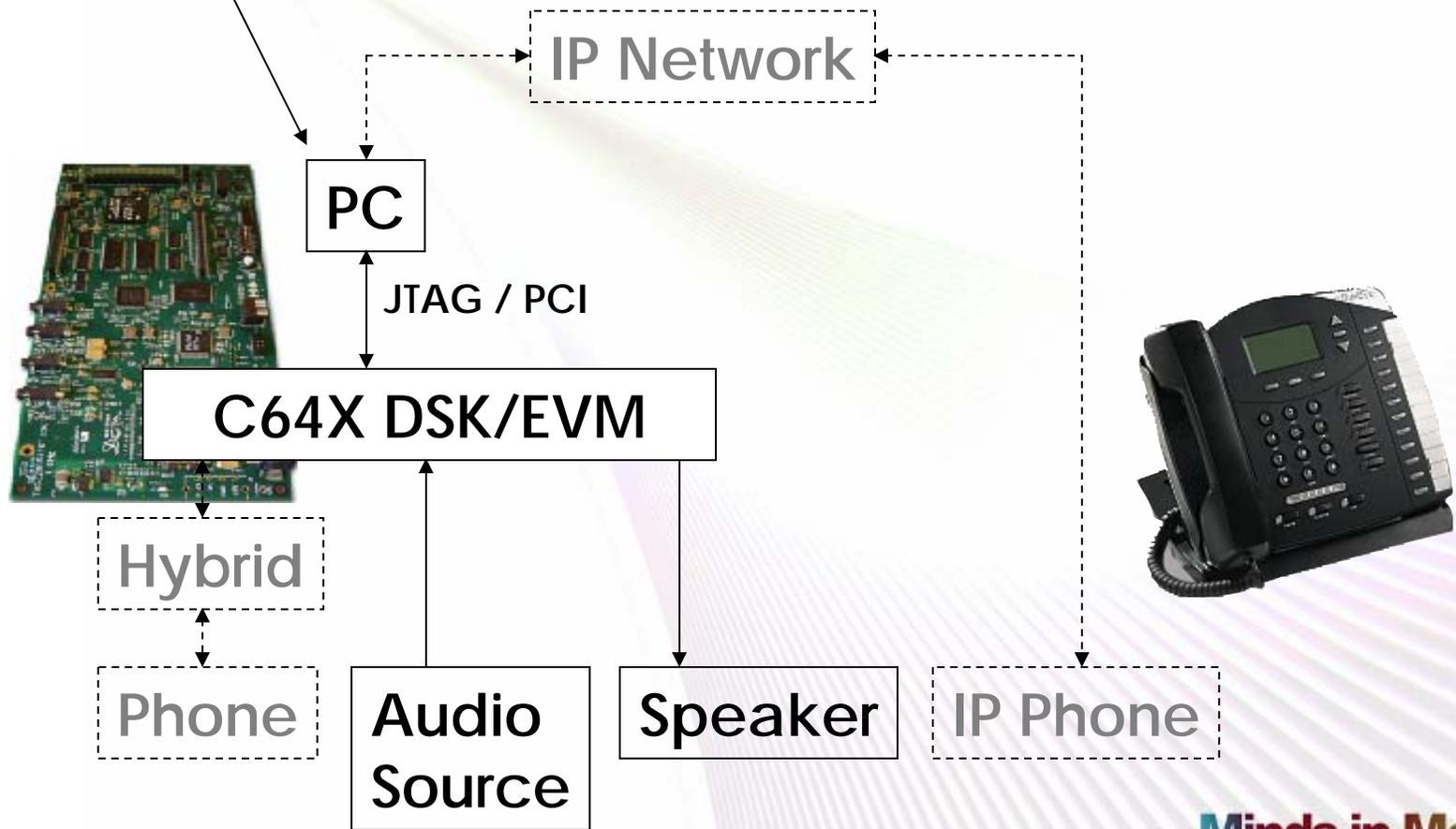
# G.PAK Configuration Builder



Minds in Motion

# G.PAK Demo

PC Emulates  
Host Processor  
Running G.PAK API



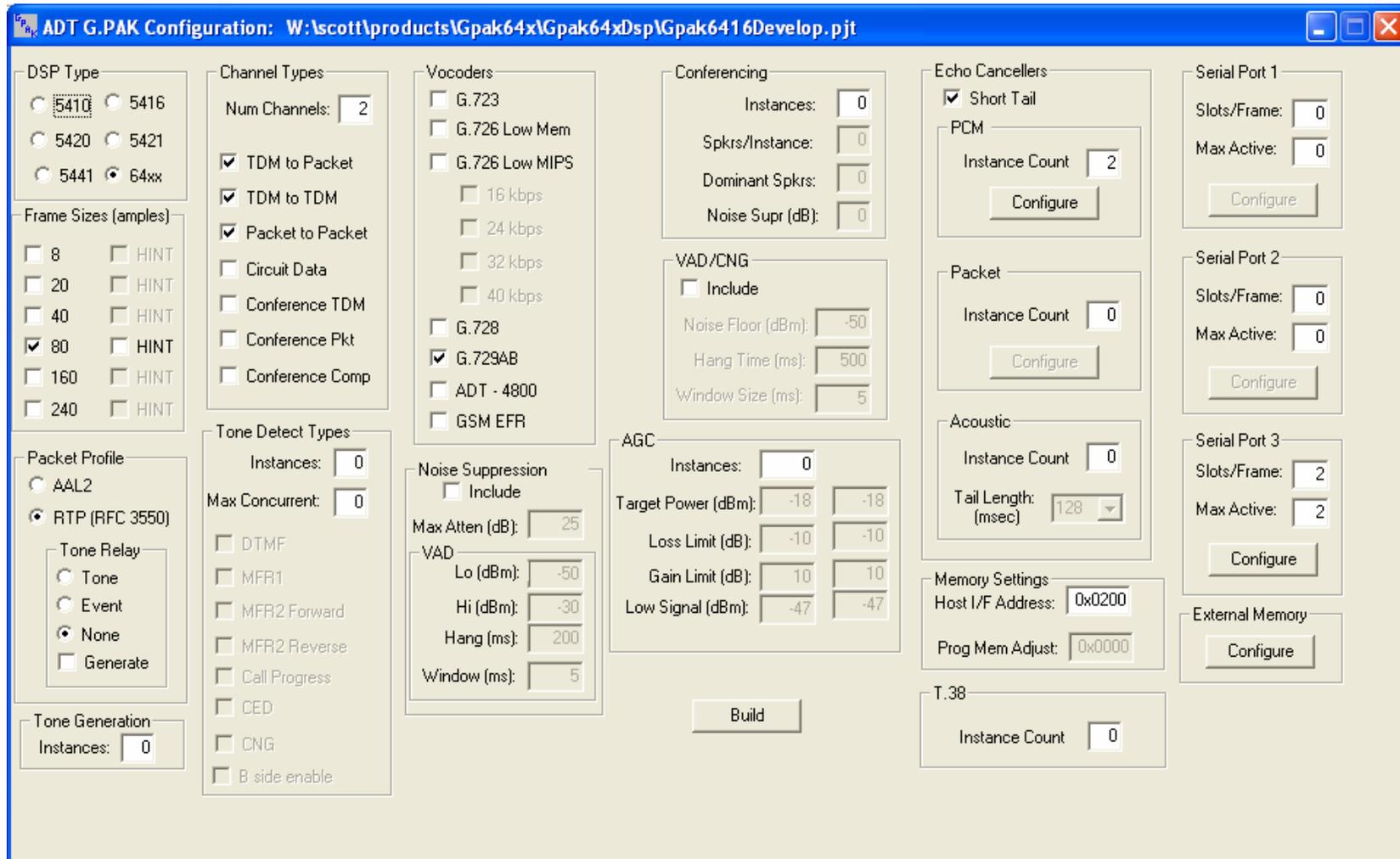
Minds in Motion

# Hands On #2: Step by Step

1. Start Code Composer Studio
2. Open the G.PAK Project
  - Project -> Open ->  
c:\adaptivedigital\tidc\gpakc64x\gpak64xdsp\gpak6416develop.pjt  
(ignore the missing library warning – we'll build that later)
3. Build the G.PAK Project
  - Project->Build  
When prompted for a G.PAK Project, click on gpak6416develop.pjt

Minds in Motion

# Hands On #2: Step by Step



Minds in Motion

## Hands on #2: Step by Step

1. Select the G.729AB Vocoder
2. Click Build
3. The program will load when the build is complete
4. Start the program – Debug -> Run

Minds in Motion

# Hands On #2: Step by Step

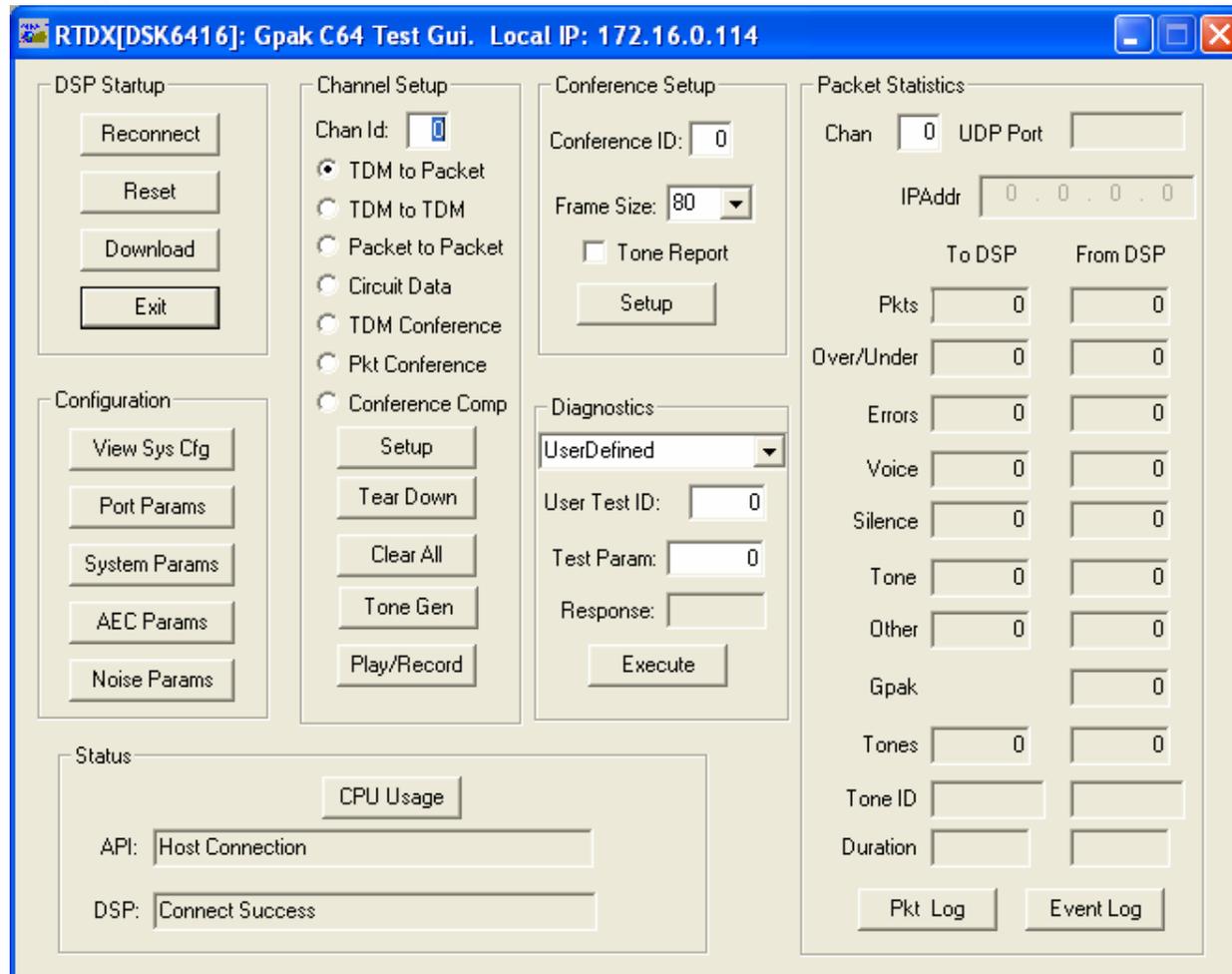
1. Start the host interface tool
  - Double-click on  
c:\adaptivedigital\tidc\Gpak64x\gpakhostinterface.exe



2. Click on RTDX
3. The host interface tool will appear

Minds in Motion

# Hands On #2: Step by Step



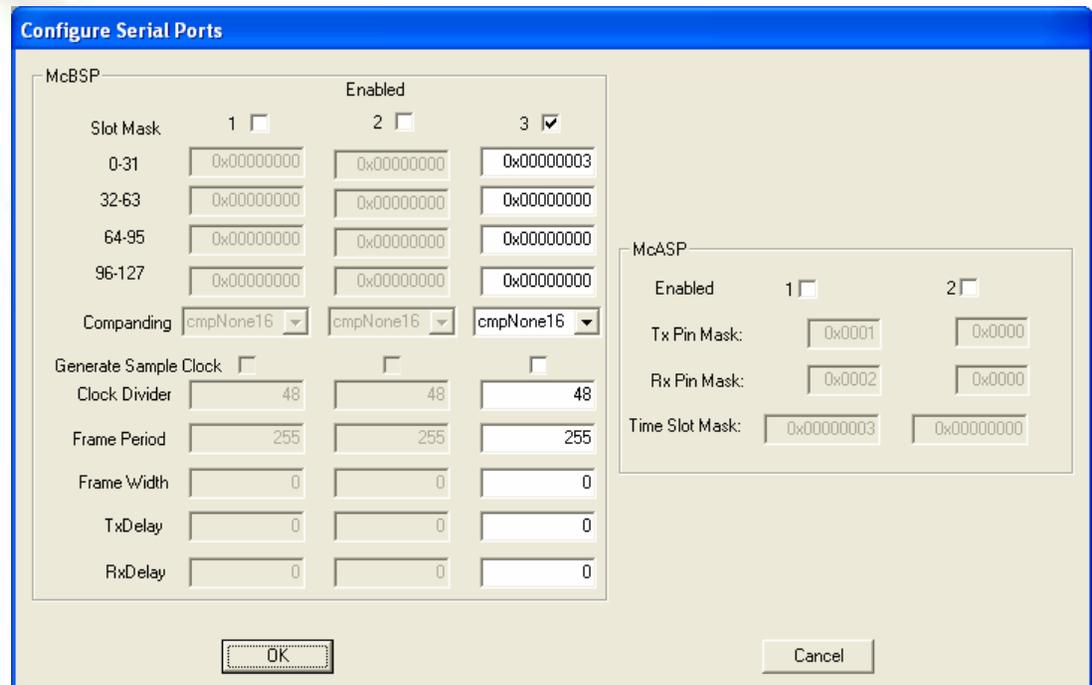
Minds in Motion

# Hands On #2: Step by Step

1. Set up the serial port (Click Port Params)  
This screen will come up:

2. Use Defaults

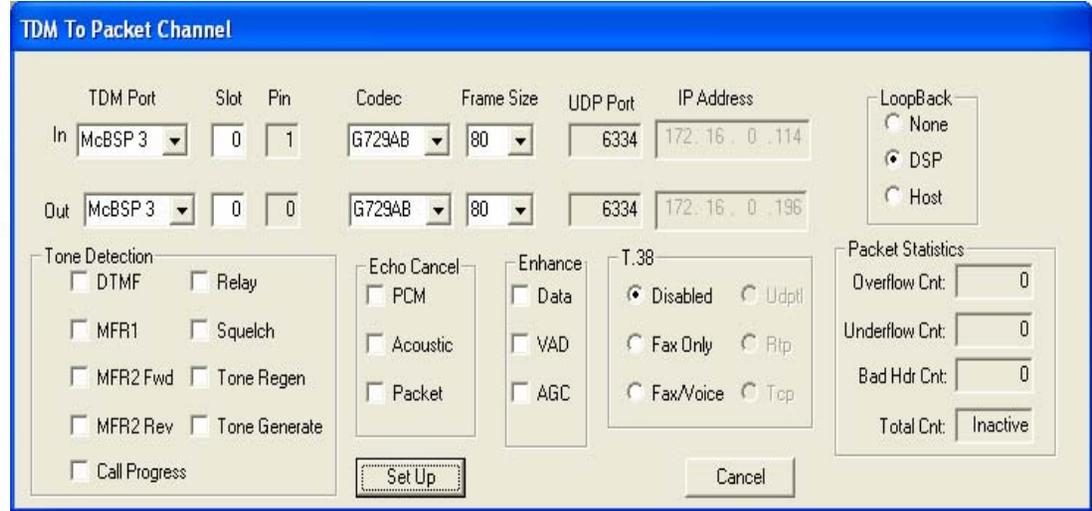
3. Click OK



# Hands On #2: Step by Step

1. Set up a TDM to Packet channel.  
The following screen will come up:

2. Use Defaults
3. Click OK



# Hands On #2: Step by Step

1. Start your audio source
2. Listen to the output
3. Other things to try:  
change the vocoder from G.729AB to G.711  
using the channel setup command

Minds in Motion

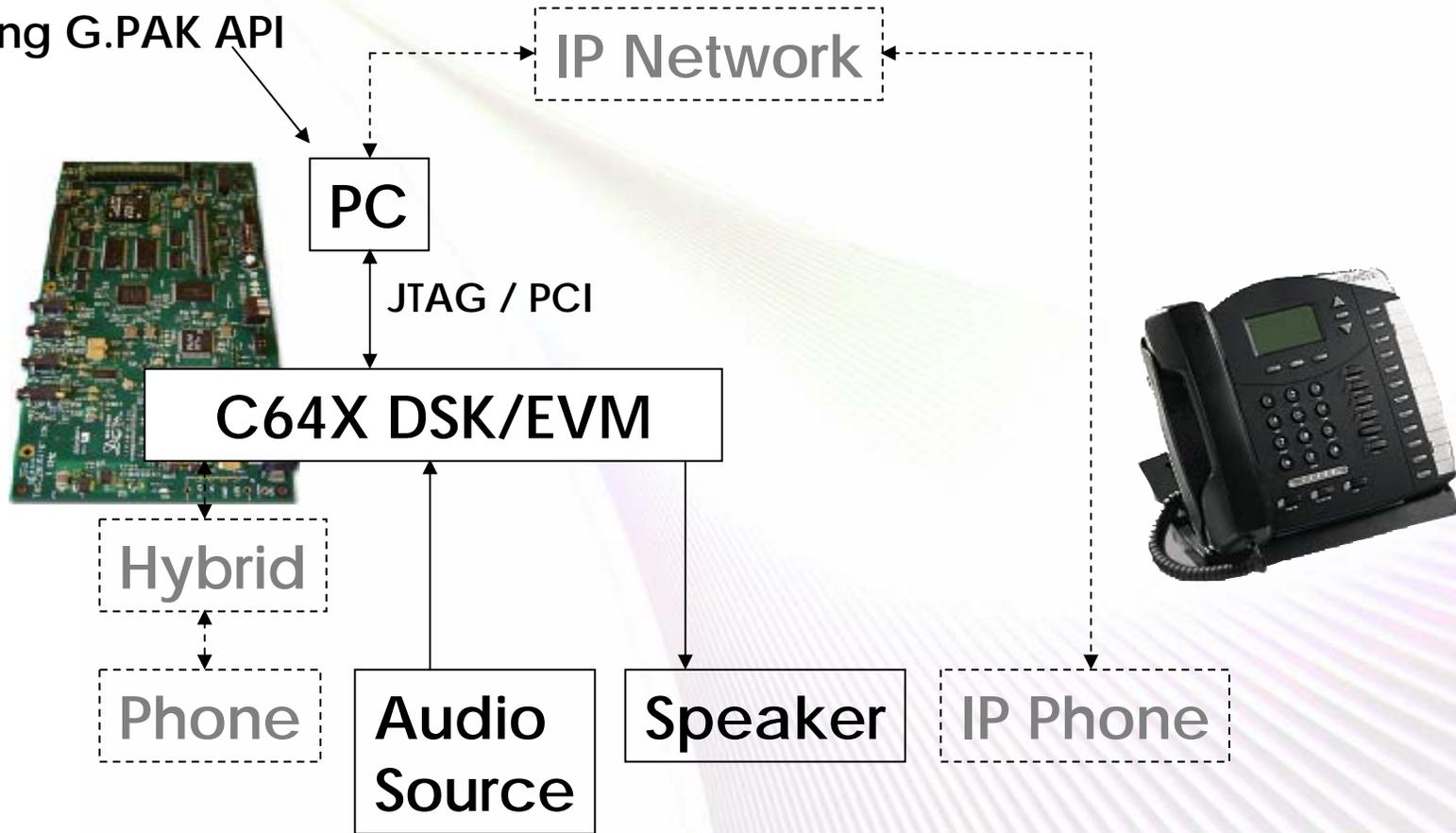
## Hands On #2: Review

- Use DSP Resource Wizard to check DSP resource requirements
- Configure and build the DSP image
- Load and run the DSP image
- Use the host interface program to configure the DSP software and set up a PCM-Packet channel

Minds in Motion

# Hands-On #2: Review

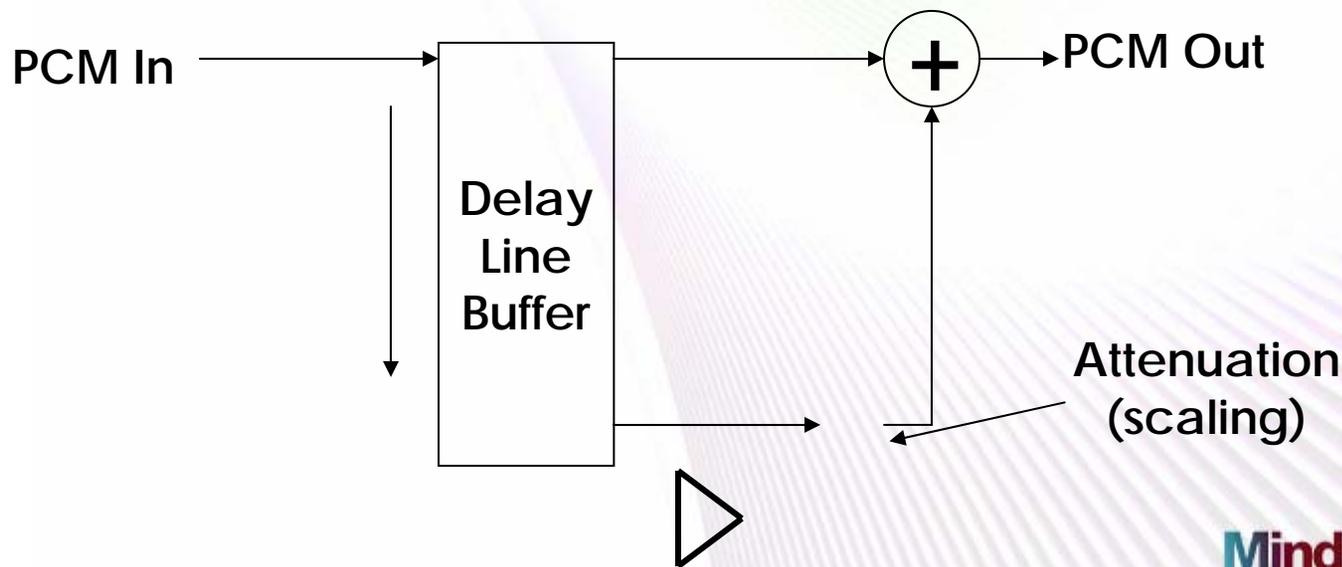
PC Emulates  
Host Processor  
Running G.PAK API



Minds in Motion

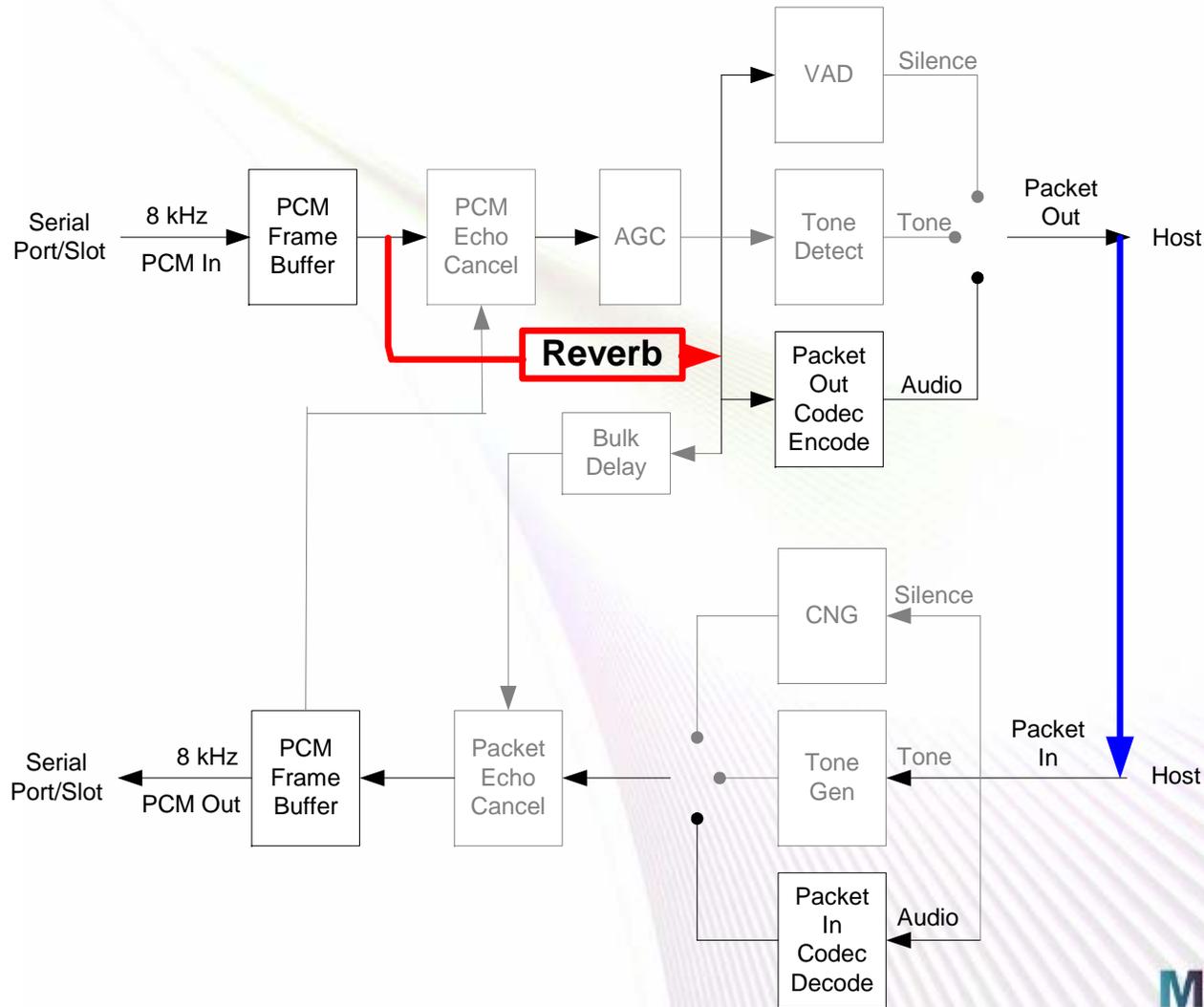
# Hands-On #3: Adding an Algorithm

- We will modify the G.PAK source code to add a reverb algorithm to the voice path
- View reverb.h



Minds in Motion

# Hands-On #3: Adding Reverb



Minds in Motion

# Adding a New Algorithm

- ✓ Important note about auto-generated files
  - The G.PAK configuration utility modifies some source files.
  - If you are modifying source files, these source files will be overwritten if you re-run the G.PAK configuration utility.

Minds in Motion

# Adding New Algorithms

- Memory Allocation
- Instantiation
- Initialization
- Signal Processing
- Teardown

Minds in Motion

# Memory Allocation / Instantiation

File	Addition
GpakDefs.h	Include file reverb.h
GPak6416Development.c	Instantiate Reverb channels and pooling data
GpakExts.h	Add references to Reverb instance and pooling information

Minds in Motion

# Initialization

File / Function	Addition
GpakMsg.c / InitGpakInterface	Mark all Reverb channels in pool as available
GpakFrame.c / initReverbChannel	New Function – Pulls a Reverb channel from pool and initializes it
Pcm2Pkt.c / SetupPcm2Pkt	Call initReverbChannel
GpakMsg.c / ProcConfigureChannelMsg	Free Reverb channel to available pool

Minds in Motion

# Signal Processing

File / Function	Addition
Pcm2Pkt.c / FrameInPcm2Pkt	Add call to the Reverb function

Minds in Motion

# Teardown

File / Function	Addition
GpakFrame.c / DeallocChanReverb	New function to free a Reverb channel to the pool
GpakMsg.c / ProcTearDownChannelMsg	Call DeallocChanReverb

Minds in Motion

## Hands On # 3: Step by Step

- Disable G.PAK Configuration builder so that it does not overwrite source files
  - From CCS:
    1. Right Click on GPAKC6416Develop project in project pane
    2. Click on Build Options
    3. Click on “General” Tab
    4. Place REM in front of the command that runs the configuration .EXE file
    5. Click OK

Minds in Motion

# Hands On # 3: Step by Step

- Replace gpak6416develop.c with gpak6416develop\_handson\_3.c
  - File -> Open gpak6416develop\_handson\_3.c
  - File -> Save As gpak6416develop.c
    - Yes to overwrite warning

- Why did I do this?
  - The G.PAK configuration program rewrites gpak6416develop.c. We need a modified version for this exercise, which has been already created as gpakdevelop\_handson\_3.c.

Minds in Motion

# Hands On # 3: Step by Step

- Open each of the following files:
  - ✓ GPAKDEFS.H
  - ✓ GPAK6416DEVELOP.C
  - ✓ PCM2PKT.C
  - ✓ PKT2PCM.C
  - ✓ GPAKFRAME.C
- In Each file, look for the occurrence of the word HANDS\_ON to understand the code that the #ifdefs are enabling

Minds in Motion

## Hands On # 3: Step by Step

- Change the project options in **each** CCS project to include the pre-processor definition `HANDS_ON`
  - Right click on a project
  - Click on Build Options
  - Click on the Preprocessor category
  - In the “Pre-defined symbols” line, append with “;HANDS\_ON”
  - Repeat for all projects

Minds in Motion

## Hands On # 3: Step by Step

- Rebuild the main project (GPAK6416Develop.pjt)
  - From the Project menu, click on Build
- Once the project builds successfully, the executable file will be loaded into the DSP on the DSK board.

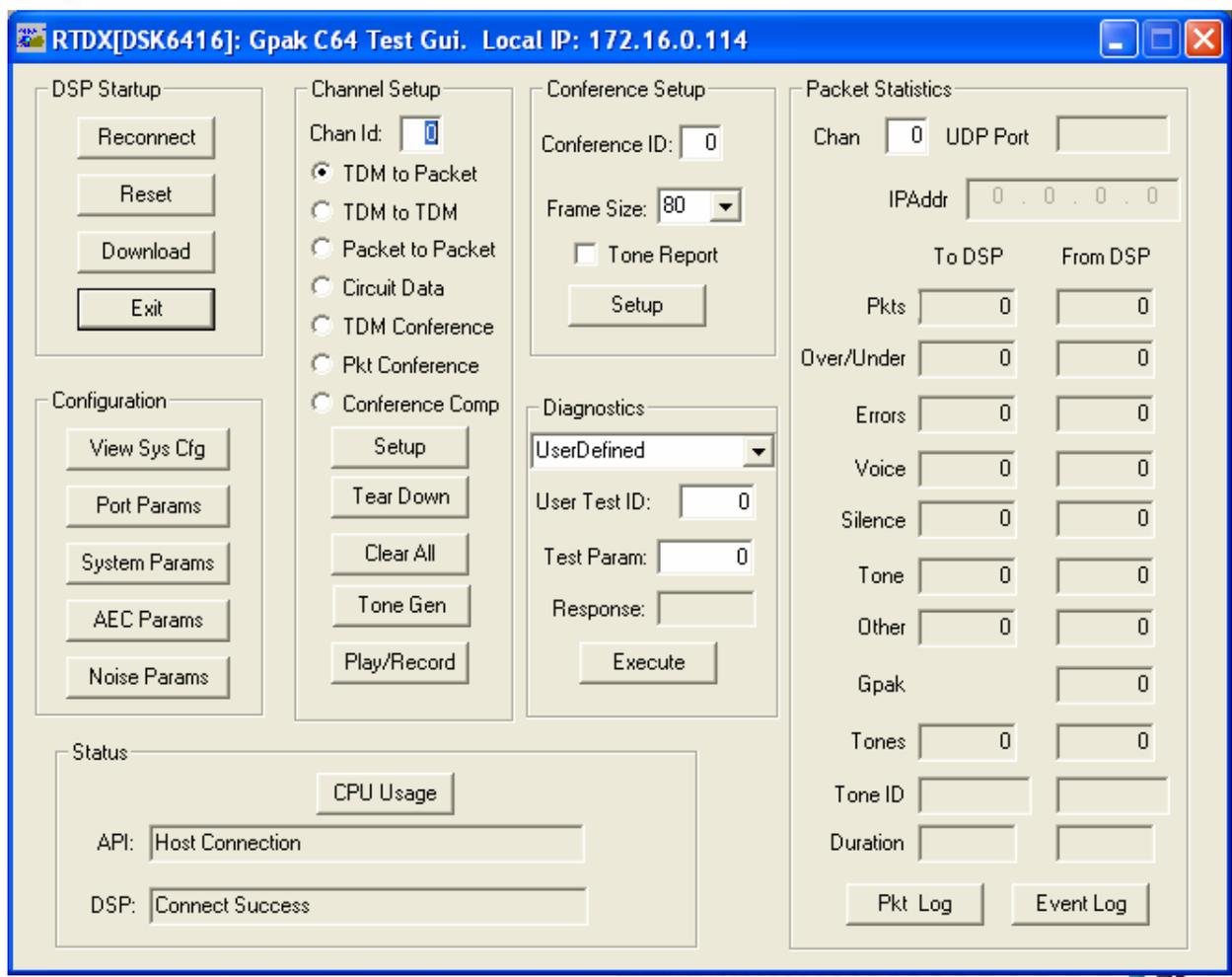
Minds in Motion

# Hands On # 3: Step by Step

1. Start the program
  - From the Debug menu click on Run
2. Switch windows tasks back to the Host Interface program.

Minds in Motion

# Hands On #3: Step by Step



Minds in Motion

## Hands On # 3: Step by Step

- Since the host program was already running, click on “Reconnect”.  
Click OK on the error messages

(We reloaded the DSP, causing the host program to become out-of-sync.)

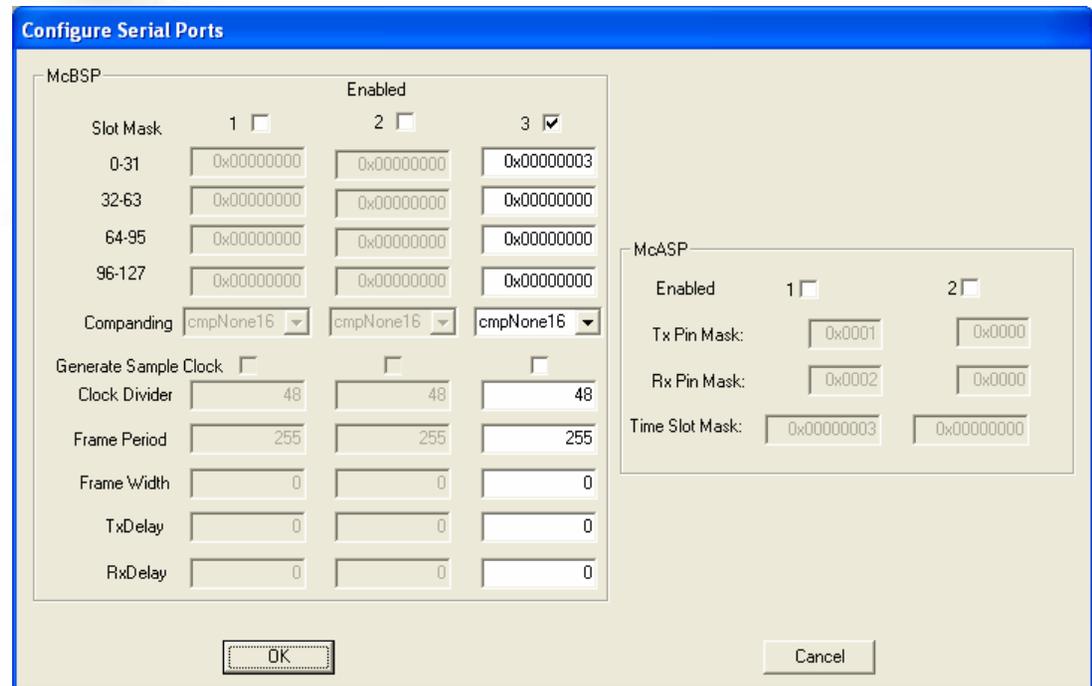
Minds in Motion

# Hands On # 3: Step by Step

1. Set up the serial port (Click Port Params)  
This screen will come up:

2. Use Defaults

3. Click OK



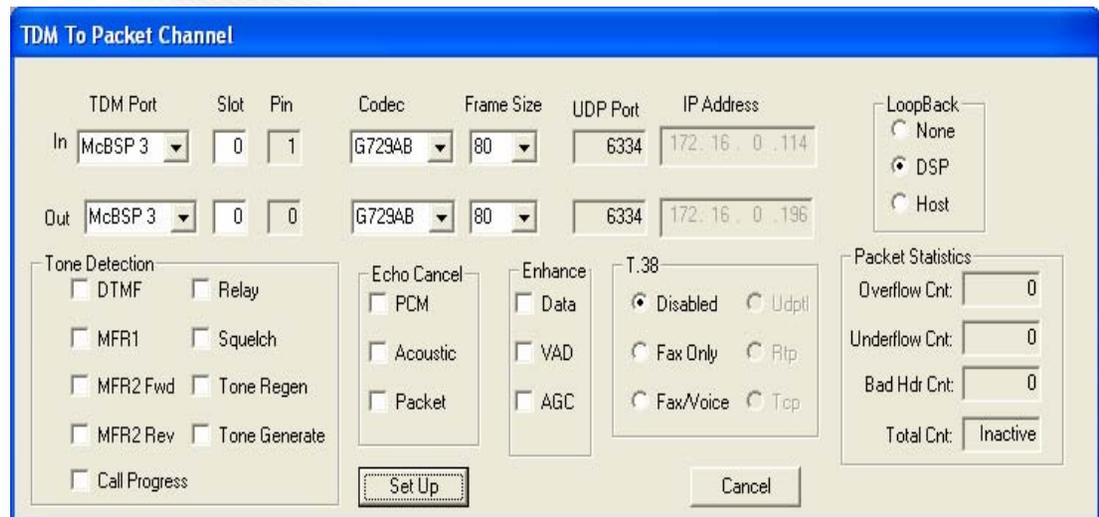
Minds in Motion

# Hands On # 3: Step by Step

1. Set up a TDM to Packet channel. The following screen will come up:

2. Use Defaults

3. Click OK



# Hands On # 3: Step by Step

1. Start your audio source
2. Listen to the output
3. Other things to try:  
change the vocoder from G.729AB to G.711  
using the channel setup command

Minds in Motion

# Hands On Review

- Use DSP Resource Wizard to check DSP resource requirements
- Configure and build the DSP program
- Modify the source code to add reverb algorithm
- Rebuild the DSP software program
- Load and run the DSP program
- Use the host interface program to configure the DSP software and set-up a PCM-Packet channel
- Listen to the result

Minds in Motion

# Summary

- Importance of leveraging existing technology
- Use of open-source to leverage existing technology
- Open Source Licensing

Minds in Motion

# Leveraging an Open-Source VoIP Application to Speed VoIP Product Development

**Scott Kurtz**

Adaptive Digital Technologies

610-825-0182 x 117

[scott.kurtz@adaptivedigital.com](mailto:scott.kurtz@adaptivedigital.com)

[www.adaptivedigital.com](http://www.adaptivedigital.com)



**Minds in Motion**