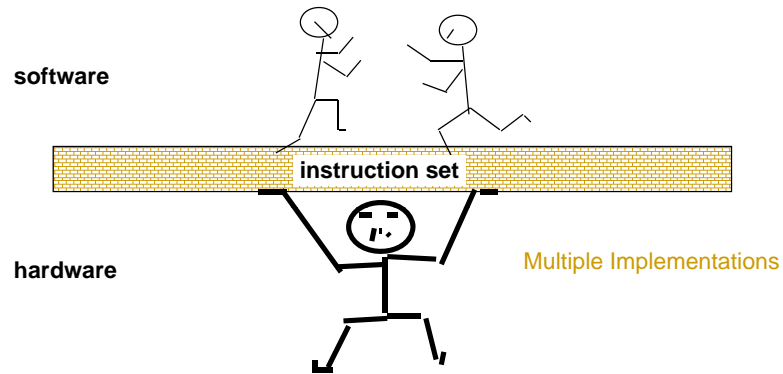


Instruction Set: a Critical Interface

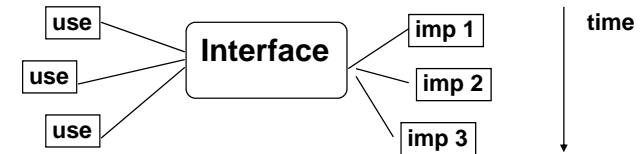


Specifies requirements for binary compatibility across implementations
 ----> New design of inst. Set architecture is not common

Interface Design

A good interface:

- Lasts through many implementations (portability, compatibility)
- Is used in many different ways (generality)
- Provides **convenient** functionality to higher levels
- Permits an **efficient** implementation at lower levels

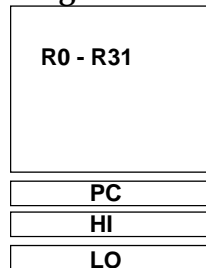


MIPS I Instruction Set Architecture

Instruction Categories

Load/Store
 Computational
 Jump and Branch
 Floating Point coprocessor
 Memory Management
 Special

Registers



Instruction Formats: all 32 bits wide

OP	rs	rt	rd	sa	funct
OP	rs	rt	immediate		
OP	jump target				

Inst. Set Design Goal

- Easy to build H/W
- Easy to build Compiler
- Maximize Performance
- Minimize Cost

RISC vs. CISC

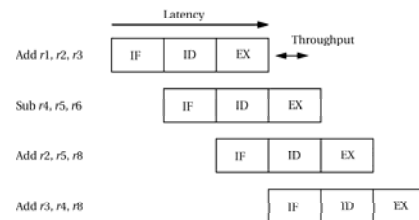
- Make common cases fast!!

Memory Related Issues

- Addressing Unit
- Little vs. Big Endian
- Aligned vs. Unaligned Accesses

Performance as a design metric

- Performance = speed:
 - Latency.
 - Throughput.
- Average vs. peak performance.
- Worst-case and best-case performance.



Other metrics

- Cost (area).
- Energy and power.
- Predictability.
- Security.

Other axes of comparison

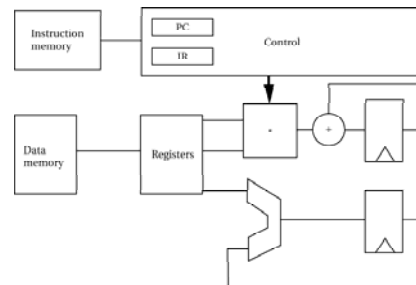
- RISC vs. CISC---Instruction set style.
- Instruction issue width.
- Static vs. dynamic scheduling for multiple-issue machines.
- Vector processing.
- Multithreading.

Embedded vs. general-purpose processors

- Embedded processors may be optimized for a category of applications.
 - Customization may be narrow or broad.
- We may judge embedded processors using different metrics:
 - Code size.
 - Memory system performance.
 - Predictability.

Digital signal processors

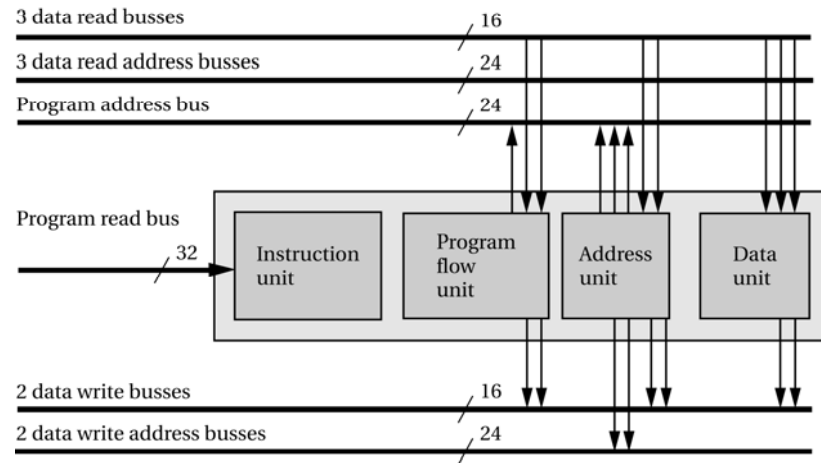
- First DSP was AT&T DSP16:
 - Hardware multiply-accumulate unit.
 - Harvard architecture.
- Today, DSP is often used as a marketing term.
- Modern DSPs are heavily pipelined.



Example: TI C5x DSP

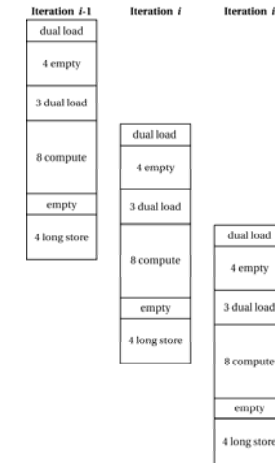
- 40-bit arithmetic unit (32-bit values with 8 guard bits).
- Barrel shifter.
- 17 x 17 multiplier.
- Comparison unit for Viterbi encoding/decoding.
- Single-cycle exponent encoder for wide-dynamic-range arithmetic.
- Two address generators.

TI C55x microarchitecture



TI C55x DCT co-processor

- Performs 1-D 8x8 DCT.
- Pipelines several DCT iterations.

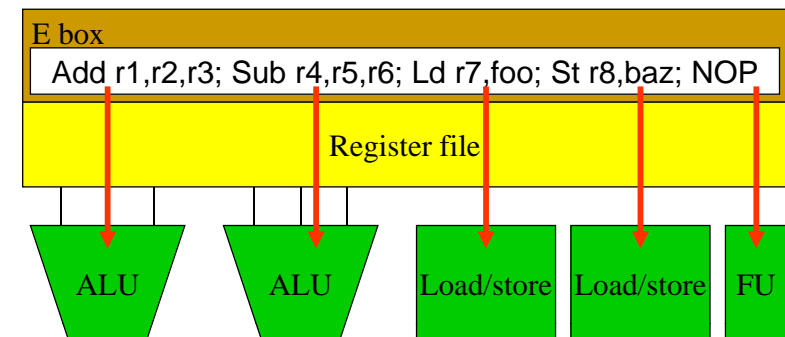


Parallelism extraction

- **Static:**
 - Use compiler to analyze program.
 - Simpler CPU.
 - Can make use of high-level language constructs.
 - Can't depend on data values.
- **Dynamic:**
 - Use hardware to identify opportunities.
 - More complex CPU.
 - Can make use of data values.

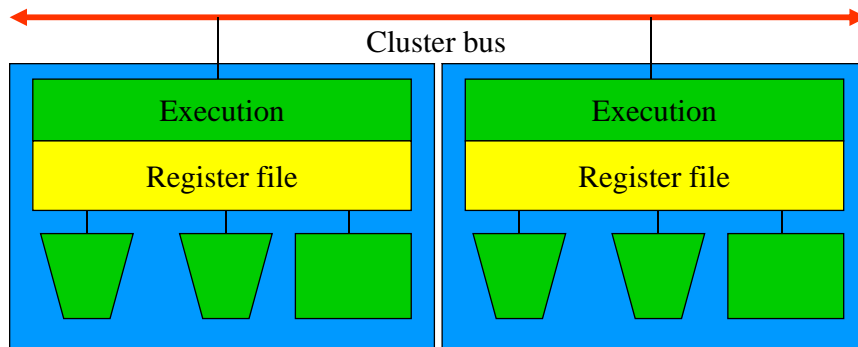
Simple VLIW architecture

- Large register file feeds multiple function units.



Clustered VLIW architecture

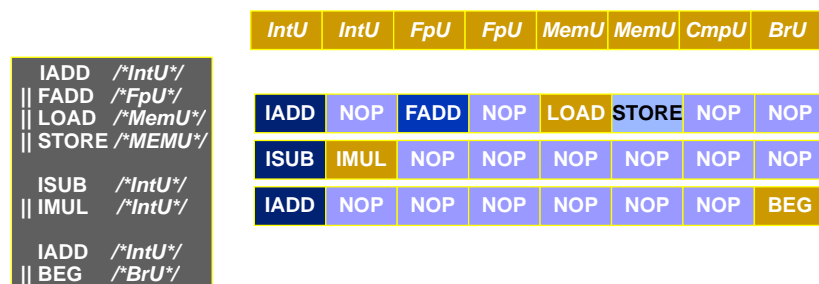
- Register file, function units divided into clusters.



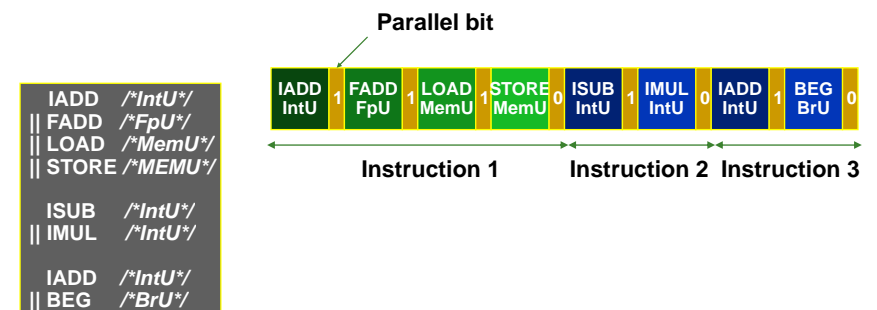
TI C62/C67

- Targeted for communication applications
 - multichannel modem
 - multichannel vocoding for telephony and wireless
 - single and multichannel ADSL modems
 - Imaging
- Up to 8 instructions/cycle (based on VLIW)
- 32 32-bit registers.
- Function units:
 - Two multipliers.
 - Six ALUs.
- All instructions execute conditionally.

VLIW instruction encoding: uncompressed



VLIW instruction encoding: compressed



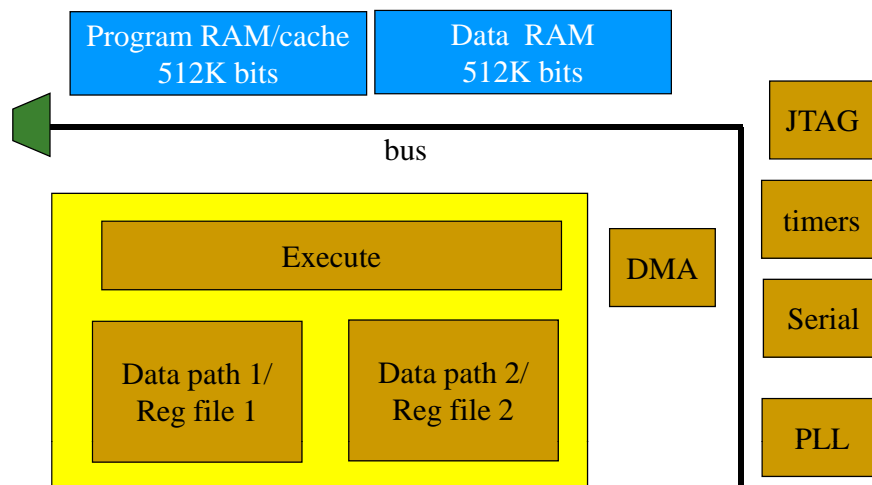
TI C6x data operations

- 8/16/32-bit arithmetic.
- 40-bit operations.
- Bit manipulation operations.

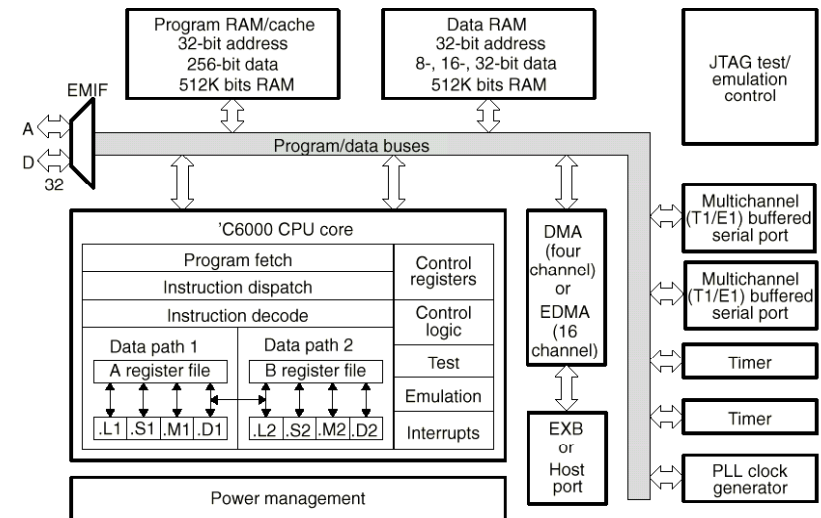
C6x system

- On-chip RAM.
- 32-bit external memory: SDRAM, SRAM, etc.
- Host port.
- Multiple serial ports.
- Multichannel DMA.
- 32-bit timer.

C6x block diagram



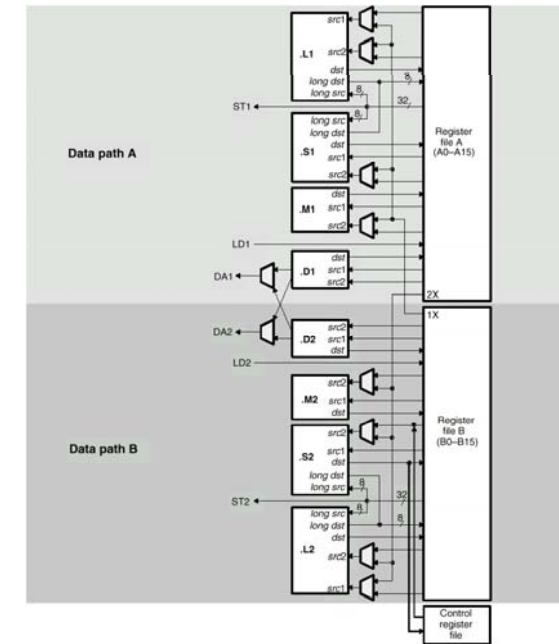
C62x Block



C6x data paths

- General-purpose register files (A and B, 16 words each).
- Eight function units:
 - .L1, .L2, .S1, .S2, .M1, .M2, .D1, .D2
- Two load units (LD1, LD2).
- Two store units (ST1, ST2).
- Two register file cross paths (1X and 2X).
- Two data address paths (DA1 and DA2).

C6x CPU Data Path



C6x function units

- .L
 - 32/40-bit arithmetic.
 - Leftmost 1 counting.
 - Logical ops.
- .S
 - 32-bit arithmetic.
 - 32/40-bit shift and 32-bit field.
 - Branches.
 - Constants.
- .M
 - 16 x 16 multiply.
- .D
 - 32-bit add, subtract, circular address.
 - Load, store with 5/15-bit constant offset.

Instruction Set

- Saturated operations
- Subtract conditional
- Bit counting
- Integer comparison
- Dual 16-bit Pair arithmetic
- Bit manipulation
- Constant generation

Instruction Packing

- In a typical VLIW, each instruction would correspond to a particular functional unit.
 - Idle functional unit == NOP in the corresponding slot
- Distinguish *Fetch Packet* from *Execution Packet*
 - The least-significant bit of C62x instruction called *p-bit*
 - Advantage: reduced code size

Pipeline

- Three Stages
 - Fetch
 - Decode
 - Execute
- Each stage subdivided into phases
 - Fetch: four phases
 - Decode: two phases
 - Execute: ten phases
- Pipeline implications
 - Multiply: 1 delay slot
 - Load: 4 delay slots
 - Branch: 5 delay slots

C6201 Memory Architecture

- External Memory Interface
- DMA Controller
- 64K bytes of on-chip program memory
 - mapped memory
 - direct mapped cache
- 64K bytes of interleaved data memory

Code-Scheduling Optimization

- Intrinsic functions
- Software pipelining
- If conversion/predicated execution
- Memory-bank disambiguation
- Memory address-dependence elimination

- Compiler achieves 72-82% of the performance of hand-optimized implementations

Superscalar processors

- Instructions are dynamically scheduled.
 - Dependencies are checked at run time in hardware.
- Used to some extent in embedded processors.
 - Embedded Pentium is two-issue in-order.

Multimedia Extensions

- Most GPPs have some form of ISA extensions for media processing
 - MAX-2 (HP PA-RISC), MMX (Intel x86), VIS (UltraSPARC) and MDMX (MIPS V)
- Target applications: signal-processing applications
 - video/image processing, graphics, communications, etc
- Underlying principle: SIMD/subword parallelism
 - A *subword* execution model based on *packed data types*

Packed Data Types

Packed byte (eight 8-bit quantities)

quant7	quant6	quant5	quant4	quant3	quant2	quant1	quant0
--------	--------	--------	--------	--------	--------	--------	--------

Packed half-word (four 16-bit quantities)

quantity3	quantity2	quantity1	quantity0
-----------	-----------	-----------	-----------

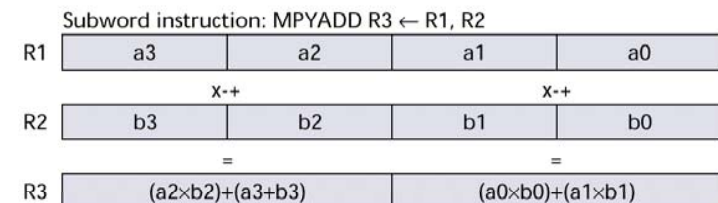
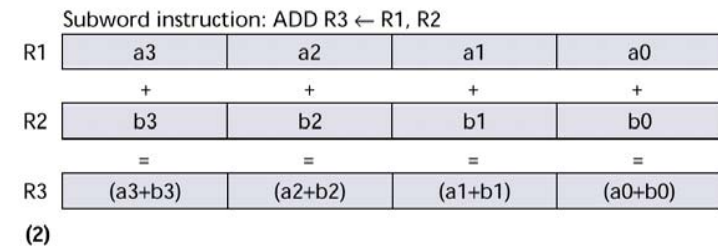
Packed word (two 32-bit quantities)

quantity1	quantity0
-----------	-----------

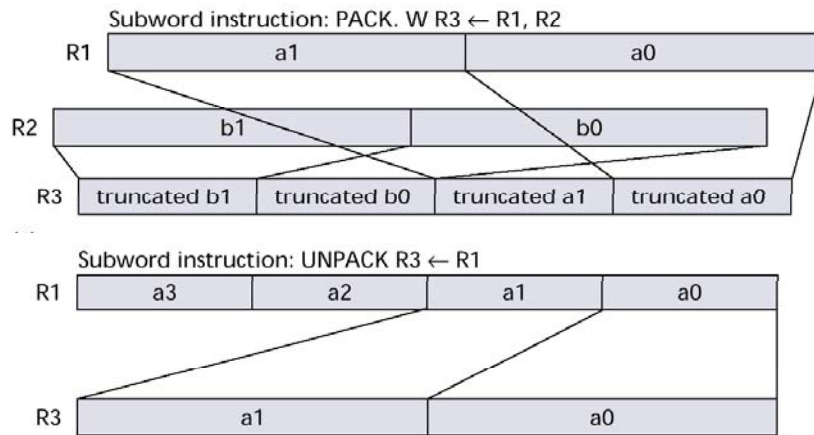
Long-word (native data type, a 64-bit quantity)

quantity0

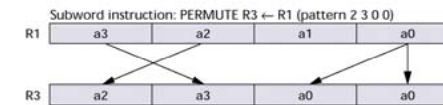
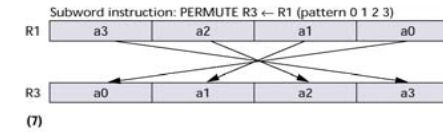
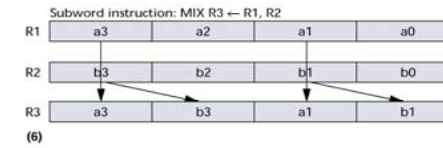
Subword Arithmetic Instructions



Data Rearrangement and Data Conversion Inst.



Data Rearrangement Instructions



Summary of MM Extensions

Instruction category	Alpha MAX	MIPS MDMX	PA-RISC MAX2	Power PC	SPARC VIS	MMX
Add/subtract		8B,4H	4H		4H,2W	8B, 4H, 2W
Saturating add/sub		8B,4H	4H			8B, 4H, 2W
Multiply		8B,4H			4B/H	4H
Compare	8B (>=)	8B,4H (=,<,<=)			4H,2W (=,not=,>,<=)	8B, 4H, 2W (=, >)
Shift right/left		8B,4H	4H			
Shift right arithmetic		4H	4H			4H, 2W (logical left & right)
Multiply and add		8B,4H				4H
Shift and add (saturating)			4H			
And/or/xor	8B,4H,2W	8B,4H,2W	8B,4H,2W		8B,4H,2W	8B,4H,2W (ANDNOT)
Absolute difference	8B				8B	
Max/min	8B, 4W	8B,4H				
Pack (2n bits → n bits)	2W→2B, 4H→4B	2*2W→4H, 2*4H→8B	2*4H→8B		2W→2H, 2W→2B, 4H→4B	2W→2H 4H→4B
Unpack/merge	2B→2W, 4B→4H	2*4B→8B, 2*2H→4H			4B→4H, 2*4B→8B	2*4B→8B 2*2H→4H 2*W→2W
Permute/shuffle		8B,4H	4H			
Register sets	Integer	Fl. Pt. + 192b Acc.	Integer		Fl. Pt.	Fl. Pt.

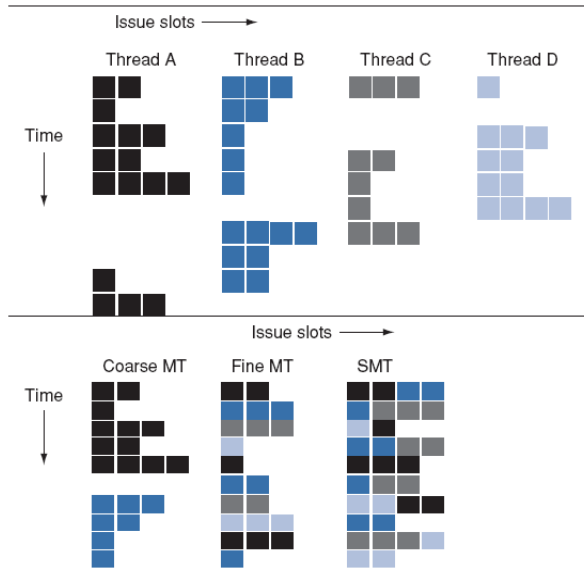
Multithreading & Multiprocessor

- Low-level parallelism mechanism.
- Hardware multithreading alternately fetches instructions from separate threads.
- Simultaneous multithreading (SMT) fetches instructions from several threads on each cycle.

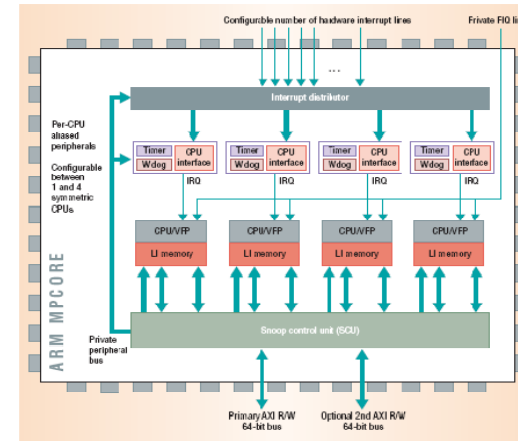
Multithreading

- Multiple threads to share the functional units of a single processor in an overlapping fashion

TLP → ILP



ARM MPCore



- 4 ARM11 CPUs
 - ARMv6K ISA
 - Vector floating-point (VFP)
 - 32KB private L1 Inst-cache, Data-cache
- 1MB unified L2 cache
- Snoop control unit
 - Maintain cache coherence
 - Use MESI protocol
- Interrupt distributor
 - SW controllable inter-processor interrupt (IPI)