Introduction to Digital Signal Processors (DSPs)

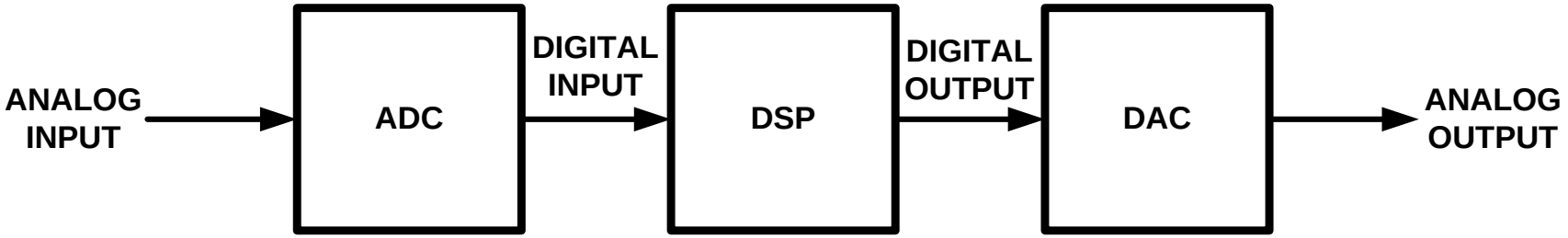
- Architectures of ADSP
- TMS series of processor

Outline/objectives

- Identify the most important DSP processor architecture features and how they relate to DSP applications
- Understand the types of code appropriate for DSP implementation

What is a DSP?

- A specialized microprocessor for real-time DSP applications
  - Digital filtering (FIR and IIR)
  - FFT
  - Convolution, Matrix Multiplication etc



Hardware used in DSP

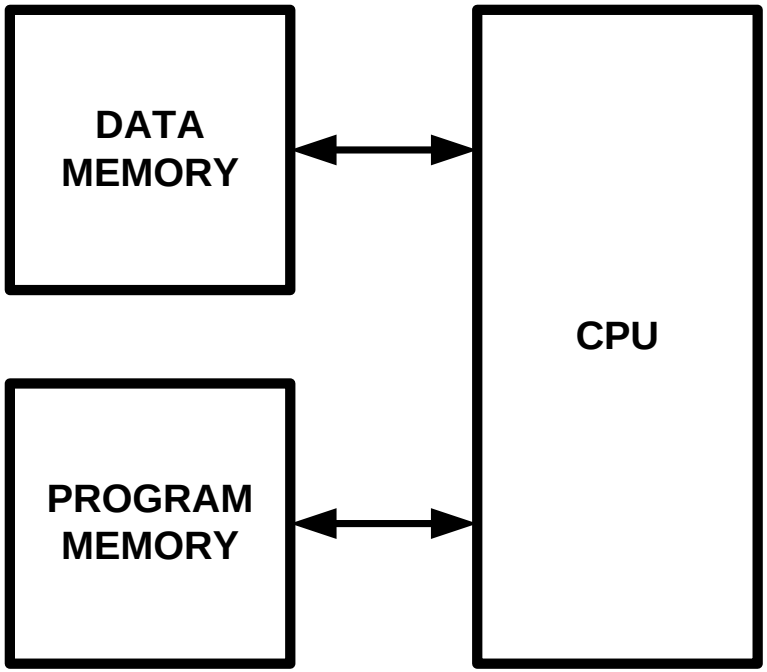
	ASIC	FPGA	GPP	DSP
Performance	**Very High**	**High**	**Medium**	**Medium High**
Flexibility	**Very low**	**High**	**High**	**High**
Power consumption	**Very low**	**low**	**Medium**	**Low Medium**
Development Time	**Long**	**Medium**	**Short**	**Short**

Common DSP features

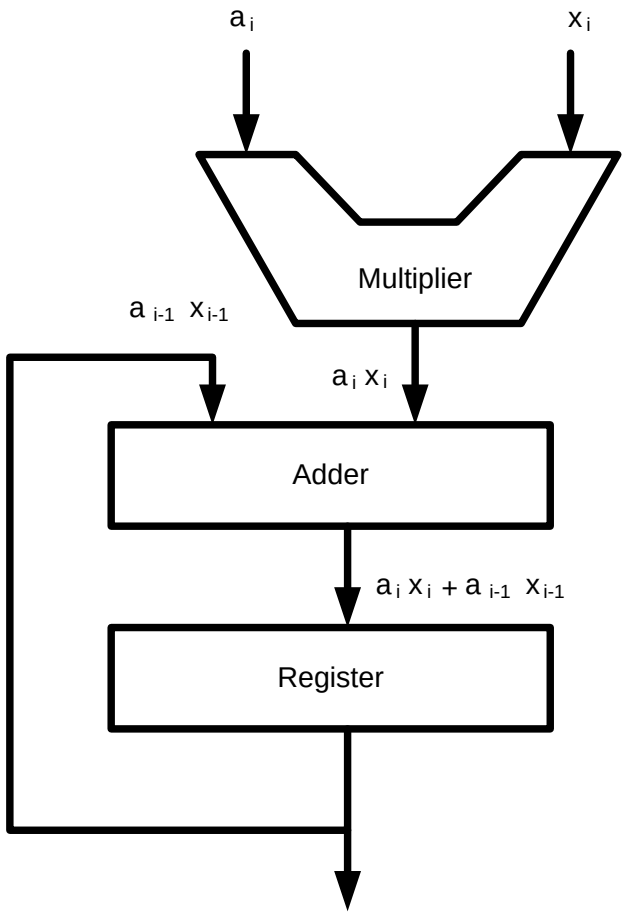
- Harvard architecture
- Dedicated single-cycle Multiply-Accumulate (MAC) instruction (hardware MAC units)
- Single-Instruction Multiple Data (SIMD) Very Large Instruction Word (VLIW) architecture
- Pipelining
- Saturation arithmetic
- Zero overhead looping
- Hardware circular addressing
- Cache
- DMA

Harvard Architecture

- Physically separate memories and paths for instruction and data



Single-Cycle MAC unit

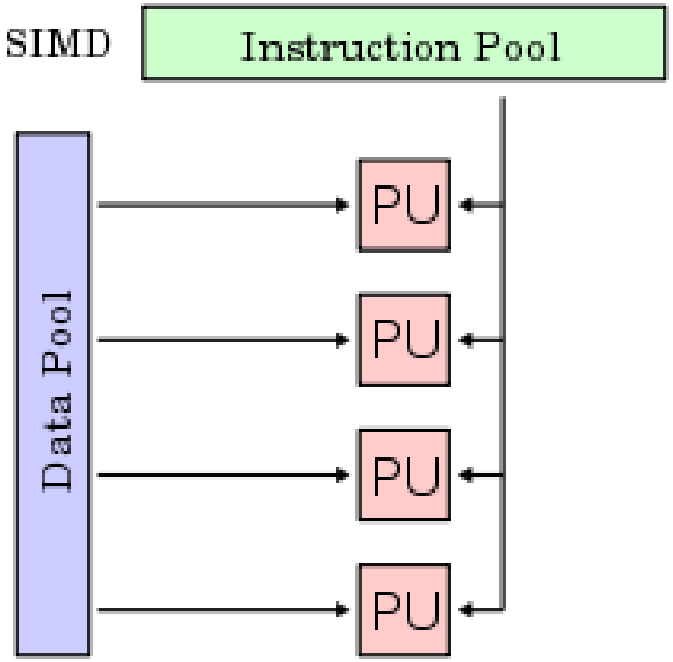


$$\sum_{i=0}^{n}(a_i x_i)$$

Can compute a sum of *n*-products in *n* cycles

Single Instruction - Multiple Data (SIMD)

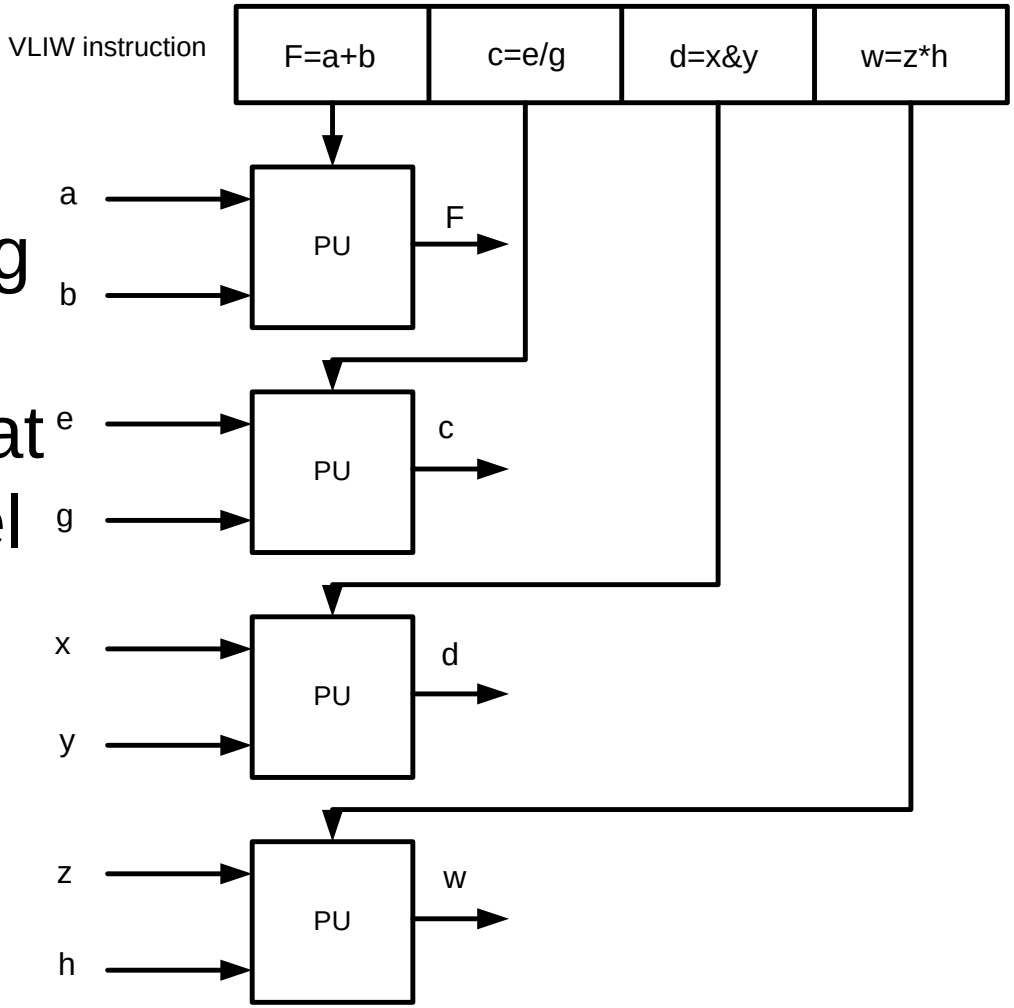
- A technique for data-level parallelism by employing a number of processing elements working in parallel



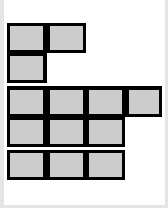
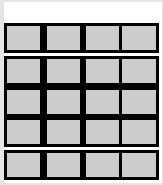
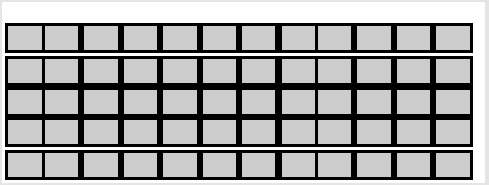
Very Long Instruction Word (VLIW)

- A technique for instruction-level parallelism by executing instructions without dependencies (known at compile-time) in parallel
- Example of a single VLIW instruction:

F=a+b; c=e/g; d=x&y; w=z*h;



CISC vs. RISC vs. VLIW

ARCHITECTURE CHARACTERISTIC	CISC	RISC	VLIW
INSTRUCTION SIZE	Varies	One size, usually 32 bits	One size
INSTRUCTION FO RMAT	Field placement varies	Regular, consistent placement of fields	Regular, consistent placement of fields
INSTRUCTION SEMANTICS	Varies from simple to complex; possibly many dependent operations per instruction	Almost always one simple operation	Many simple, independent operations
REGISTERS	Few, sometimes special	Many, general-purpose	Many, general-purpose
MEMORY REFE RENCES	Bundled with operations in many different types of instructions	Not bundled with operations, i.e., load/store architecture	Not bundled with operations, i.e., load/store architecture
HARDWARE DESIGN FOCUS	Exploit microcoded implementations	Exploit implementations with one pipeline and & no microcode	Exploit implementations with multiple pipelines, no microcode & no complex dispatch logic
PICTURE OF FIVE TYPICAL INSTRU CTIONS = 1 BYTE			

Pipelining

- DSPs commonly feature deep pipelines
- TMS320C6x processors have 3 pipeline stages with a number of phases (cycles):
  - Fetch
    - Program Address Generate (PG)
    - Program Address Send (PS)
    - Program ready wait (PW)
    - Program receive (PR)
  - Decode
    - Dispatch (DP)
    - Decode (DC)
  - Execute
    - 6 to 10 phases

Zero Overhead Looping

- Hardware support for loops with a constant number of iterations using hardware loop counters and loop buffers
- No branching
- No loop overhead
- No pipeline stalls or branch prediction
- No need for loop unrolling

Direct Memory Access (DMA)

- The feature that allows peripherals to access main memory without the intervention of the CPU
- Typically, the CPU initiates DMA transfer, does other operations while the transfer is in progress, and receives an interrupt from the DMA controller once the operation is complete.
- Can create cache coherency problems (the data in the cache may be different from the data in the external memory after DMA)
- Requires a DMA controller

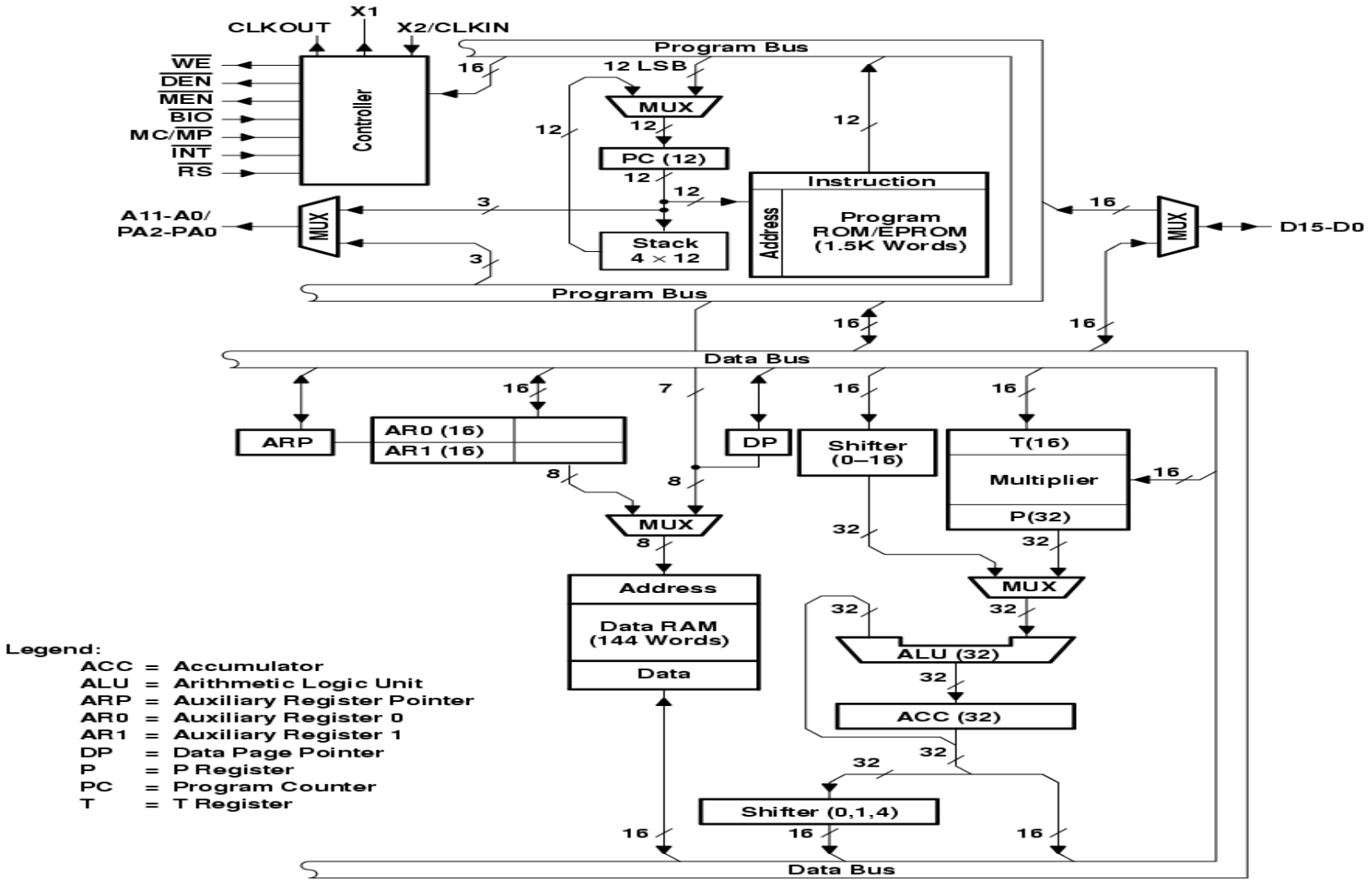
DSP vs. Microcontroller

- DSP
  - Harvard Architecture
  - VLIW/SIMD (parallel execution units)
  - No bit level operations
  - Hardware MACs
  - DSP applications
- Microcontroller
  - Mostly von Neumann Architecture
  - Single execution unit
  - Flexible bit-level operations
  - No hardware MACs
  - Control applications

Texas Instruments TMS320 Family Multiple DSP μP Generations

	First Sample	Bit Size	Clock speed (MHz)	Instruction Throughput	MAC execution (ns)	MOPS	Device density (# of transistors)
Uniprocessor Based (Harvard Architecture)							
TMS32010	1982	16 integer	20	5 MIPS	400	5	58,000 (3μ)
TMS320C25	1985	16 integer	40	10 MIPS	100	20	160,000 (2μ)
TMS320C30	1988	32 flt.pt.	33	17 MIPS	60	33	695,000 (1μ)
TMS320C50	1991	16 integer	57	29 MIPS	35	60	1,000,000 (0.5μ)
TMS320C2XXX	1995	16 integer		40 MIPS	25	80	
Multiprocessor Based							
TMS320C80	1996	32 integer/flt.				2 GOPS 120 MFLOP	MIMD
TMS320C62XX	1997	16 integer		1600 MIPS	5	20 GOPS	VLIW
TMS310C67XX	1997	32 flt. pt.			5	1 GFLOP	VLIW

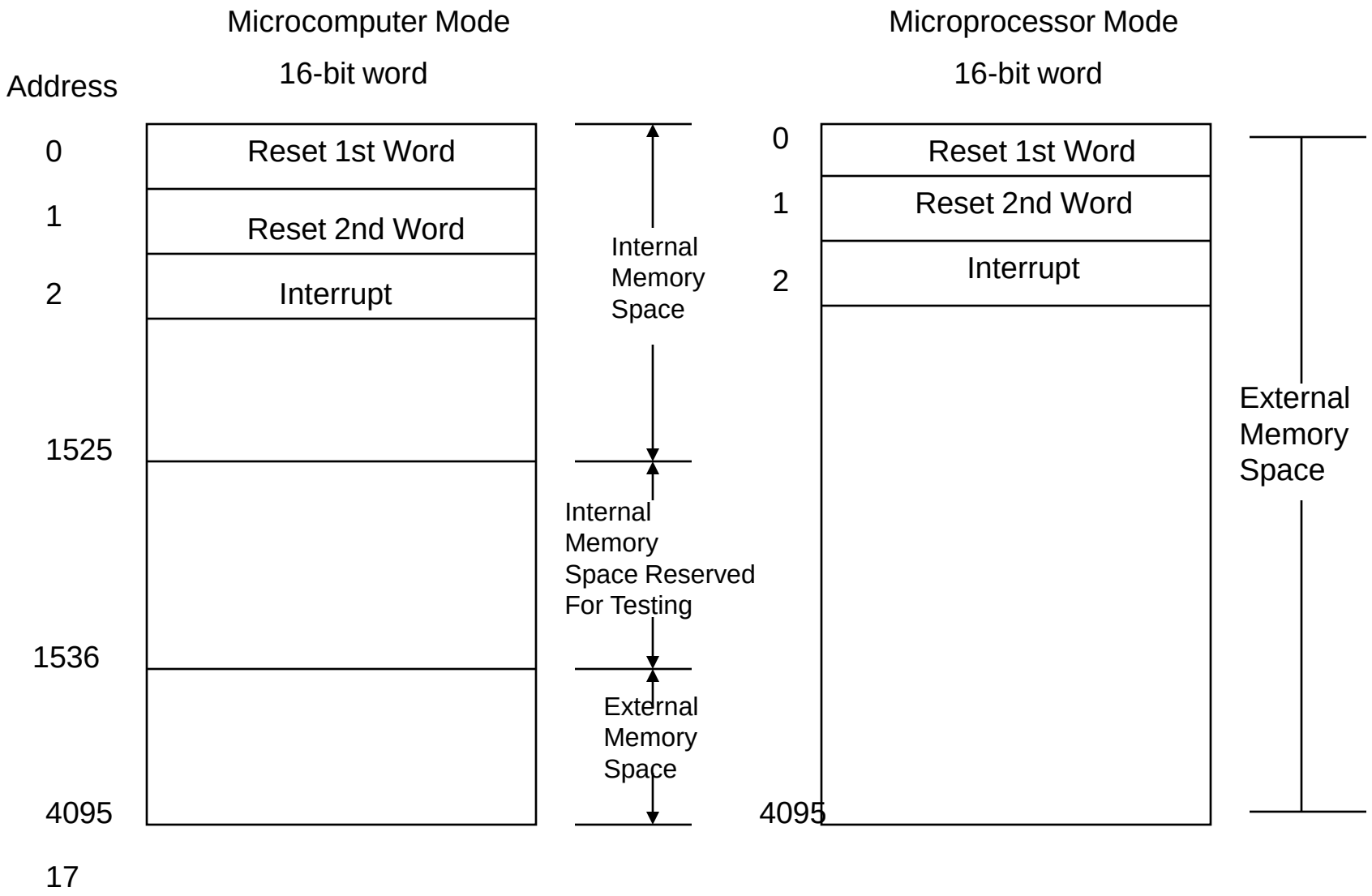
TMS32010 BLOCK DIAGRAM



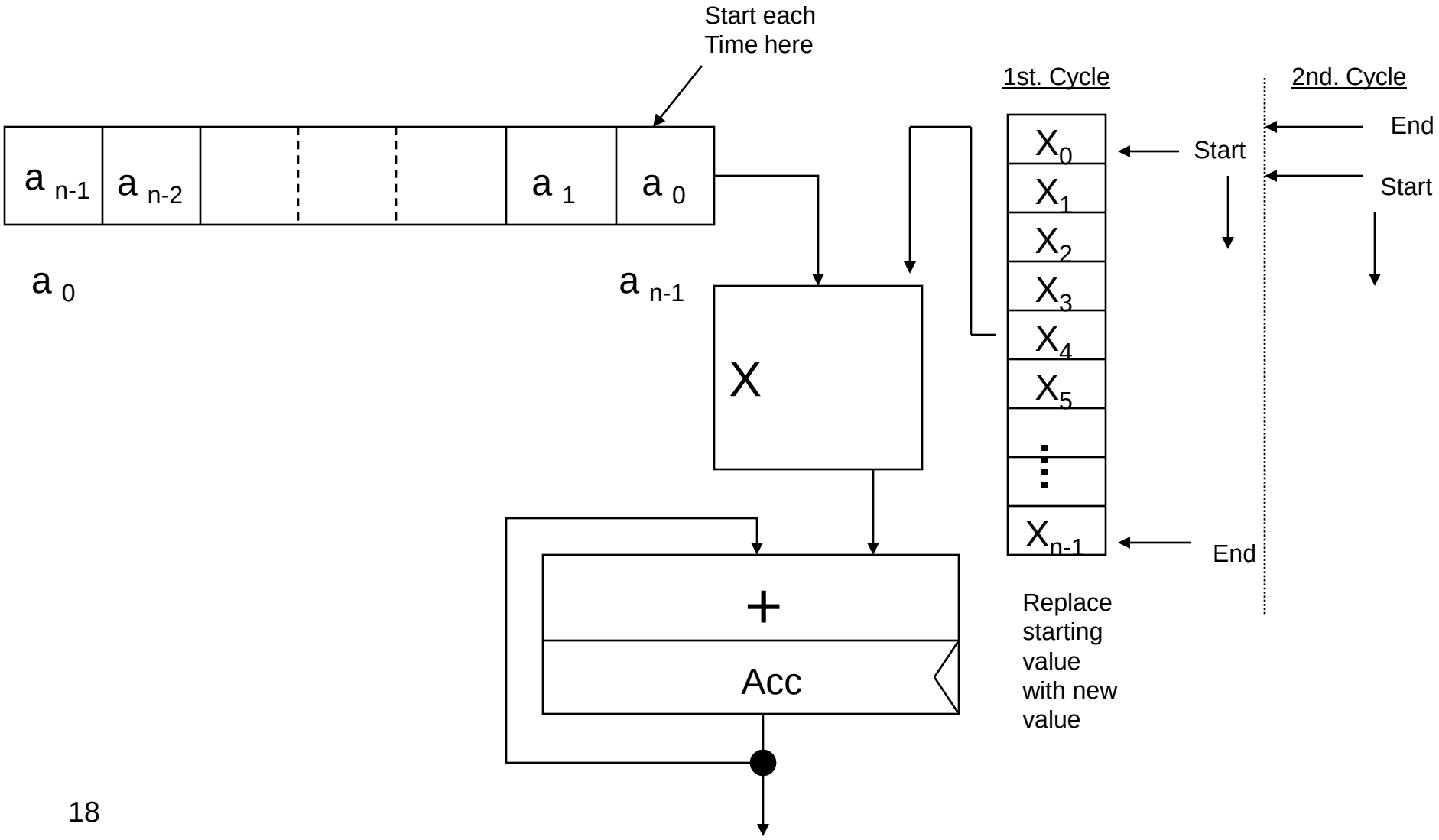
Legend:

ACC	=	Accumulator
ALU	=	Arithmetic Logic Unit
ARP	=	Auxiliary Register Pointer
AR0	=	Auxiliary Register 0
AR1	=	Auxiliary Register 1
DP	=	Data Page Pointer
P	=	P Register
PC	=	Program Counter
T	=	T Register

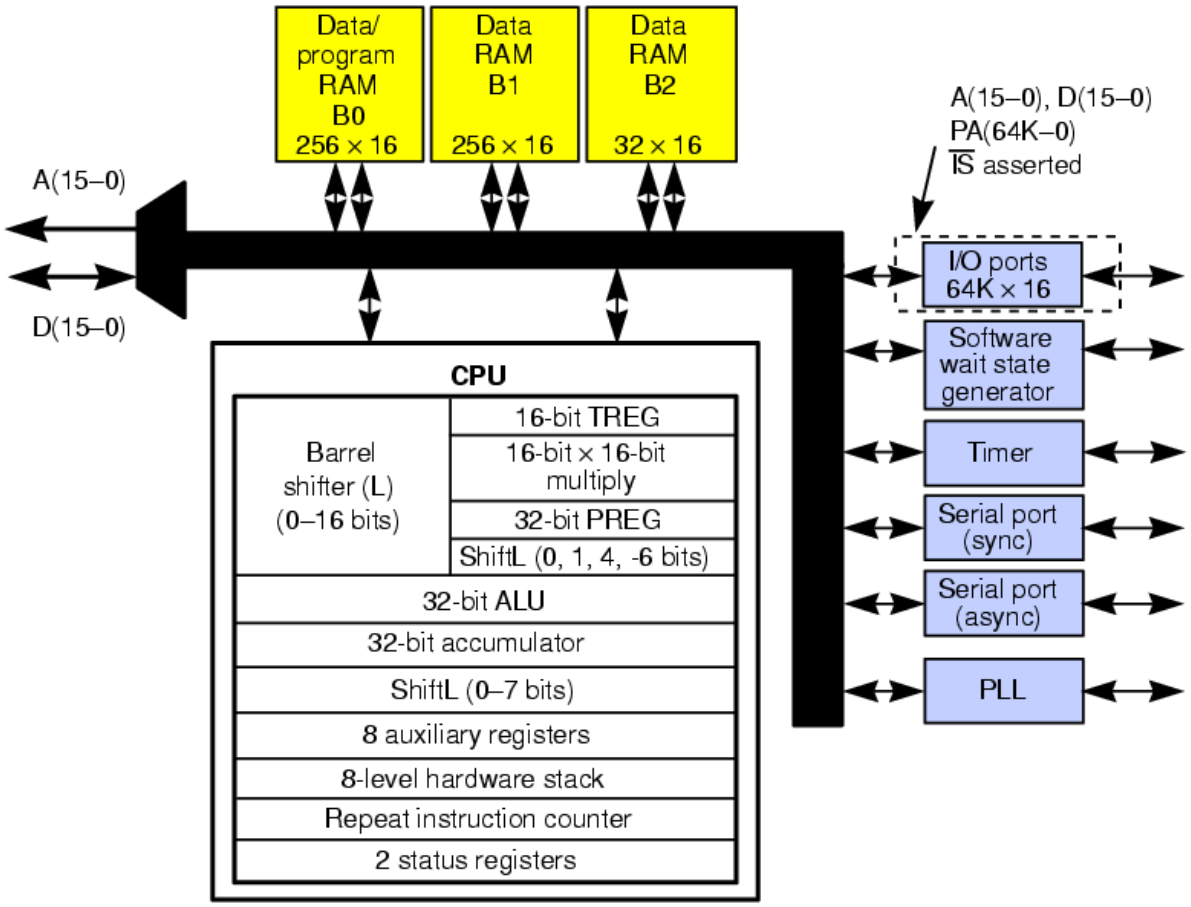
TMS32010 Program Memory Maps



Digital FIR Filter Implementation (Uniprocessor-Circular Buffer)



TMS320C203/LC203 BLOCK DIAGRAM



Third Generation DSP μP Case Study

TMS320C30 Key Features

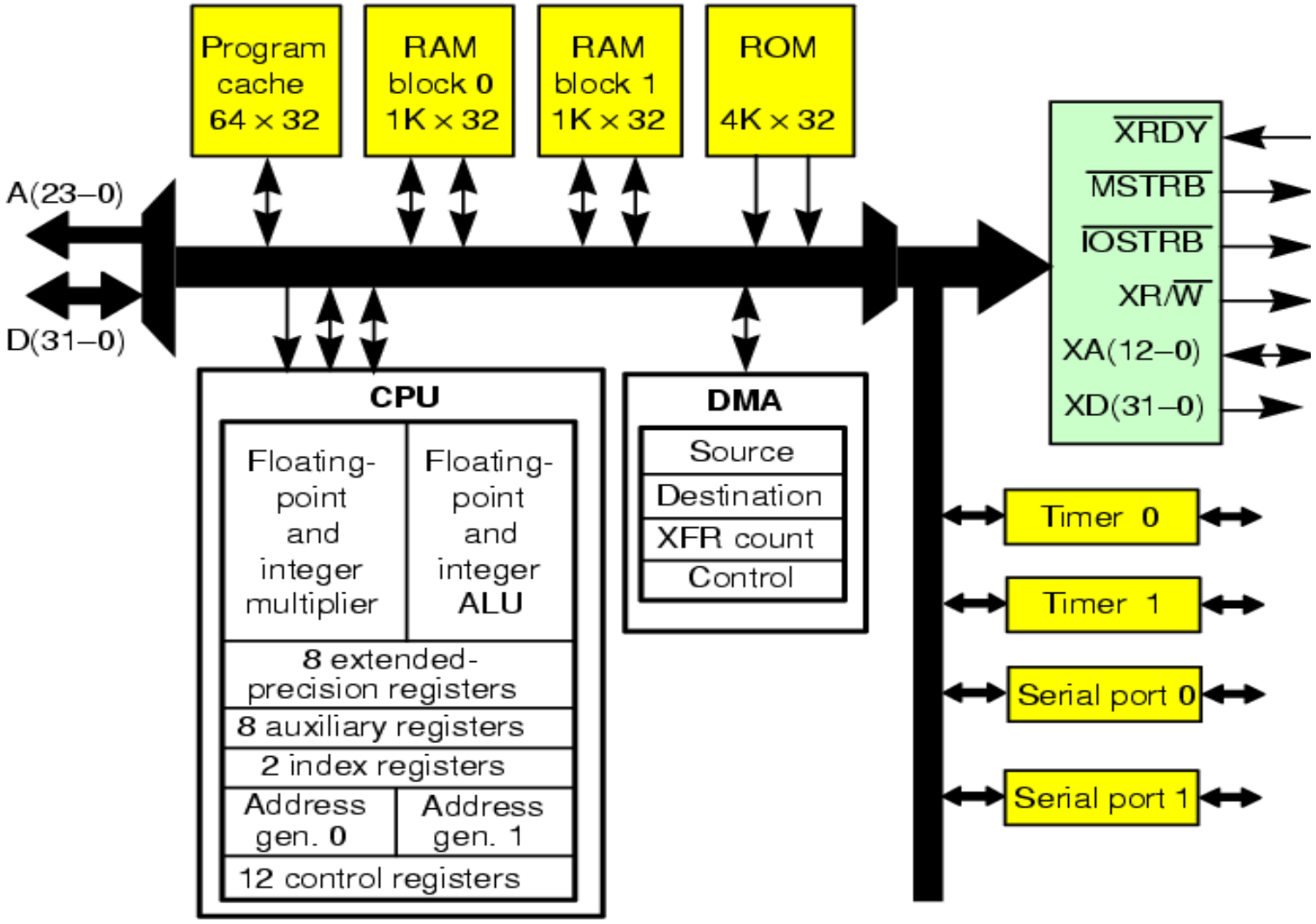
- 60 ns single-cycle instruction execution time
  - 33.3 MFLOPS (million floating-point operations per second)
  - 16.7 MIPS (million instructions per second)
- One 4K x 32-bit single-cycle dual-access on-chip ROM block
- Two 1K x 32-bit single-cycle dual-access on-chip RAM blocks
- 64 x 32-bit instruction cache
- 32-bit instruction and data words, 24-bit addresses
- 40/32-bit floating-point/integer multiplier and ALU
- 32-bit barrel shifter

Third Generation DSP μP Case Study

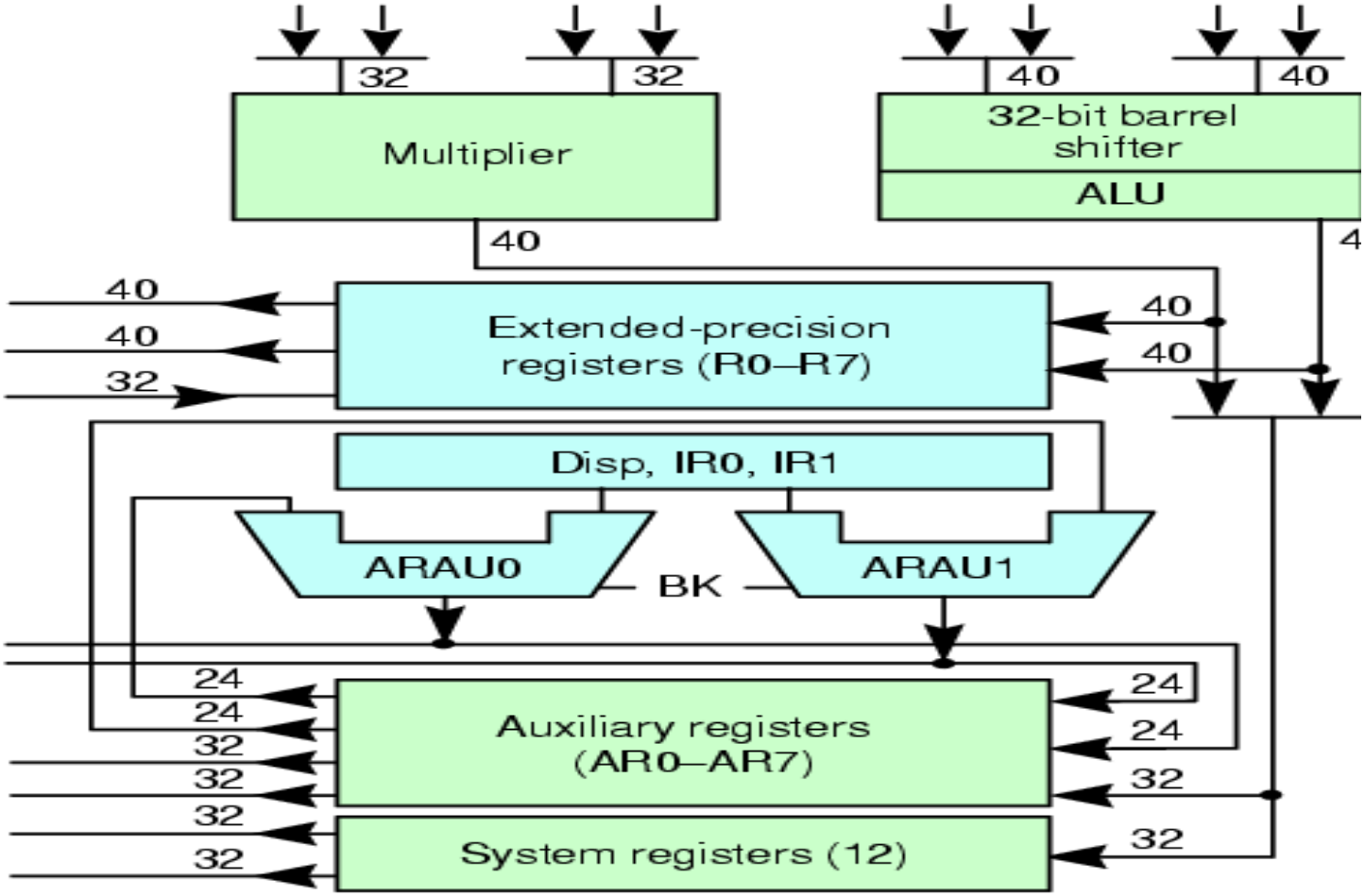
TMS320C30 Key Features *(cont.)*

- Eight extended precision registers (accumulators)
- Two address generators with eight auxiliary registers and two auxiliary register arithmetic units
- On-chip direct memory Access (DMA) controller for concurrent I/O and CPU operation
- Parallel ALU and multiplier instructions
- Block repeat capability
- Interlocked instructions for multiprocessing support
- Two serial ports to support 8/16/32-bit transfers
- Two 32-bit timers
- 1 μ CDMOS Process

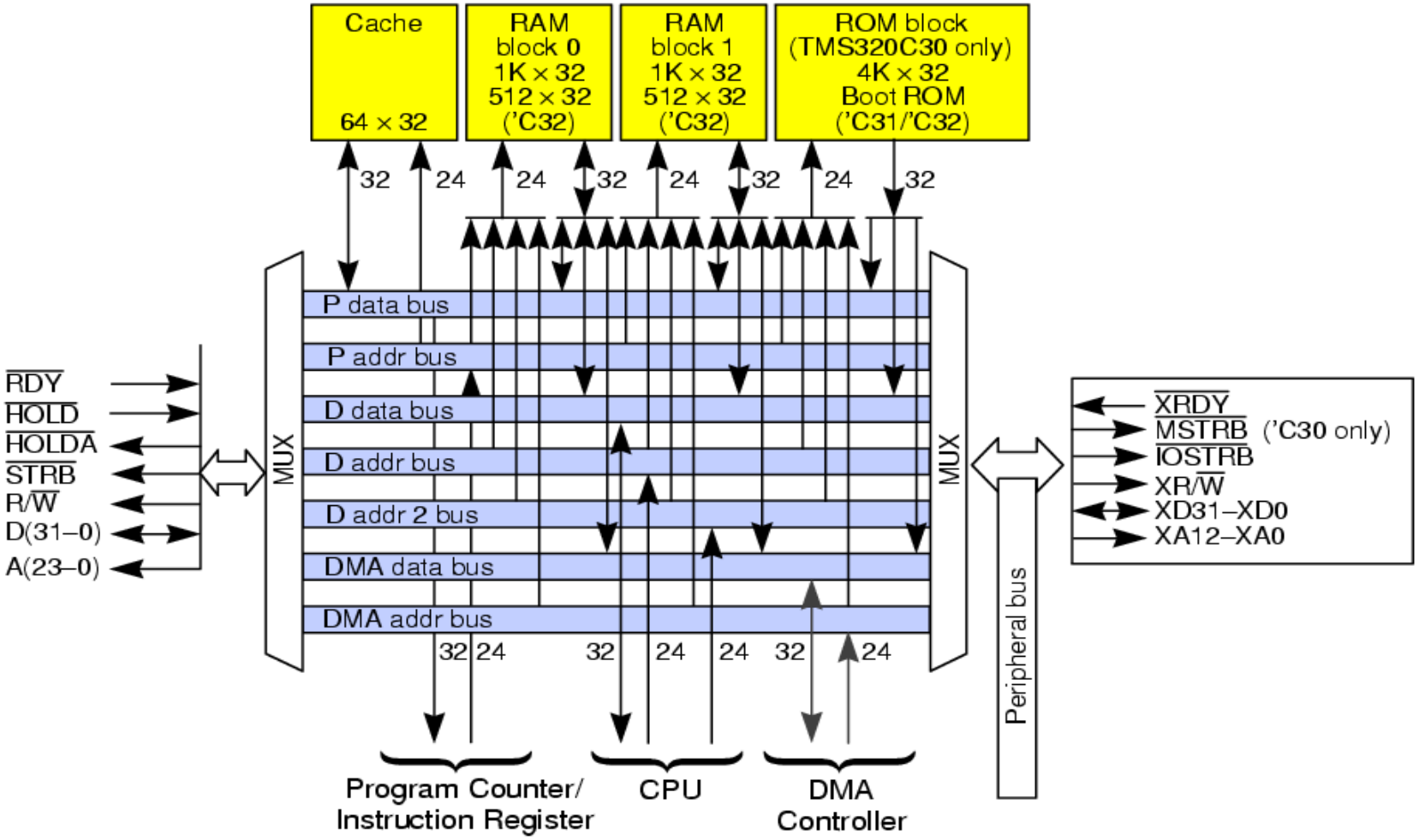
TMS320C30 BLOCK DIAGRAM



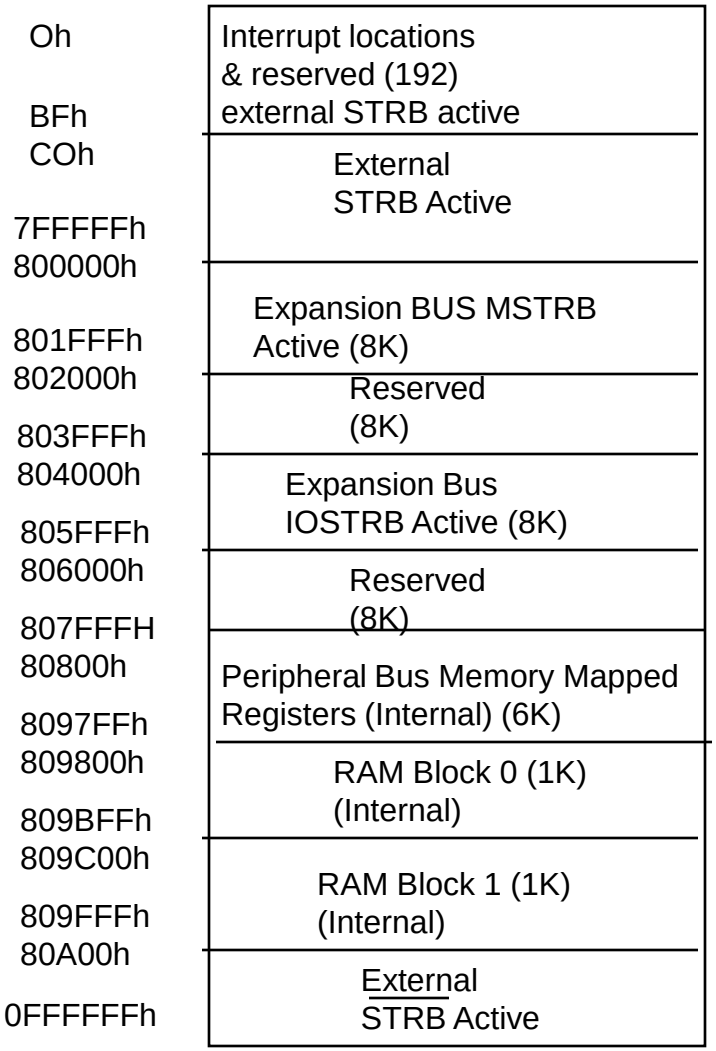
TMS320C3x CPU BLOCK DIAGRAM



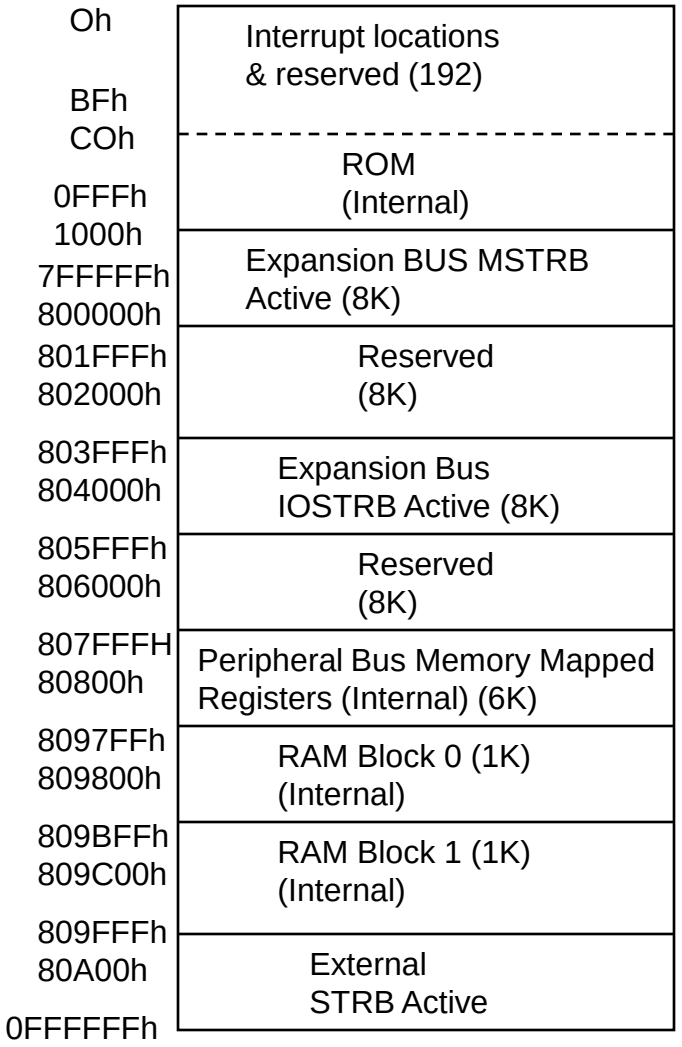
TMS320C3x MEMORY BLOCK DIAGRAM



TMS320C30 Memory Organization



Microprocessor Mode



Microcomputer Mode

Thank You