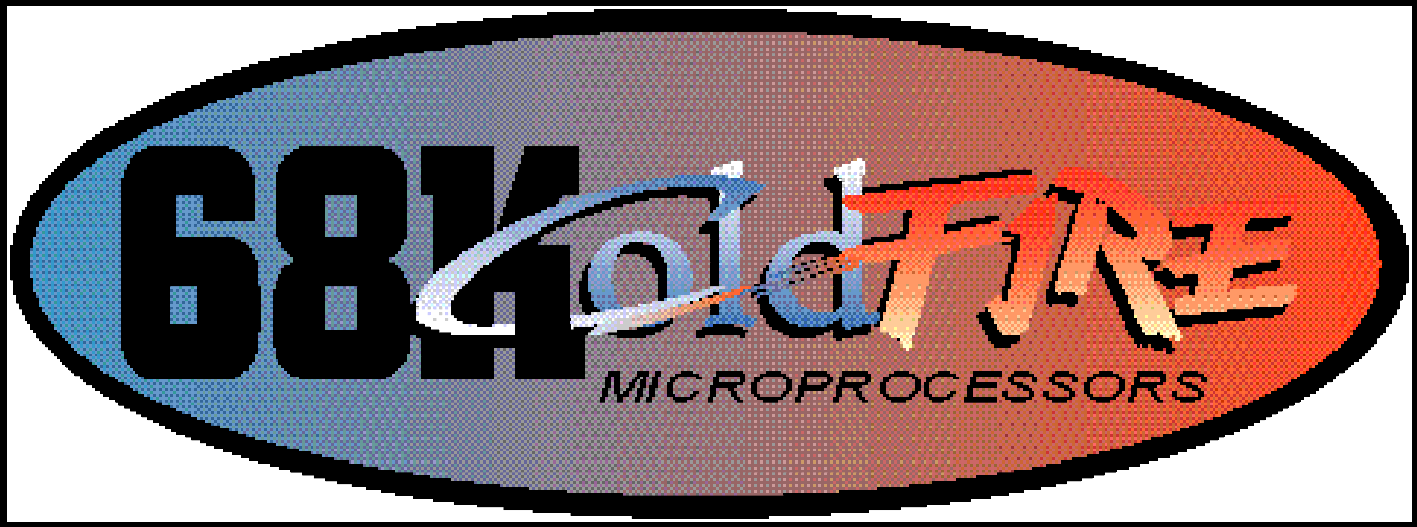


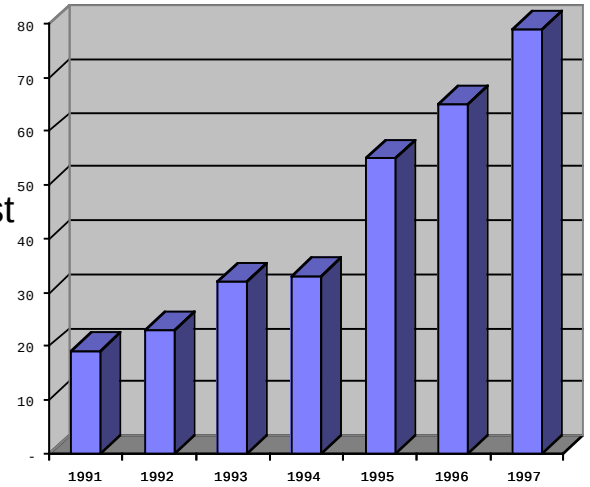
Version 4 ColdFire Core Announcement

October, 1998

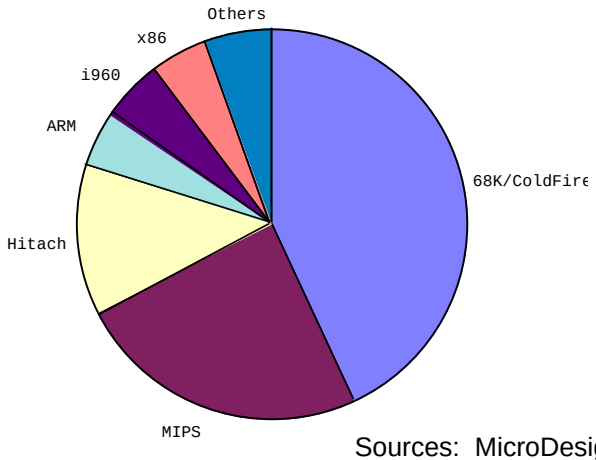
ColdFire® Is...

- An Instruction Set
  - ColdFire is 68K
    - Binary compatible with MicroAPL emulator and translator
    - 68K/ColdFire is most widely used 32-bit embedded architecture
  - Desktop-specific functions eliminated to minimize complexity/cost
    - SW supported functions: BCD, Rotate, etc.
    - Simplified exception model
- An Implementation Methodology Designed for Reuse
  - All ColdFire cores are 100% fully synthesizable
  - Parameterizable - all options exist within a single design description
  - Configurable
    - Generic local-memory controllers support a range of sizes
    - Choose size using compiled memory arrays
  - Hierarchical architecture
    - Multiple buses provide layers of bandwidth + modularity
    - Standard internal bus structure provides simple interface
  - Design-for-Test
    - Muxed D-FF rising-edge clocked design
    - ATPG scan vectors for stuck-at, speed testing
    - BIST test methodology for memories
  - Deployment focused on soft macro RTL + support of hard macros

68K/ColdFire Millions of Units Shipped

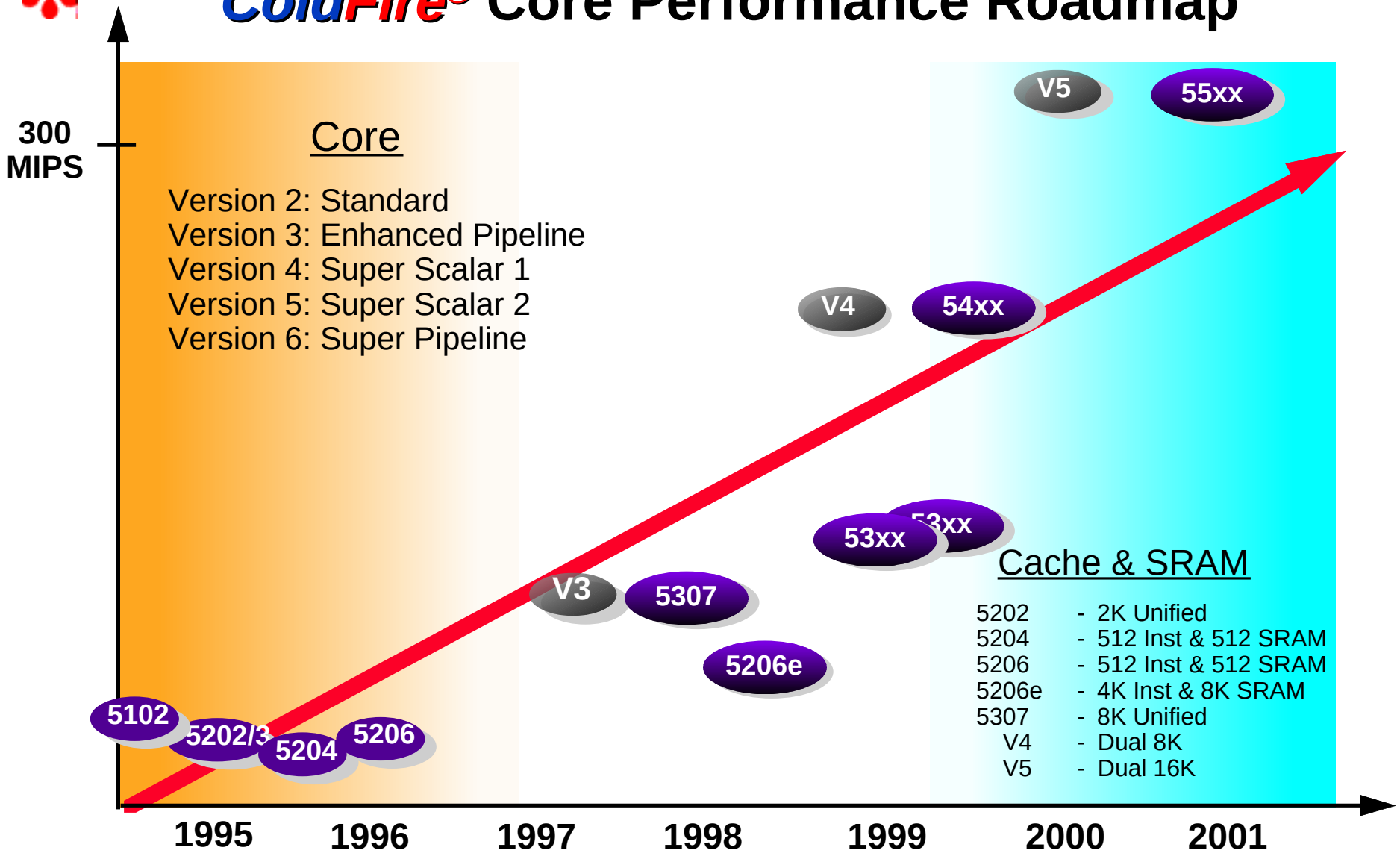


1997 Market Share



Sources: MicroDesign Resources, Motorola

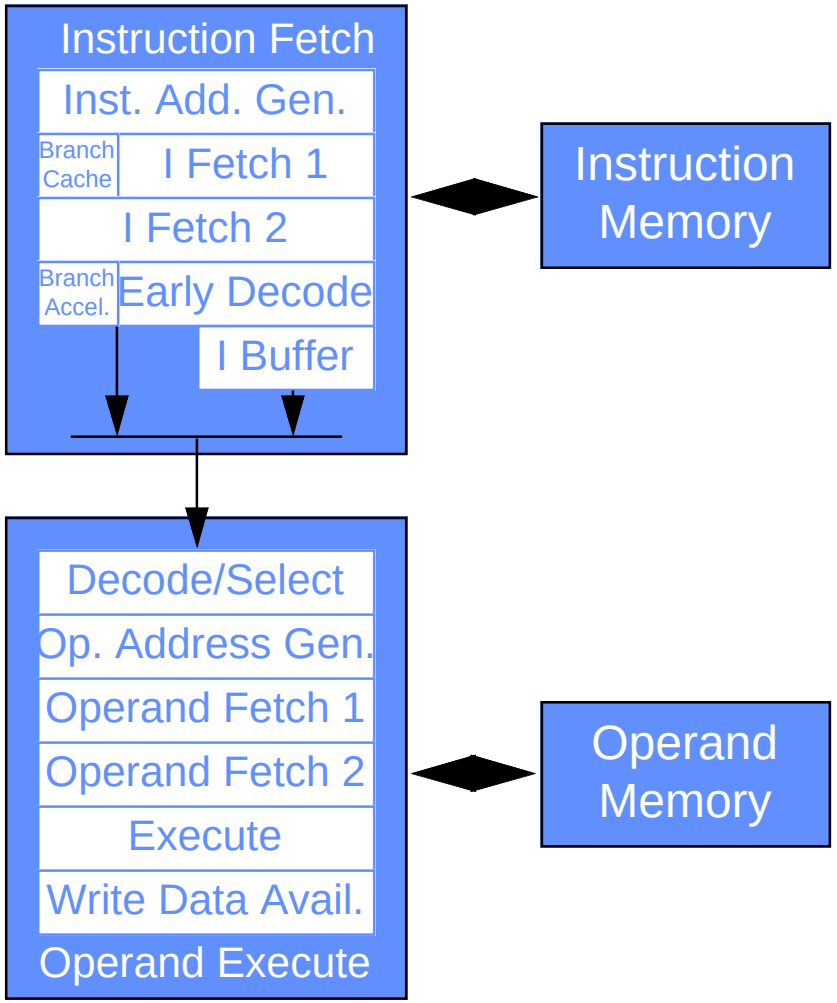
ColdFire® Core Performance Roadmap



MOTOROLA

Digital DNA™ from Motorola

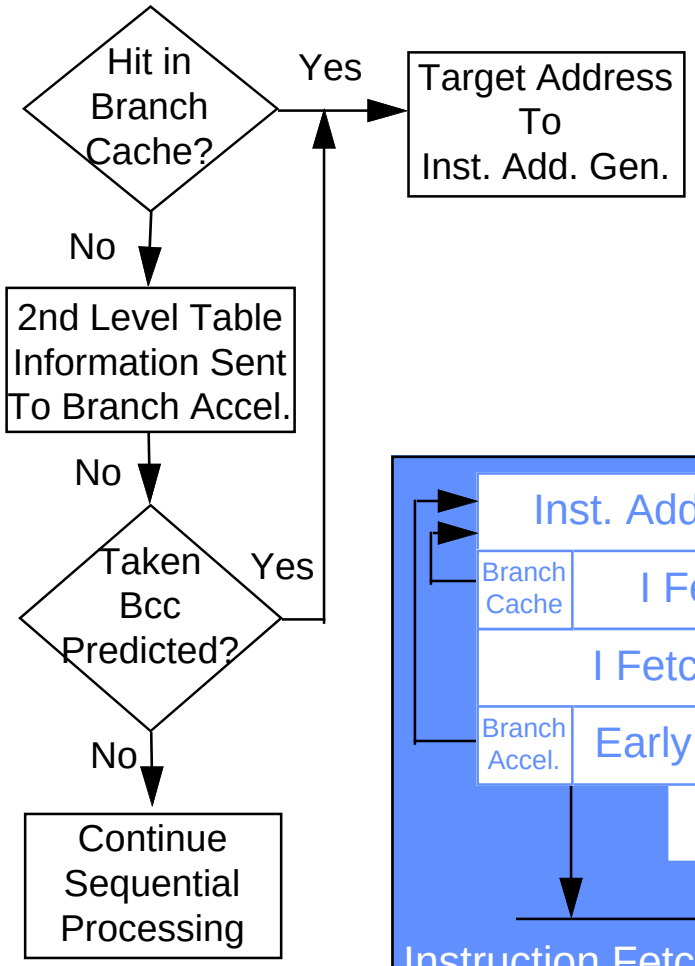
Version 4 ColdFire® Core



- Independent, decoupled pipelines
  - 4-stage Instruction Fetch Pipeline (IFP)
  - 5-stage Operand Execution Pipeline
  - FIFO I-Buffer is the decoupling mechanism
- Limited superscalar execution through use of instruction folding
  - Approaches dual-issue performance but at a much lower silicon cost
- Harvard memory architecture
- Most instructions execute in 1 cycle
- Sophisticated 2-level branch acceleration mechanisms in the IFP minimize execution time of change-of-flow instructions

V4 Branch Execution

- Two-level hierarchical scheme:
  - 8-entry, direct-mapped branch cache
  - 128-entry, direct-mapped table
- 2-bit prediction per entry
- Instruction folding in branch cache on conditional branches for zero-cycle execution
- Unconditional branch calculates target address in Early Decode stage and sends to Inst. Addr. Gen. stage
- 4-entry LIFO hardware stack in Early Decode stage to accelerate subroutine return instructions

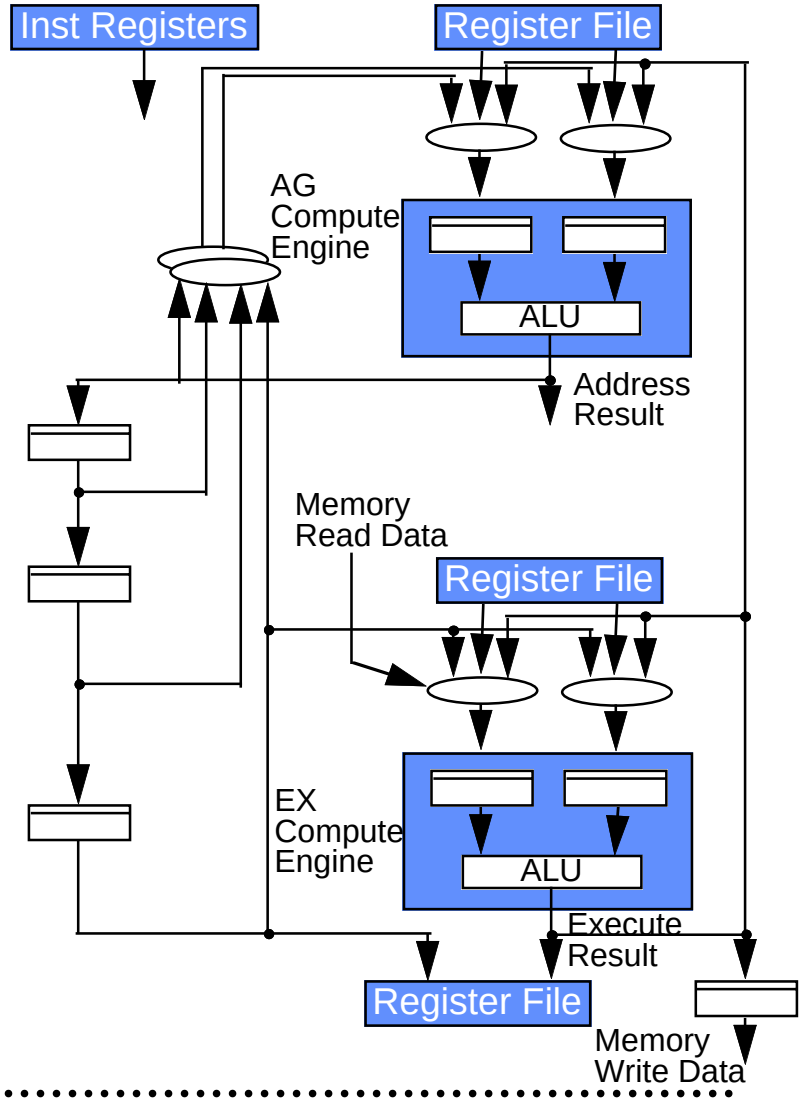


V4 Operand Execution Pipeline (OEP)

- Basic 5-stage execution pipeline
  - Built upon two 2-stage “compute engines”
  - 1st stage = Register File, 2nd stage = ALU
  - Implemented in 1st/2nd stages, 4th/5th stages
  - Optional 6th stage for instructions that store results in memory
- Optimized OAG compute engine
  - Normally used for operand address generation
  - Also used for instruction execution for certain opcodes
  - ISA broadly grouped into 3 categories
    - Always executed in EX (e.g., add.l <mem>,Rx)
    - Always executed in OAG (e.g., lea <ea>,Rx)
    - Can be executed in either (e.g., add.l Ry,Rx)
  - OEP always attempts to relocate execution to OAG engine
- OAG execution plus register renaming resources provides flexibility to minimize pipeline stalls and increase performance

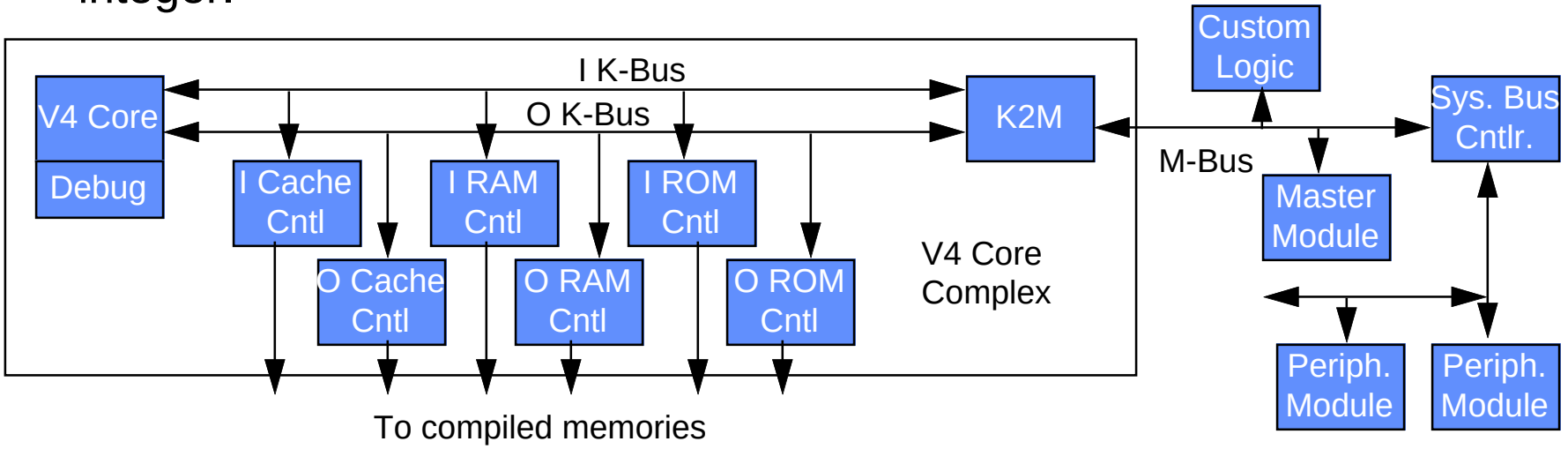
More on the V4 Operand Execution Pipeline

- Instruction folding on heavily used constructs
  - Used with “assignment” opcodes like move.l
  - Combine two instructions and issue into OAG stage together
    - Example: `mov.l <mem>,Rx`
      `op.l Ry,Rx`
    - Combine into a single 3-operand function:
      `op.l <mem>,Ry,Rx`
  - Also used on 2-inst combinations involving different destination registers
- Instruction folding in combination with folding on conditional branches provides limited superscalar functionality
  - Maximize multiple instruction dispatch within a single pipeline



V4 Memory Structure

- V4 Operand Execution Pipeline requires higher KBus bandwidth than previous generations plus operates at higher frequencies.
- Harvard bus architecture doubles available KBus bandwidth, removes instruction and operand conflicts, and provides more flexibility for local memory configurations.
- Design supports multiple clock domains, where core complex operates at n:1 speed compared to the M-Bus speed where n is any integer.



Predicted Version 4 Performance

- V4 predicted performance analysis
  - Use of architectural models similar to those used in V2, V3 analysis work
  - Large set of embedded applications and benchmarks
- Configurations
  - Version 4
    150/75 MHz, 8 KByte I- & O-Caches, 6-3-3-3 (70 ns DRAM) external memory system
  - Version 3
    90/45 MHz, 8 KByte U-Cache, 4-2-2-2 (70 ns DRAM) external memory system
  - Copyback cache mode, no internal SRAM used, no MAC instructions, no new instructions included
- V4 performance provided by a combination of factors:
  - 1.6x frequency factor
  - 1.3x - 1.5x architectural improvement factor
- 150/75 MHz Version 4 = 2.1 - 2.5 x 90/45 MHz Version 3

ColdFire® ISA Additions

Added Instructions

Move 3-bit Data Quick
Move with Sign Extend
Move with Zero-Fill
Move Data Source to Destination, byte and word*
Compare, byte and word*
Compare Immediate, byte and word*
Branch Conditionally, 32-bit displacement*
Branch Always, 32-bit displacement*
Branch to Subroutine, 32-bit displacement*
Signed Saturate
Test and Set an Operand*

* Implementation of 68K opcodes

Functionality added based on input from tool developers and customers to:

- Enhance code density
- Improve performance
- Enhance support for byte, word size operands
- Enhance support for position-independent code
- Assist 68K assembly code migration to ColdFire code

By incorporating these ISA changes, the following architectural metrics were measured across a wide range of applications:

- **Static code size reduction up to 6%**
- **Dynamic pathlength reduction up to 12%**
- **Performance improvement up to 10%**

ColdFire® MAC and Debug Additions

- Multiply-Accumulate Unit Enhancement
  - Addition of signed, fixed-point fractionals to extend the precision range
  - Provides better results for exacting applications (e.g., Digital Audio)
- Debug Enhancements for Rev C
  - Additional on-chip breakpoint registers
    - 3 more PC breakpoint registers
    - Another address range & Data+Mask
  - Ability to service I/O interrupts while in “emulator mode”
  - Functionality added based on input from tool developers and customers to enhance support for real-time debug
- As with all optional ColdFire modules, these modules are easily incorporated into any version of ColdFire core
  - For example, enhanced MAC unit will be included in next mask set of MCF5307

Version 4 *ColdFire®* Core Summary

- Next-generation *ColdFire* microarchitecture provides a 2.5x performance improvement over current V3 designs
  - Improved microarchitecture for higher performance: greater than 200 Dhrystone 2.1 MIPS @ 150 MHz
  - Small set of new instructions for better code density and performance
  - Enhanced MAC and debug functionality
  - Core size = 4.5 mm^2 in 0.25 micron drawn process
  - Preliminary model already deployed to Hewlett-Packard's Integrated Circuit Business Division
  - Version 4 core design complete in November 1998
- First standard V4 integrated microprocessors available 2Q99
- Binary compatibility between 68K and *ColdFire* using software emulation
- Approaching $3 Billion in committed design wins