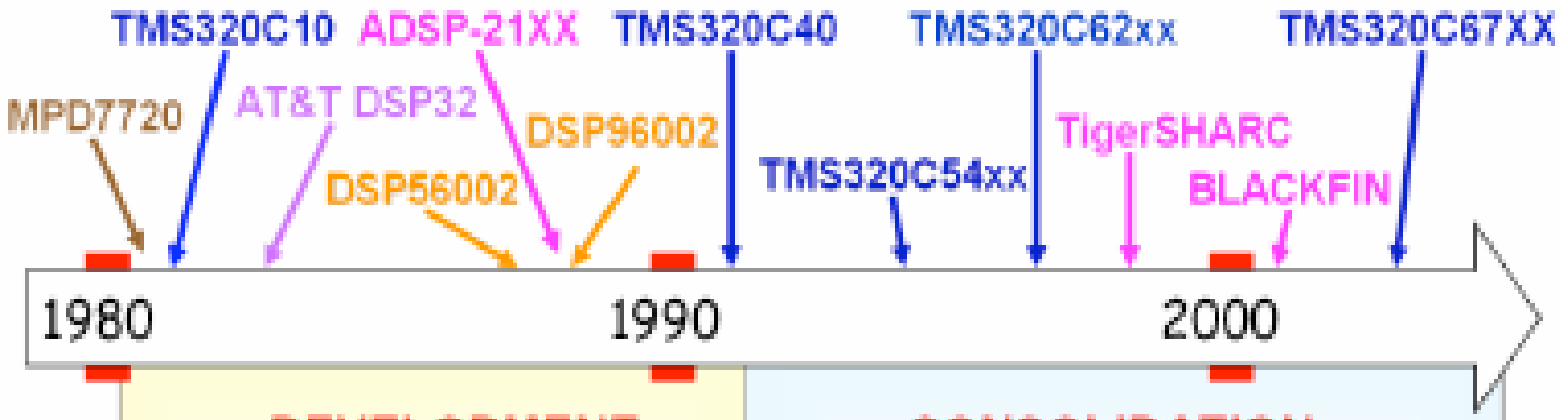
C10. VLIW DSPs

(based on TI educational material)

Outline:

- DSP architecture roadmap
- TMS 320C6000 OVERVIEW
- C6X Architecture
- C6X Instructions
- C6x Programing
- Trends C66X, C67X
- Reference: SPRU731.pdf



DEVELOPMENT

- Harvard architecture
- Data format:
  - early '80s: fixed point
  - late '80s: floating point (often *non* IEEE).
- DMA
- Fixed-width instruction set

CONSOLIDATION

- Parallel architectures
- Many on-chip peripherals
- Multiprocessing support
- Late '90s: improved debug capabilities (ex: TI RTDX)
- Fewer manufacturers
- Wider/few families (code compatibility).
- Specialised families.

Performance

- **Low** : ~ 25 to 50 MHz clock, low cost / power consumption.
- **Mid** : ~ 150 MHz clock, multiprocessing support.
- **High** : Enhanced architectures - *VLIW* (Very Long Instruction Word) or *SIMD* (Single Input Multiple Data).

Trend

- Blurred borderline to µ-processors, µ-controllers.
- FPGAs / SOPCs as alternative / help.

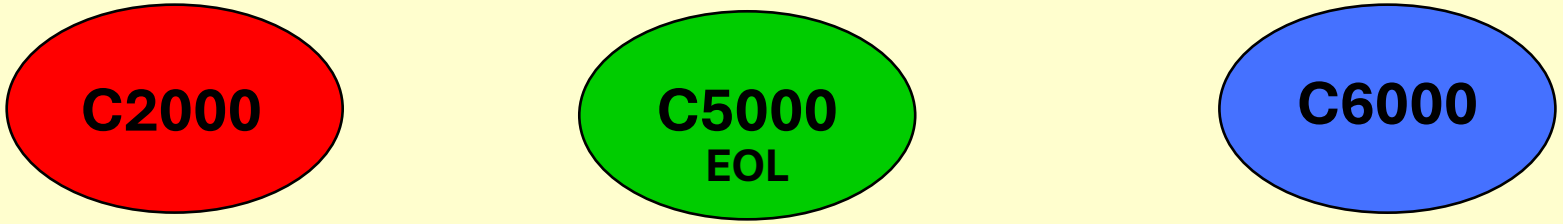
If / which DSP ?

- Global cost & performances ➡ difficult choice.
- Ex: Help from PS DSP Advisory committee.

SOPC=System On Programmable Chips

Texas Instruments' TMS320 family

- Different families and sub-families exist to support different markets



Lowest Cost

Control Systems

- Motor Control
- Storage
- Digital Ctrl Systems

Efficiency

Best MIPS per Watt / Dollar / Size

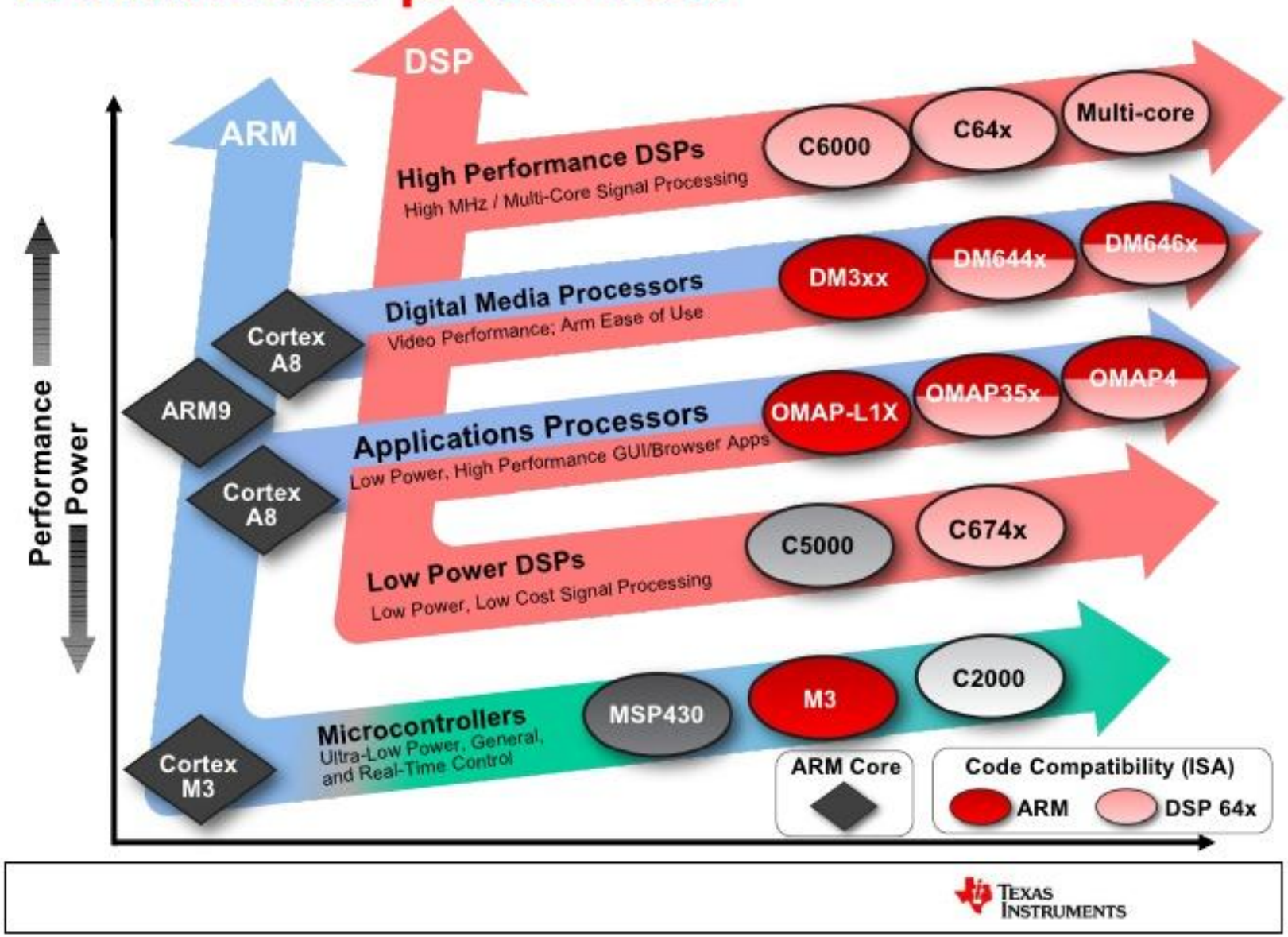
- Wireless phones
- Internet audio players
- Digital still cameras
- Modems
- Telephony
- VoIP

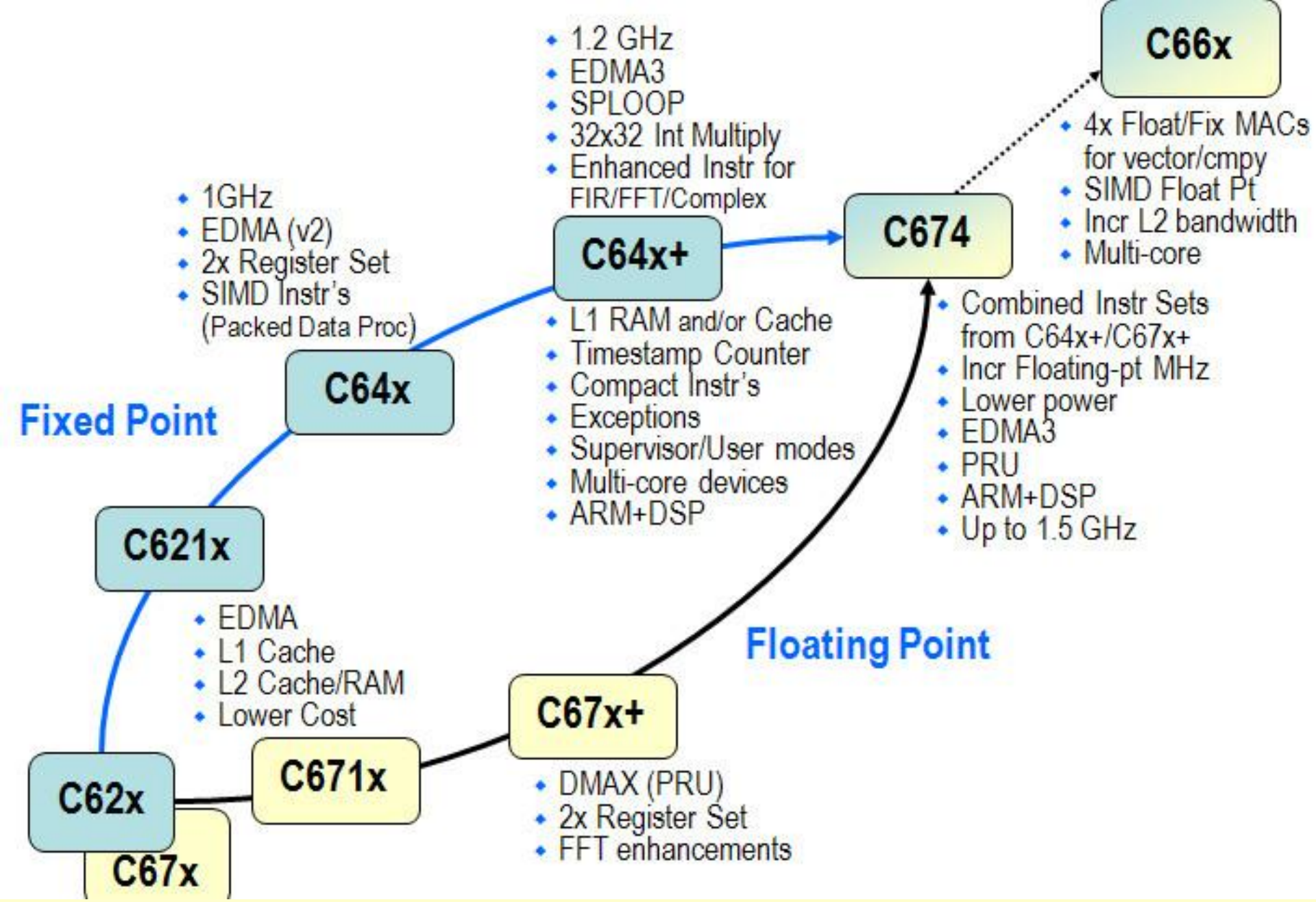
Performance & Best Ease-of-Use

- **Multi Channel and Multi Function App's**
- Comm Infrastructure
- Wireless Base-stations
- DSL
- Imaging
- Multi-media Servers
- Video etc

- TI TMS320 C62x, C64x……..C66x – multicore dsp
- ADI TigerSHARC ADS-TS20x
- Freescale (Motorola) MSC71xx and MSC81xx
- StarCore SC1400 Agere/Motorola (DSP core)

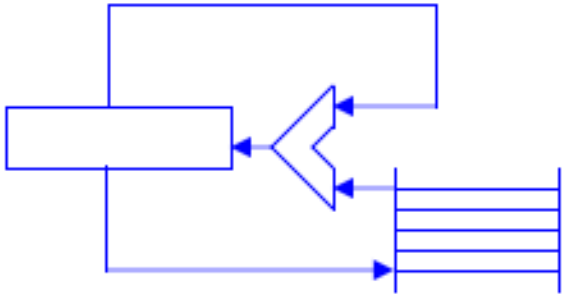
TI embedded processors



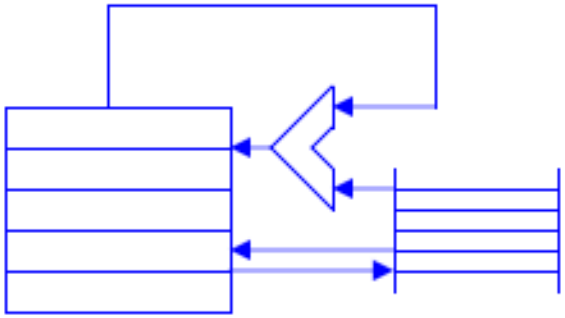


C6000 Roadmap

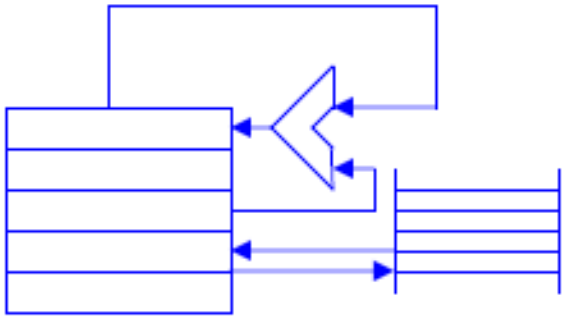
Accumulator architecture



Memory-register architecture



Load-store architecture



TMS 320C6000 VLIW DSP OVERVIEW

- Different from the conventional DSP architecture
- MIMD type architecture
- HLL programming
- Code optimization made during the compiling phase

REMEMBER!

Conventional DSP Architecture

- Multiply-accumulate (MAC) in 1 instruction cycle
- Harvard architecture for fast on-chip I/O
  - Data memory/bus separate from program memory/bus
  - One read from program memory per instruction cycle
  - Two reads/writes from/to data memory per inst. cycle
- Instructions to keep pipeline (3-6 stages) full
  - Zero-overhead looping (one pipeline flush to set up)
  - Delayed branches
- Special addressing modes supported in hardware
  - Bit-reversed addressing (e.g. fast Fourier transforms)
  - Modulo addressing for circular buffers (e.g. filters)

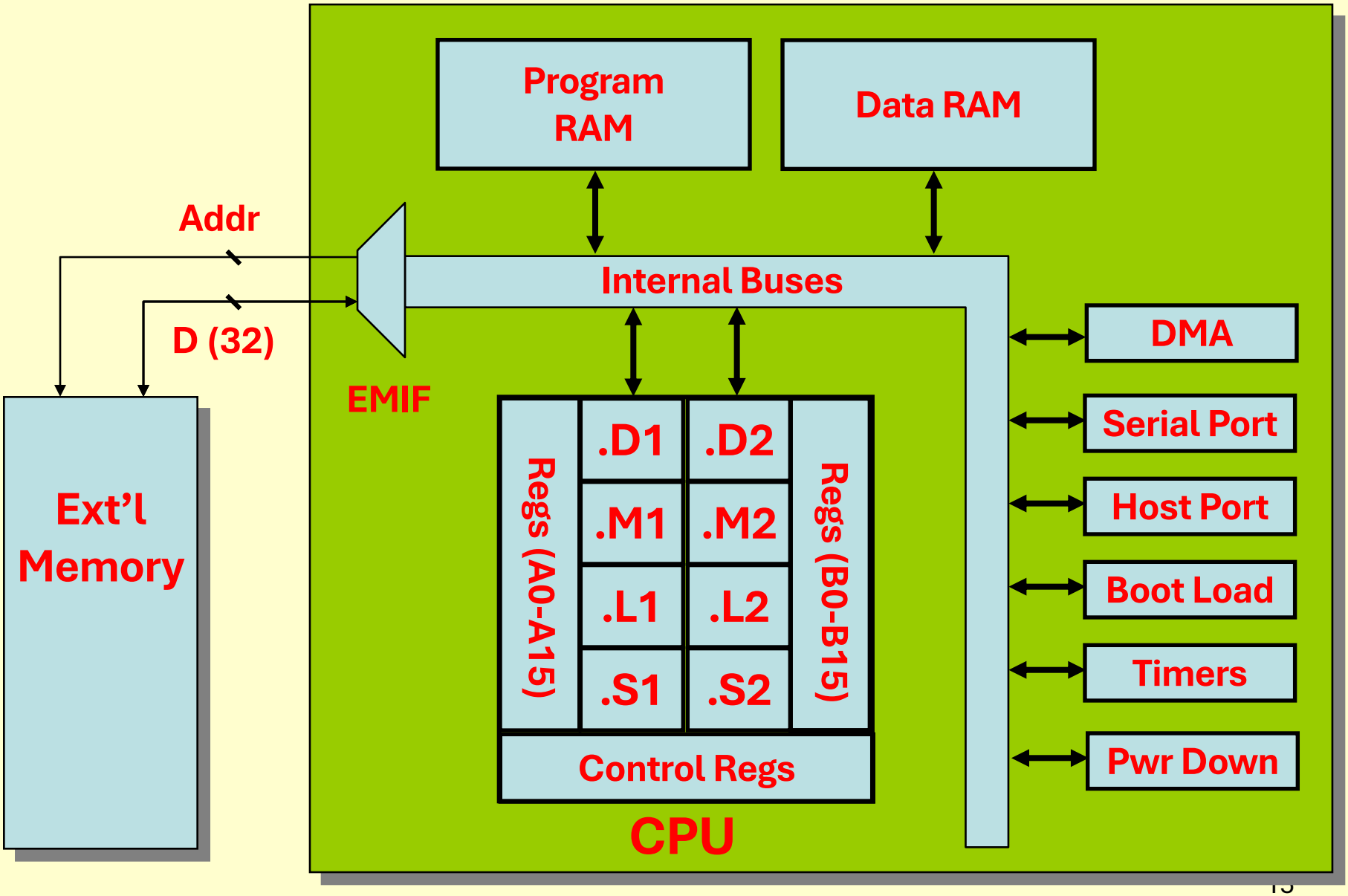
VLIW Architecture - Core Principles

- Very Long Instruction Word (VLIW) - key concept:
  - Introduced by J. Fisher (1983), adopted by TI for C6000 family
- Multiple operations packed into a single wide instruction word
- Compiler determines parallelism at compile-time (static scheduling)
- No complex hardware scheduling logic needed at runtime
- Advantages of VLIW over traditional architectures:
  - Simpler hardware (no scoreboarding, no reservation stations)
  - Lower power consumption due to reduced control logic
  - Deterministic execution timing (important for real-time DSP)
  - Higher clock frequencies possible with simpler pipeline control
  - TI C6000: 256-bit fetch packet = 8 x 32-bit instructions
  - Compiler responsibility: detect dependencies, schedule operations, insert NOPs

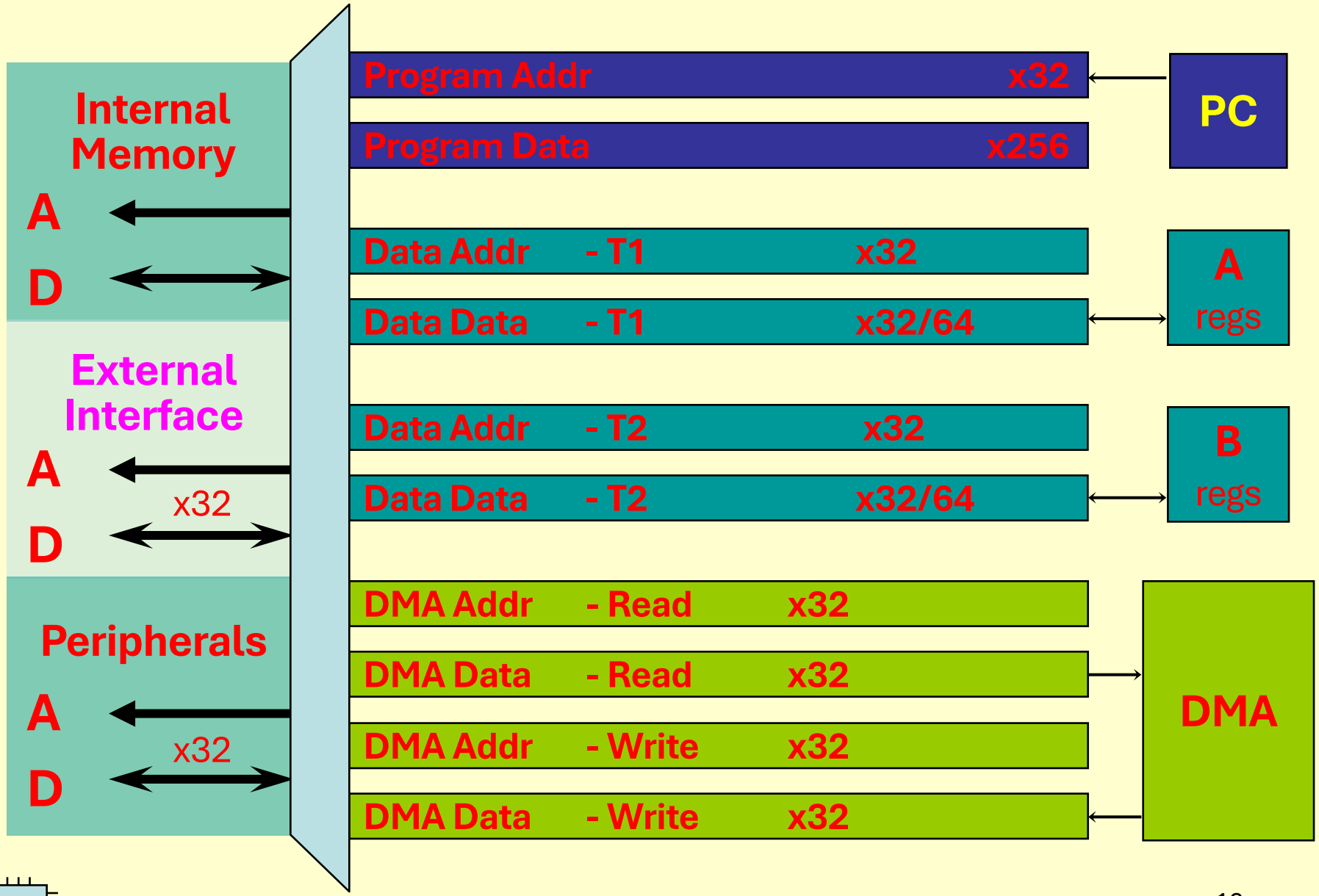
VLIW vs Superscalar vs EPIC Architectures

- **Superscalar (e.g., Intel Pentium, ARM Cortex-A):**
  - Hardware detects parallelism at runtime (dynamic scheduling)
  - Complex out-of-order execution logic, branch prediction
  - Flexible but high silicon area and power cost
- **VLIW (e.g., TI C6000, Analog Devices TigerSHARC):**
  - Compiler detects and encodes parallelism statically
  - Simple hardware, predictable timing, lower power
  - Code is architecture-specific - recompile needed for new variants
- **EPIC - Explicitly Parallel Instruction Computing (e.g., Intel Itanium):**
  - Hybrid: compiler exposes parallelism, hardware assists
  - Predication and speculation support in ISA
  - More complex than pure VLIW, less complex than superscalar
- **Key trade-offs for DSP applications:**
  - VLIW wins for DSP: deterministic timing, low power, high throughput
  - Compiler quality is critical for VLIW performance

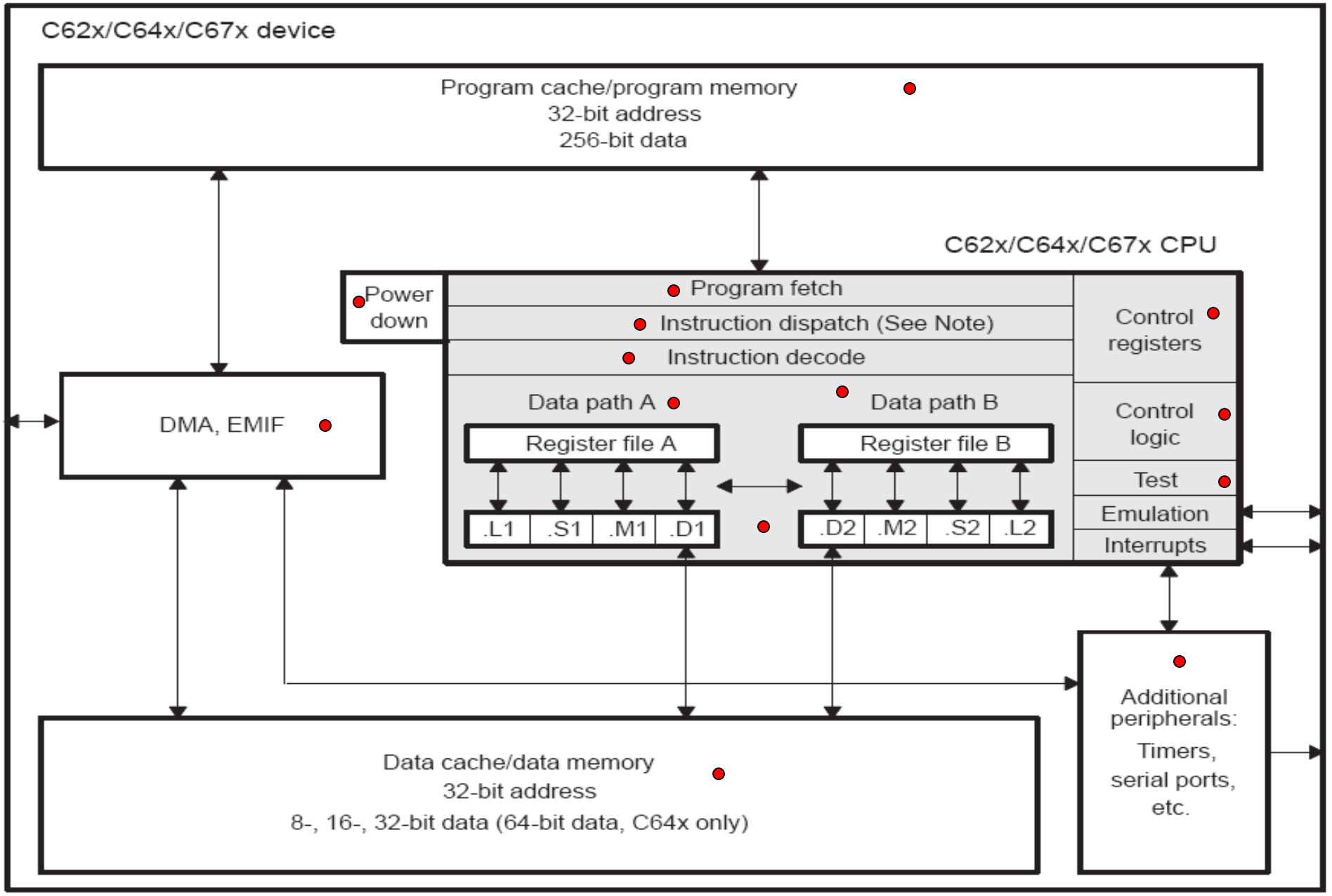
'C6x DSP Block Diagram



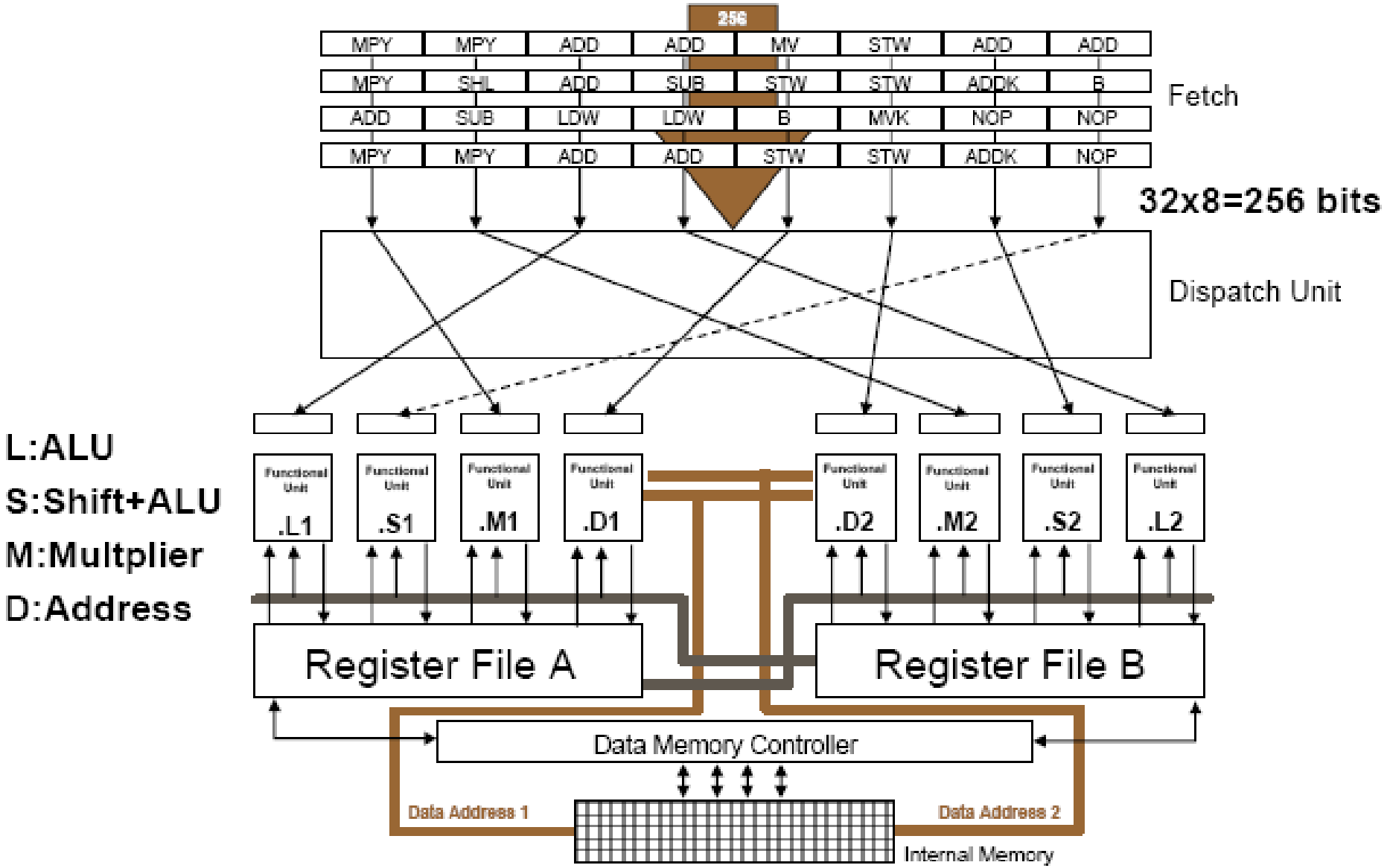
‘C6x Internal Buses



‘C67x can perform 64-bit data loads.



TMS320C62x/C64x/C67x Block Diagram



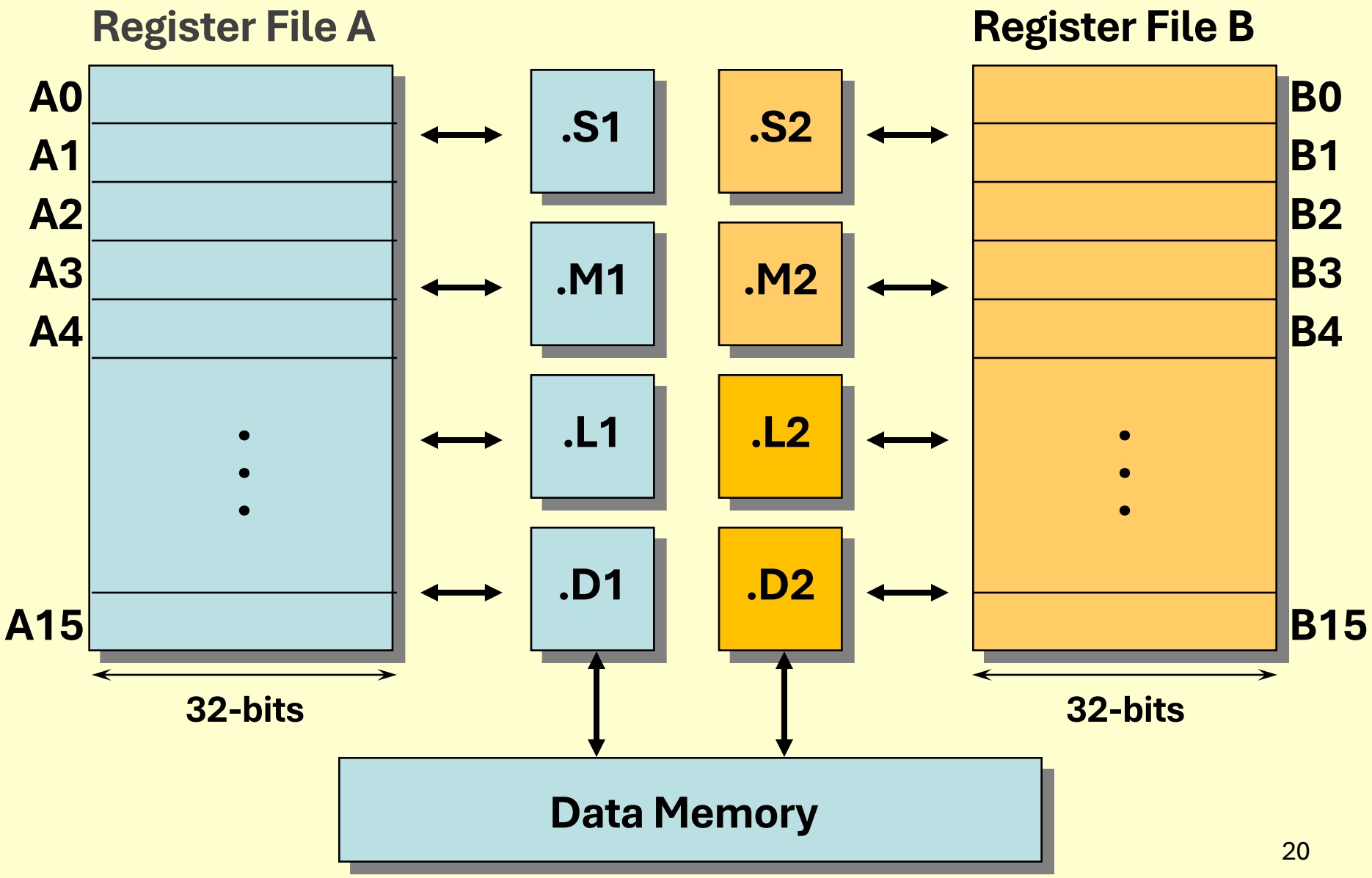
Features • More instructions/ cycle, packed in a "super-long instruction"

• Regular Architecture, more orthogonal, RISC like

• Uniform Instruction set, more instructions.

Main features of C6000 family architecture

- Very long instruction word (VLIW) size of 256 bits
  - Eight 32-bit functional units with single cycle throughput
  - One instruction cycle per clock cycle
- Data word size is 32 bits
  - 16 (32 on C6400) 32-bit registers in each of 2 data paths
  - 40 bits can be stored in adjacent even/odd registers
- Two parallel data paths
  - Data unit - 32-bit address calculations (modulo, linear)
  - Multiplier unit - 16 bit × 16 bit with 32-bit result
  - Logical unit - 40-bit (saturation) arithmetic & compares
  - Shifter unit - 32-bit integer ALU and 40-bit shifter


Register File A
Register File B
A0
A1
A2
A3
A4
A15
B0
B1
B2
B3
B4
B15
.S1
.S2
.M1
.M2
.L1
.L2
.D1
.D2
32-bits
32-bits
Data Memory

Functional Units and Operations Performed

Functional Unit	Fixed-Point Operations
.L unit (.L1, .L2)	32/40-bit arithmetic and compare operations 32-bit logical operations Leftmost 1 or 0 counting for 32 bits Normalization count for 32 and 40 bits
.S unit (.S1, .S2)	32-bit arithmetic operations 32/40-bit shifts and 32-bit bit-field operations 32-bit logical operations Branches Constant generation Register transfers to/from control register file (.S2 only)
.M unit (.M1, .M2)	16 × 16-bit multiply operations
.D unit (.D1, .D2)	32-bit add, subtract, linear and circular address calculation Loads and stores with 5-bit constant offset Loads and stores with 15-bit constant offset (.D2 only)

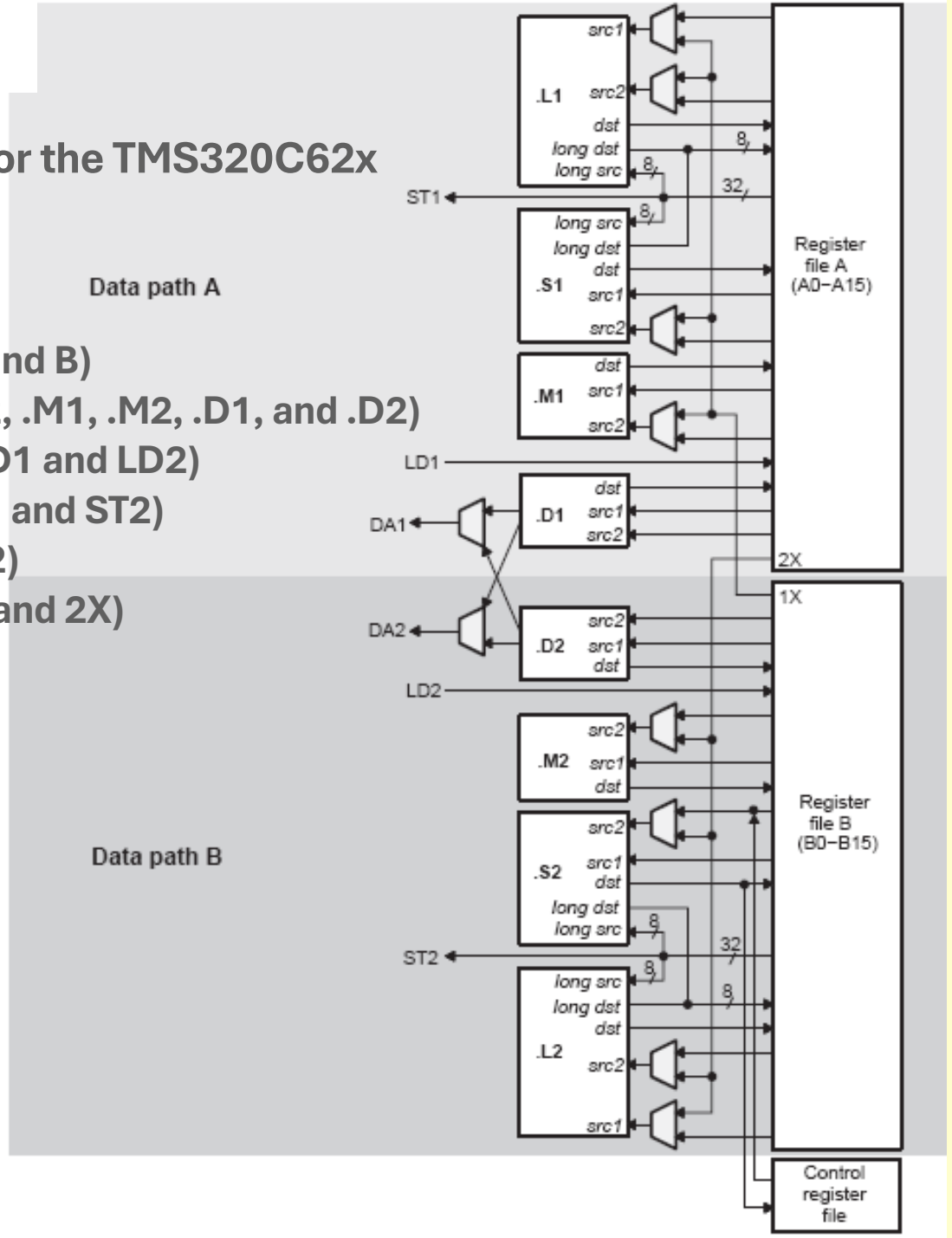
- In the best case, all units operate in parallel, a processor performs :

- four arithmetic operations,
- two multiplications,
- two address calculations in one instruction cycle.

TMS320C62x CPU Data Paths

- The components of the data path for the TMS320C62x The figure shows the CPUs.

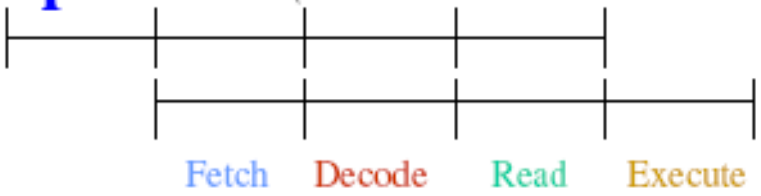
- Two general-purpose register files (A and B)
- Eight functional units (.L1, .L2, .S1, .S2, .M1, .M2, .D1, and .D2)
- Two load-from-memory data paths (LD1 and LD2)
- Two store-to-memory data paths (ST1 and ST2)
- Two data address paths (DA1 and DA2)
- Two register file data cross paths (1X and 2X)



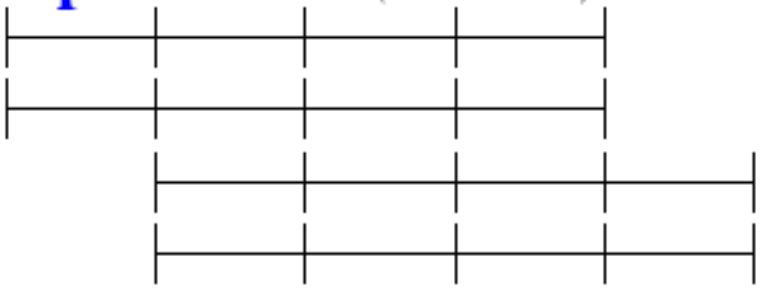
Pipelining

Sequential *(Freescale 56000)*

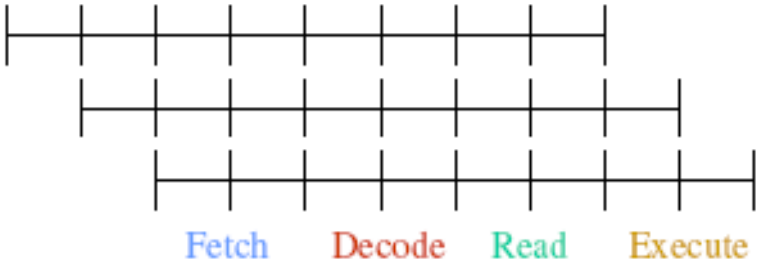
Pipelined *(Most conventional DSPs)*



Superscalar *(Pentium)*



Superpipelined *(TMS320C6000)*



Pipelining

- Process instruction stream in stages (as stages of assembly on a manufacturing line)
- Increase throughput

Managing Pipelines

- Compiler or programmer
- Pipeline interlocking

https://ebrary.net/206273/computer_science/superpipeline_architecture

Pipelining to TMS 320C6000

- **1 instruction / every machine cycle**
- **Pipeline depth**

- 7-11 stages C62x : fetch 4; decode 2; execute 1-5

-7-16 stages to C67x: fetch4; decode2; execute 1-10

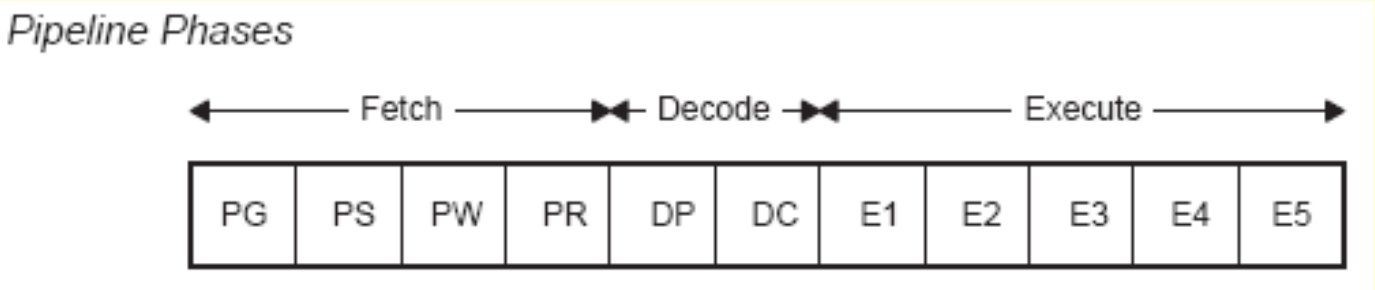
- a loop in a pipeline will disable interrupts

- avoid loop usage by employing conditional execution!

- No hardware protection against pipeline incidents!

- compiler/assembler must warn the pipeline incidents

- Instruction dispatching

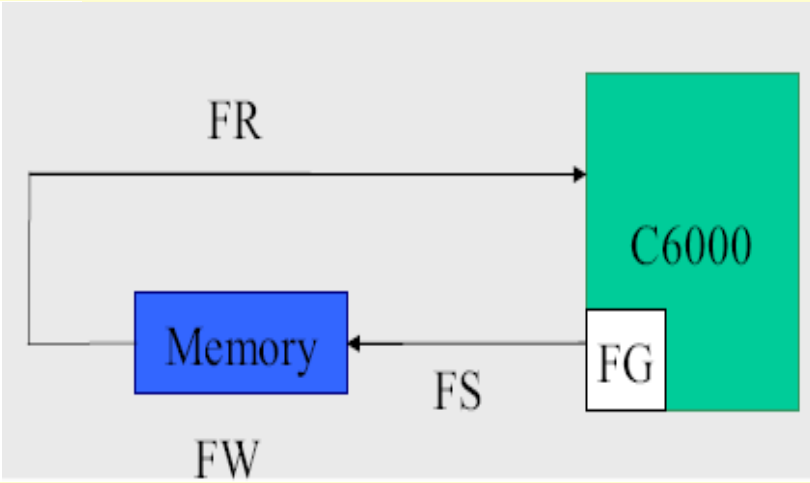


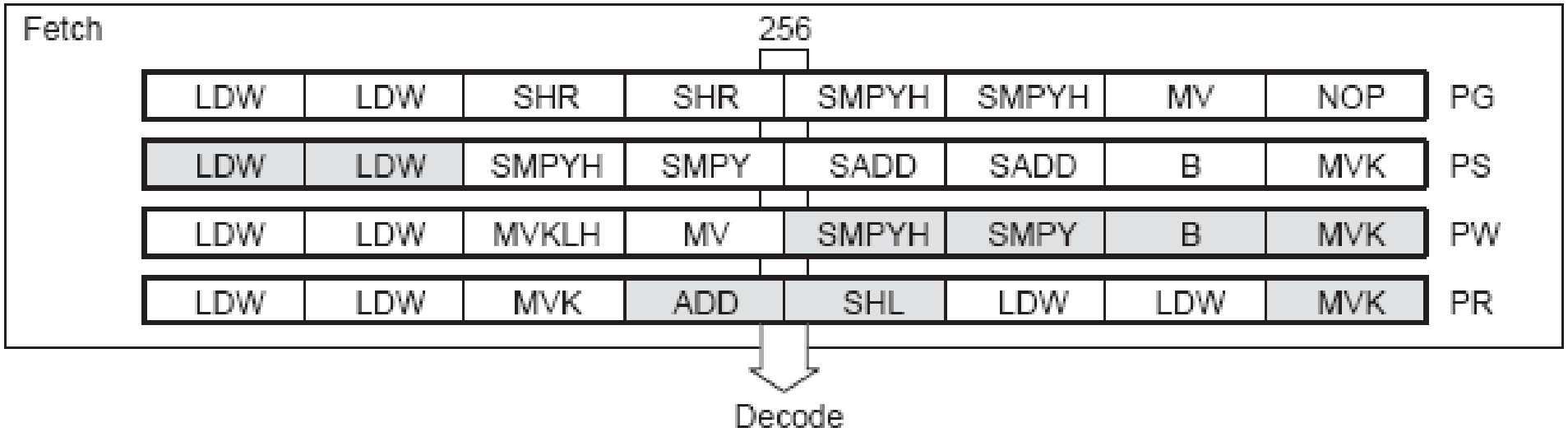
Pipelining to TMS 320C6000

Fetch

The fetch phases of the pipeline are:

- **PG:** Program address generate
- **PS:** Program address send
- **PW:** Program access ready wait
- **PR:** Program fetch packet receive



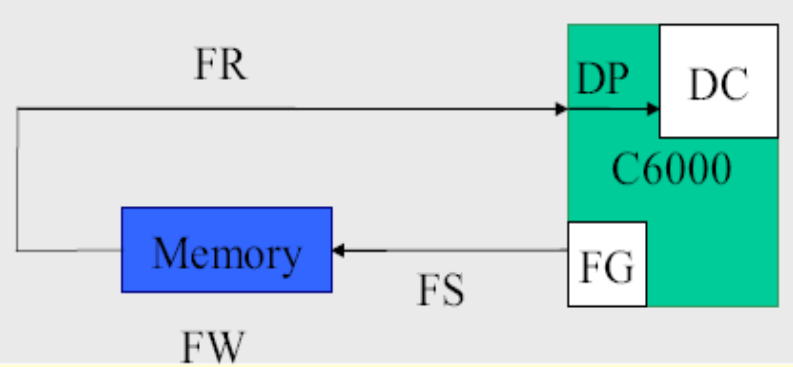


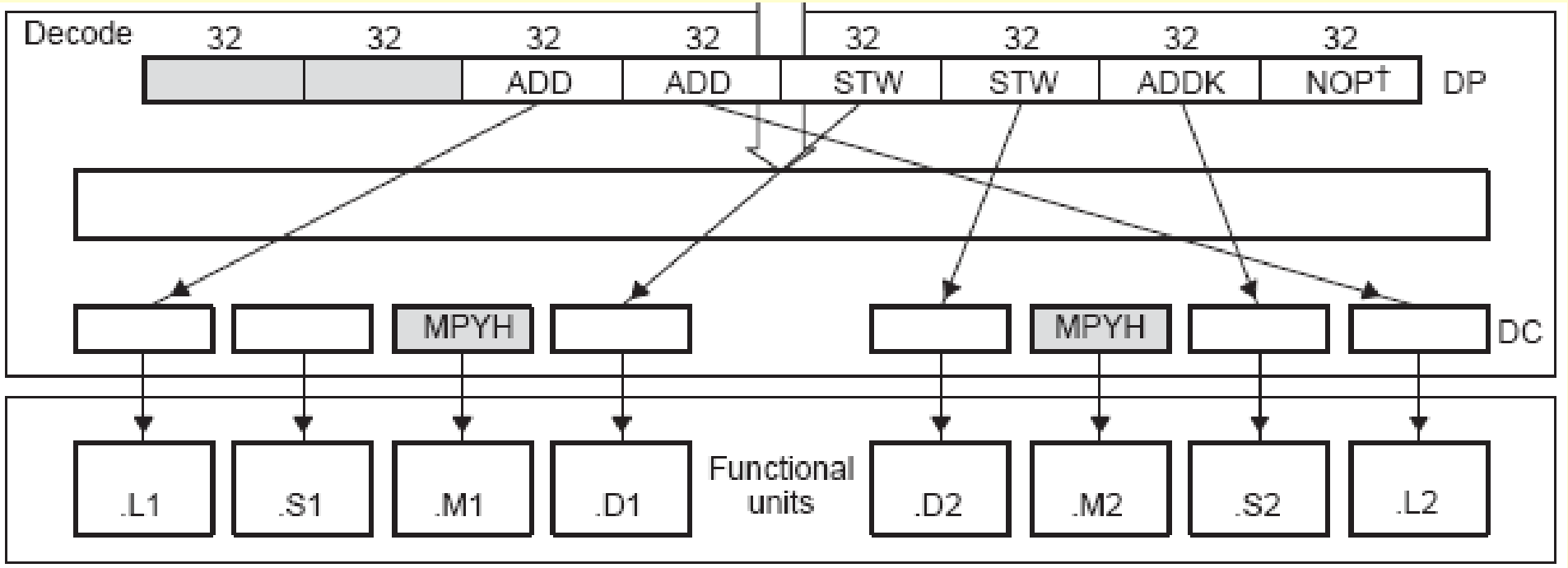
Pipelining to TMS 320C6000

Decode

The decode phases of the pipeline are:

- DP: Instruction dispatch
- DC: Instruction decode

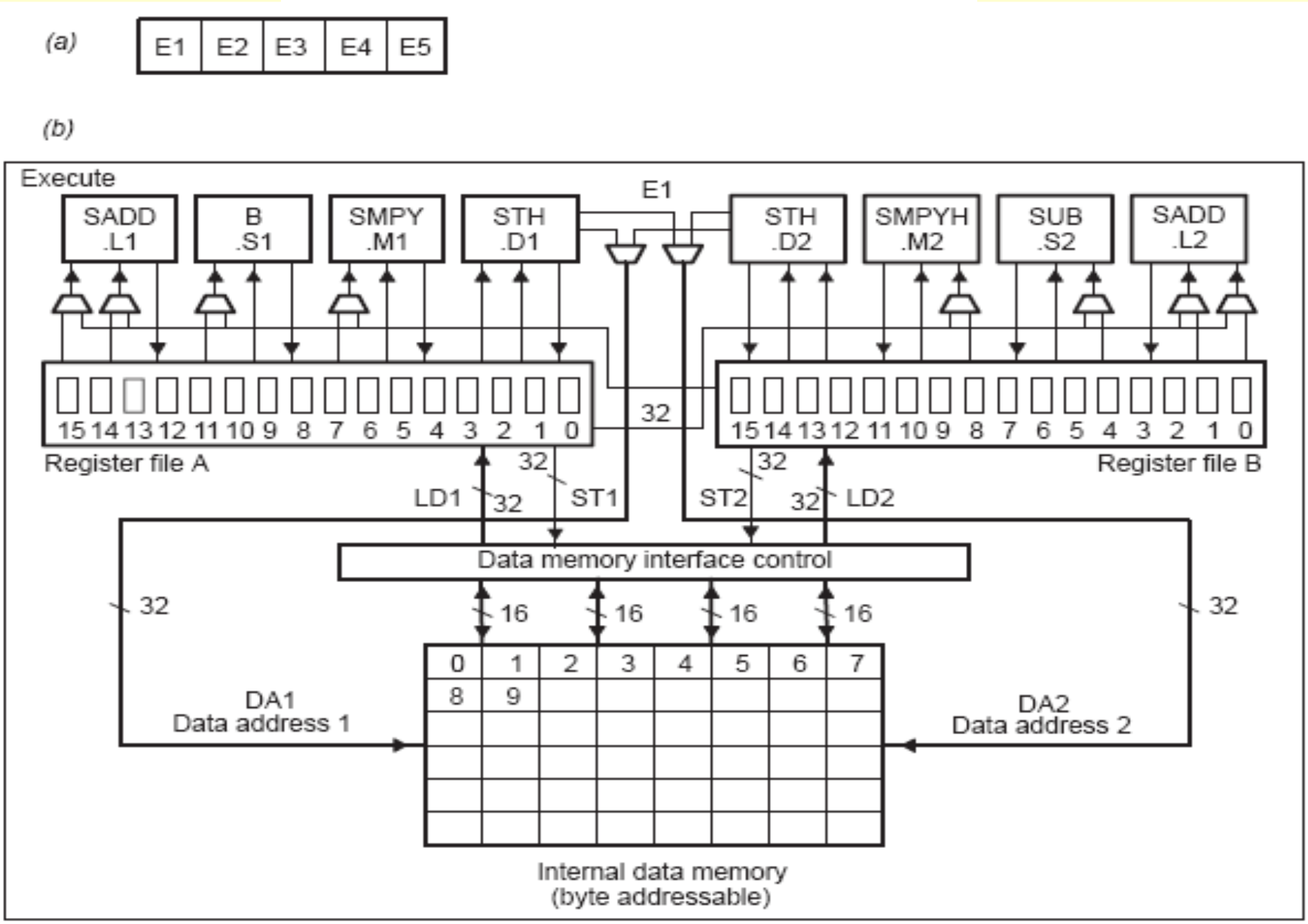




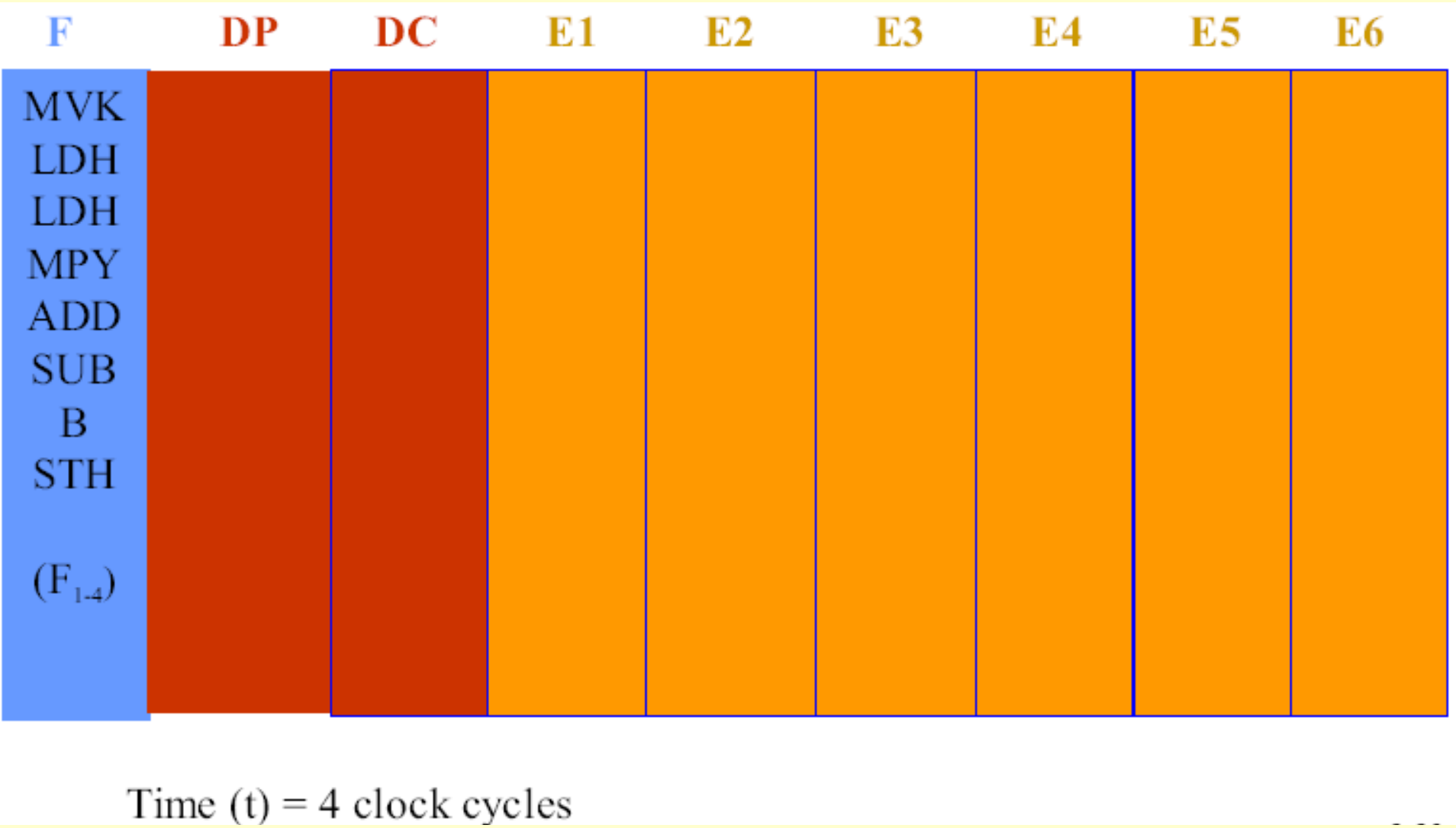
† NOP is not dispatched to a functional unit.

Pipelining to TMS 320C6000

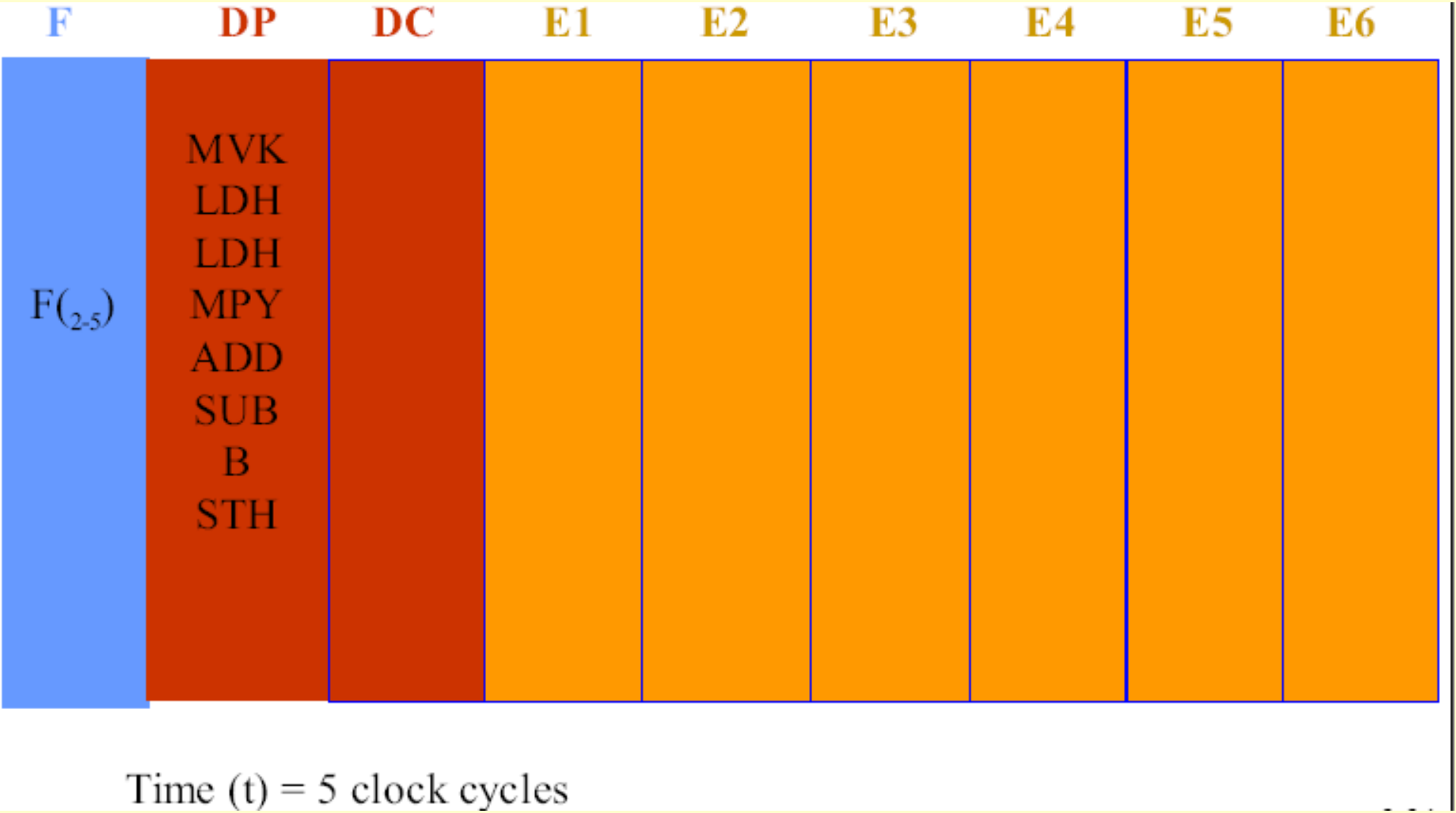
Execute Phases of the Pipeline



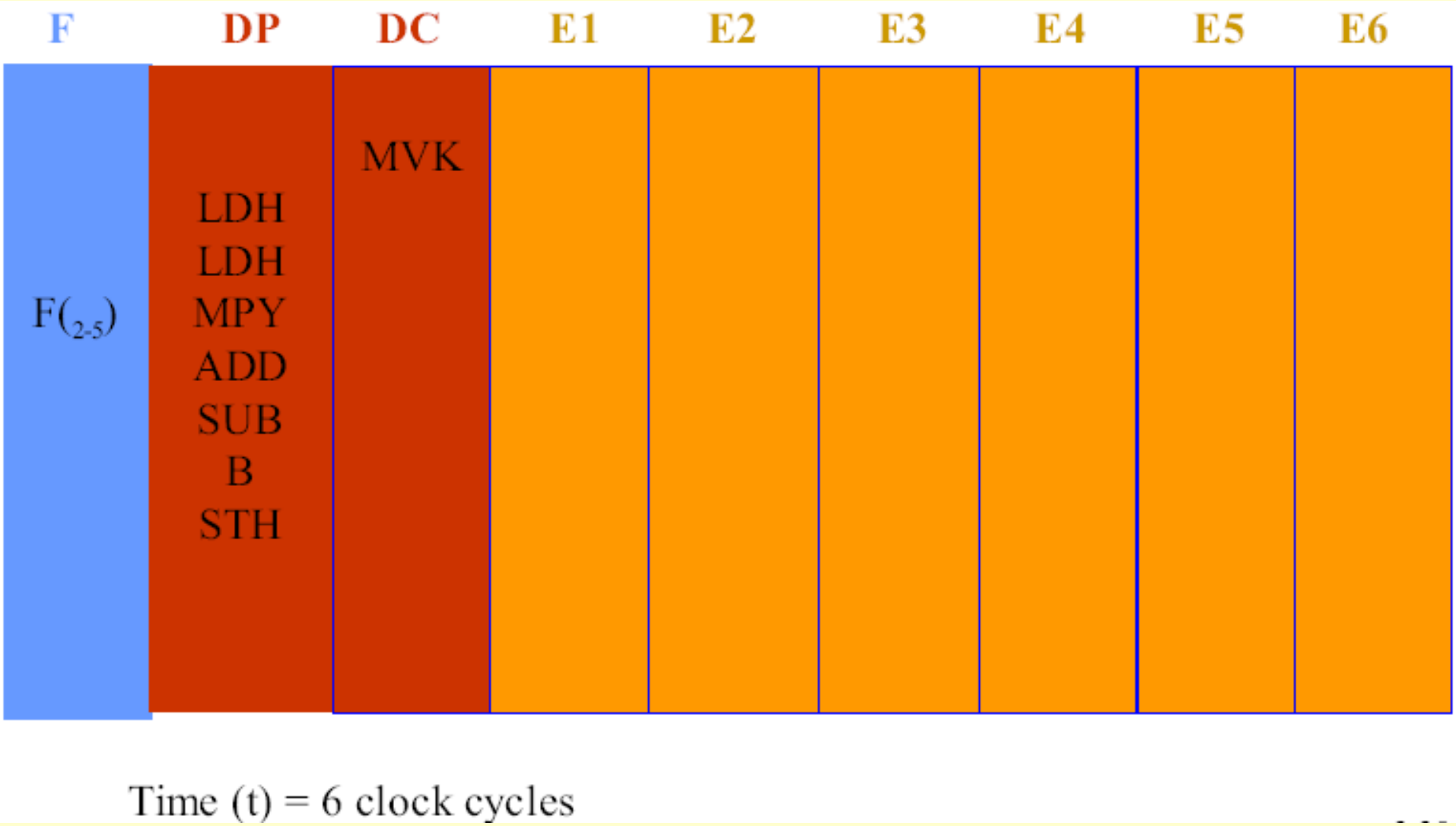
FETCH PACKET



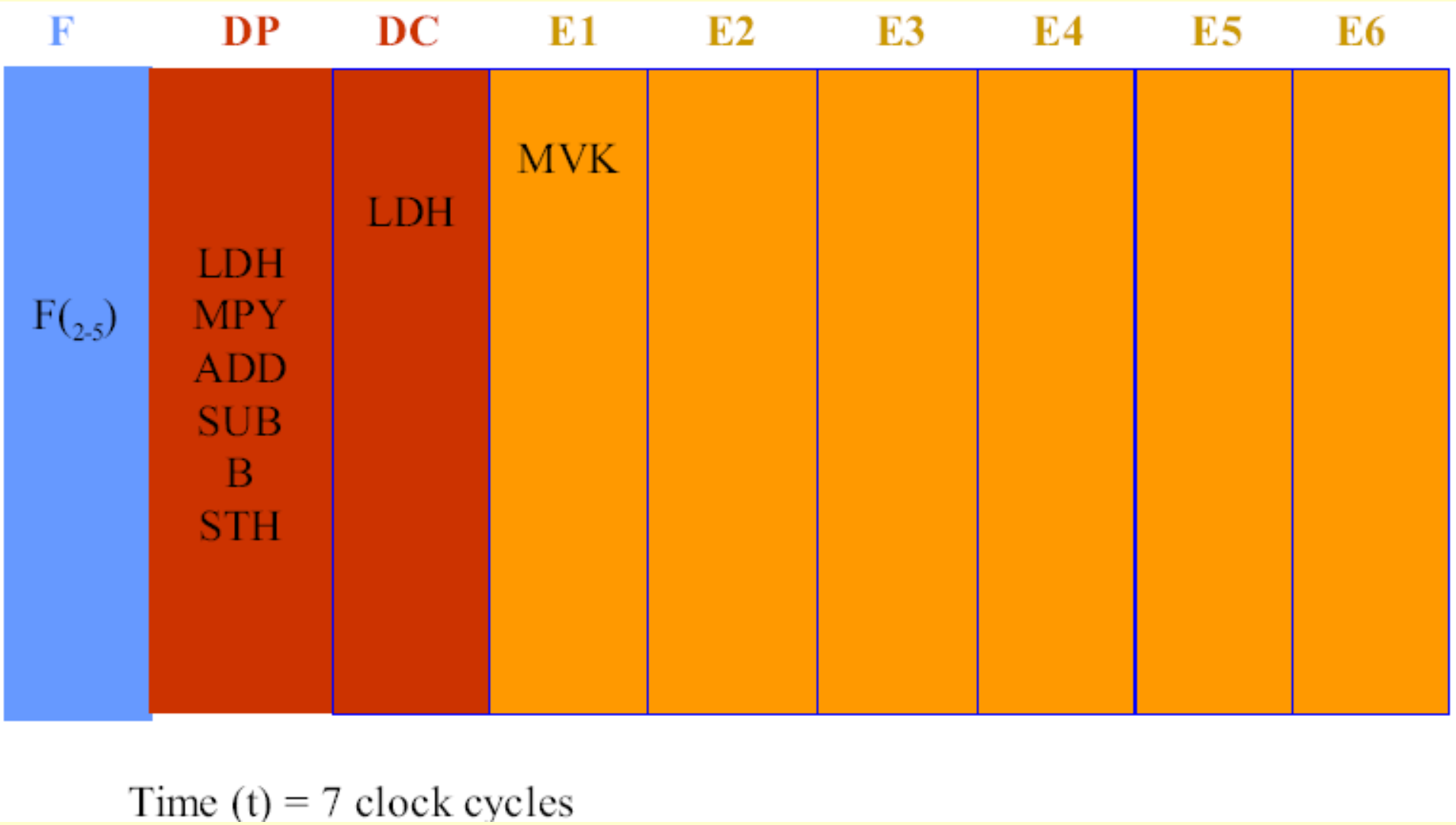
DISPATCHING



DECODING

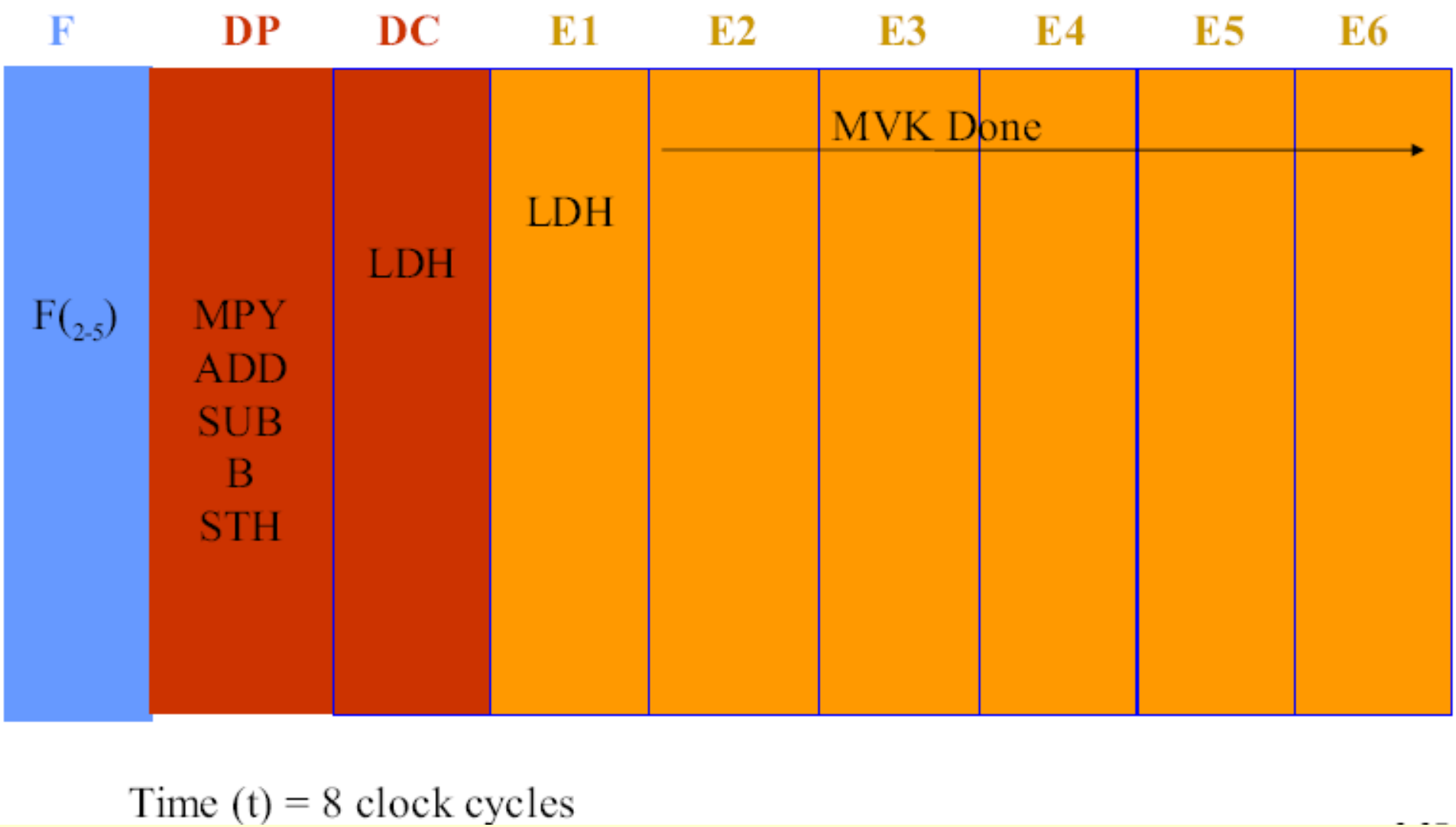

F
DP
DC
E1
E2
E3
E4
E5
E6
F(2-5)
LDH
LDH
MPY
ADD
SUB
B
STH
MVK
Time (t) = 6 clock cycles

EXECUTE -1



Time (t) = 7 clock cycles

Execute (MVK done LDH in E1)



Time (t) = 8 clock cycles

Vector Dot Product with pipeline effects

```
; clear A4 and initialize pointers A5, A6, and A7
    MVK   .S1   40,A2    ; A2 = 40 (loop counter)
loop   LDH   .D1   *A5++,A0    ; A0 = a(n)
    LDH   .D1   *A6++,A1    ; A1 = x(n)
    NOP   4
    MPY   .M1   A0,A1,A3    ; A3 = a(n) * x(n)
    NOP
    ADD   .L1   A3,A4,A4    ; Y = Y + A3
    SUB   .L1   A2,1,A2 ; decrement loop counter
[A2]   B      .S1   loop    ; if A2 != 0, then branch
    NOP   5
    STH   .D1   A4,*A7  ; *A7 = Y
```

- Assembler will introduce automatic NOP !
- Assembler may transform sequential code to parallel code !

'C62x Instruction Set (by category)

Arithmetic
ABS
ADD
ADDA
ADDK
ADD2
MPY
MPYH
NEG
SMPY
SMPYH
SADD
SAT
SSUB
SUB
SUBA
SUBC
SUB2
ZERO

Logical
AND
CMPEQ
CMPGT
CMPLT
NOT
OR
SHL
SHR
SSHL
XOR

Bit Mgmt
CLR
EXT
LMBD
NORM
SET

Data Mgmt
LDB/H/W
MV
MVC
MVK
MVKL
MVKH
MVKLH
STB/H/W

Program Ctrl
B
IDLE
NOP

Note: Refer to the 'C6000 CPU Reference Guide for more details

'C62x Instruction Set (by unit)

.S Unit	
ADD	MVKLH
ADDK	NEG
ADD2	NOT
AND	OR
B	SET
CLR	SHL
EXT	SHR
MV	SSHL
MVC	SUB
MVK	SUB2
MVKL	XOR
MVKH	ZERO

.L Unit	
ABS	NOT
ADD	OR
AND	SADD
CMPEQ	SAT
CMPGT	SSUB
CMPLT	SUB
LMBD	SUBC
MV	XOR
NEG	ZERO
NORM	

.M Unit	
MPY	SMPY
MPYH	SMPYH

.D Unit	
ADD	STB/H/W
ADDA	SUB
LDB/H/W	SUBA
MV	ZERO
NEG	

Other	
NOP	IDLE

Note: Refer to the 'C6000 CPU Reference Guide for more details.

' C6700: Superset of Fixed-Point (by unit)

.S

.L

.D

.M

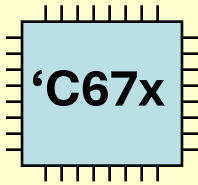
.S Unit		
ADD	NEG	ABSSP
ADDK	NOT	ABSDP
ADD2	OR	CMPGTSP
AND	SET	CMPEQSP
B	SHL	CMPLTSP
CLR	SHR	CMPGTDP
EXT	SSHL	CMPEQDP
MV	SUB	CMPLTDP
MVC	SUB2	RCPSP
MVK	XOR	RCPDP
MVKL	ZERO	RSQRSP
MVKH		RSQRDP
		SPDP

.D Unit			
ADD		NEG	
ADDAB	(B/H/W)	STB	(B/H/W)
ADDAD		SUB	
LDB	(B/H/W)	SUBAB	(B/H/W)
LDDW		ZERO	
MV			

.L Unit		
ABS	NOT	ADDSP
ADD	OR	ADDDP
AND	SADD	SUBSP
CMPEQ	SAT	SUBDP
CMPGT	SSUB	INTSP
CMPLT	SUB	INTDP
LMBD	SUBC	SPINT
MV	XOR	DPINT
NEG	ZERO	SPRTUNC
NORM		DPTRUNC
		DPSP

.M Unit		
MPY	SMPY	MPYSP
MPYH	SMPYH	MPYDP
MPYLH		MPYI
MPYHL		MPYID

No Unit Used	
NOP	IDLE



Note: Refer to the 'C6000 CPU Reference Guide for more details.

'C64x: Superset of 'C62x

.S

Dual/Quad Arith
SADD2
SADDUS2
SADD4

Bitwise Logical
ANDN

Shifts & Merge
SHR2
SHRU2
SHLMB
SHRMB

Data Pack/Un
PACK2
PACKH2
PACKLH2
PACKHL2
UNPKHU4
UNPKLU4
SWAP2
SPACK2
SPACKU4

Compares
CMPEQ2
CMPEQ4
CMPGT2
CMPGT4

Branches/PC
BDEC
BPOS
BNOP
ADDKPC

.L

Dual/Quad Arith
ABS2
ADD2
ADD4
MAX
MIN
SUB2
SUB4
SUBABS4

Bitwise Logical
ANDN

Shift & Merge
SHLMB
SHRMB

Load Constant
MVK (5-bit)

Data Pack/Un
PACK2
PACKH2
PACKLH2
PACKHL2
PACKH4
PACKL4
UNPKHU4
UNPKLU4
SWAP2/4

.D

Dual Arithmetic
ADD2
SUB2

Bitwise Logical
AND
ANDN
OR
XOR

Address Calc.
ADDAD

Mem Access
LDDW
LDNW
LDNDW
STDW
STNW
STNDW

Load Constant
MVK (5-bit)

.M

Multiplies
MPYHI
MPYLI
MPYHIR
MPYLIR
MPY2
SMPY2
DOTP2
DOTPN2
DOTPRSU2
DOTPNRSU2
DOTPU4
DOTPSU4
GMPY4
XPND2/4

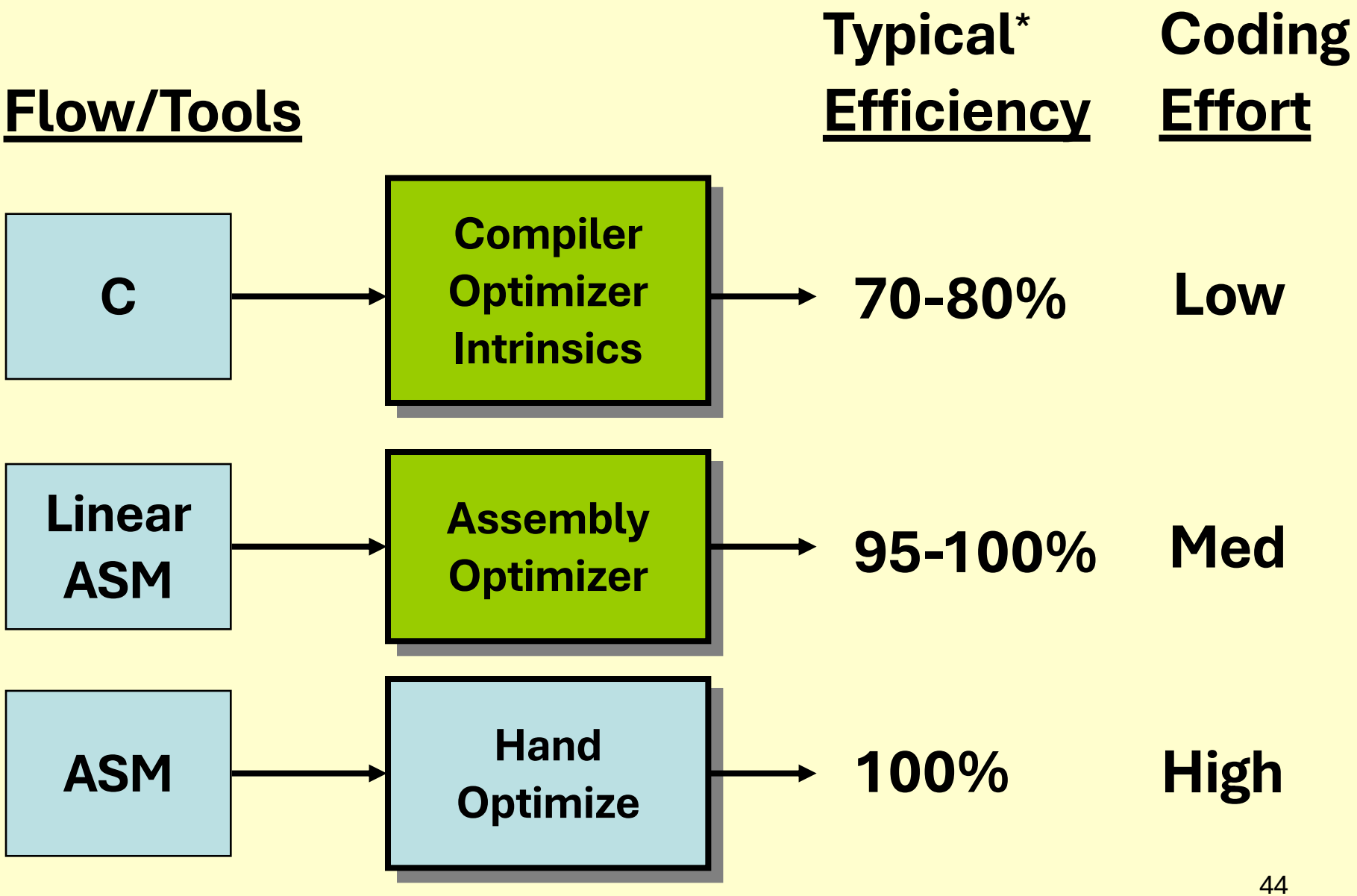
Average
AVG2
AVG4

Shifts
ROTL
SSHVL
SSHVR

Bit Operations
BITC4
BITR
DEAL
SHFL

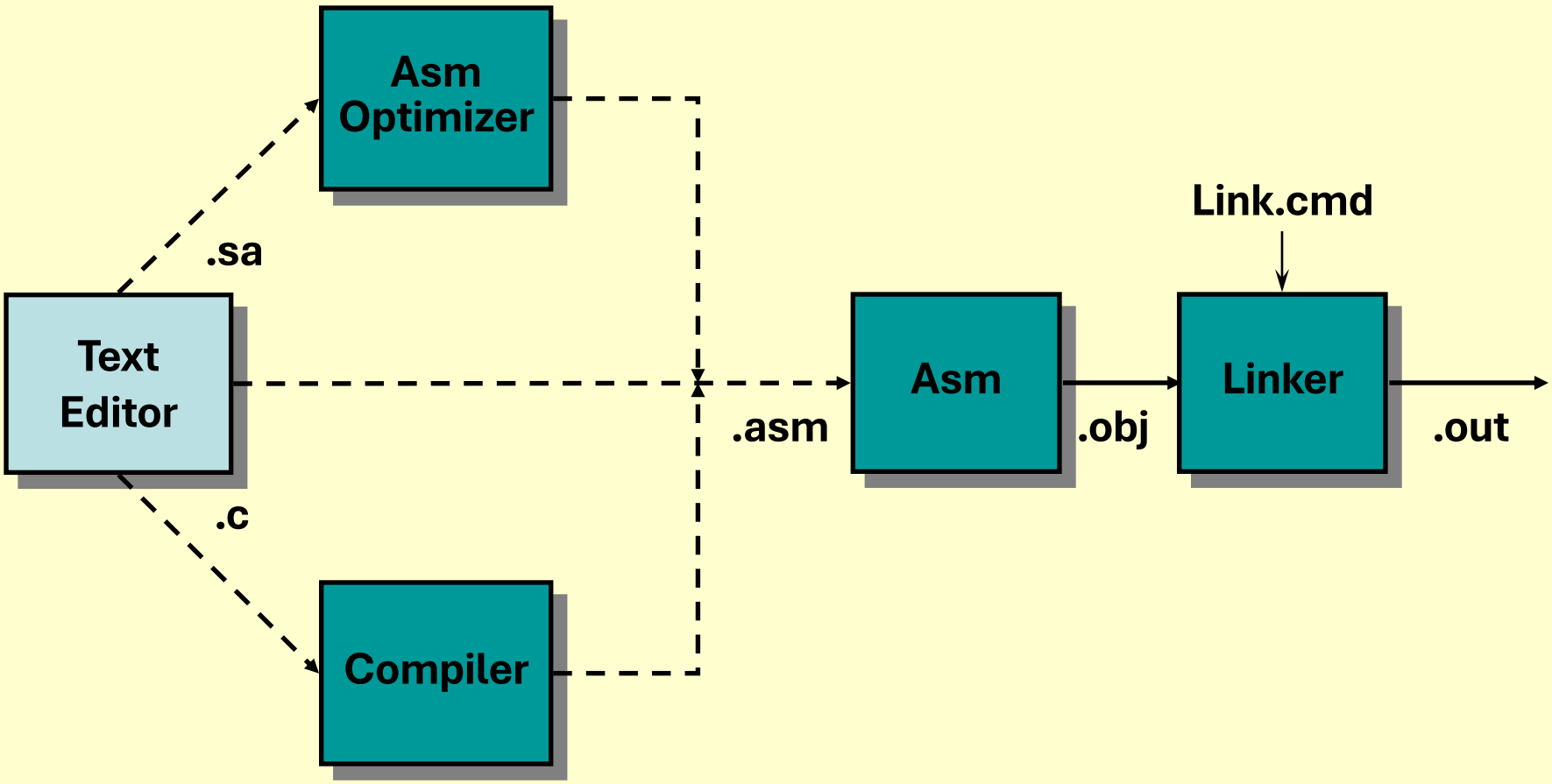
Move
MVD

'C6x Programing



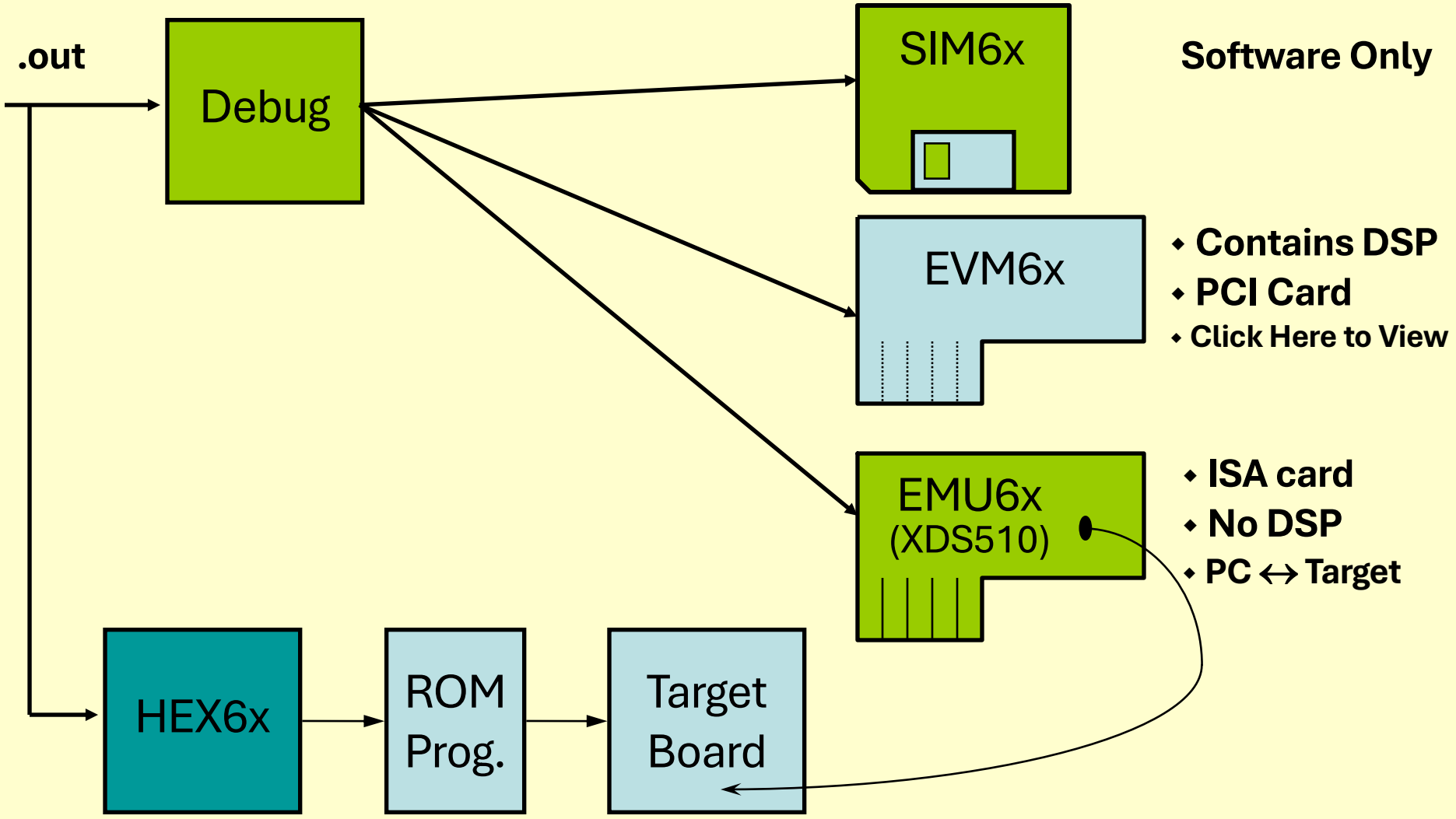
* Typical efficiency vs. hand optimized assembly

Software Tool Flow



Compl. 6x runs all the code generation tools

Debug Tools Flow



The C6000 as standalone DSP chips is essentially legacy. As of August 2023, standalone C6000 DSPs are no longer available as individual chips — they are only available bundled inside other SoCs. The C62x, C64x, C67x, and even the C66x multicore parts are either end-of-life or in last-time-buy status. You can't design a new product around a standalone C6678, for example.

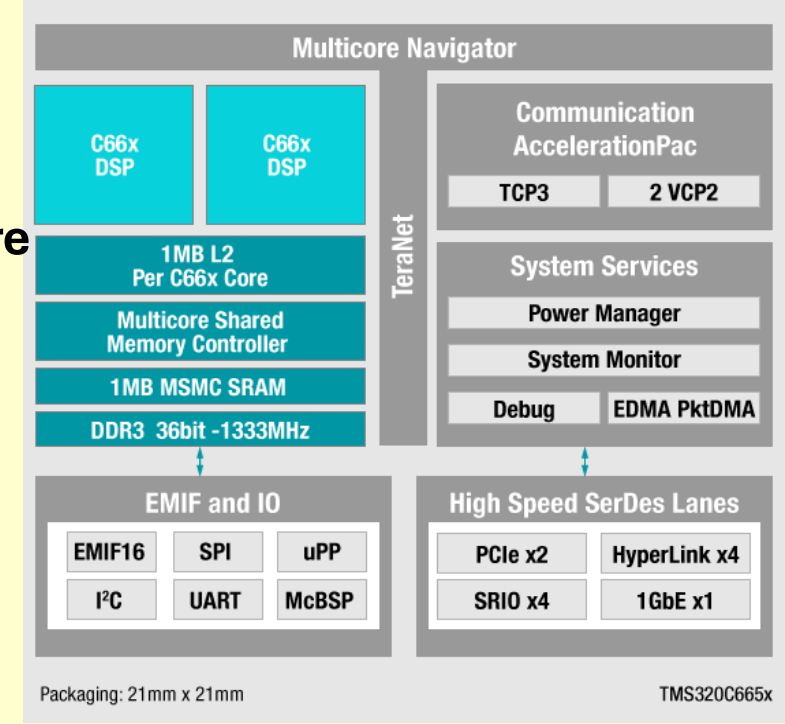
However, the C6000 VLIW architecture lives on inside modern TI SoCs — and it's very active:

1. **C66x cores** are still embedded in current TI Jacinto automotive SoCs. The TDA4VM EVM itself includes both C7x and C66x DSP cores, where the C66x handles computer vision tasks like pre/post-processing.
2. **C7x is the direct evolution** of the C6000 VLIW lineage. The C7x next-generation DSP combines TI's industry-leading DSP and EVE cores into a single higher-performance core with floating-point vector capabilities, enabling backward compatibility for legacy code. It has a 512-bit SIMD datapath (vs 64-bit for C6x) and a Matrix Multiply Accelerator.
3. **C7x is in active, current products as of 2025-2026:** TI's AM275x-Q1 MCUs and AM62D-Q1 processors integrate TI's vector-based C7x DSP core for automotive audio applications including Dolby Atmos, active noise cancellation, and spatial audio. TI's C7x vector DSP delivers up to 40 GFLOPS each, enabled by integrated hardware accelerators.

- The **VLIW principles** (static scheduling, fetch/execute packets, p-bit encoding, software pipelining, no hardware interlocking) are **identical** in concept from C62x through C7x. The C6000 is a cleaner, simpler platform to teach these on.
- The **C6000 instruction set** is well-documented with decades of educational material (SPRU731, etc.), making it ideal for a university course.
- Understanding C6000 gives students a direct path to understanding C7x, which is TI's current and future DSP — used in ADAS, robotics (Amazon Proteus), radar, and automotive audio.
- The VLIW concepts also apply broadly to other architectures (Qualcomm Hexagon DSP, Andes NDS, etc.).

C66x Multicore DSP

C66x – world's fastest floating-point DSP core with devices ranging from **single core C6654** to **octal core C6678** and supporting core speeds up to 1.4GHz



Main Features

- Up to 1.4GHz of fixed and floating-point performance per DSP core
- Single core to eight core scalability
- KeyStone™ architecture for enhanced multicore performance
- Large embedded memory and high bandwidth DDR3/DDR3L interface
- Network Coprocessor (NetCP) option including security and packet acceleration
- High Speed I/O including PCIe, Serial RapidI/O, Gigabit Ethernet, Hyperlink

http://www.ti.com/lsds/ti/processors/dsp/c6000_dsp/c66x/products.page

APPLICATIONS

- Avionics and defense
- Communications systems
- Machine vision
- Embedded and cloud analytics
- High performance computing
- Multimedia infrastructure
- Medical imaging
- Test and measurement
- Surveillance and security
- Software defined radio (SDR)

- TI's new TMS320C66x (aka C66x) series, a multicore chip they ***designed for 4G cellular base stations and radio network controllers***. The C66x is a 40nm chip that comes in single-core, dual-core, quad-core and eight-core variations. Its most distinguishing feature is *the addition of floating-point instructions*, which were incorporated to support the more complex processing required for 4G wireless communications. The previous generation C64x series DSPs supported only fixed-point math.
- The C66x is implemented with TI's new ***KeyStone architecture***, which incorporates an *eight-way VLIW architecture, a high-speed switch fabric called TeraNet, and a multicore navigator and DMA system that manages packet sending to other cores and peripherals.* All the C66x products come with 512 KB L2 cache/core, along with 32 KB L1 cache for both instructions and data.
- In its eight-core 1.25 GHz implementation, the C66x delivers 160 single precision (SP) Gflops, while sucking up just 10W of power. That works out to an impressive 16 SP Gflops/watt. Energy efficiency is a hallmark of DSPs, in general, since they typically populate systems (like the aforementioned cellular base station towers and radio network controllers), where power and cooling is in short supply.
- The first HPC (High Performance Computing)-friendly C66x-based device is a PCIe card, which sports four of the eight-core DSPs running at 1.0 GHz. Built by Advantech, a TI parter, the half-length PCIe card delivers 512 SP Gflops at a modest 50W.
- On-board memory consists of 4 GB DDR3 RAM (1333 MHz), with full ECC support. They're also working on a full-length card, with eight DSPs, twice as much memory, and twice the performance.

http://processors.wiki.ti.com/index.php/Keystone_Device_Architecture
http://www.hpcwire.com/2011/10/27/texas_instruments_makes_hpc_play_with_new_multicore_dsp_chips/

C66x KeyStone Multicore Architecture

- KeyStone I Architecture (C6670, C6672, C6674, C6678):
- TeraNet: high-speed non-blocking switch fabric interconnect
- Multicore Navigator: hardware-accelerated packet routing between cores
- EDMA3: Enhanced DMA with 64 channels, linked/chained transfers
- Memory Hierarchy:
  - 32 KB L1P (program) + 32 KB L1D (data) per core
  - 512 KB - 1 MB L2 cache/SRAM per core
- Multicore Shared Memory Controller (MSMC): 4-8 MB shared SRAM
- High-speed peripherals: SRIO, PCIe Gen2, GbE, HyperLink
- KeyStone II (66AK2Hx): ARM Cortex-A15 + C66x DSP cores
- TMS320C6678: 8 cores at 1.25 GHz = 320 GMAC/s fixed-point
- Total: 160 GFLOPS single-precision in ~10W power envelope
- Applications: 4G/5G baseband, radar, sonar, medical imaging, HPC

TI C7x DSP - Next Generation VLIW

- C7x: TI's latest DSP architecture (C7100, C7120, C7504)
- 512-bit wide VLIW with SIMD and vector processing
- Streaming Engine (SE): hardware-accelerated data streaming
  - 2 SEs provide conflict-free data access patterns
  - Eliminates load instructions for regular data access
- Matrix Multiply Accelerator (MMA): dedicated tensor operations
- Key features of C7x architecture:
- 64 vector registers (512-bit each)
- Predicate registers for vector lane masking
- Native support for 8/16/32-bit integer and FP16/FP32/FP64
- Backwards compatible with C66x code (binary compatibility)
- C7x in TDA4x SoC: ADAS and autonomous driving applications
- Peak: 80 TOPS (INT8) with MMA, 40 GFLOPS (FP32) per core
- Evolution: C6000 VLIW -> C66x multicore -> C7x vector VLIW + MMA