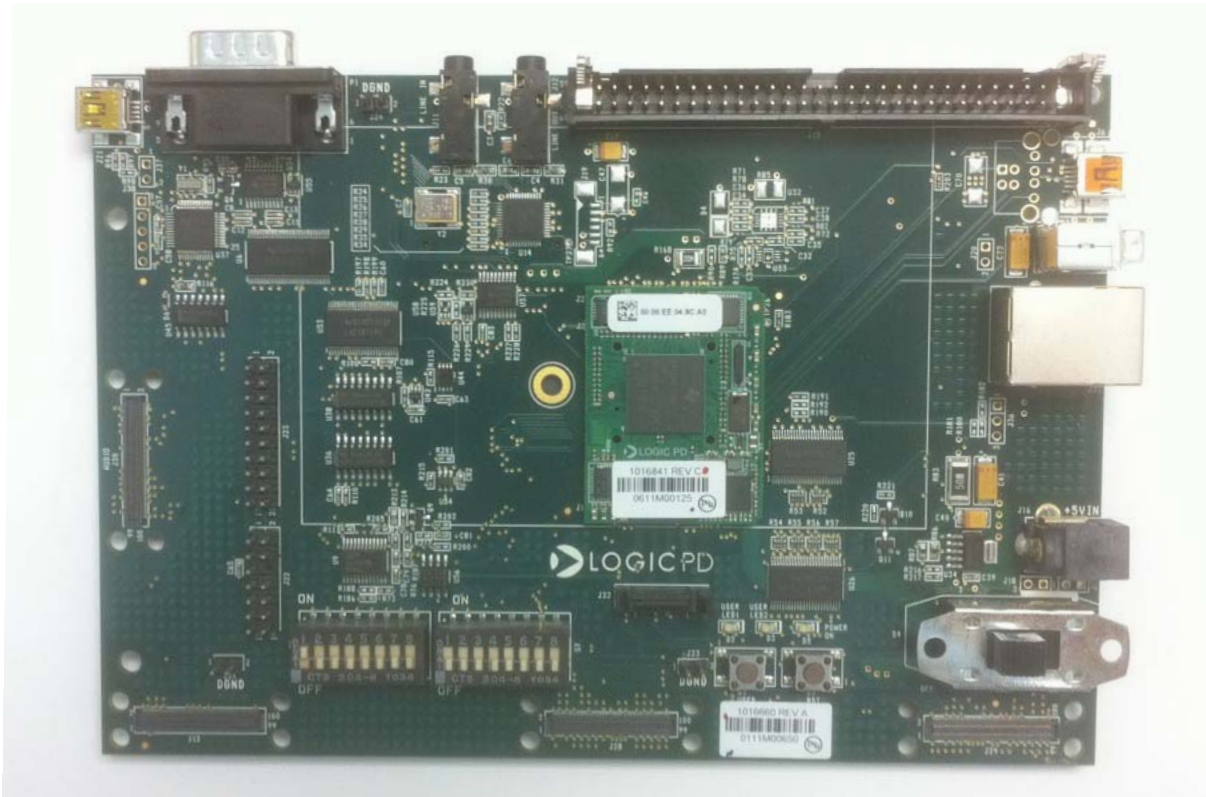


# Real-Time DSP

## ECE 5655/4655 Lecture Notes



© 1998 – 2014

Mark A. Wickert

Portions "Courtesy of Texas Instruments"



# Table of Contents

## Overview of Real-Time Digital Signal Processing

Introduction . . . . .	1–1
A Brief Description of the Course	1–1
Background Requirements	1–4
A Brief History of DSP in the Context of Real-Time Processing . . . . .	1–4
Great Moments in DSPS History	1–6
Who Are the Players?	1–7
DSP Integration (Frantz)	1–7
Communications and Wireless	1–8
DSP Hardware Design Issues . . . . .	1–11
DSP System Design Flow (Marven & Ewers). . . . .	1–12
Course Laboratory Foundations. . . . .	1–13
Software	1–14
Test Equipment	1–14

## The TMS320C6x Family: Hardware and Software

Introduction . . . . .	2–1
C62/64 Product Specifications	2–2
C67 Product Specifications	2–3
C6x Family Feature Overview. . . . .	2–4
C6x Architecture Overview . . . . .	2–5
A High-Level Look	2–5
C64x <sup>TM</sup> Enhancements Include:	2–6
A Detailed Look at the C6x Architecture:	2–7
Functional Unit Operations	2–8
C62/67 and C64 Data Paths	2–9
The Internal Bus Structure	2–10
C6x Internal Buses for 6201 and 6701	2–10
Internal Memory . . . . .	2–11
C6x11 Double Level Memory	2–11
Cache versus RAM	2–12
Internal Memory Summary	2–13
External Memory Interface (EMIF). . . . .	2–13
SDRAM	2–14
SBSRAM	2–14
Asynchronous Memory	2–14

Host Port and Expansion Bus (HPI/XB) .....	2-14
Host Port 2-14	
Expansion Bus 2-15	
Serial Port (McBSP) .....	2-15
DMA and Enhanced DMA (EDMA) .....	2-16
Timer/Counter .....	2-17
Laboratory Hardware Targets .....	2-18
The C6201 and C6701 EVM Block Diagram 2-19	
EVM Codec Block Diagram 2-20	
The C6711 DSP Starter Kit (DSK) Board Layout 2-21	
AD535 Telephony Codec on the 6711 DSK 2-22	
The C6713 DSK Block Diagram 2-23	
AIC23 Codec on the 6713 DSK 2-24	
6713 DSK Board Layout 2-25	
Memory Mapping for the EVM and DSK's .....	2-27
The C713 and C6416 DSK Memory Maps 2-28	
Software Development Overview .....	2-29
C6x Code Generation Overview 2-30	
Code Development Flow Chart 2-31	
DSP/BIOS 2-32	
DSP/BIOS Summary for Now 2-33	

## **TMS320C6x Programming**

Introduction .....	3-1
Programming Alternatives 3-1	
Introduction to Assembly Language Programming .....	3-2
A Dot Product Example 3-2	
Instruction Set Summary by Category 3-10	
C62xx and C67xx Instruction Set Summary by Unit 3-11	
Introduction to the Pipeline .....	3-12
Pipelined and Non-Pipelined 3-13	
Program Fetch Stage 3-13	
Decode Stage 3-14	
Execute Stage 3-14	
Summary of Pipeline Phases 3-15	
Sending Code Through the Pipeline 3-16	
Pipeline Code Example 3-16	
Use of Parallel Instructions 3-22	
C67x Exceptions 3-26	
C Programming .....	3-27
Debug options 3-29	
Optimize Options 3-29	

Code Size	3–30
Assembler Options	3–30
Linker Options	3–31
Summary of Popular Options	3–31
Embedded Systems with C	3–32
Initialization Under C	3–34
Compiler Sections	3–36
Memory Management	3–39
Linker Options	3–42
Calling Assembly with C	3–44
Linear Assembly and Assembly Optimization	3–48
Calling from Linear Assembly	3–51
Linear Assembly Compiler Settings	3–51
Example: Vector Norm Squared	3–51
C Version	3–52
Assembly Version	3–55
The Linear Assembly Version	3–60

## Analog Input and Output

Introduction	4–1
Analog I/O Basics	4–1
Analog I/O on the C6x	4–3
Interrupt Steps	4–3
Source Selection	4–4
Interrupt Vector Table	4–6
Enabling Interrupts	4–10
Interrupt Service Routines in C	4–13
Serial Port Communication	4–15
On-Board TLC320AD535 Telephony Codec	4–20
A Simple Interrupt Program	4–23
PCM3006 Stereo Daughter Card Codec	4–29
Daughtercard Physical Layout (Mike Morrow)	4–31
Supplied Software (Mike Morrow)	4–32
MATLAB API (Mike Morrow)	4–42
TI PCM 3003 Audio Daughter Card	4–48
Default Jumper Settings	4–49
Jumper Descriptions	4–50
Board Schematics	4–52
A Simple Interrupt Program	4–55
6713 DSK with On-Board TLV320AIC23	
Stereo Audio Codec	4–60
AIC23 Portion of the 6713 DSK	4–65

The AIC23 hardware interface on 6713 DSK	4–66
Board Usage Cautions	4–66
A Simple Loop Program	4–67
Appendix: Mike Morrow C6x11 DSK Support Tools	4–72
winDSK6	4–73
winDSK6 Control	4–73
DSK6x MATLAB API	4–74

## Fixed-Point Considerations

Introduction	5–1
C6x Data Types	5–2
Q-Format Number Representation	5–2
Multiplicative Overflow	5–5
Accumulative Overflow	5–10
Saturating the Result	5–11
Use Guard Bits	5–13
Overflow Allowed by Design	5–14
Floating Point	5–15
Single Precision	5–15
Double Precision	5–16
Addition	5–16

## Code Optimization and Advanced C

Introduction	6–1
C Runtime Environment	6–1
Stack Pointer	6–1
Global Pointer	6–2
Memory Management	6–4
Use of Volatile	6–5
Optimizing C Code	6–6
Memory Aliases	6–6
C Intrinsic Functions	6–6
Intrinsics Reference Guide	6–9

## Real-Time FIR Digital Filters

Introduction	7–1
Basics of Digital Filter Design	7–2
Overview of Approximation Techniques	7–4
Basic FIR Filter Topologies	7–5

Overview of FIR Filter Design .....	7-7
Why or Why Not Choose FIR? 7-7	
FIR Design Using Windowing 7-7	
Example: Use of the Hanning Window 7-10	
Lowpass Design 7-14	
Optimum Approximations of FIR Filters 7-17	
MATLAB Basic Filter Design Functions .....	7-19
Windowed FIR Design From Amplitude Specifications 7-20	
Using MATLAB's <code>fdatool</code> .....	7-23
Example: Windowed FIR Design 7-23	
Example: Equiripple FIR Design 7-28	
Writing C Coefficient Files 7-31	
Filter Implementation Code Examples .....	7-35
A Simple Fixed-Point Implementation 7-35	
A Simple Floating Point Implementation 7-40	
Direct Export Into Code Composer 7-42	
Circular Addressing .....	7-45
Circular Buffer in C 7-45	
C Circular Buffer 7-46	
Performance of Interrupt Driven Programs .....	7-55
Traditional Code Timing Approach 7-56	
Using Counters in the Wait Loop 7-57	
MATLAB CCS Link Example .....	7-61

## Real-Time IIR Digital Filters

Introduction .....	8-1
Basic IIR Filter Topologies .....	8-1
Direct Form I 8-2	
Direct Form II 8-3	
Cascade Form 8-5	
Parallel Form 8-6	
Transposed Forms 8-8	
Pole-Zero Lattice 8-8	
Digital Sinusoidal Oscillators .....	8-8
Overview of IIR Filter Design .....	8-11
IIR Approximation Approaches 8-11	
Converting to 8-13	
Classical Design from Analog Prototypes 8-18	
Obtaining the Second-Order Section Coefficients 8-19	
MATLAB Design Functions .....	8-19
MATLAB Filter Design Examples 8-21	
Writing C Coefficient Files 8-33	

Filter Implementation Code Examples .....	8–35
A Simple Floating-Point Cascade of Biquads Implementation	8–35
Direct Export Into Code Composer	8–39
Coupled Form Oscillator in C	8–41
MATLAB CCS Link Example .....	8–45
Command Line Interfacing	8–45
MATLAB GUI Interfacing	8–52

## Adaptive Filters

Introduction .....	9–1
Wiener Filter .....	9–2
Adaptive Wiener Filter.....	9–4
Least-Mean-Square Adaptation	9–6
Adaptive Filter Variations	9–8
Adaptive Line Enhancement .....	9–9
C6x Code Examples.....	9–13
A Two Channel Input Signal + Signal or Signal + Noise Canceller	9–13
Adaptive Line Enhancer	9–18

## Real-Time Fast Fourier Transform

Introduction .....	10–1
DTFT, DFT, and FFT Theory Overview.....	10–1
The DTFT	10–2
The DFT and its Relation to the DTFT	10–2
Simple Application Examples .....	10–5
Spectral Analysis	10–5
Transform Domain Linear Filtering	10–9
Radix 2 FFT.....	10–10
Decimation-in-Time	10–11
Decimation-in-Frequency	10–15
Computing the IDFT	10–15
Frame Processing.....	10–16
Radix-2 FFT Implementation on the C6x .....	10–17
A C Based Real-Time FFT	10–17
Transform Domain Filtering using a Linear Assembly Based FFT and IFFT	10–21

## Real-time Analysis and Scheduling

Introduction .....	11–1
Features .....	11–1



The DSP/BIOS API Modules .....	11-3
A Case Study: Audio Player with DTMF .....	11-4
Run them together under <code>main()</code> :	11-5
A second solution is to use two interrupts under <code>main()</code> :	11-5
Interrupt driven state machine	11-6
The DSP/BIOS Solution	11-7
Getting Started with DSP/BIOS	11-10

## Communications Applications

Introduction .....	12-1
Summary of Factors Relating to Implementation Choices	12-2
Transmitting Signals .....	12-3
Baseband/IF Transmitter	12-3
Complex Baseband Transmitter	12-6
Practical Pulse Shaping	12-8
Receiving Signals. ....	12-15
Complex Envelope Representation	12-15
Standard I-Q Demodulation	12-16
Using the Hilbert Transform	12-17
System Application: A DSP Based Costas Loop for Coherent Carrier Recovery. ....	12-23
MATLAB Simulation	12-28
Real-Time Implementation on the C6711 DSK	12-32
Commercial Receiver Architectures .....	12-42

## Reference Frameworks

Introduction .....	13-1
--------------------	------

## Simulink Embedded Target

Introduction .....	14-1
--------------------	------



# Overview of Real-Time Digital Signal Processing

## Introduction

In this first chapter we provide motivation for the topics to be addressed in this course. Before going any further let us first give a short description of the course and the assumed background for the course

### A Brief Description of the Course

- A course in real-time DSP brings together the following:
  - Continuous- and discrete-time systems theory (in particular knowledge from a first DSP course)
  - Software engineering concepts
  - Microprocessor programming and hardware interfacing
- The interest in doing this stems from the increase in real-time DSP applications headed for the consumer market, and the ever improving device VLSI designs for implementing powerful DSP microprocessors

- Texas Instruments has created the following long list of DSP real-time application areas

<b>Automotive</b>	<b>Consumer</b>	<b>Control</b>
Adaptive ride control Antiskid brakes Cellular telephones Digital radios Engine control Navigation and global positioning Vibration analysis Voice commands Anticollision radar	Digital radios/TVs Educational toys Music synthesizers Pagers Power tools Radar detectors Solid-state answering machines	Disk drive control Engine control Laser printer control Motor control Robotics control Servo control
<b>General-Purpose</b>	<b>Graphics/Imaging</b>	<b>Industrial</b>
Adaptive filtering Convolution Correlation Digital filtering Fast Fourier transforms Hilbert transforms Waveform generation Windowing	3-D rotation Animation/digital maps Homomorphic processing Image compression/transmission Image enhancement Pattern recognition Robot vision Workstations	Numeric control Power-line monitoring Robotics Security access
<b>Instrumentation</b>	<b>Medical</b>	<b>Military</b>
Digital filtering Function generation Pattern matching Phase-locked loops Seismic processing Spectrum analysis Transient analysis	Diagnostic equipment Fetal monitoring Hearing aids Patient monitoring Prosthetics Ultrasound equipment	Image processing Missile guidance Navigation Radar processing Radio frequency modems Secure communications Sonar processing
<b>Telecommunications</b>		<b>Voice/Speech</b>
1200- to 33 600-bps modems Adaptive equalizers ADPCM transcoders Cellular telephones Channel multiplexing Data encryption Digital PBXs Digital speech interpolation (DSI) DTMF encoding/decoding Echo cancellation	Faxing Line repeaters Personal communications systems (PCS) Personal digital assistants (PDA) Speaker phones Spread spectrum communications Video conferencing X.25 packet switching	Speaker verification Speech enhancement Speech recognition Speech synthesis Speech vocoding Text-to-speech Voice mail

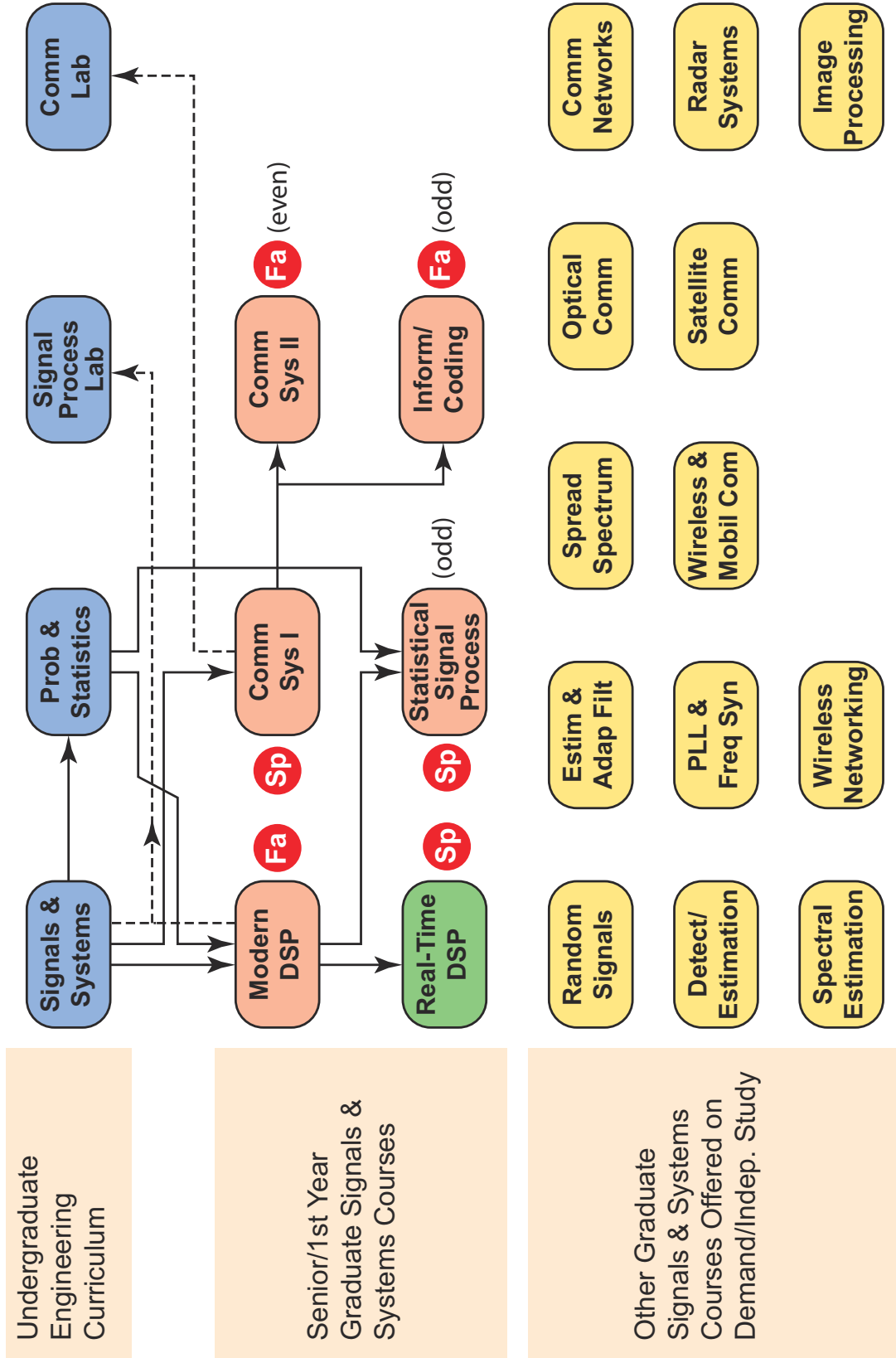
- This course is about the use of general purpose digital signal processing microprocessors for solving signal processing problems in real-time
- The course focus will be on using the Texas Instruments (TI) C6x family of fixed/floating processors, and in particular implementing programs written in assembly, linear assembly, and ANSI C, for the TI OMAP-L138 (ARM Cortex-A9 + C6748 floating-point DSP) ZOOM™ OMAP-L138 Experimenter Kit (TMDSEXPL138 \$495 at [www.ti-estore.com](http://www.ti-estore.com))
  - A lower cost version of this board that does not include the touch screen is available for ~\$150 (where?)
- Other platforms that will be explored this semester include:
  - TI C6713 DSK (digital signal processing starter kit available for ~\$395.00 at [www.ti-estore.com](http://www.ti-estore.com))
  - TI C55x fixed-point family via the VC5505/VC5515 eZDSP USB Stick (\$49 at [www.ti-estore.com](http://www.ti-estore.com))
  - TI OMAP-3530 (ARM Cortex-A8 + C64x+ DSP) Beagle-Board-xM (\$149 from links at <http://beagleboard.org/>)
- The course will start out considering general signal processing applications of real-time DSP and the associated programming issues
  - Later in the course, a focus applications area will be wireless communication system design using DSP algorithms

- In addition to the text book, we will be using TI training materials from the courses
  - TMS320C6000 DSP Workshop (normally a 4 day course)
  - TI C6000/OMAL-L138 Teaching Materials DVD v3
- The course meeting time will be used for lecturing and laboratory time (about 60/40) using hardware/software development tools, primarily Code Composer Studio (4.2 & 5.1)

## **Background Requirements**

- The required background for all students taking the course is an introductory graduate or junior/senior level undergraduate course in DSP and experience programming in ANSI C
- Knowledge of ANSI C is required in order to develop real-time algorithm in a high level language (HLL)
- TMS320C6x assembly and linear assembly language programming will be covered in the course
- Eventually, and in practice, we will be using a combination of C and assembly, e.g., mixed language programming.
- Programming the PC host in C/C++, or perhaps some other language, may be useful for developing a user interactive application for the final project
  - A MATLAB API (application programming interface) for the DSK is also available
- Familiarity with test equipment e.g., signal generators, digital scopes, and a spectrum/network analyzer would be helpful

# Course Perspective



## A Brief History of DSP in the Context of Real-Time Processing

- From a systems engineering point of view DSP stands for digital signal processing, but in the world of real-time hardware, systems people often refer to digital signal processing solutions (DSPS)
- The decades of DSP/DSPS have brought the following<sup>1</sup>

Table 1.1: Decades of DSP

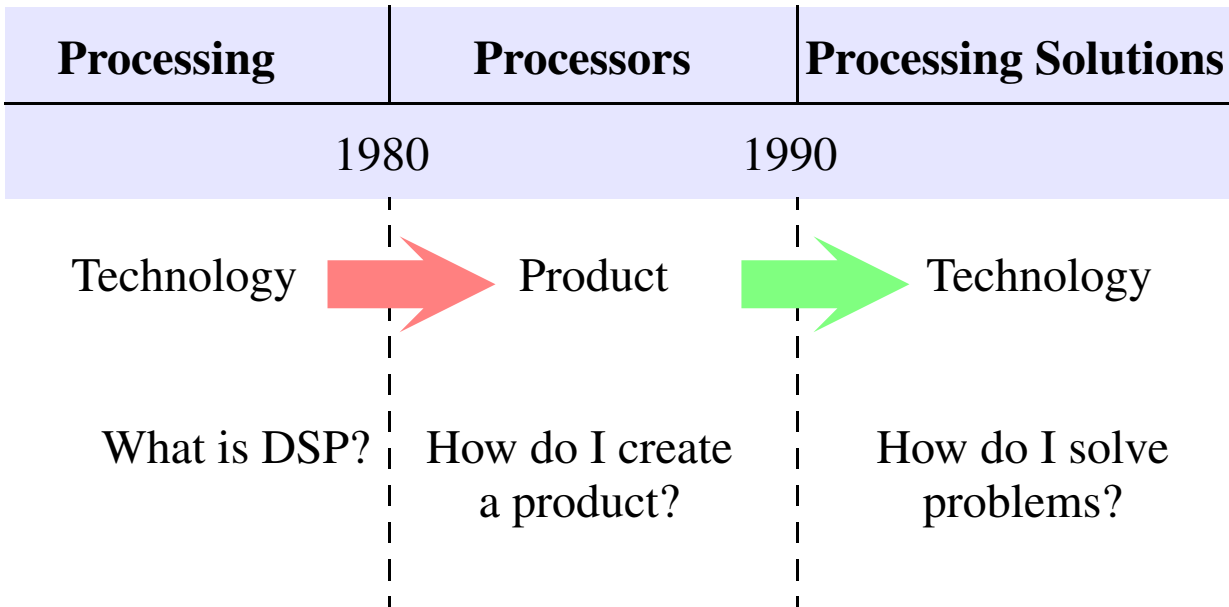
Decade	Characteristic	\$/MIPS
'60s	University Curiosity	\$100–\$1,000
'70s	Military Advantage	\$10–\$100
'80s	Commercial Success	\$1–\$10
'90s	Consumer Enabler	\$0.10–\$1
Beyond	Expected Part of Daily Life?	\$0.01–\$0.10

---

1. Gene A. Frantz, “TI’s DSPS Future,” *Texas Instruments DSPS Fest ‘97*, Houston, TX, July 1997.



- Another way of looking at this is (Frantz)



- We are well beyond the time line shown above, what has been happening in the last 12 years?
  - DSP is definitely mainstream in most everything electronic
  - general purpose DSP processors are only part of the picture; FPGA drives the government/military applications side, and ARM processors drive the consumer product side
  - In the last few years we have seen more multi-core devices
  - The open media application platform, OMAP-L138 used for this course is an example of this

## Great Moments in DSPS History<sup>1</sup>

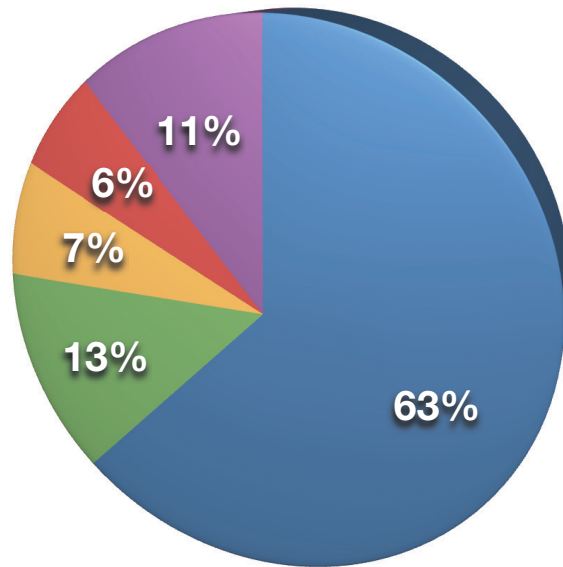
- 1976 – DSP is used to simulate a voice in the educational product “Speak and Spell”
- 1982 – TI announces details on the TMS32010 which executes at 5 MIPS
- 1985 – DSP is used in a modem for the first time
- 1986 – Lotus automotive uses DSP in active suspension and noise abatement in racing cars
- 1988 – First DSP hearing aid introduced
- 1991 – First TI sponsored educators conference
- 1993 – Cadillac automotive introduces a DSP-based ride control system
- 1995 – TI implements the *On-line DSP Lab<sup>TM</sup>* for Web based testing of DSP applications
- 1997 – 15 years of DSPs for TI and
  - The introduction of the TMS320C6x family of DSPs 1,600 MIPS performance (C62x)
  - The 1-v barrier in power consumption reached
  - In late 1997 the first C6x floating point part is announced (C67x) with 1 GFLOPS performance
- 2012?

---

1. “Great Moments in DSPS History,” *Integration: An Update on Texas Instruments Semiconductors*, vol. 14, No. 4, June 1997.

## DSP Vendor Market Share for 2006

Source: Forward Concepts, 2007

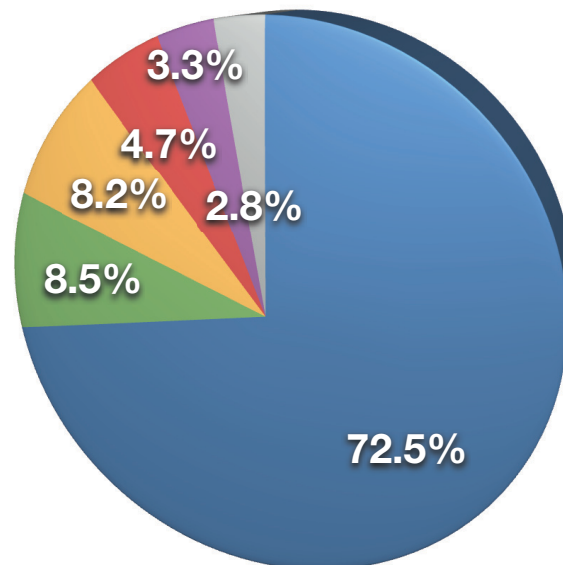


● TI ● Freescale ● Agere ● ADI ● Other

- TI increased by 17% in 2006 over 2005; 17% increase in cellular shipments and 9% in the DSP market

## DSP Markets for 2006

Source: Forward Concepts, 2007



● Wireless ● Multipurpose ● Consumer ● Wireline  
● Computer ● Automotive

**DSP Integration (Frantz)**

Table 1.2: DSP Integration over three decades.

	Typical 1982 DSP	Typical 1992 DSP (97)	Typical 2002 DSP
Die size	50 mm	50 mm	50 mm
Technology size	3 $\mu$	0.8 (.35) $\mu$	0.18 $\mu$
MIPS	5	40 (100)	2000
MHz	20	80 (200)	500
RAM (words)	144	1 K	16 K
ROM (words)	1.5 K	4 K	64 K
Price	\$150.00	\$15.00	\$1.50
Pwr. Dissp.	250 mW/ MIP	12.5 mW/ MIP (1)	0.1 mW/MIP
Transistors	50 K	500 K	5 M
Wafer Size	3" (75 mm)	6" (150 mm)	12" (300 mm)

- In Spring 2000 TI also announced the C55x family, which offers 0.05 mW/MIPS and 600–800 MIPS
- Late 2004, C67 at 1350 MFLOPs at \$36.50, 800 MFLOPs at \$21.07
- Late 2004, 1GHz C64 which offers fixed-point processing for communications and imaging applications (eight 32-bit instructions/cycle, 28 operations/cycle, 8000 MIPS)

Microcontrollers (MCUs)	ARM®-Based Processors	Digital Signal Processors (DSPs)
16-bit ultra-low power MCUs	32-bit ARM Cortex™-M3 MCUs	DSP+ARM
<b>MSP430™</b> Up to 25 MHz Flash 1 KB to 256 KB Analog I/O, ADC, LCD, USB, RF Measurement, Sensing, General Purpose \$0.25 to \$9.00	<b>C2000™ Delfino™ Piccolo™</b> 40MHz to 300 MHz Flash, RAM 16 KB to 512 KB PWM, ADC, CAN, SPI, I²C Motor Control, Digital Power, Lighting, Ren. Enrgy \$1.50 to \$20.00	<b>Ultra Low power DSP</b>
<b>C2000™ Delfino™ Piccolo™</b> 40MHz to 300 MHz Flash, RAM 16 KB to 512 KB PWM, ADC, CAN, SPI, I²C Motor Control, Digital Power, Lighting, Ren. Enrgy \$1.50 to \$20.00	<b>Stellaris® ARM® Cortex™-M3</b> Up to 100 MHz Flash 8 KB to 256 KB USB, ENET, MAC-PHY, CAN, ADC, PWM, SPI Connectivity, Security, Motion Control, HMI, Industrial Automation \$1.00 to \$8.00	<b>Multi-core DSP</b>
<b>C2000™ Delfino™ Piccolo™</b> 40MHz to 300 MHz Flash, RAM 16 KB to 512 KB PWM, ADC, CAN, SPI, I²C Motor Control, Digital Power, Lighting, Ren. Enrgy \$1.50 to \$20.00	<b>Sitara™ ARM® Cortex™-A8 &amp; ARM9</b> 300MHz to >1GHz Cache, RAM, ROM USB, CAN, PCIe, EMAC Industrial computing, POS & portable data terminals \$5.00 to \$20.00	<b>C5000™</b> Up to 300 MHz +Accelerator Up to 320KB RAM Up to 128KB ROM USB, ADC McBSP, SPI, I²C Audio, Voice Medical, Biometrics \$3.00 to \$10.00
<b>C2000™ Delfino™ Piccolo™</b> 40MHz to 300 MHz Flash, RAM 16 KB to 512 KB PWM, ADC, CAN, SPI, I²C Motor Control, Digital Power, Lighting, Ren. Enrgy \$1.50 to \$20.00	<b>ARM® Cortex™-A8 &amp; ARM9</b> 300MHz to >1GHz Cache, RAM, ROM USB, ENET, PCIe, SATA, SPI Floating/Fixed Point Video, Audio, Voice, Security, Conferencing \$5.00 to \$200.00	<b>C6000™</b> 24,000 MMACS Cache, RAM, ROM SRIO, EMAC DMA, PCIe Telecom test & meas, media gateways, base stations \$40 to \$200.00
<b>C2000™ Delfino™ Piccolo™</b> 40MHz to 300 MHz Flash, RAM 16 KB to 512 KB PWM, ADC, CAN, SPI, I²C Motor Control, Digital Power, Lighting, Ren. Enrgy \$1.50 to \$20.00	<b>C6000™ DaVinci™ video processors Integra™</b> 300MHz to >1GHz +Accelerator Cache, RAM, ROM USB, ENET, PCIe, SATA, SPI Floating/Fixed Point Video, Audio, Voice, Security, Conferencing \$5.00 to \$200.00	<b>Ultra Low power DSP</b>

Software & Dev. Tools

MCUs – Microprocessors

## – C67x floating-point family

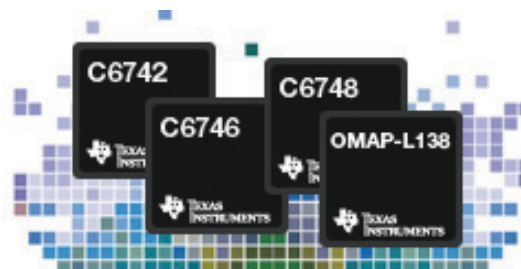
New OMAP-L138, TMS320C6748/6/2 power-efficient processors offer unmatched connectivity options for industrial and communication applications

With these four new processors from TI based on the **OMAP-L13x/C674x** platforms, developers can:

- Access flexible networking and interface options (SATA, uPP, Ethernet MAC and more)

- Optimize their designs for power and performance utilizing the industry's lowest power floating-point DSP

- Develop and design products easier with low-cost development tools (starting at \$149) and pin-for-pin and software compatible options.



All the devices feature:

- C674x fixed- and floating-point DSP core which provides high-precision and wide dynamic range as well as higher performance, efficiency and low power

- Power management software with DVFS to enable various power down modes

- Complementary analog products including the TI TPS65070 power management device with integrated 10-bit, 4-channel ADC, touch screen controller and battery charger implements all sequencing and default options and supports the devices' power modes

- Object code compatible with all TMS320C67x+™ and TMS320C64x+™ devices so developers can leverage existing software code

- Pin-for-pin compatibility (**C6742**, **C6746**, **C6748** and **OMAP-L138**) allows for the ability to expand an entire product portfolio using the same hardware and software platform

The OMAP-L138 applications processor consists of an ARM9 and C674x fixed and floating point DSP combined with interface options for SATA, Universal Parallel Port (uPP), Ethernet MAC, USB 2.0HS, USB1.1 FS, Video Input/Output, LCD Controller and much more.

The TMS320C6748/6/2 processors consist of a C674x fixed- and floating-point DSP combined with a variety of networking and interface features including many, if not all, of those found in the OMAP-L138.

– C64x fixed-point multi-core family

## **TMS320C64x DSPs Performance Value DSPs**

C64x Performance Value DSPs offer speeds up to 1 GHz offer a wide range of performance and peripheral options.

- Up to 1GHz / 8000 MMACS of processing performance
- Up to 1MB of L2 memory on-chip
- Starting at ~\$10 (1ku quantity)
- Smallest package C64x DSP achieved with compact 16mm x 16mm footprint
- Significant BOM savings from on-chip integration of EMAC, PCI, HPI, McBSP, McA I2C, UART, PWM, GPIO, Utopia, EMIF, and 64-bit timers.
- C64x+ core enables 20% higher cycle performance compared to the previous generation C64x core
- 20-30% smaller code size from 16-bit compact instructions and SP LOOP buffer

### **C64x™ DSP Applications**

- Networked applications, printers, scanners, copiers, Software Defined Radio, pub safety, medical diagnostics, industrial, Military/Aero/Space, communications and emerging.

– Multi-core versions

## **TMS320C647x Multicore DSPs**

**Devices**    Benchmarks

### **TMS320C647x™ high-performance DSP**

Options of up to 6 integrated cores on a single die

Highest performing multicore DSP with 4.2 GHz performance

Includes industry's best power efficiency with 3 GHz at 0.15 mW/MIPS

100% code compatible with TI's single core DSPs based on C64x or C64x+ cores

TI's SmartReflex™ technology reduces device power consumption by tweaking the core voltage once the chip is integrated into the system.

Serializer/deserializer (SERDES) interfaces

SGMII Ethernet MAC (EMAC)

Antennae Interface (AIF)

Serial Rapid I/O (SRIO)

Up to 4.8 MB of L1/L2 RAM

TI's fastest DDR2 memory interface



– C66x fixed/float dual-core family

## TMS320C66x high-performance multicore DSPs combine fixed- and floating-point capabilities, delivering innovation and performance at unmatched levels

The industry's highest performing multicore DSPs combine both fixed- and floating-point capabilities for the first time in the TMS320C66x generation of scalable devices. The new generation includes three new multicore processors, the **TMS320C6672**, **TMS320C6674** and **TMS320C6678**, and a four-core **TMS320C6670** communications system-on-a-chip (SoC).

The devices deliver up to 10 GHz of performance at the lowest power levels, making them suitable for developing products for applications such as mission critical, medical and high-end imaging, test and automation, high-performance computing and core networking.



### TMS320C66x high-performance multicore DSPs key features include:

- Coprocessing, fixed and floating-point operation and optimized inter-element communication due to TI's KeyStone architecture
- Variety of processor options integrating up to eight C66x cores, including:
  - Value-optimized, dual-core **C6672** in 1 GHz or 1.25 GHz speed options
  - Power-optimized **C6674** with four cores or **C6678** with up to eight cores. Both are available in 1 GHz, or 1.25 GHz speeds
  - Performance-optimized, four core **C6670** SoC, in 1 GHz or 1.2 GHz speeds
- Industry's first 10 GHz DSP featuring 320 GMACs and 160 GFLOPs of fixed- and floating-point performance
- 100% code compatibility with C64x™ fixed-point and C67x™ floating-point devices
- Multicore entitlement with features such as Multicore Navigator, Layer 1/2 acceleration, PCI Express, and improved memory architecture
- Full range of peripherals and power optimization techniques
- Field-hardened compiler and development tools, robust ecosystem and numerous hardware vendor partnerships

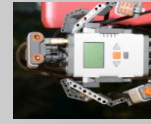





**Design considerations:**

- **Ease of programming shortens development time by weeks or months resulting in reduced time to market**
- **Energy efficient DSP devices as low as 38mW standby, 467mW active power consumption**
- **Floating point instructions for applications that require high precision, wide dynamic range**
- **Connectivity peripherals for BOM cost reduction, including Ethernet MAC, UPP, SATA, USB, VPIF, etc.**
- **Code compatibility across full line of C6000 DSPs – C64x, C64x+, C67x and C674x DSPs**
- **Industrial/automotive temp support from -40 to 125 deg C**

**Applications such as:**

- **Medical imaging**
- **Test and measurement**
- **Military**
- **Audio**
- **Communication**
- **Industrial**



DSP	DSP + ARM	DSP/ARM + Video
<p><b>C645X</b> <b>C641X</b> <b>C674X</b> ★ <b>C642X</b> <b>C672X</b> <b>C671X</b></p>  <p><b>DSP with focus on intensive Signal Processing</b></p> <ul style="list-style-type: none"> <li>• Power-efficient fixed/floating pt DSP devices; 38mW to 467mW</li> <li>• High-performance DSP devices; up to 8000 MMAC @ 1GHz</li> <li>• Excels at communications, industrial, military, medical, T&amp;M and audio processing</li> </ul>	<p><b>C6L138/OMAP-L138</b> ★ <b>C6L137/OMAP-L137</b></p>  <p><b>DSP with integrated Controller/Host Processing</b></p> <ul style="list-style-type: none"> <li>• Low power (38mW standby; &lt;520mW typical active) fixed and floating point DSP with ARM core</li> <li>• Ideal for industrial, audio and communications apps</li> </ul>	<p><b>DM6467/T</b> <b>OMAP3530/25</b> <b>DM644X</b> <b>DM3xx</b> ★ <b>DM643X</b> <b>DM64X</b></p>  <p><b>Video and Imaging Processors</b></p> <ul style="list-style-type: none"> <li>• High performance multi-format video up to 1080p</li> <li>• Ideal for Video, Imaging and Vision applications</li> </ul>

★ Featured Product

– C55x fixed-point family

**TMS320C55x™ DSPs - featuring the TMS320C5505, TMS320C5504, TMS320C5514 and TMS320C5515 DSPs**

- 100% code-compatible with C55x devices
- Broad portfolio of the industry's most power-efficient 16-bit DSPs with standby power as low as 0.15 mW/MHz and performance up to 600 MIPS
- Lowest industry standby power greatly extends battery life
- Software compatible with all C5000™ DSPs
- Easy-to-use software and development tools speed time-to-market

**C55x DSP Applications**

- Feature-rich, miniaturized personal and portable products; 2G, 2.5G and 3G cell phones and basestations; digital audio players; digital still cameras; electronic books, voice recognition, GPS receivers; fingerprint / pattern recognition; wireless modems; headsets; biometrics.

## Communications and Wireless

- In the past two key application areas, telephone line modems and cellular voice/data communications, have had a major impact on DSPS sales
- In 1994 when the V.34 modem standard (28.8 kbps) was approved fixed point (integer) DSPs became a popular design route
  - The V.34 standard allowed for data rates from 2400 to 28,800 bps
  - The modulation scheme at 28,800 is a high-order quadrature amplitude modulation (QAM) employing *shell mapping*, *trellis encoding*, *nonlinear precoding*, and *soft decision Viterbi decoding*<sup>1</sup>
  - All of the above and backward compatibility (lower rate schemes) can be performed by one 50 MIPS fixed point DSP
  - In late 1994 these modems cost ~\$400, today we know that 28,800 modems (dial-up) are not used much at all
- In the cellular telephone arena fixed point DSPs are widely used in the portable handsets
  - For example the TI C55x series is popular because it has features such as low power dissipation, integration of

---

1. Steven Tretter, *Communication System Design using DSP Algorithms with Laboratory Experiments for the TMS320C6713*, Plenum Press, New York, 2008.

memory and peripherals, integrated Viterbi accelerator, and small packaging, just to name a few

- Other vendors have tailored their DSP product lines for the high volume cellular handset markets as well
- DSPs are also utilized in VoIP and cellular basestations
  - In a portable device market, such as cell phones, the efficiency of a particular processor can be measured in mW/algorithm, which is obtained via

$$\frac{\text{mW}}{\text{algorithm}} = \frac{\text{mA}}{\text{MIPS}} \times \frac{\text{MIPS}}{\text{algorithm}} \times \text{voltage}$$

- MIPS/algorithm requirements for an IS95 handset<sup>1</sup>

Table 1.3: MIPS requirements for an IS95 handset.

Algorithm	Implementation	MIPS
Correlator	Hardware	5
Automatic frequency control (AFC)	Hardware	5
Automatic gain control (AGC)	Hardware	5
Transmit filter	Hardware	30
128-pt FFT	Software	1
Viterbi decoder (length 9, rate 1/2)	Software	6
Vocoder (8-Kbps Qualcomm code excited linear prediction (QCELP))	Software	20
Vocoder (enhanced variable rate coder (EVRC))	Software	30

# DSP Hardware Design Issues<sup>1</sup>

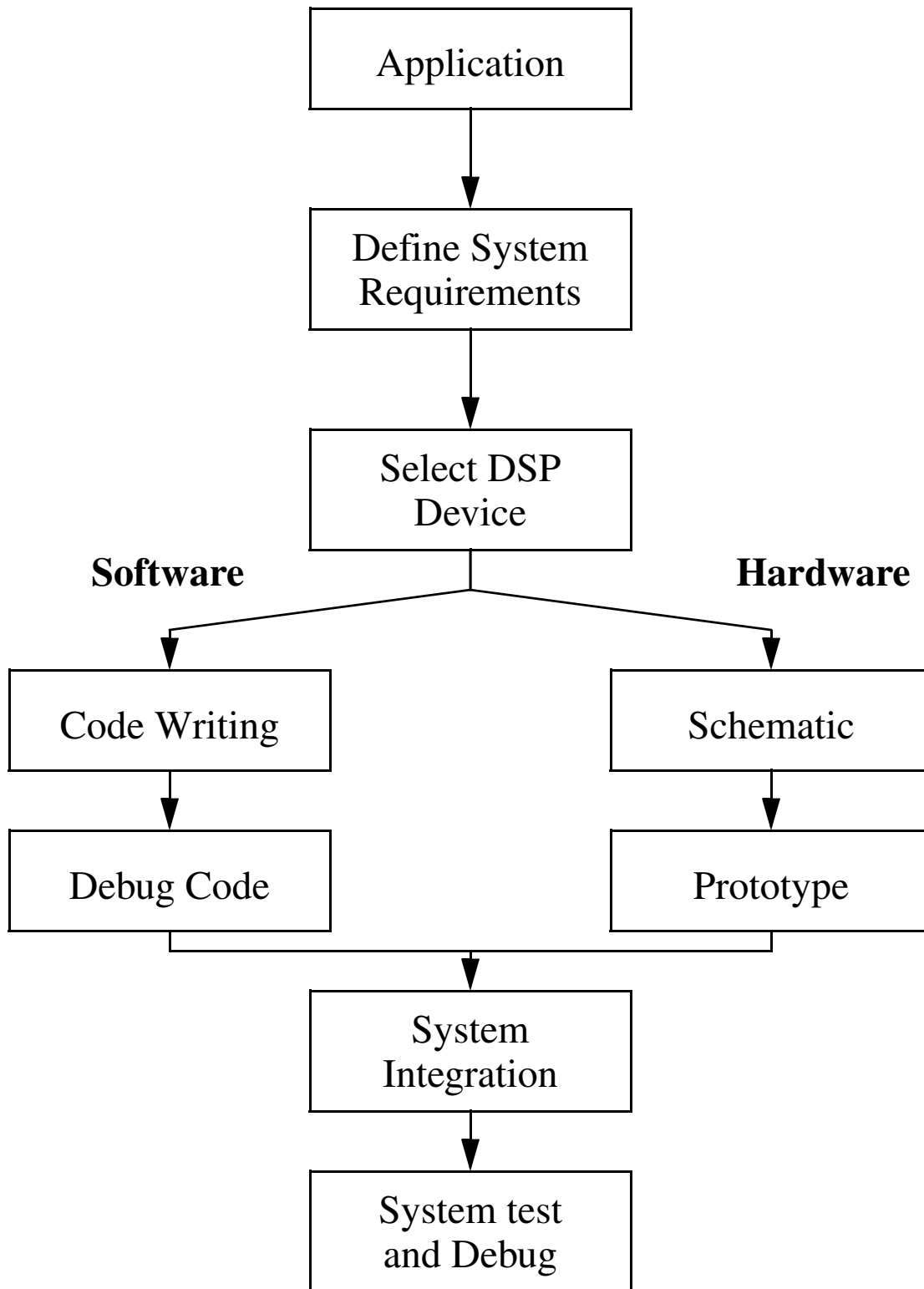
- Hardware alternatives, FPGA, ASIC
- Using a general purpose DSP
- Fixed-Point DSPS
  - Fixed-point arithmetic
  - Quantization effects and scaling in fixed point
  - Typically a lower power design
  - Typically a lower cost design
- Floating Point DSPS
  - A more rapid design due to the ease of floating point algorithm implementation
  - With the powerful C6x family a few floating point calculations can easily be software emulated on the C62x or C64x
  - With the C67x the floating point units are available, so efficient floating point code can be easily written
- Accessing memory resources
- Integration of peripheral devices
- A very practical alternative today is to consider a hybrid FPGA/general purpose DSP architecture

---

1. John Groe and Lawrence Larsen, *CDMA Mobile Radio Design*, Artech House, Boston, MA, 2000.

1. Craig Marven and Gillian Ewers, *A Simple Approach to Digital Signal Processing*, John Wiley, 1996. ISBN 0-471-15243-9.

# DSP System Design Flow (Marven & Ewers)



- Application Design
  - Software design
  - Generating assembler source and using C where possible
  - Testing the code
  - Hardware design
  - System integration

## Course Laboratory Foundations

- Of primary focus is the ZOOM OMAP-L138/C6748 DSK (open multimedia application platform = OMAP)
- TI TMS320C6713 DSP Starter Kit (DSK)
  - In the past we have used the C6713 DSK with CCS 3.3; this target is supported by the text, and can be programmed using CCS 4.2 (in theory CCS 5.3 but have not tried yet)
- In the fixed-point arena, we will also spend some time with the C55x platform on CCS 5.3, specifically either the C5505 or C5515
- The OMAP3530 based *Beagleboard* is available for the final project, for those interested in embedded Linux development
- In the future we may have a second daughter card which is suitable for software radio work involving in-phase/quadrature (I/Q) sampling and/or zero intermediate frequency (IF) processing



- Zero IF processing is becoming more popular in wireless chip sets, so this front-end processing approach and others will hopefully be part of this class in the future

## Software

- TI integrated development environment *Code Composer Studio*; CCS 5.3 Platinum (all cores) which supports TI DSPs and MSP430 devices
  - The above IDE is based *Eclipse*
- MATLAB release 2012b with the following enhancement for DSP
  - Signal processing toolbox
  - DSP System Toolbox, (includes Filter Design Toolbox)
  - Fixed-Point Toolbox
  - The text supported *MATLAB Real-Time Interface* (documentation to be released very soon)
- *GoldWave* or *Audacity* shareware for .wav audio file manipulation

## Test Equipment

- Agilent two-channel 300 MHz digital scope (2 Gsps)
- Agilent function/arbitrary waveform generators; 15 MHz and 80 MHz models
- Agilent 10 Hz – 500 MHz spectrum/vector network analyzers

