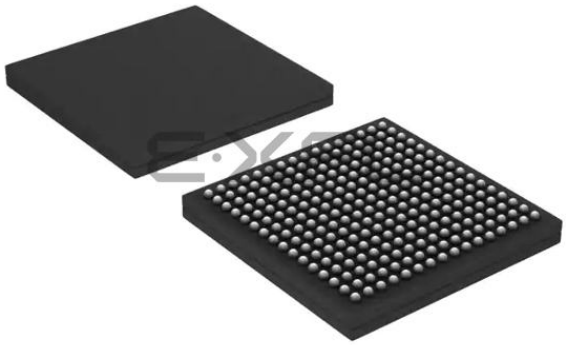

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	Coldfire V2
Core Size	32-Bit Single-Core
Speed	66MHz
Connectivity	CANbus, EBI/EMI, Ethernet, I²C, SPI, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	150
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	256-LBGA
Supplier Device Package	256-MAPBGA (17x17)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mcf5282cvf66j

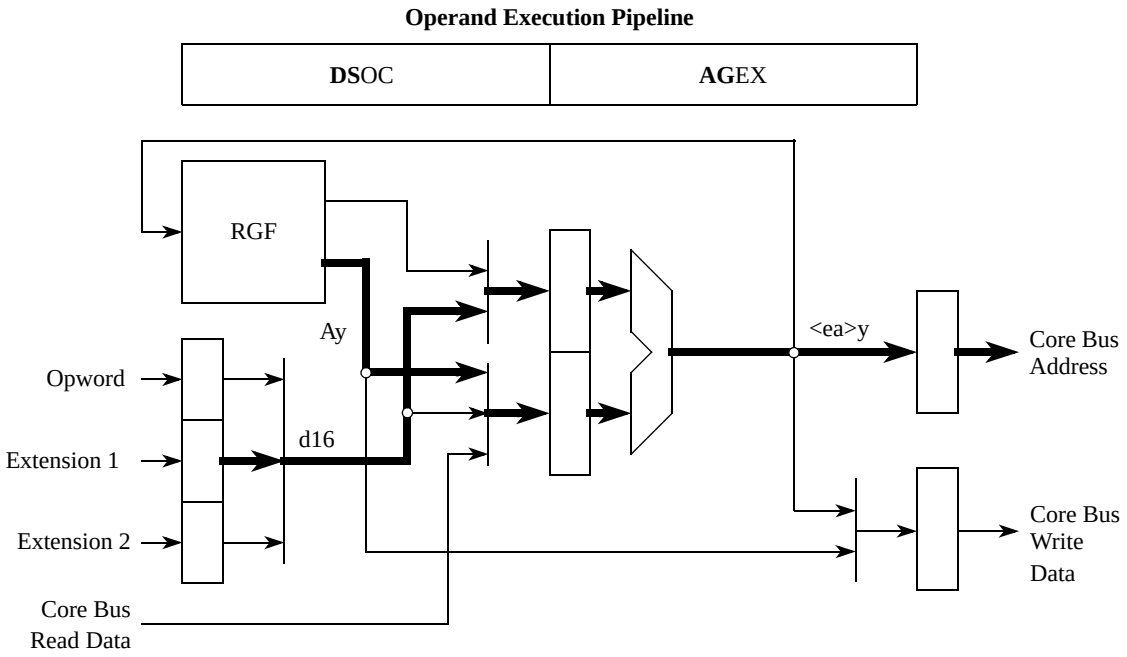

Figure 2-12. V2 OEP Embedded-Load Part 1

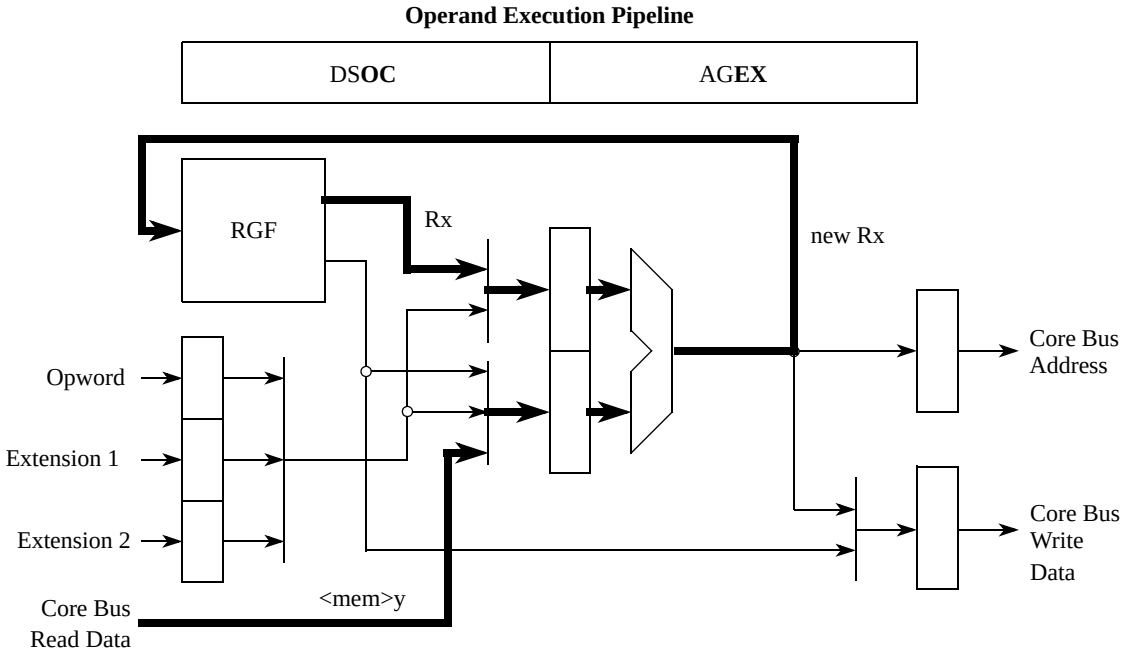

Figure 2-13. V2 OEP Embedded-Load Part 2

For register-to-memory (store) operations, the stage functions (DS/OC, AG/EX) are effectively performed simultaneously allowing single-cycle execution. See Figure 2-14 where the effective address is of the form <ea>x = (d16,Ax), i.e., a 16-bit signed displacement added to a base register Ax.

3. The processor then generates a trace exception. The PC in the exception stack frame points to the instruction after the stop, and the SR reflects the value loaded in the previous step.

If the processor is not in trace mode and executes a stop instruction where the immediate operand sets SR[T], hardware loads the SR and generates a trace exception. The PC in the exception stack frame points to the instruction after the stop, and the SR reflects the value loaded in step 2.

Because ColdFire processors do not support any hardware stacking of multiple exceptions, it is the responsibility of the operating system to check for trace mode after processing other exception types. As an example, consider a TRAP instruction execution while in trace mode. The processor initiates the trap exception and then passes control to the corresponding handler. If the system requires that a trace exception be processed, it is the responsibility of the trap exception handler to check for this condition (SR[T] in the exception stack frame set) and pass control to the trace handler before returning from the original exception.

2.3.4.7 Unimplemented Line-A Opcode

A line-A opcode is defined when bits 15-12 of the opword are 0b1010. This exception is generated by the attempted execution of an undefined line-A opcode.

2.3.4.8 Unimplemented Line-F Opcode

A line-F opcode is defined when bits 15-12 of the opword are 0b1111. This exception is generated when attempting to execute an undefined line-F opcode.

2.3.4.9 Debug Interrupt

See Chapter 30, “Debug Support,” for a detailed explanation of this exception, which is generated in response to a hardware breakpoint register trigger. The processor does not generate an IACK cycle, but rather calculates the vector number internally (vector number 12). Additionally, SR[M,I] are unaffected by the interrupt.

2.3.4.10 RTE and Format Error Exception

When an RTE instruction is executed, the processor first examines the 4-bit format field to validate the frame type. For a ColdFire core, any attempted RTE execution (where the format is not equal to {4,5,6,7}) generates a format error. The exception stack frame for the format error is created without disturbing the original RTE frame and the stacked PC pointing to the RTE instruction.

The selection of the format value provides some limited debug support for porting code from M68000 applications. On M68000 family processors, the SR was located at the top of the stack. On those processors, bit 30 of the longword addressed by the system stack pointer is typically zero. Thus, if an RTE is attempted using this old format, it generates a format error on a ColdFire processor.

If the format field defines a valid type, the processor: (1) reloads the SR operand, (2) fetches the second longword operand, (3) adjusts the stack pointer by adding the format value to the auto-incremented address after the fetch of the first longword, and then (4) transfers control to the instruction address defined by the second longword operand within the stack frame.

NOTE

The execution times for moving the contents of the Racc, Raccext[01,23], MACSR, or Rmask into a destination location <ea>x shown in this table represent the best-case scenario when the store instruction is executed and there are no load or M{S}AC instructions in the EMAC execution pipeline. In general, these store operations require only a single cycle for execution, but if preceded immediately by a load, MAC, or MSAC instruction, the depth of the EMAC pipeline is exposed and the execution time is four cycles.

2.3.5.7 Branch Instruction Execution Times

Table 2-18. General Branch Instruction Execution Times

Opcode	<EA>	Effective Address							
		Rn	(An)	(An)+	-(An)	(d16,An) (d16,PC)	(d8,An,Xi*SF) (d8,PC,Xi*SF)	xxx.wl	#xxx
BRA		—	—	—	—	2(0/1)	—	—	—
BSR		—	—	—	—	3(0/1)	—	—	—
JMP	<ea>	—	3(0/0)	—	—	3(0/0)	4(0/0)	3(0/0)	—
JSR	<ea>	—	3(0/1)	—	—	3(0/1)	4(0/1)	3(0/1)	—
RTE		—	—	10(2/0)	—	—	—	—	—
RTS		—	—	5(1/0)	—	—	—	—	—

Table 2-19. Bcc Instruction Execution Times

Opcode	Forward Taken	Forward Not Taken	Backward Taken	Backward Not Taken
Bcc	3(0/0)	1(0/0)	2(0/0)	3(0/0)

6.4.3.4 Flash User Mode Illegal Operations

The ACCERR flag will be set during a command write sequence if any of the illegal operations below are performed. Such operations will cause the command sequence to immediately abort.

1. Writing to the CFM array before initializing CFMCLKD.
2. Writing to the CFM array while in emulation mode.
3. Writing a byte or a word to the CFM array. Only 32-bit longword programming is allowed.
4. Writing to the CFM array while CBEIF is not set.
5. Writing an invalid user command to the CFMCMD.
6. Writing to any CFM other than CFMCMD after writing a longword to the CFM array.
7. Writing a second command to CFMCMD before executing the previously written command.
8. Writing to any CFM register other than CFMUSTAT (to clear CBEIF) after writing to the command register.
9. Entering stop mode while a program or erase command is in progress.
10. Aborting a command sequence by writing a 0 to CBEIF after the longword write to the CFM array or after writing a command to CFMCMD and before launching it.

The PVIOL flag will be set during a command write sequence after the longword write to the CFM array if any of the illegal operations below are performed. Such operations will cause the command sequence to immediately abort.

1. Writing to an address in a protected area of the CFM array.
2. Writing a mass erase command to CFMCMD while any logical sector is protected (see Section 6.3.4.4, “CFM Protection Register (CFMPROT)”).

If a Flash physical block is read during a program or erase operation on that block (CFMUSTAT bit CCIF = 0), the read will return non-valid data and the ACCERR flag will not be set.

6.4.4 Stop Mode

If a command is active (CCIF = 0) when the MCU enters stop mode, the command sequence monitor performs the following:

1. The command in progress aborts
2. The Flash high voltage circuitry switches off and any pending command (CBEIF = 0) does not executed when the MCU exits stop mode.
3. The CCIF and ACCERR flags are set if a command is active when the MCU enters stop mode.

NOTE

The state of any longword(s) being programmed or any erase pages/physical blocks being erased is not guaranteed if the MCU enters stop mode with a command in progress.

WARNING

Active commands are immediately aborted when the MCU enters stop mode. Do not execute the STOP instruction during program and erase operations.

[4] The BDM logic is clocked by a separate TCLK clock. Entering halt mode via the BDM port exits any low-power mode. Upon exit from halt mode, the previous low-power mode will be re-entered and changes made in halt mode will remain in effect.

8.6.2 Features

Each bus transfer can be classified by its privilege level and the reference type. The complete set of access types includes:

- Supervisor instruction fetch
- Supervisor operand read
- Supervisor operand write
- User instruction fetch
- User operand read
- User operand write

Instruction fetch accesses are associated with the execute attribute.

It should be noted that while the bus does not implement the concept of reference type (code versus data) and only supports the user/supervisor privilege level, the reference type attribute is supported by the system bus. Accordingly, the access checking associated with both privilege level and reference type is performed in the IPS controller using the attributes associated with the reference from the system bus.

The SACU partitions the access control mechanisms into three distinct functions:

- Master privilege register (MPR)
  - Allows each bus master to be assigned a privilege level:
    - Disable the master's user/supervisor attribute and force to user mode access
    - Enable the master's user/supervisor attribute
  - The reset state provides supervisor privilege to the processor core (bus master 0).
  - Input signals allow the non-core bus masters to have their user/supervisor attribute enabled at reset. This is intended to support the concept of a trusted bus master, and also controls the ability of a bus master to modify the register state of any of the SACU control registers; that is, only trusted masters can modify the control registers.
- Peripheral access control registers (PACRs)
  - Nine 8-bit registers control access to 17 of the on-chip peripheral modules.
  - Provides read/write access rights, supervisor/user privilege levels
  - Reset state provides supervisor-only read/write access to these modules
  - Grouped peripheral access control registers (GPACR0, GPACR1)
  - One single register (GPACR0) controls access to 14 of the on-chip peripheral modules
  - One register (GPACR1) controls access for IPS reads and writes to the Flash module
  - Provide read/write/execute access rights, supervisor/user privilege levels
  - Reset state provides supervisor-only read/write access to each of these peripheral spaces

8.6.3 Memory Map/Register Definition

The memory map for the SACU program-visible registers within the System Control Module (SCM) is shown in Figure 8-7. The MPR, PACR, and GPACRs are 8 bits in width.

and writes. Each PACR follows the format illustrated in Figure 8-9. For a list of PACRs and the modules that they control, refer to Table 8-11.

	7	6–4	3	2–0
Field	LOCK1	ACCESS_CTRL1	LOCK0	ACCESS_CTRL0
Reset	0000_0000			
R/W	R/W			
Address	IPSBAR + 0x24 + Offset			

Figure 8-9. Peripheral Access Control Register (PACR*n*)

Table 8-9. PACR Field Descriptions

Bits	Name	Description
7	LOCK1	This bit, when set, prevents subsequent writes to ACCESSCTRL1. Any attempted write to the PACR generates an error termination and the contents of the register are not affected. Only a system reset clears this flag.
6–4	ACCESS_CTRL1	This 3-bit field defines the access control for the given platform peripheral. The encodings for this field are shown in Table 8-10.
3	LOCK0	This bit, when set, prevents subsequent writes to ACCESSCTRL0. Any attempted write to the PACR generates an error termination and the contents of the register are not affected. Only a system reset clears this flag.
2–0	ACCESS_CTRL0	This 3-bit field defines the access control for the given platform peripheral. The encodings for this field are shown in Table 8-10.

Table 8-10. PACR ACCESSCTRL Bit Encodings

Bits	Supervisor Mode	User Mode
000	Read/Write	No Access
001	Read	No Access
010	Read	Read
011	Read	No Access
100	Read/Write	Read/Write
101	Read/Write	Read
110	Read/Write	Read/Write
111	No Access	No Access

Table 8-11. Peripheral Access Control Registers (PACRs)

IPSBAR Offset	Name	Modules Controlled	
		ACCESS_CTRL1	ACCESS_CTRL0
0x024	PACR0	SCM	SDRAMC
0x025	PACR1	EIM	DMA
0x026	PACR2	UART0	UART1

Table 9-10. Stop Mode Operation (Sheet 4 of 5)

MODE In	LOCEN	LOCRE	LOLRE	PLL	OSC	EWKUP	Expected PLL Action at Stop	PLL Action During Stop	MODE Out	LOCKSS	LOCK	LOCS	Comments
NRM	1	0	1	On	On	X	—	—	NRM	‘LK	1	‘LC	
								Lose lock or clock	RESET	—	—	—	Reset immediately
NRM	1	1	X	Off	X	X	Lose lock, f.b. clock, reference clock	RESET	RESET	—	—	—	Reset immediately
NRM	1	1	0	On	On	0	—	—	NRM	‘LK	1	‘LC	
								Lose clock	RESET	—	—	—	Reset immediately
								Lose lock	Stuck	—	—	—	
								Lose lock, regain	NRM	0	1	‘LC	
NRM	1	1	0	On	On	1	—	—	NRM	‘LK	1	‘LC	
								Lose clock	RESET	—	—	—	Reset immediately
								Lose lock	Unstable NRM	0	0–> 1	‘LC	
								Lose lock, regain	NRM	0	1	‘LC	
NRM	1	1	1	On	On	X	—	—	NRM	‘LK	1	‘LC	
								Lose clock or lock	RESET	—	—	—	Reset immediately
REF	1	0	0	X	X	X	—	—	REF	0	X	1	
								Lose reference clock	Stuck	—	—	—	
SCM	1	0	0	Off	X	0	PLL disabled	Regain SCM	SCM	0	0	1	Wakeup without lock
SCM	1	0	0	Off	X	1	PLL disabled	Regain SCM	SCM	0	0	1	
SCM	1	0	0	On	On	0	—	—	SCM	0	0	1	Wakeup without lock
								Lose reference clock	SCM				

Table 12-8. CSCR*n* Field Descriptions (continued)

Bits	Name	Description
3	BSTW	Burst write enable. Specifies whether burst writes are used for memory associated with each $\overline{CS}n$. 0 Break data larger than the specified port size into individual port-sized, non-burst writes. For example, a longword write to an 8-bit port takes four byte writes. 1 Enables burst write of data larger than the specified port size, including longword writes to 8 and 16-bit ports, word writes to 8-bit ports and line writes to 8-, 16-, and 32-bit ports.
2–0	—	Reserved, should be cleared.

14.2.14.7 Debug Data (DDATA[3:0])

Debug data signals (DDATA[3:0]) display captured processor addresses, data and breakpoint status.

These pins can also be configured as GPIO PDD[7:4].

14.2.14.8 Processor Status Outputs (PST[3:0])

PST[3:0] outputs indicate core status, as shown below in Table 14-7. Debug mode timing is synchronous with the processor clock; status is unrelated to the current bus transfer.

These pins can also be configured as GPIO PDD[3:0].

Table 14-7. Processor Status Encoding

PST[3:0]	Definition
0000	Continue execution
0001	Begin execution of an instruction
0010	Reserved
0011	Entry into user mode
0100	Begin execution of PULSE and WDDATA instruction
0101	Begin execution of taken branch
0110	Reserved
0111	Begin execution of RTE instruction
1000	Begin one-byte transfer on DDATA
1001	Begin two-byte transfer on DDATA
1010	Begin three-byte transfer on DDATA
1011	Begin four-byte transfer on DDATA
1100	Exception Processing
1101	Emulator-Mode Exception Processing
1110	Processor is stopped
1111	Processor is halted

14.2.15 Test Signals

14.2.15.1 Test (TEST)

This input signal is reserved for factory testing only and should be connected to VSS to prevent unintentional activation of test functions.

IPSBAR Offset: 0x1084 Access: User read/write

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	0	0	0	0	MAX_FL										
W																
Reset	0	0	0	0	0	1	0	1	1	1	1	0	1	1	1	0

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0	0	0	FCE	BC_REJ	PROM	MII_MODE	DRT	LOOP
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Figure 17-10. Receive Control Register (RCR)

Table 17-14. RCR Field Descriptions

Field	Description
31–27	Reserved, must be cleared.
26–16 MAX_FL	Maximum frame length. Resets to decimal 1518. Length is measured starting at DA and includes the CRC at the end of the frame. Transmit frames longer than MAX_FL causes the BABT interrupt to occur. Receive frames longer than MAX_FL causes the BABR interrupt to occur and sets the LG bit in the end of frame receive buffer descriptor. The recommended default value to be programmed is 1518 or 1522 if VLAN tags are supported.
15–6	Reserved, must be cleared.
5 FCE	Flow control enable. If asserted, the receiver detects PAUSE frames. Upon PAUSE frame detection, the transmitter stops transmitting data frames for a given duration.
4 BC_REJ	Broadcast frame reject. If asserted, frames with DA (destination address) equal to FFFF_FFFF_FFFF are rejected unless the PROM bit is set. If BC_REJ and PROM are set, frames with broadcast DA are accepted and the M (MISS) is set in the receive buffer descriptor.
3 PROM	Promiscuous mode. All frames are accepted regardless of address matching.
2 MII_MODE	Media independent interface mode. Selects the external interface mode for transmit and receive blocks. 0 7-wire mode (used only for serial 10 Mbps) 1 MII mode
1 DRT	Disable receive on transmit. 0 Receive path operates independently of transmit (use for full duplex or to monitor transmit activity in half duplex mode). 1 Disable reception of frames while transmitting (normally used for half duplex mode).
0 LOOP	Internal loopback. If set, transmitted frames are looped back internal to the device and transmit output signals are not asserted. The internal bus clock substitutes for the FEC_TXCLK when LOOP is asserted. DRT must be set to 0 when setting LOOP.

17.4.11 Transmit Control Register (TCR)

TCR is read/write and configures the transmit block. This register is cleared at system reset. Bits 2 and 1 must be modified only when ECR[ETHER_EN] is cleared.

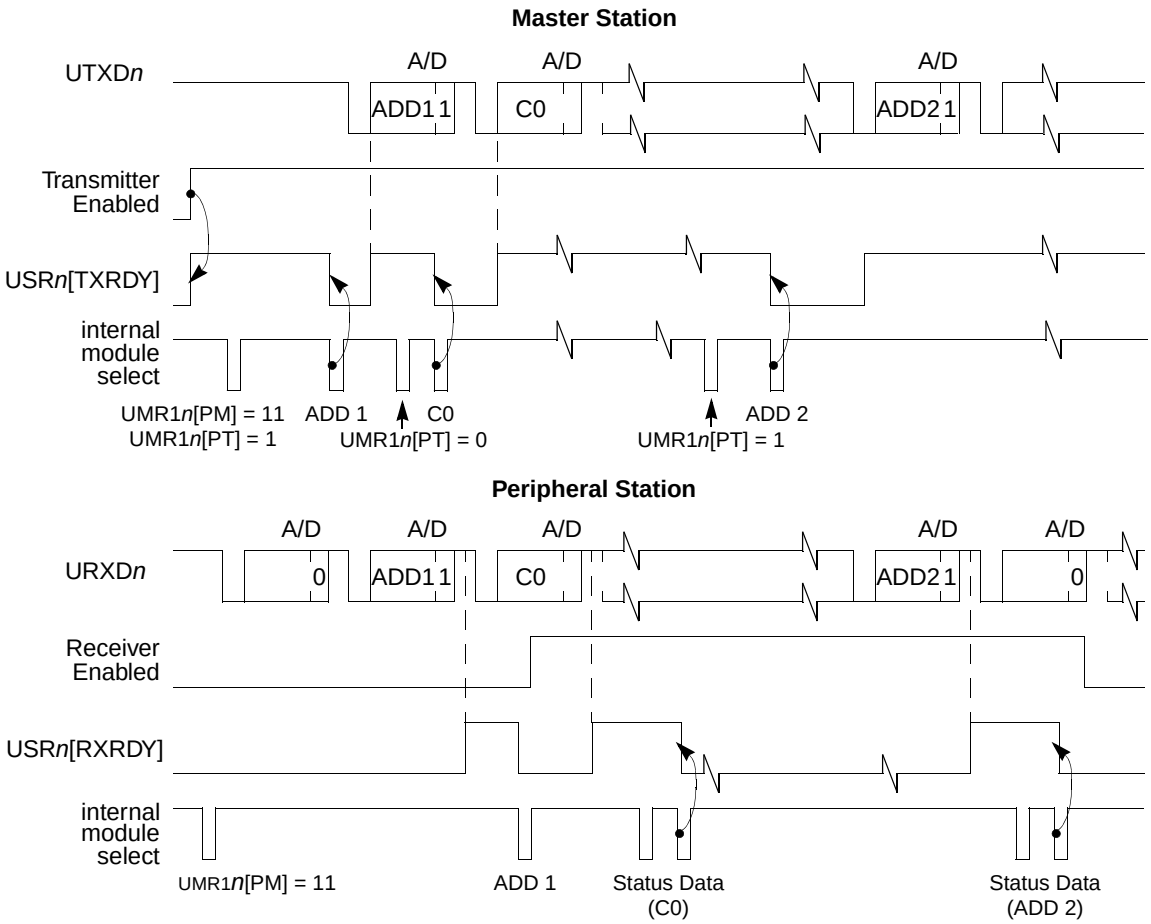

Figure 23-24. Multidrop Mode Timing Diagram

A character sent from the master station consists of a start bit, a programmed number of data bits, an address/data (A/D) bit flag, and a programmed number of stop bits. A/D equals 1 indicates an address character; A/D equals 0 indicates a data character. The polarity of A/D is selected through UMR1*n*[PT]. UMR1*n* should be programmed before enabling the transmitter and loading the corresponding data bits into the transmit buffer.

In multidrop mode, the receiver continuously monitors the received data stream, regardless of whether it is enabled or disabled. If the receiver is disabled, it sets the RXRDY bit and loads the character into the receiver holding register FIFO provided the received A/D bit is a 1 (address tag). The character is discarded if the received A/D bit is 0 (data tag). If the receiver is enabled, all received characters are transferred to the CPU through the receiver holding register during read operations.

In either case, data bits load into the data portion of the FIFO while the A/D bit loads into the status portion of the FIFO normally used for a parity error (USR*n*[PE]).

Framing error, overrun error, and break detection operate normally. The A/D bit takes the place of the parity bit; therefore, parity is neither calculated nor checked. Messages in this mode may continues containing error detection and correction information. If 8-bit characters are not required, one way to provide error detection is to use software to calculate parity and append it to the 5-, 6-, or 7-bit character.

Note that the received identifier field is always stored in the matching MB, thus the contents of the identifier field in a MB may change if the match was due to mask.

25.4.2.1 Self-Received Frames

The FlexCAN receives self-transmitted frames if there exists a matching receive MB.

25.4.3 Message Buffer Handling

To maintain data coherency and proper FlexCAN operation, the CPU must obey the rules listed in Section 25.4.1, “Transmit Process” and in Section 25.4.2, “Receive Process.” Deactivation of a message buffer (MB) is a host action that causes that message buffer to be excluded from FlexCAN transmit or receive processes. Any CPU write access to a control/status word of MB structure deactivates that MB, thus excluding it from Rx/Tx processes. Any form of CPU MB structure access within the FlexCAN (other than those specified in Section 25.4.1, “Transmit Process” and in Section 25.4.2, “Receive Process”) may cause the FlexCAN to behave in an unpredictable manner.

The match/arbitration processes are performed only during one period by the FlexCAN. Once a winner or match is determined, there is no re-evaluation whatsoever, in order to ensure that a receive frame is not lost. Two receive MBs or more that hold a matching ID to a received frame do not assure reception in the FlexCAN if the user has deactivated the matching MB after FlexCAN has scanned the second.

25.4.3.1 Serial Message Buffers (SMBs)

To allow double buffering of messages, the FlexCAN has two shadow buffers called serial message buffers. These two buffers are used by the FlexCAN for buffering both received messages and messages to be transmitted. Only one SMB is active at a time, and its function depends upon the operation of the FlexCAN at that time. At no time does the user have access to or visibility of these two buffers.

25.4.3.2 Transmit Message Buffer Deactivation

Any write access to the control/status word of a transmit message buffer during the process of selecting a message buffer for transmission immediately deactivates that message buffer, removing it from the transmission process.

If the user deactivates the transmit MB while a message is being transferred from a transmit message buffer to a SMB the message will not be transmitted.

If the user deactivates the transmit message buffer after the message is transferred to the SMB, the message will be transmitted, but no interrupt will be requested and the transmit code will not be updated.

If a message buffer containing the lowest ID is deactivated while that message is undergoing the internal arbitration process to determine which message should be sent, then that message may not be transmitted.

25.4.3.3 Receive Message Buffer Deactivation

Any write access to the control/status word of a receive message buffer during the process of selecting a message buffer for reception immediately deactivates that message buffer, removing it from the reception process.

If a receive message buffer is deactivated while a message is being transferred into it, the transfer is halted and no interrupt is requested. If this occurs, that receive message buffer may contain mixed data from two different frames.

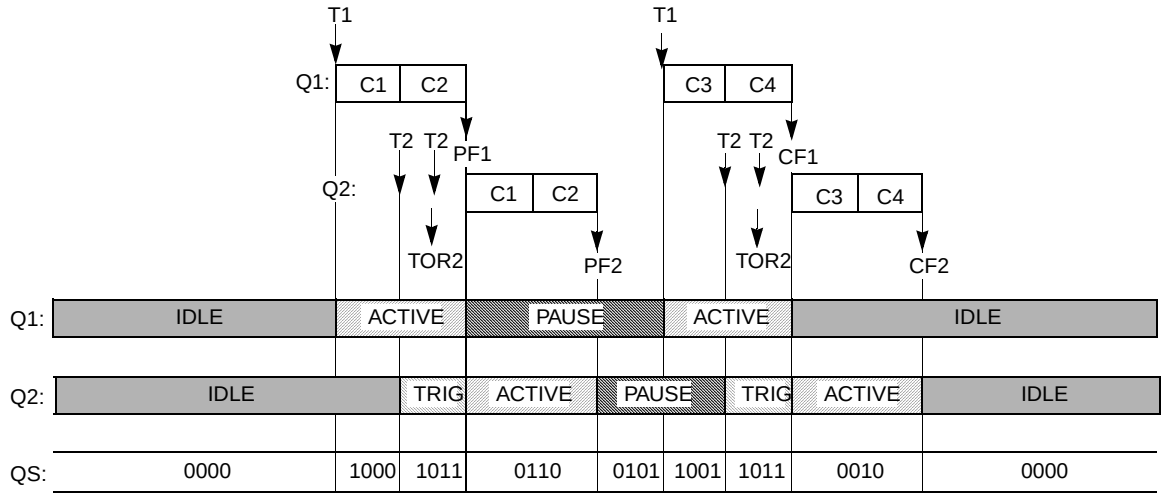

Figure 28-27. CCW Priority Situation 5

The remaining situations, S6 through S11, show the impact of a queue 1 trigger event occurring during queue 2 execution. Because queue 1 has higher priority, the conversion taking place in queue 2 is aborted so that there is no variable latency time in responding to queue 1 trigger events.

In situation 6 (Figure 28-28), the conversion initiated by the second CCW in queue 2 is aborted just before the conversion is complete, so that queue 1 execution can begin. Queue 2 is considered suspended. After queue 1 is finished, queue 2 starts over with the first CCW, when the RESUME control bit is set to 0. Situation S7 (Figure 28-29) shows that when pause operation is not used with queue 2, queue 2 suspension works the same way.

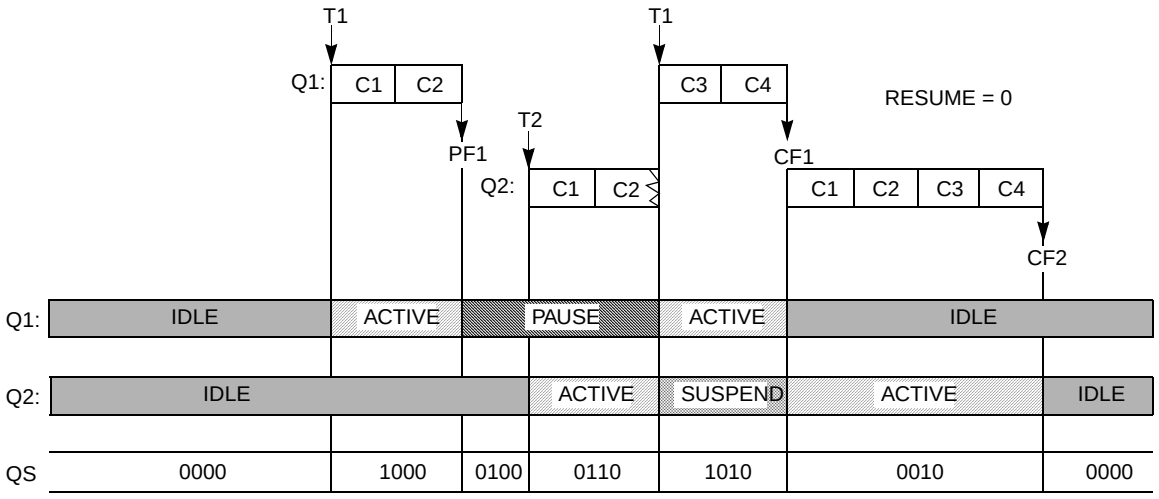

Figure 28-28. CCW Priority Situation 6

Asynchronous reset sources usually indicate a catastrophic failure. Therefore, the reset control logic does not wait for the current bus cycle to complete. Reset is asserted immediately to the system.

29.5.1.1 Power-On Reset

At power up, the reset controller asserts $\overline{\text{RSTO}}$. $\overline{\text{RSTO}}$ continues to be asserted until V_{DD} has reached a minimum acceptable level and, if PLL clock mode is selected, until the PLL achieves phase lock. Then after approximately another 512 cycles, $\overline{\text{RSTO}}$ is negated and the part begins operation.

29.5.1.2 External Reset

Asserting the external $\overline{\text{RSTI}}$ for at least four rising CLKOUT edges causes the external reset request to be recognized and latched. The bus monitor is enabled and the current bus cycle is completed. The reset controller asserts $\overline{\text{RSTO}}$ for approximately 512 cycles after $\overline{\text{RSTI}}$ is negated and the PLL has acquired lock. The part then exits reset and begins operation.

In low-power stop mode, the system clocks are stopped. Asserting the external $\overline{\text{RSTI}}$ in stop mode causes an external reset to be recognized.

29.5.1.3 Watchdog Timer Reset

A watchdog timer timeout causes timer reset request to be recognized and latched. The bus monitor is enabled and the current bus cycle is completed. If the $\overline{\text{RSTI}}$ is negated and the PLL has acquired lock, the reset controller asserts $\overline{\text{RSTO}}$ for approximately 512 cycles. Then the part exits reset and begins operation.

29.5.1.4 Loss-of-Clock Reset

This reset condition occurs in PLL clock mode when the LOCRE bit in the SYNCR is set and either the PLL reference or the PLL itself fails. The reset controller asserts $\overline{\text{RSTO}}$ for approximately 512 cycles after the PLL has acquired lock. The part then exits reset and begins operation.

29.5.1.5 Loss-of-Lock Reset

This reset condition occurs in PLL clock mode when the LOLRE bit in the SYNCR is set and the PLL loses lock. The reset controller asserts $\overline{\text{RSTO}}$ for approximately 512 cycles after the PLL has acquired lock. The part then exits reset and resumes operation.

29.5.1.6 Software Reset

A software reset occurs when the SOFTRST bit is set. If the $\overline{\text{RSTI}}$ is negated and the PLL has acquired lock, the reset controller asserts $\overline{\text{RSTO}}$ for approximately 512 cycles. Then the part exits reset and resumes operation.

29.5.1.7 LVD Reset

The LVD reset will occur when the supply input voltage, $V_{DD,}$ drops below V_{LVD} (minimum).

30.5.3.3.7 Resume Execution (GO)

The pipeline is flushed and refilled before normal instruction execution resumes. Prefetching begins at the current address in the PC and at the current privilege level. If any register (such as the PC or SR) is altered by a BDM command while the processor is halted, the updated value is used when prefetching resumes. If a GO command is issued and the CPU is not halted, the command is ignored.

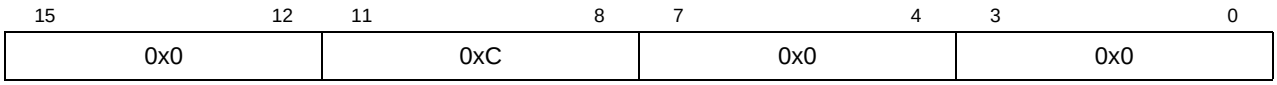

Figure 30-29. GO Command Format

Command Sequence:

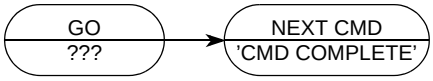

Figure 30-30. GO Command Sequence

Operand Data: None

Result Data: The command-complete response (0xFFFF) is returned during the next shift operation.

30.5.3.3.8 No Operation (NOP)

NOP performs no operation and may be used as a null command where required.

Command Formats:

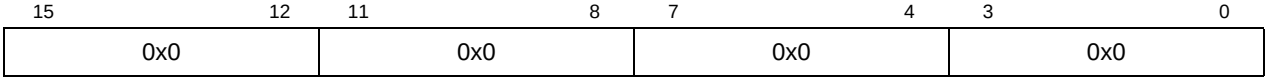

Figure 30-31. NOP Command Format

Command Sequence:

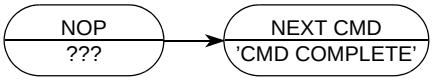

Figure 30-32. NOP Command Sequence

Operand Data: None

Result Data: The command-complete response, 0xFFFF (with S cleared), is returned during the next shift operation.

30.5.3.3.9 Read Control Register (RCREG)

Reads the selected control register and returns the 32-bit result. Accesses to the processor/memory control registers are always 32 bits wide, regardless of register width. The second and third words of the command form a 32-bit address, which the debug module uses to generate a special bus cycle to access the specified control register. The 12-bit Rc field is the same as that used by the MOVEC instruction.

Table 33-6. Typical Application Power Consumption

Application	Current
Dhrystone benchmark running from cache and on-chip SRAM (running at 64 MHz f_{sys})	181.6 mA
dBUG ROM monitor running from external flash and SDRAM (running at 64 MHz f_{sys})	155 mA

Table 33-7 lists the maximum power consumption specifications.

Table 33-7. Maximum Power Consumption Specifications

Characteristic	Symbol	Typical	Max	Unit
Operating Supply Current [1]	I_{DD}			
Master Mode				
• 66 MHz		—	200	mA
• 80 MHz (MCF528*x* only)		—	240	mA
Single Chip Mode		—	150	mA
WAIT/DOZE		—		
• 66 MHz		—	125	mA
• 80 MHz (MCF528*x* only)		—	150	mA
Clock Synthesizer Supply Current	I_{DDPLL}			
Normal Operation 8.25 MHz crystal, VCO on, Max f_{sys}		—	4	mA
STOP (OSC and PLL enabled)		—	2	mA
STOP (OSC enable, PLL disabled)		—	1	mA
STOP (OSC and PLL disabled)		—	10	µA
RAM Memory Standby Supply Current	I_{STBY}			
Normal Operation: $V_{DD} > V_{STBY} - 0.3$ V		—	10	µA
Transient Condition: $V_{STBY} - 0.3$ V > $V_{DD} > V_{SS} + 0.5$ V		—	7	mA
Standby Operation: $V_{DD} < V_{SS} + 0.5$ V		—	20	µA
Flash Memory Supply Current	I_{DDF}			
Read		16.5[3]	30	mA
Program or Erase[2]		25[4]	64	mA
Idle		1.6[4]	20	mA
STOP		0.2	10	µA
Analog Supply Current	I_{DDA}			
Normal Operation		—	5.0	mA
Low-Power Stop		—	10.0	µA

1 Current measured at maximum system clock frequency, all modules active, and default drive strength with matching load.

2 Programming and erasing all 8 blocks of the Flash.

3 Measured with f_{sys} of 64 MHz.

4 Measured with f_{sys} of 32 MHz and f_{clk} of 187.5 kHz.

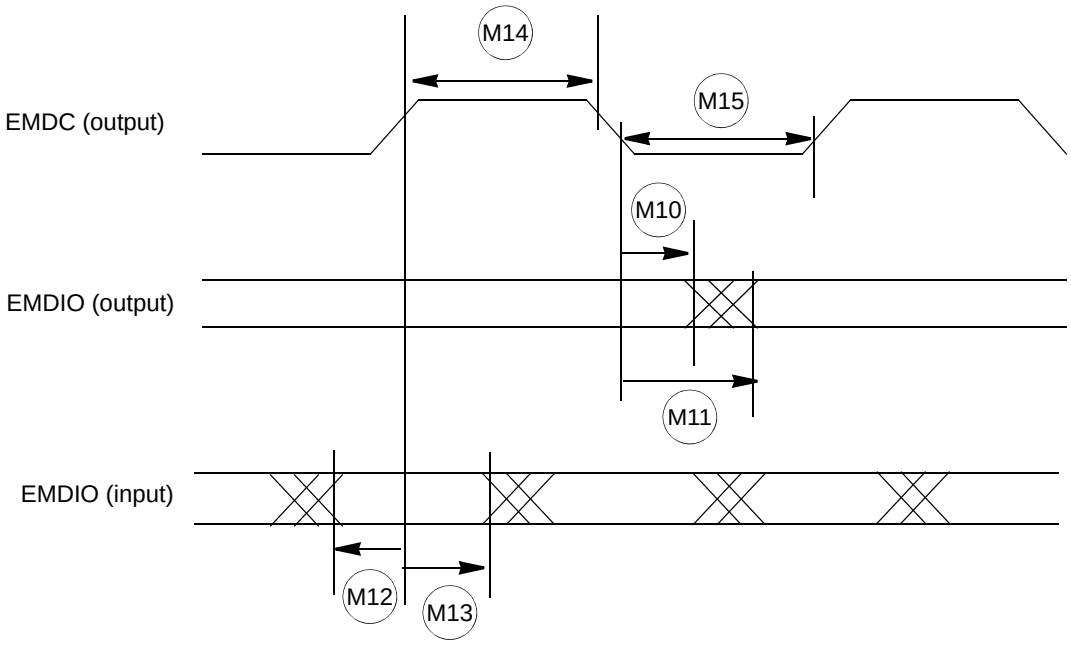

Figure 33-14. MII Serial Management Channel Timing Diagram

33.14 DMA Timer Module AC Timing Specifications

Table 33-25 lists timer module AC timings.

Table 33-25. Timer Module AC Timing Specifications

Name	Characteristic [1]	Min	Max	Unit
T1	DTIN0 / DTIN1 / DTIN2 / DTIN3 cycle time	3	—	t_{CYC}
T2	DTIN0 / DTIN1 / DTIN2 / DTIN3 pulse width	1	—	t_{CYC}

[1] All timing references to CLKOUT are given to its rising edge when bit 3 of the SDRAM control register is 0.

33.15 QSPI Electrical Specifications

Table 33-26 lists QSPI timings.

Table 33-26. QSPI Modules AC Timing Specifications

Name	Characteristic	Min	Max	Unit
QS1	QSPI_CS[3:0] to QSPI_CLK	1 × tcyc	510 × tcyc	ns
QS2	QSPI_CLK high to QSPI_DOUT valid.	—	12	ns
QS3	QSPI_CLK high to QSPI_DOUT invalid. (Output hold)	2	—	ns
QS4	QSPI_DIN to QSPI_CLK (Input setup)	10	—	ns
QS5	QSPI_DIN to QSPI_CLK (Input hold)	10	—	ns

The values in Table 33-26 correspond to Figure 33-15.

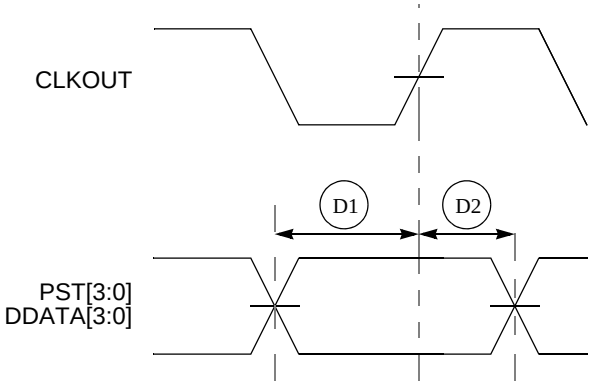

Figure 33-21. Real-Time Trace AC Timing

Figure 33-22 shows BDM serial port AC timing for the values in Table 33-28.

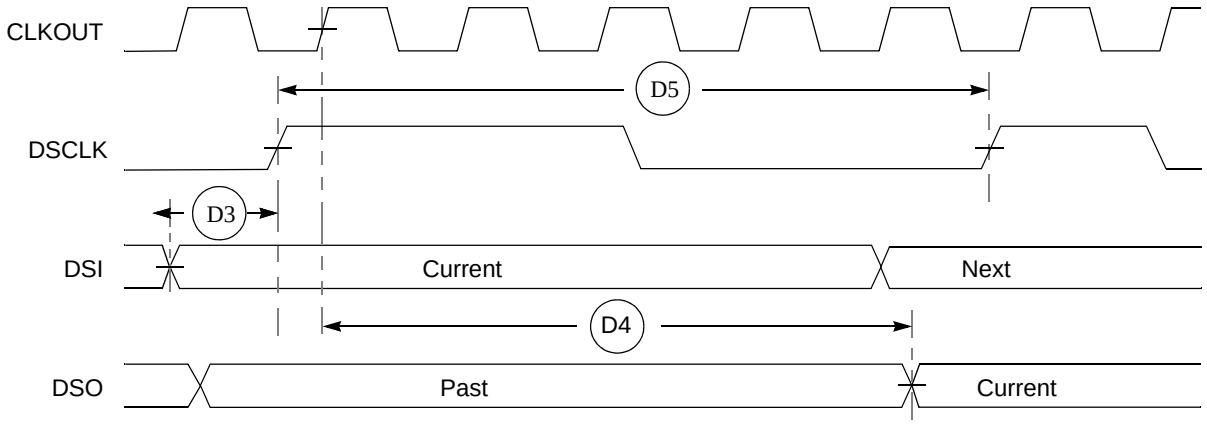

Figure 33-22. BDM Serial Port AC Timing