SECTION 1
INTRODUCTION

1.1 BACKGROUND

The MCF5206 integrated microprocessor combines a ColdFire™ processor core with several peripheral functions such as a DRAM controller, timers, general-purpose I/O and serial interfaces, debug module, and system integration. Designed for embedded control applications, the ColdFire core delivers enhanced performance while maintaining low system costs. To speed program execution, the on-chip instruction cache and SRAM provide one-cycle access to critical code and data. The MCF5206 greatly reduces the time required for system design and implementation by packaging common system functions on-chip and providing glueless interfaces to 8-, 16-, and 32-bit DRAM, SRAM, ROM, and I/O devices.

The revolutionary ColdFire microprocessor architecture gives cost-sensitive, high-volume applications new levels of price and performance. Based on the concept of variable-length RISC technology, ColdFire combines the architectural simplicity of conventional 32-bit RISC with a memory-saving, variable-length instruction set. The denser binary code for ColdFire processors consumes less valuable memory than any fixed-length instruction set RISC processor available. This improved code density means more efficient system memory use for a given application and requires slower, less costly memory to help achieve a target performance level.

The integrated peripheral functions provide high performance and flexibility: The DRAM controller supports as much as 512 Mbytes of DRAM; support for both page-mode and extended-data-out DRAMs; programmable full duplex DUART and a separate I^2C[1]-compatible Motorola bus (M-Bus interface). Two 16-bit general-purpose multimode timers provide separate input and output signals. For system protection, the processor includes a programmable 16-bit software watchdog timer and several bus monitors. In addition, common system functions such as chip-selects, interrupt control, bus arbitration, and IEEE 1149.1 Test (JTAG) support are included.

A sophisticated debug interface supports both background-debug mode and real-time trace. This interface is common to all ColdFire-based processors and allows common emulator support across the entire ColdFire family.

1.2 MCF5206 FEATURES

The primary features of the MCF5206 integrated processor include the following:

---

1. I^2C is a trademark of Phillips.

- ColdFire Processor Core
  - Variable-length RISC
  - 32-bit internal address bus with 28 bit external bus
  - chip-select and DRAM
  - internal 32-bit decoding
  - 32-bit data bus
  - 16 user-visible 32-bit wide registers
  - Supervisor / User modes for system protection
  - Vector base register to relocate exception-vector table
  - Optimized for high-level language constructs
  - 17 MIPS at 33Mhz
- 512-Byte Direct-mapped instruction cache
- 512-Byte on-chip SRAM
  - Provides one-cycle access to critical code and data
- DRAM Controller
  - Programmable refresh timer provides CAS-before-RAS refresh
  - Support for 2 separate memory banks
  - Support for page-mode DRAMs and extended-data-out (EDO) DRAMs
  - Allows external bus master access
- Dual Universal Synchronous/Asynchronous Receiver/Transmitter (DUART)
  - Full duplex operation
  - Baud-rate generator
  - Modem control signals available ($\overline{\text{CTS}}$, $\overline{\text{RTS}}$)
  - Processor-interrupt capability
- Dual 16-Bit General-Purpose Multimode Timers
  - 8-bit prescaler
  - Timer input and output pins
  - 30ns resolution with 33MHz system clock
  - Processor-interrupt capability
- Motorola Bus (M-Bus) Module
  - Interchip bus interface for EEPROMs, LCD controllers, A/D converters, keypads
  - Compatible with industry-standard I^2C Bus
  - Master or slave modes support multiple masters
  - Automatic interrupt generation with programmable level
- System Interface
  - Glueless bus interface to 8-, 16-, and 32-bit DRAM, SRAM, ROM, and I/O devices
  - 8 programmable chip-select signals
  - Programmable wait states and port sizes
  - Allows external bus masters to access chip-selects
  - System protection

    - 16-bit software watchdog timer with prescaler
    - Double bus fault monitor
    - Bus timeout monitor
    - Spurious interrupt monitor
  - Programmable interrupt controller
    - Low interrupt latency
    - 3 external interrupt inputs
    - Programmable interrupt priority and autovector generator
  - IEEE 1149.1 test (JTAG) support
  - 8-bit general-purpose I/O interface

- System Debug Support
  - Real-time trace
  - Background debug interface

- Fully Static 5.0-Volt Operation
- 160 Pin QFP Package

1.3 FUNCTIONAL BLOCKS

Figure 1-1 is a block diagram of the MCF5206 processor. The paragraphs that follow provide an overview of the integrated processor.

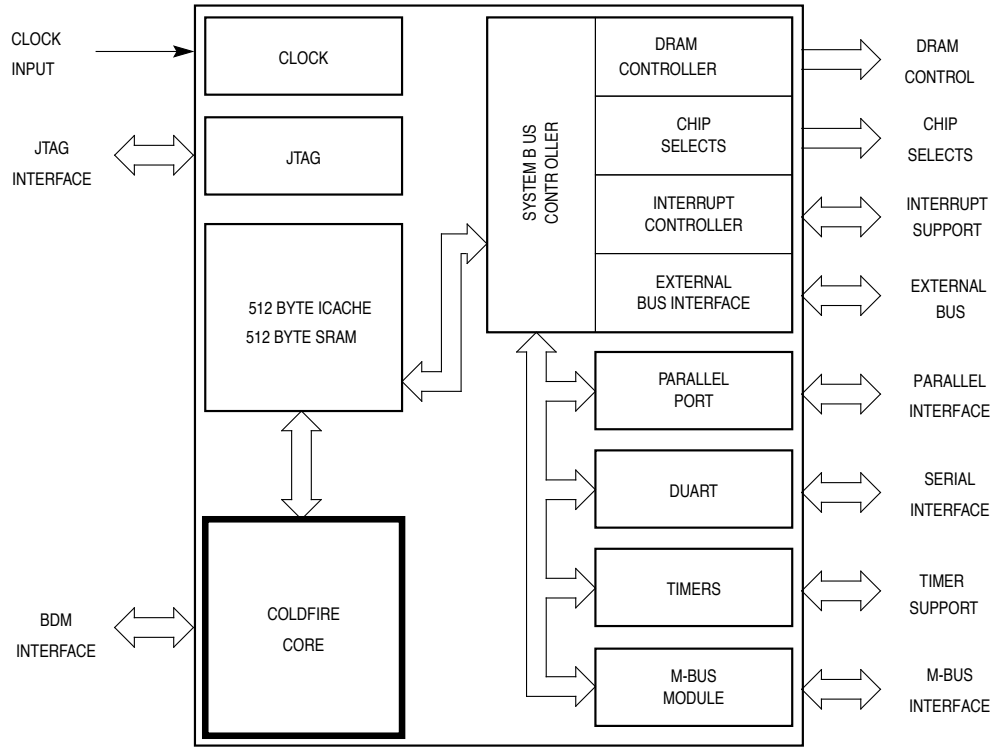

Figure 1-1. MCF5206 Block Diagram

1.3.1 ColdFire Processor Core

The ColdFire processor core consists of two independent, decoupled pipeline structures to maximize performance while minimizing core size.The instruction fetch pipeline (IFP) is a two-stage pipeline for prefetching instructions. The prefetched instruction stream is then gated into the two-stage operand execution pipeline (OEP), which decodes the instruction, fetches the required operands and then executes the required function. Because the IFP and OEP pipelines are decoupled by an instruction buffer that serves as a FIFO queue, the IFP can prefetch instructions in advance of their actual use by the OEP, thereby minimizing time stalled waiting for instructions. The OEP is implemented in a two-stage pipeline featuring a traditional RISC datapath with a dual-read-ported register file feeding an arithmetic/logic unit.

1.3.1.1 PROCESSOR STATES. The processor is always in one of four states: normal processing, exception processing, stopped, or halted. It is in the normal processing state

when executing instructions, fetching instructions and operands, and storing instruction results.

Exception processing is the transition from program processing to system, interrupt, and exception handling; it includes fetching the exception vector, stacking operations, and refilling the instruction fetch pipe after an exception. The processor enters exception processing when an exceptional internal condition arises, such as tracing an instruction, an instruction resulting in a trap, or executing specific instructions; (External conditions, such as interrupts and access errors, also cause exceptions) and ends when the first instruction of the exception handler enters the operand execution pipeline.

Stopped mode is a reduced power operation mode that causes the processor to remain quiescent until either a reset or nonmasked interrupt occurs. The STOP instruction is used to enter this operation mode.

The processor halts when it receives an access error or generates an address error while in the exception processing state. For example, if during exception processing of one access error another access error occurs, the MCF5206 processor cannot complete the transition to normal processing nor can it save the internal machine state. The processor assumes that the system is not operational and halts. Only an external reset can restart a halted processor. When the processor executes a STOP instruction, it is in a special type of normal processing state, e.g., one without bus cycles. The processor stops but it does not halt.

The processor can also halt in a restart mode because of Background-Debug mode events.

1.3.1.2 PROGRAMMING MODEL. The ColdFire programming model is separated into two privilege modes: supervisor and user. The S-bit in the status register (SR) indicates the current privilege mode. The processor identifies a logical address by accessing either the supervisor or user address space, which differentiates between supervisor and user modes.

User programs can access only registers specific to the user mode. System software executing in the supervisor mode can access all registers using the control registers to perform supervisory functions. User programs are thus restricted from accessing privileged information. The operating system performs management and service tasks for user programs by coordinating their activities. This difference allows the supervisor mode to protect system resources from uncontrolled accesses.

Most instructions execute in either mode but some instructions that have important system effects are privileged and can execute only in the supervisor mode. For instance, user programs cannot execute the STOP instructions. To prevent a program executing in user mode from entering the supervisor mode, instructions that can alter the S-bit in the SR are privileged. The TRAP instructions provide controlled access to operating system services for user programs.

When in normal processing, the processor employs the user mode and the user programming model . During exception processing, the processor changes from user to supervisor mode. The current SR value on the stack is saved and then the S-bit is set, forcing the processor into the supervisor mode. To return to the user mode, a system routine must execute a MOVE to SR, or an RTE, which operate in the supervisor mode, modifying

the S-bit of the SR. After these instructions execute, the instruction fetch pipeline flushes and is refilled from the appropriate address space.

The registers depicted in the programming model (see Figure 1-2) provide operand storage and control for the ColdFire processor core. The registers are also partitioned into user and supervisor privilege modes. The user programming model consists of 16 general-purpose, 32-bit registers and two control registers. The supervisor model consists of five more registers that can be accessed only by code running in supervisor mode.

Only system programmers can use the supervisor programming model to implement operating system functions and I/O control. This supervisor/user distinction allows for the coding of application software that will run without modification on any ColdFire Family processor. The supervisor programming model contains the control features that system designers would not want user code to erroneously access as this might effect normal system operation. Furthermore, the supervisor programming model may need to change slightly from ColdFire generation to generation to add features or improve performance as the architecture evolves.

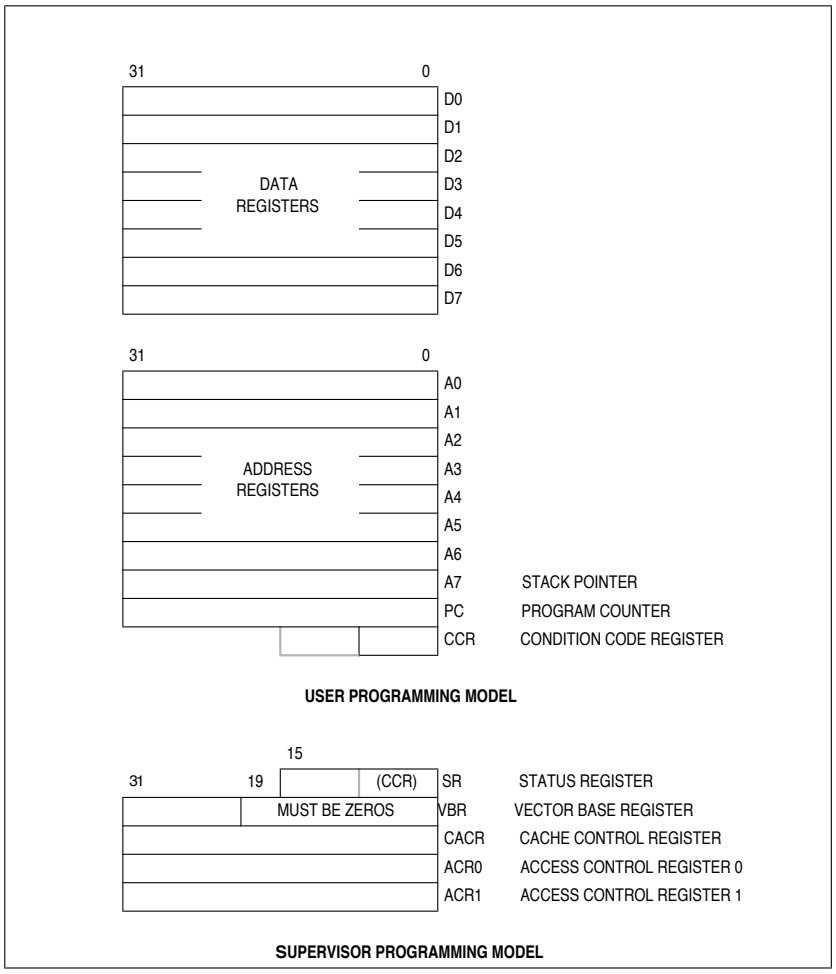

Figure 1-2. Programming Model

The user programming model includes eight data registers, seven address registers, and a stack pointer register. The address registers and stack pointer can be used as base address registers or software stack pointers, and any of the 16 registers can be used as index registers. Two control registers are available in the user mode: the program counter (PC), which contains the address of the instruction that the MCF5206 device is executing, and the lower byte of the SR, which is accessible as the Condition Code Register (CCR). The CCR contains the condition codes that reflect the results of a previous operation and can be used for conditional instruction execution in a program.

The supervisor programming model includes the upper byte of the SR, which contains operation control information. The Vector Base Register (VBR) contains the upper 12 bits of the base address of the exception vector table, which is used in exception processing. The lower 20 bits of the VBR are forced to zero, allowing the vector table to reside on any 1 Mbyte memory boundary.

The Cache Control Register (CACR) controls enabling of the on-chip cache. Two access control registers (ACR1, ACR0) allow portions of the address space to be mapped as noncacheable. See subsections 4.3 and 4.4 for details on these registers.

1.3.1.3 DATA FORMAT SUMMARY. The processor performs all arithmetic using 2's complement, but operands can be signed or unsigned. Registers, memory, or instructions themselves can contain operands. The operand size for each instruction is either explicitly encoded in the instruction or implicitly defined by the instruction operation. Table1-1 summarizes the MCF5206 data formats.

Table 1-1. ColdFire MCF5206 Data Formats

OPERAND DATA FORMAT	SIZE
Bit	1-Bit
Byte	8-Bits
Word	16-Bits
Longword	32-Bits

1.3.1.4 ADDRESSING CAPABILITIES SUMMARY. The MCF5206 processor supports seven addressing modes. The register indirect addressing modes support postincrement, predecrement, offset, and indexing, which are particularly useful for handling data structures common to sophisticated embedded applications and high-level languages. The program counter indirect mode also has indexing and offset capabilities. This addressing mode is typically required to support position-independent software. Besides these addressing modes, the MCF5206 processor provides index scaling features.

An instruction's addressing mode can specify the value of an operand or a register containing the operand. It can also specify how to derive the effective address of an operand in memory. Each addressing mode has an assembler syntax. Some instructions imply the addressing mode for an operand. These instructions include the appropriate fields for operands that use only one addressing mode. Table 1-2 summarizes the effective addressing modes of ColdFire processors. Table 1-3 summarizes specific effective addressing modes. Table 1-4 summarizes the MOVE specific effective addressing modes.

1.3.1.5 NOTATIONAL CONVENTIONS. Table 1-5 lists the notation conventions used throughout this manual, unless otherwise specified.

1.3.1.6 INSTRUCTION SET OVERVIEW. The ColdFire instruction set supports high-level languages and is optimized for those instructions embedded code most commonly executes. Table 1-6 and Table 1-7 provide an alphabetized listing of the ColdFire instruction

Table 1-2. ColdFire Effective Addressing Modes

ADDRESSING MODES	SYNTAX
Register Direct Data Address	 Dn An
Register Indirect Address Address with Postincrement Address with Predecrement Address with Displacement	 (An) (An)+ –(An) (d16,An)
Address Register Indirect with Index 8-Bit Displacement	 (d_8,An,Xi)
Program Counter Indirect with Displacement	 (d_{16},PC)
Program Counter Indirect with Index 8-Bit Displacement	 (d_8,PC,Xi)
Absolute Data Addressing Short Long	 (xxx).W (xxx).L
Immediate	#<xxx>

Table 1-3. MOVE Specific Effective Addressing Modes

SOURCE <EA>	DESTINATION <EA>
Dn	All
An	All
(An)	All
(An)+	All
-(An)	All
(d_{16},An) (d_{16},PC)	Dn An (An) (An)+ -(An) (d_{16},An)
(d_8,An,Xi) (d_8,PC,Xi)	Dn An (An) (An)+ -(An)
(xxx).W (xxx).L	Dn An (An) (An)+ -(An)
#<xxx>	Dn An (An) (An)+ -(An)

Table 1-4. Notational Conventions

SINGLE- AND DOUBLE-OPERAND OPERATIONS	
+	Arithmetic addition or postincrement indicator
–	Arithmetic subtraction or predecrement indicator
¥	Arithmetic multiplication
∏	Arithmetic division or conjunction symbol
~	Invert; operand is logically complemented
L	Logical AND
V	Logical OR
≈	Logical exclusive OR
–>	Source operand is moved to destination operand
<–>	Two operands are exchanged
<op>	Any double-operand operation
<operand>tested	Operand is compared to zero and the condition codes are set appropriately
sign-extended	All bits of the upper portion are made equal to the high-order bit of the lower portion
OTHER OPERATIONS	
STOP	Enter the stopped state, waiting for interrupts
TRAP	Equivalent to Format ' Offset Word ˇ (SSP); SP – 4ˇ SP PCˇ (SP) SP – 2ˇ SP SRˇ (SP) SP – 2ˇ SP Format + Offsetˇ (SP) (Vector)ˇ PC
If <condition> then <operations> else <operations>	Test the condition. If true, the operations after "then" are performed. If the condition is false and the optional "else" clause is present, the operations after "else" are performed. If the condition is false and else is omitted, the instruction performs no operation. Refer to the Bcc instruction description as an example.

REGISTER SPECIFICATION	
An	Any Address Register n (example: A3 is address register 3)
Ay, Ax	Source and destination address registers, respectively
BR	Base Register—An or PC
Dn	Any Data Register n (example: D5 is data register 5)
Dy, Dx	Source and destination data registers, respectively
Rn	Any Address or Data Register
Ry, Rx	Any source and destination registers, respectively
Xi	Index Register—An, Dn, or suppressed

DATA SIZE AND TYPE	
<size>	Operand Data Size: Byte (B), Word (W), Long (L)
B, W, L	Specifies a signed integer data type of byte, word, or longword

SUBFIELDS AND QUALIFIERS

Table 1-4. Notational Conventions (Continued)

#<xxx> or #<data>	Immediate data following the instruction word(s)
()	Identifies an indirect address in a register
d_n	Displacement Value, n Bits Wide (example: d_{16} is a 16-bit displacement)
LSB	Least Significant Bit
LSW	Least Significant Word
MSB	Most Significant Bit
MSW	Most Significant Word
SCALE	A scale factor (1, 2, or 4)
REGISTER NAMES	
CCR	Condition Code Register (lower byte of status register)
PC	Program Counter
SR	Status Register
Rc	Any Control Register (e.g., VBR, CACR)
REGISTER CODES	
*	General Case
C	Carry Bit in CCR
cc	Condition Codes from CCR (c, n, v, x, z) c = carry bit is set n = negative number v = overflow x = sign extended z = zero
FC	Function Code
N	Negative Bit in CCR
U	Undefined, Reserved for Motorola Use.
V	Overflow Bit in CCR
X	Extend Bit in CCR
Z	Zero Bit in CCR
—	Not Affected or Applicable

MISCELLANEOUS	
<ea>y, <ea>x	Any source or destination effective address, respectively
<label>	Assembly Program Label
<list>	List of registers, for example D3–D0
m	Bit m of an Operand
m–n	Bits m through n of Operand

set opcode, operation, and syntax. The left operand in the syntax is always the source operand and the right operand is the destination operand.

OPCODE	SUPPORTED OPERAND SIZES	ADDRESSING MODES	OPERATION
ADD	.L	Dy,<ea>,x <ea>y,Dx	Source + Destination - > Destination
ADDA	.L	<ea>y,Ax	Source + Destination - > Destination
ADDI	.L	#<data>,Dx	Immediate Data + Destination -> Destination
ADDQ	.L	#<data>,<ea>x	Immediate Data + Destination -> Destination
ADDX	.L	Dy,Dx	
AND	.L	Dy,<ea>x <ea>y, Dx	Source L Destination -> Destination
ANDI	.L	#<data>,Dx	Immediate Data L Destination -> Destination
ASL	.L	Dx,Dy #<data>,Dx	
ASR	.L	Dx,Dy #<data>,Dx	
Bcc	.B,.W	<label>	If Condition true, then PC + dn -> PC
BCHG	.B,.L	Dy,<ea>x #<data>,<ea>x	~(<Bit Number> of Destination) -> Z; Bit of Destination
BCLR	.B,.L	Dy,<ea>x #<data>,<ea>x	~ (<Bit number> of Destination) -> Z; 0 -> Bit of Destination
BRA	.B,.W	<label>	PC + dn -> PC
BSET	.B,.L	Dy,<ea>x #<data>,<ea>x	~(<Bit number> of Destination) -> Z; 1 -> Bit of Destination
BSR	.B,.W	<label>	SP - 4 -> SP; Next PC -> (SP); PC + dn -> PC
BTST	.B,.L	Dy,<ea>x #<data>,<ea>x	~ (<Bit number> of Destination) -> Z
CLR	.B,.W,.L	<ea>x	0 -> Destination
CMP	.L	<ea>y,Dx	Destination - Source
CMPA	.L	<ea>y,Ax	Destination - Source
CMPI	.L	#<data>,Dx	Destination - Immediate Data
EOR	.L	Dy,<ea>x	Source ≈ Destination -> Destination
EORI	.L	#<data>,Dx	Immediate Data ≈ Destination -> Destination
EXT	.W,.L	Dx	Sign-extended Destination -> Destination
EXTB	.L	Dx	Sign-extended Destination -> Destination

OPCODE	SUPPORTED OPERAND SIZES	ADDRESSING MODES	OPERATION
JMP	Unsized	<ea>y	<ea>y -> PC
JSR	Unsized	<ea>y	SP - 4 -> SP;Next PC -> (SP); <ea>y -> PC
LEA	.L	<ea>y,Ax	<ea>-> An
LINK	.W	Ax,#data	SP - 4 -> SP;Ax -> (SP); SP -> Ax; SP + d16 -> SP
LSL	.L	Dx,Dy #<data>,Dx	
LSR	.L	Dx,Dy #<data>,Dx	
MOVE	.B,.W,.L	<ea>y,<ea>x	Source -> Destination
MOVE from CCR	.W	Dx	CCR -> Destination
MOVE to CCR	.W	Dy,CCR #<data>,CCR	Source -> CCR
MOVEM	.L	list,<ea>x <ea>y,list	Listed Registers -> Destination Source -> Listed Registers
MOVEQ	.L	#<data>,Dx	Sign-extended Immediate Data -> Destination
MULS	.W	<ea>y,Dx	Source x Destination -> Destination
MULS	.L	<ea>y,Dx	Source x Destination -> Destination
MULU	.W	<ea>y,Dx	Source x Destination -> Destination
MULU	.L	<ea>y,Dx	Source x Destination -> Destination
NEG	.L	<ea>x	0 - Destination -> Destination
NEGX	.L	<ea>x	0 - Destination - X -> Destination
NOP	Unsized		
NOT	.L	<ea>x	~ Destination -> Destination
OR	.L	Dy,<ea>x <ea>y,Dx	Source V Destination -> Destination
ORI	.L	#<data>,Dx	Immediate Data V Destination -> Destination
PEA	.L	<ea>y	SP - 4 -> SP; <ea>y -> (SP)
PULSE	None		Generate unique PST value
RTS	None		(SP) -> PC; SP + 4 -> SP
Scc	.B	Dx	If condition true, then 1's -> Destination; else o's -> Destination
SUB	.L	Dy,<ea>x <ea>y,Dx	Destination - Source -> Destination
SUBA	.L	<ea>,Ax	Destination - Source -> Destination
SUBI	.L	#<data>,Dx	Destination - Immediate Data -> Destination

OPCODE	SUPPORTED OPERAND SIZES	ADDRESSING MODES	OPERATION
SUBQ	.L	#<data>,<ea>x	Destination - Immediate Data -> Destination
SUBX	.L	Dy,Dx	Destination - Source - X -> Destination
SWAP	.W	Dx	
TRAP	None	#<vector>	
TRAPF	None		PC + 2 -> PC
TRAPF	.W,.L	#<data>	PC + 4 -> PC PC + 6 -> PC
TST	.B,.W,.L	<ea>y	Set Condition Codes
UNLK	Unsized	Ax	Ax -> SP; (SP) -> Ax; SP + 4 -> SP
WDDATA	.B,.W,.L	<ea>y	(<ea>y) -> DDATA port

Table 1-6. Supervisor-Mode Instruction Summary

OPCODE	SUPPORTED OPERAND SIZES	ADDRESSING MODES	OPERATION
CPUSHL	Unsized	Ax	
HALT*	Unsized		
MOVE from SR	.W	Dx	SR -> Destination
MOVE to SR	.W	Dy,SR #<data>,SR	Source -> SR
MOVEC	.L	Rn,Rc	Ry -> Rc
RTE	None		2(SP) -> SR; 4 (SP) -> PC; SP + 8 -> SP. Adjust stack according to format
STOP	Unsized	#<data>	Immediate Data -> SR; STOP
WDEBUG	.L	<ea>y	<ea>y -> DEBUG;<ea>y+4 -> DEBUG
*The HALT instruction can be changed to allow supervisor or user mode execution by setting the UHE bit in the Real-Time Debug Support Configuration/Status Register (CSR).			

1.3.2 Instruction Cache

The instruction cache improves system performance by providing cached instructions to the execution unit in a single clock. The MCF5206 processor uses a 512-byte, direct-mapped instruction cache to achieve 17 MIPS at 33 MHz.The cache is accessed by physical addresses, where each 16-byte line consists of an address tag and a valid bit.

The instruction cache also includes a bursting interface for 32-, 16-, and 8-bit port sizes to quickly fill cache lines.

1.3.3 Internal SRAM

The 512-byte on-chip SRAM provides one clock-cycle access for the ColdFire core. This SRAM can store processor stack and critical code or data segments to maximize performance.

1.3.4 DRAM Controller

The MCF5206 DRAM controller provides a glueless interface for as many as two banks of DRAM, each of which can be from 128 Kbytes to 256 Mbytes in size. The controller supports an 8-, 16-, or 32-bit data bus. A unique addressing scheme allows for increases in system memory size without rerouting address lines and rewiring boards. The controller operates in fast page mode, burst-page mode, or normal mode, and supports extended-data-out (EDO) DRAMs.

DRAM operations are available to other external bus masters. The DRAM controller can generate $\overline{\text{CAS}}$ and $\overline{\text{RAS}}$ for an external master and can continue to manage refresh requests.

1.3.5 DUART Module

A full duplex DUART module contains independent receivers and transmitters that can be clocked by the DUART internal timer. This timer is clocked by the system clock or an external clock supplied by the TIN pin. Data formats can be 5, 6, 7, or 8 bits with even, odd, or no parity, and as many as 2 stop bits in 1/16 increments. Four-byte receive buffers and two-byte transmit buffers minimize CPU service calls. The DUART module also provides several error-detection and maskable-interrupt capabilities. Modem support includes request-to-send ($\overline{\text{RTS}}$) and clear-to-send ($\overline{\text{CTS}}$) lines.

The system clock provides the clocking function via a programmable prescaler. You can select full duplex, autoecho loopback, local loopback, and remote loopback modes. The programmable DUART can interrupt the CPU on various normal or error-condition events.

1.3.6 Timer Module

The timer module includes two general-purpose timers, each of which contains a free-running 16-bit timer for use in any of three modes. One mode captures the timer value with an external event. Another mode triggers an external signal or interrupts the CPU when the timer reaches a set value, while a third mode counts external events. The timer unit has an 8-bit prescaler that allows for programming the clock input frequency, which is derived from the system clock. The programmable timer-output pin generates either an active-low pulse or toggles the output.

1.3.7 Motorola Bus (M-Bus) Module

The M-Bus interface is a two-wire, bidirectional serial bus that exchanges data between devices and is compatible with the I^2C Bus standard. The M-Bus minimizes the interconnection between devices in the end system and is best suited for applications that need occasional bursts of rapid communication over short distances among several devices. The number of devices that can be connected is limited by bus capacitance and the number of unique addresses.

1.3.8 System Interface

The MCF5206 processor provides a glueless interface to 8-, 16-, and 32-bit port size SRAM, ROM, and peripheral devices with independent programmable control of the assertion and negation of chip-selects and write-enables. Programmable address and data-hold times can be extended for a compatible interface to external devices and memory. The MCF5206 also supports bursting ROMs.

1.3.8.1 EXTERNAL BUS INTERFACE. The bus interface controller transfers information between the ColdFire core and memory, peripherals, or other masters on the external bus. The external bus interface provides as many as 28 bits of address bus space, a 32-bit data bus, and all associated control signals. This interface implements an extended synchronous protocol that supports bursting operations. For nonsynchronous external memory and peripherals, the MCF5206 processor provides an alternate asynchronous bus transfer acknowledgment signal.

Simple two-wire request/acknowledge bus arbitration between the MCF5206 processor and another bus master, such as a DMA device, is glueless with arbitration handled internal to the MCF5206 processor. Alternately, an external bus arbiter can control more complex three-wire (request, grant, busy) multiple-master bus arbitration, allowing overlapped bus arbitration with one clock-bus handovers.

1.3.8.2 CHIP-SELECTS . Eight programmable chip-select outputs provide signals that enable external memory and peripheral circuits for automatic wait-state insertion. These signals also interface to 8-, 16-, or 32-bit ports. In addition, other external bus masters can access chip-selects. The upper four chip-selects are multiplexed with A[27:24] of the address bus and the four write-enable signals. The base address, access permissions, and timing waveforms are all programmable with configuration registers.

1.3.9 8-Bit Parallel Port (General-Purpose I/O)

An 8-bit general-purpose programmable parallel port serves as either an input or an output on a bit-by-bit basis. The parallel port is multiplexed with PST[3:0] and DDATA[3:0] debug signals.

1.3.10 Interrupt Controller

The interrupt controller provides user-programmable control of three or seven external interrupt and five internal peripheral interrupts. You can program each internal interrupt to any one of seven interrupt levels and four priority levels within each of these levels. You can configure the three external interrupt signals as either fixed interrupt levels 1, 4, and 7, or as a seven-level encoded interrupt. You can program the external interrupts to any one of the four priority levels within the respective interrupt levels.

1.3.11 System Protection

The MCF5206 processor contains a 16-bit software watchdog timer with an 8-bit prescaler. The programmable software watchdog timer provides either a level 7 interrupt or a hardware reset on timeout. The MCF5206 processor also contains a reset status register that indicates the cause of the last reset.

1.3.12 JTAG

To help with system diagnostics and manufacturing testing, the MCF5206 processor includes dedicated user-accessible test logic that complies with the IEEE 1149.1 standard for boundary scan testability, often referred to as Joint Test Action Group, or JTAG. For more information, refer to the IEEE 1149.1 standard.

1.3.13 System Debug Interface

The ColdFire processor core debug interface supports real-time trace and Background-Debug Mode. A four-pin Background Debug Mode (BDM) interface provides system debug. The BDM is a proper subset of the BDM interface provided on Motorola's 683XX Family of parts.

In real-time trace, four status lines provide information on processor activity in real time (PST pins). A 4-bit wide debug data bus (DDATA) displays operand data, which helps track the machine's dynamic execution path as the change-of-flow instructions execute. These signals are multiplexed with an 8-bit parallel port for application development, which does not use real-time trace.

1.3.14 Pinout and Package

The MCF5206 device is supplied in a 160-pin plastic quad flat pack package.