Application Note

AN2393/D
Rev. 0, 11/2002

Migrating from the MMC2114 to the MCF5282

MOTOROLA
intelligence everywhere™
digital dna™

Grant Whitacre
TECD Applications

This application note describes what designers and programmers should consider when migrating from the M•CORE family to the ColdFire family. It addresses general concerns about M•CORE to ColdFire migration as well as specific concerns about migrating from the MMC2114 to the MCF5282.

There are a number of advantages that make ColdFire an attractive choice for current M•CORE customers. This document highlights these advantages and provides a checklist for managing the differences between the two architectures. Although this application note addresses specific topics for MMC2114 to MCF5282 migration, M•CORE users will find it useful when migrating from any M•CORE device to any ColdFire device.

This document organizes the differences between these two families and devices into two major categories: hardware and code. Specifically, this document:

- Examines parametric differences, i.e., electrical parameters that specify the conditions under which the device must operate, and discusses performance and power requirements
- Compares mechanical characteristics, in particular, packaging and external signals
- Examines on-chip modules, contrasting shared modules and introducing modules not found on the MMC2114
- Discusses operating modes and how they are entered
- Discusses differences between the MMC2114 and MCF5282 memory maps

Because this application note is not intended to be an all-encompassing reference for either of these two devices or families, there are frequent references to more detailed information throughout each of the following sections. A list of all suggested references is given at the end of this document.

This document contains the following topics:

To locate any published errata or updates for this document, refer to the web site at http://www.motorola.com/semiconductors.

1 Comparison Overview

Many current users of the MMC2114 and MMC2107 will find several advantages in migrating to the MCF5282. These advantages fall into three categories:

- Increases in device performance and functionality, including expanded on-chip memory size for both static RAM and Flash, and external peripheral control
- Enhancement of core architectural features and ability of the core's register set, inherent execution control, and instruction set to support efficient code execution in an embedded application
- Improvements in system control capabilities such as module-level low-power control and memory/module access control

Migration from the MMC2114 to the MCF5282 includes the following increases in device performance and functionality:

- Doubling of the system bus frequency from 33 MHz to 66 MHz
- Doubling of on-chip SRAM memory from 32 Kbytes to 64 Kbytes
- Doubling of on-chip Flash memory from 256 Kbytes to 512 Kbytes
- Addition of 2 Kbytes of configurable instruction/data cache RAM for faster code execution, especially when executing from external Flash memory
- Doubling of the on-chip Programmable Interrupt timers (PITs), from 2 to 4 modules
- 50% increase in general purpose input/output (GPIO) ports from a maximum of 104 pins on the MMC2114 to a maximum of 150 pins on the MCF5282
- Two interrupt controllers rather than one to support the increase in interrupt sources from 43 to 72 sources and to provide expanded exception handling functionality
- Exact module reuse of the Queued Analog to Digital Converter (QADC), the General Purpose timers (called timer modules in the MMC2114), the PITs, Watchdog module, Clock module and Reset Controller module with low-voltage detection
- Minor module differences in reuse of the Chip Configuration module and the Edge Port module
- Full-function asynchronous communication with three Universal Asynchronous Receiver/Transmitter (UART) modules rather than limited handshaking functionality of the MMC2114's dual Serial Communication Interface (SCI) modules
- Improvement over the MMC2114's Serial Peripheral Interface (SPI) module with queuing and improved buffering support in the MCF5282's Queued Serial Peripheral Interface (QSPI) module
- Addition of several modules:
  - 10/100 Fast Ethernet (media access) controller (FEC)
  - Controller Area Network (CAN) controller
  - I^{2}C communication controller
  - 4-channel Direct Memory Access controller (DMAC)
  - 4 DMA-supported 32-bit DMA timer modules
  - Synchronous/asynchronous Dynamic Random Access Memory (SDRAM) controller

The MCF5282 core architectural features include the following:

- The MCF5282's Multiply-Accumulate controller (MAC) unit provides a common set of simple DSP operations, and speeds the execution of the integer multiply instructions for both signed and unsigned operands in the ColdFire core.

- Multiplies of 16x16 and 32x32 with 32-bit accumulates are supported. The MAC unit is tightly coupled to the Operand Execution Pipeline (OEP) and features a three-stage execution pipeline.
- The OEP can issue a 16 x 16 multiply with a 32-bit accumulation and fetch a 32-bit operand in the same cycle.
- A hardware divide module is also coupled to the core's OEP which allows the processor to support signed divides, unsigned divides, and remainder instructions.

The following system control options and capabilities are offered in the MCF5282:

- A base address register which provides relocation of internal resources
- An interrupt controller that supports seven programmable interrupt levels for internal peripheral interrupts and seven external pin interrupts, and has eight programmable and one fixed priority level within each interrupt level
- A software watchdog timer to prevent erratic operation caused by runaway code execution
- Seven types of resets, including low power supply voltage detection, and a status register to indicate the cause of last reset
- Chip selects with programmable wait states, port sizes, and transfer burstability on reads or writes
- Doze, wait, and stop power-saving operation modes, with exit control and control of mode/clock behavior for several on-chip modules
- Individual module access control, including read/write access in either user or supervisor mode, for 17 of the MCF5282's on-chip modules
- Grouped module and address space access control, including the above access differentiation plus code execution access for 13 of the MCF5282's on-chip modules
- Bus master arbitration and prioritization for 3 on-chip masters (core, FEC, and DMA controller)

Table 1 provides a brief look at the design considerations and benefits of migration from the MMC2114 to the MCF5282. It also provides a page and section reference to this document's detailed description of the specific migrating issue.

Table 1. MMC2114 to MCF5282 Migration Issue Summary

Device Feature	Device Implementation		Impact		Migration Criticality	Section/Page Number
	MMC2114	MCF5282	Hardware	Code		
System Frequency	33 MHz	66 MHz	√	√	Low	Section 2.1.2, "System Clocks" on page 6
Power Requirement	200 mW at 3.6V / 33 MHz	750 mW at 3.6V / 66 MHz	√		Medium	Section 2.1.3, "Power Requirements" on page 6
Packaging	144 LQFP, 100 LQFP, 196 MAPBGA	256 MAPBGA	√		Medium	Section 2.2.1, "Packaging" on page 6
SRAM Size	32 Kbytes	64 Kbytes		√	Low	Section 2.3.1.1, "SRAM" on page 9
Flash Size	256 Kbytes	512 Kbytes		√	Low	Section 2.3.1.2, "Flash" on page 9

Table 1. MMC2114 to MCF5282 Migration Issue Summary (continued)

Device Feature	Device Implementation		Impact		Migration Criticality	Section/Page Number
	MMC2114	**MCF5282**	**Hardware**	**Code**		
SRAM Location	Fixed at 0x80_0000	Set by RAMBAR		√	Low	Section 2.3.1.1, “SRAM” on page 9
Flash Location	Fixed at 0x00	Set by FLASHBAR		√	Low	Section 2.3.1.2, “Flash” on page 9
Differences in Flash Configuration Field Size and Content	0x200–0x22b	0x400–0x417			Low	Section 2.3.1.2, “Flash” on page 9
Low Power Mode Entry	By Wait, Doze, and Stop Instructions	By Low Power Control Register [LPMD] Setting		√	Low	Section 2.3.1.5, “Power Management” on page 10
Low Power Mode Exit by Interrupt	By Any Enabled Interrupt	By Low Power Interrupt Control Register [XLPM_IPL] Setting		√	Low	Section 2.3.1.5, “Power Management” on page 10
Number of External Interrupt Pins	8	7		√	Low	Section 2.3.1.7, “Edge Port” on page 100
Number of PITs	2	4			Low	Section 2.3.1.10, “Programmable Interrupt Timers (PITs)” on page 11
Difference in Interrupt Control	Various	Various		√	Medium	Section 2.3.2.2, “Interrupt Controller” on page 12
Differences in Operation and Number of SCIs and UARTs	2 SCIs	4 UARTs		√	Medium	Section 2.3.2.3, “SCI vs UART” on page 13
Differences in Chip Configuration	Various	Various	√	√	Medium	Section 2.4.1, “Chip Configuration” on page 16
Differences in Memory Maps	Various	Various		√	Medium	Section 2.5, “MMC2114 and MCF5282 Memory Maps” on page 19
FEC, I2C, FlexCAN, SDRAMC, DMA Timers	No	Yes	√	√	Medium	Section 2.3.3, “New/Additional Modules” on page 15

2 Device Differences

This section addresses differences between the MMC2114 and the MCF5282 and highlights what the user needs to consider when making this migration. The categories discussed in this section include electrical characteristics, mechanical characteristics, on-chip modules, and memory maps.

2.1 Electrical Characteristics

The following sections discuss differences in electrical characteristics between the two devices.

2.1.1 Supply Voltages and References

All voltage supplies and reference voltage ranges for the two devices are the same. They are shown here for reference.

2.1.1.1 Core and Pads Supply ($V_{DD)}$

MMC2114/MCF5282: 2.7 to 3.6V

2.1.1.2 Flash Module Supply (V_{DDF})

MMC2114/MCF5282: 2.7 to 3.6V

2.1.1.3 PLL Supply (V_{DDPLL})

MMC2114/MCF5282: 2.7 to 3.6V

2.1.1.4 Standby Mode Supply (V_{STBY}) in Standby Mode

MMC2114/MCF5282: 2.7 to 3.6V

2.1.1.5 Analog Supply (V_{DDA})

MMC2114/MCF5282: 4.5 to 5.5V

2.1.1.6 ESD Supply (V_{DDH})

MMC2114/MCF5282: 4.5 to 5.5V

2.1.1.7 Programming Supply (V_{PP})

MMC2114/MCF5282: Not connected in normal user mode

2.1.1.8 Negative Power References (V_{SS}, V_{SSF}, V_{SSPLL}, V_{SSA}, V_{SSH})

MMC2114/MCF5282: 0V

2.1.1.9 QADC High Voltage Reference (V_{RH})

MMC2114/MCF5282: V_{DDA} - 0.1V to V_{DDA}

2.1.1.10 QADC Low Voltage Reference (V_{RL})

MMC2114/MCF5282: V_{SSA} to V_{SSA}+ 0.1V

2.1.2 System Clocks

The following sections discuss differences in clocking characteristics between the two devices.

2.1.2.1 Internal System Frequency in External Clock Mode (f_{SYS})

MMC2114: 0 to 33 MHz

MCF5282: 0 to 66 MHz

2.1.2.2 Internal System Frequency in Internal Clock Mode (f_{SYS})

MMC2114: 3/64 to 33 MHz

MCF5282: f_{REF}/32 to 66 MHz

2.1.2.3 Crystal Reference for PLL ($f_{REF\_CRYSTAL}$)

MMC2114/MCF5282: 2 to 10 MHz

2.1.2.4 External Reference for PLL (f_{REF_EXT})

MMC2114/MCF5282: 2 to 10 MHz

2.1.2.5 External Clock for 1:1 PLL ($f_{REF\_1:1}$)

MMC2114: 10 to 33 MHz

MCF5282: 33 to 66 MHz

2.1.3 Power Requirements

Power requirements for the two devices are as follows:

MMC2114: 200 mW at 3.6V/33MHz

MCF5282: 750 mW at 3.6V/66MHz (Maximum)

2.2 Mechanical Characteristics

2.2.1 Packaging

MMC2114: 196-ball plastic mold array process ball grid array (MAPBGA), 144-pin low-profile quad flat pack (LQFP), 100-pin LQFP (single-chip operation only)

MCF5282: 256-ball MAPBGA

2.2.2 External Signal Comparison

Because of the devices’ different packages, it is not useful to compare their pinouts. It is useful to discuss the commonality of signals brought out of each device. Table 2 is a summary of each device’s externally-accessible signals.

Table 2. MMC2114/MCF5282 External Signal Comparison

Module	Signals	MMC2114	MCF5282	Comment
Reset/ Clock	$\overline{\text{RSTI}}$, $\overline{\text{RSTO}}$, EXTAL, XTAL, CLKOUT	√	√	
Chip Configuration	$\overline{\text{RCON}}$, D[26,19:16]	√	√	
	PLLEN, D[23:21]	√		
	CLKMOD1, CLKMOD0, D[25:24]		√	
External Interface Module/ Chip Select	A[22:0], D[31:0], $\overline{\text{BS}}$[3:0], $\overline{\text{OE}}$, $\overline{\text{TA}}$, $\overline{\text{TEA}}$, R/$\overline{\text{W}}$, TSIZ1, TSIZ0, $\overline{\text{CS}}$[3:0]	√	√	$\overline{\text{BS}}$x (MCF5282) = $\overline{\text{EB}}$x (MMC2114); $\overline{\text{CS}}$[6:4] pins can be configured either as chip selects or A[23:21].
	$\overline{\text{SHS}}$, TC[2:0], CSE[1:0]	√		
	A23, $\overline{\text{TS}}$, $\overline{\text{TIP}}$, $\overline{\text{CS}}$[6:4]		√	
EPORT	$\overline{\text{IRQ}}$[7:1]	√	√	
	$\overline{\text{IRQ0}}$	√		
QADC	AN[56, 55, 53, 52, 3:0], VRH, VRL	√	√	Analog inputs are multiplexed with ETRIG[2:1], MA[1:0] and AN[Z:W].
QSPI (SPI)	QSPI_DOUT, QSPI_DIN, QSPI_CLK	√	√	MOSI (MMC2114) => QSPI_DOUT (MCF5282); MISO (MMC2114) => QSPI_DIN (MCF5282); SCK (MMC2114) => QSPI_CLK (MCF5282)
	QSPI_CS[3:0]		√	
	$\overline{\text{SS}}$	√		
GP Timers	GPTA[3:0], GPTB[3:0]	√	√	ICOCnx (MMC2114) => GPTnx; SYNCA is muxed on both the TSIZ1 and TS pins and can be selectively enabled on either of these pins; SYNCB is muxed on both the TSIZ0 and TIP pins and can be selectively enabled on either of these pins.
	SYNCA, SYNCB		√	
SCIs	TXD1, RXD1, TXD2, RXD2	√		

Table 2. MMC2114/MCF5282 External Signal Comparison

Module	Signals	MMC2114	MCF5282	Comment
UARTs	URXD[2:0], UTXD[2:0], URTS[1:0], UCTS[1:0]		√	URXD2 is muxed on the EMDIO, CANRX, and SDA pins and can be selectively enabled on any of these pins; UTXD2 is muxed on the EMDC, CANTX, and SCL pins and can be selectively enabled on any of these pins; URTS1 and URTS0 are each muxed on the DTIN3, DTOUT3, DTIN1, and DTOUT1 pins and each can be selectively enabled on these pins; UCTS1 and UCTS0 are each muxed on the DTIN2, DTOUT2, DTIN0, and DTOUT0 pins and each can be selectively enabled on these pins.
OnCE/ JTAG Port	TRST, TCLK, TMS, TDI, TDO, DE	√		
BDM/ JTAG Port	JTAG_EN, DSCLK/TRST, DSI/TDI, DSO/TDO, BKPT/TMS, DDATA[3:0], PST[3:0], TCLK		√	BDM and JTAG functions share many of the same pins; the device is either in BDM or JTAG mode.
DMA Timers	DTIN[3:0]; DTOUT[3:0]		√	
SDRAM Controller	SRAS, SCAS, SDWE, SDRAM_CS[1:0], SCKE		√	
FEC	EMDIO, EMDC, ETXCLK, ETXEN, ETXD[3:0], ECOL, ERXCLK, ERXDV, ERXER, ERXD[3:0], ECRS, ETXER		√	
FlexCAN	CANRX, CANTX		√	
I2C	SDA, SCL		√	
Power Supplies	V_{DD}/V_{SS}, V_{DDF}/V_{SSF}, V_{DDA}/V_{SSA}, V_{DDH}, V_{STBY}	√	√	
	V_{DDPLL}/V_{SSPLL}		√	

2.3 On-chip Modules

This section examines the MCF5282's on-chip modules and highlights differences between these modules and those found on the MMC2114. Modules common to both devices are discussed first, followed by a comparison of modules on the two devices that, though similar in functionality, are significantly different. That section will be followed by a discussion of the MCF5282 modules that are not found on the MMC2114. Note that for all modules, registers are mapped at different places on the two devices which will require the update of existing user code.

2.3.1 Common Modules

The MCF5282 and MMC2114 share several identical or very similar modules. Each of these modules is discussed in the following sections.

2.3.1.1 SRAM

The MMC2114 has 32 Kbytes of SRAM while the MCF5282 has 64 Kbytes. On the MMC2114, the SRAM has a fixed address starting at 0x80_0000. On the MCF5282, the SRAM location is dictated by the value programmed into the RAM base address register, RAMBAR. RAMBAR must be programmed before accesses to SRAM can be made.

2.3.1.2 Flash

The MMC2114 has 256 Kbytes of internal Flash while the MCF5282 has 512 Kbytes. On the MMC2114, the Flash has a fixed address starting at 0x00. On the MCF5282, the Flash location is dictated by the value programmed into the Flash base address register, FLASHBAR. FLASHBAR is initialized automatically to 0x00 if the internal Flash is the boot device, otherwise FLASHBAR must be programmed before accesses to the Flash module can be made.

Each version of Flash on the two devices has a configuration field. The values programmed into this configuration field are automatically loaded into protection and security registers at reset. The configuration field in the MMC2114's Flash is located at 0x0200 and occupies 44 bytes, while the configuration field in the MCF5282 is located at 0x0400 and occupies 24 bytes. The configuration field for each device holds five types of configuration information:

- Back door comparison key
- Flash program/data space restriction
- Flash program/erase sector protection
- Flash supervisor/user space restriction
- Flash security word

The difference in these two configuration fields is in the construction of the three types of protection values. On the MMC2114, there are four sets of these values, but only two of the sets actually apply to the 256-Kbyte array. Each 16-bit value applies to a separate 128-Kbyte bank of the 256-Kbyte Flash array, where each bit protects an 8-Kbyte Flash logical sector.

On the MCF5282, there is a single set of these protection values (each value 32 bits in length) and each bit specifies the protection for a 16-Kbyte Flash logical sector in the 512-Kbyte array.

For both devices, each protection value is copied into equivalently-sized registers upon reset. These registers are readable, but they are writable to change protection only when the lock bit in the Flash's configuration register is cleared.

The method of programming and erasing the Flash on the MCF5282 is the same as for the MMC2114.

2.3.1.3 Clock Module

The clock module is identical for each of the two devices. The only difference for clock selection is in the signals used during chip configuration to select the clock mode. See Section 2.4.1, “Chip Configuration” for details.

2.3.1.4 Reset Controller Module

The Reset Controller modules in the two devices are very similar. One difference is in the reset control register (RCR). All bits in the RCR are the same for the two devices except for the low voltage detect stop enable (LVDSE) bit. This bit is in the MMC2114's RCR, but it is not present in the MCF5282's RCR. In the

MCF5282, the LVDSE bit can be found in the low power control register (LPCR). For the MCF5282, LPCR[LVDSE] actually stands for “low voltage detect standby enable”; it controls whether the Power Management module (PMM) enters standby mode (LVD disabled) or pseudo-standby mode (LVD enabled) when the PMM receives a power-down request. This bit has no effect if the LPCR[LVDE] bit is a logic 0.

2.3.1.5 Power Management

One difference between the two devices is in their ability to enter low-power operation. The reason for this difference is the fact that the M•CORE architecture supports low-power mode entry by implementing specific instructions, DOZE, WAIT, and STOP, to enter the doze, wait, and stop modes. The ColdFire instruction set, on the other hand, has only a STOP instruction. Therefore, the MCF5282’s Power Management Module provides the capability to enter all three low-power modes by allowing the application to specify how the STOP instruction is interpreted. The LPCR[LPMD] field, which can be set by the user, specifies the low-power mode that the device is to enter when a STOP instruction is executed. This bit can be changed dynamically depending on the needs of the application.

Another way that the devices’ low-power operation differs is that the MCF5282 allows the user to specify the minimum interrupt level required to exit from low-power operation. This is accomplished by setting the XLPM_IPL field in the low power interrupt control register (LPICR). This field has 8 possible settings, and can specify any minimum level interrupt from all levels (0–7) to just a level-7 interrupt for exit from low power. Level-7 interrupts are non-maskable and can always bring the device out of low-power mode. The value programmed in this field must be higher than the value programmed in the interrupt priority mask field of the status register in order for the specified level to be recognized and to bring the device out of low-power mode. On the MMC2114, the only interrupt that can bring the device out of stop mode is an external EPORT interrupt.

For both devices, there is user control over whether the external CLKOUT signal, the PLL, and the internal oscillator module are disabled in stop mode.

See the MMC2114 and MCF5282 user’s manuals for specific module operation in each of the low-power modes.

2.3.1.6 I/O Ports

Operation and characteristics of the general purpose digital input/output (I/O) ports in the two modules are very similar. They differ only in the amount of digital I/O present on each device. The MMC2114 has eight full 8-bit ports and one 5-bit port. By contrast, the MCF5282 has twelve full 8-bit ports, one 7-bit port, two 6-bit ports and three 4-bit ports. On both devices, most I/O ports are multiplexed with other selectable primary functions.

For both devices, each port has an output data register, a data direction register, a port pin data /set data register and a clear output data register. Many ports, which also have primary functions associated with one or more of its pins, also have or share a port pin assignment register. Aside from the differences in the primary functions that these pins might have, digital I/O ports are handled exactly the same on the two devices.

2.3.1.7 Edge Port

The main difference in the Edge Port for these two devices is that the EPORT on the MMC2114 has eight pins and MCF5282’s EPORT has seven pins. The MMC2114’s EPORT pins are named INT0–INT7, while the pins on the MCF5282 are named $\overline{\text{IRQ1}}$–$\overline{\text{IRQ7}}$. The setup and control of these pins as well as the interrupts that can be generated are identical for both devices.

Though the MMC2114 has eight designated EPORT pins and the MCF5282 has only seven, the MCF5282 can use a timer pin with its channel configured for input capture as an external interrupt pin. Either type of timer on the MCF5282, the General Purpose timer or the DMA timer, can have a channel configured for input capture for the purpose of using the pin as an external interrupt pin. For example, the DTIN*n* pin of a DTIM*n* DMA Timer can be configured to generate an interrupt upon the sensing of either or both edges of an external signal. Similarly, the GPT*x* pin of one of the General Purpose Timers (GPTA or GPTB), where *x* designates channels 0-3, can be configured as input capture. Here again, the pin can be configured to generate an interrupt upon the sensing of either or both edges of an external signal. Be aware that timer-generated interrupts are maskable and that their priority and level are programmable, not unmaskable and fixed as is the case with EPORT interrupts. Also note that while the EPORT's $\overline{\text{IRQ}}$ pins can be configured for level-sensed interrupts, the timers' input capture pins cannot. This means that the $\overline{\text{IRQ}}$ pins can detect a level-sense interrupt in stop mode when there are no clocks running in the device, but the timers will not be able to detect an edge-sensitive interrupt under the same conditions.

2.3.1.8 QADC

The QADC module is exactly the same for both the MMC2114 and the MCF5282.

2.3.1.9 General Purpose Timers

The timer modules, TIM1 and TIM2, on the MMC2114 and the General Purpose timers, GPTA and GPTB, on the MCF5282 are identical in functionality. The timer names have been changed so that the General Purpose timers of the MCF5282 can be distinguished from the DMA timers. Register names have also been renamed to coincide with the name of the timer. The MMC2114's timer pin names, ICOC*xy*, where *x* specifies the timer number (1-2) and *y* specifies the channel number (0-3), are also different from the MCF5282, which has timer pins named GPT*xy*, where *x* specifies the timer designation (A or B) and *y* specifies the channel number (0-3).

2.3.1.10 Programmable Interrupt Timers (PITs)

The PITs are identical for both devices. The only difference is that a bit name in the PIT status and control register (PSCR*x*) has been changed from PDBG to HALTED to be consistent with ColdFire terminology. The control of this bit, and the subsequent operation of the module, are the same for the two devices despite the bit name difference.

Note that there are four PITs on the MCF5282 while there are only two on the MMC2114. There should be no migration issues with the PITs.

2.3.1.11 Watchdog

The Watchdog timers on each of the two modules are identical. Like the PITs, the only difference is that a bit name in the Watchdog status and control register (WSCR) has been changed from DBG to HALTED to be consistent with ColdFire terminology. The control of this bit, and the subsequent operation of the module, are the same for the two devices despite the bit name difference.

2.3.2 Different Modules

Several of the modules on the MMC2114 and MCF5282 are very similar in functionality, but have some significant differences. Each will be discussed in the following sections.

2.3.2.1 M•CORE vs ColdFire Version 2 Core

See the M•CORE and ColdFire user’s manuals for features of each device’s core.

2.3.2.2 Interrupt Controller

There are a few fundamental differences between the two devices’ interrupt controllers of which the user should be aware when migrating from the MMC2114 to the MCF5282. These differences arise primarily because of the differences in the ColdFire and M•CORE architectures and how each core supports exception handling. This document does not go into detail about the procedure that each core follows when an exception state exists or when an interrupt occurs. That information is readily available in the reference manual for each core. What will be discussed here are the programming considerations to use and control each device’s interrupt controller.

Interrupt control and the vector table for the MMC2114 are based on interrupt priority. The MMC2114’s core supports 32 levels of priority for both normal and fast interrupts. Therefore, many of the interrupt controller’s registers, as well as the vector table, are structured with this in mind. Interrupt sources, of which there are 40 on the MMC2114, can each be mapped to one of the 32 interrupt priorities, either as a normal or fast interrupt. They are not explicitly fixed to a particular interrupt priority or a particular location in the vector table. An interrupt source is associated with an interrupt priority by programming its priority level select register (PLSR) with the desired priority level for the interrupt. Additionally, an interrupt is enabled on a priority basis by setting the corresponding bit in either the normal interrupt enable register (NIER) or the fast interrupt enable register (FIER). When an interrupt is pending, the priority level’s corresponding bit is set in either the NIER or FIER.

The interrupt controller for the MCF5282 uses a different model, one that is based on the fixed mapping of interrupt sources rather than the interrupt priority selected for each source. Before describing the controller, it should noted that there are two interrupt controllers on the MCF5282, each one capable of handling 63 interrupt sources. In actuality, there are 80 sources of interrupts on the MCF5282. Everything discussed in this section pertains to both of the MCF5282’s interrupt controllers.

Like the MMC2114’s PLSRs, the MCF5282 has a set of interrupt control registers (ICR1–ICR63) for each controller that assign an interrupt priority to each of the interrupt sources. The difference here is that the ColdFire architecture supports a priority scheme in which there is both a priority and a level that define an interrupt’s overall priority. There are 9 priorities associated with each of the 7 levels of interrupts for a total of 63 unique interrupt priority levels. (It is important that each interrupt source be assigned a unique priority level to guarantee proper operation of the interrupt controller.) Each of the eight (of the nine) priorities is assignable to each of the interrupt sources. The ninth priority within each interrupt level is a fixed priority interrupt and is automatically assigned to the Edge Port’s $\overline{\text{IRQ}}$ interrupts. The priority and level for these interrupts are fixed such that $\overline{\text{IRQ1}}$ is the fixed priority interrupt within level 1, $\overline{\text{IRQ2}}$ is the fixed priority interrupt within level 2, and so on through $\overline{\text{IRQ7}}$ for level 7. These fixed-priority interrupts have a priority at the midpoint of the 9 priorities within each level such that priority 0 (lowest priority) through priority 3 are lower priority than the fixed priority ($\overline{\text{IRQ}x}$) for a particular level, while priority 4 through priority 7 (highest priority) have a higher priority than the fixed priority for a particular level. Also, level 7 is the highest level and level 0 is the lowest. Therefore, the interrupt source assigned to priority 7 within interrupt level 7 has the highest priority for this interrupt controller. To take this one step further, an interrupt controller 1 interrupt at or above the level of an interrupt controller 2 interrupt has priority over that controller 2 interrupt.

Fixed priority interrupts are designated interrupt sources 1 through 7 for each interrupt controller. IRQ1–IRQ7 are assigned to these slots for interrupt controller 1. Interrupt controller 2 does not have any fixed priority interrupts.

Once the MCF5282's ICRs are programmed with the desired priority and level for each interrupt source, then that interrupt source's interrupts are enabled by clearing the corresponding bit in the interrupt mask registers (IMRH and IMRL). There is a bit in these two 32-bit registers for each possible interrupt source associated with the interrupt controller, plus another bit that masks all interrupts regardless of the individual mask bit settings.

Another method of masking interrupts in the MCF5282 is by way of the interrupt priority mask (I) field in the core's status register. This field allows interrupt masking on a level basis, such that all levels at or below the level programmed into this 3-bit field are masked. Level-7 interrupts are unmaskable regardless of the setting in this field. In the MMC2114, this masking is performed on a priority basis of the 32 levels of interrupt priority, and is set in the ICR. All normal interrupts with a priority at or lower than what is programmed in the 5-bit ICR[MASK] field are masked. If the mask fast interrupts bit, ICR[MFI], is set then all normal interrupts are masked and all fast interrupts at or below what is programmed in the MASK field are masked.

Similar to the MMC2114's normal interrupt pending register (NIPR) and fast interrupt pending register (FIPR), each interrupt controller in the MCF5282 has two 32-bit registers called the interrupt pending register, high and low (IPRH and IPRL). Each bit in these registers specifies the pending state for the corresponding interrupt source.

These are the basic control differences between the interrupt controllers in these two devices. This section is not the extent of the interrupt servicing and exception handling considerations that need to be examined. It does highlight the differences in interrupt control methodology between the two devices.

2.3.2.3 SCI vs UART

Moving from the two MMC2114's SCIs to the three MCF5282 UARTs is a move toward more functionality, so there are very few migration issues as a result of differences between these two asynchronous communication modules. The biggest advantage in moving from the SCI to a UART is the ability to incorporate flow control in the communication link, thereby preventing receiver overruns. This can be accomplished with the $\overline{\text{RTS}}$ and $\overline{\text{CTS}}$ handshaking signals of the MCF5282's UART0 and UART1. The third UART, UART2, is intended for fairly low-speed and local asynchronous communication, and does not have flow control signals.

Table 3 contrasts the features of the MMC2114's SCIs and the MCF5282's UARTs.

Table 3. SCI and UART Feature Differences

Feature	SCI	UART	
Clock Source	f_{SYS}/16	f_{SYS}/32, DTIN	DMA Timer input can be used as external UART clock reference on the MCF5282
Baud Rate Divider Range	13-bit Prescaler	16-bit Prescaler	Baud rate = Clock source divided by any value in specified range
Hardware Flow Control		√	For the MCF5282’s UART0 and UART1 only
Programmable Number of Data Bits	8,9	5,6,7,8	
Programmable Number of Stop Bits	√	√	
Programmable Parity Enable	√	√	
Programmable Parity Type	√	√	Even or odd parity
Force Polarity of Parity Bit		√	
Send Break	√	√	MCF5282’s break signal may be variable length
Indication of Break Received		√	
Framing, Parity, and Overrun Error Indications	√	√	
Transmitter Empty, Transmission Complete, Receiver Ready/Full Indications	√	√	
Idle Line, Noise Indication	√		
Receive FIFO	1 shift register, 1 shared data register	1 shift register, 3 dedicated Rx data registers	
Transmit FIFO	1 shift register, 1 shared data register	1 shift register, 1 dedicated Tx data register	
Multidrop Mode Support		√	Parity bit setting to indicate data or address character
Automatic Echo Mode		√	
Local Loop-back Mode		√	
Remote Loop-back Mode		√	

As indicated in the table, the three UARTs on the MCF5282 can be externally clocked via the corresponding DMA timer’s DTIN inputs. This allows UART baud rate generation at frequencies totally independent of the internal clock frequency.

One final feature of the MCF5282 UARTs is their ability to trigger a DMA transfer from any programmed source and destination addresses upon a UART’s Receive FIFO becoming not empty or becoming full.

2.3.3 New/Additional Modules

There are six modules on the MCF5282 that have no functional equivalent on the MMC2114. They are the I^{2}C, FlexCAN, Fast Ethernet controller (FEC), the SDRAM controller, the Direct Memory Access controller (DMAC) and the DMA timers. The following sections give a high-level description of these modules. The only migration issue with these modules is whether or not they could be of use in an existing design when porting to the MCF5282. If an existing MMC2114 design has similar functionality implemented off-chip, then it may be beneficial to eliminate the external components and bring the functionality on chip.

2.3.3.1 I^{2}C

The I^{2}C bus is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange, minimizing the interconnection between devices. This bus is suitable for applications requiring occasional communications over a short distance between many devices. It is used primarily as an inter-chip bus interface for EEPROMs, LCD controllers, A/D converters, and keypads, and is fully compatible with the industry-standard I^{2}C bus. Both master and slave modes support multiple masters. Automatic interrupt generation is supported with programmable interrupt level and priority.

2.3.3.2 FlexCAN

The FlexCAN module is a communication controller that implements the CAN 2.0B protocol. The CAN protocol can be used as an industrial control serial data bus, meeting the specific requirements of real-time processing, reliable operation in harsh EMI environments, cost-effectiveness, and required bandwidth. The FlexCAN is based on, and includes, all existing features of the Motorola TouCAN module. It is a full implementation of the CAN protocol specification version 2.0B. The communication data structure supports both standard data and remote frames (up to 109 bits long) and extended data and remote frames (up to 127 bits long). Each message’s data block size is programmable at 0–8 bytes in length, and the programmable bit rate is settable up to 1 Mbit/sec.

A total of 16 flexible message buffers (MBs) of 0–8 byte data length each, are configurable for received or transmitted messages, all supporting standard and extended messages. Unused MB space can be used as general purpose RAM space.

2.3.3.3 Fast Ethernet Controller (FEC)

The MCF5282’s integrated Fast Ethernet controller (FEC) performs the full set of IEEE 802.3/Ethernet CSMA/CD media access control and channel-interface functions. The FEC supports connection and functionality for either 10 or 100 Mbps 802.3 media independent interface (MII), with either half or full duplex capability. It requires an external transceiver (PHY) to complete the interface to the media. There are on-chip transmit and receive FIFOs, a built-in dedicated DMA controller, and memory-based flexible descriptor rings.

2.3.3.4 SDRAM Controller

The SDRAM controller provides all required signals for glueless interfacing to a variety of JEDEC-compliant SDRAM devices. SRAS/SCAS address multiplexing is software-configurable for different page sizes. To maintain refresh capability without conflicting with concurrent accesses on the address and data buses, $\overline{\text{SRAS}}$, $\overline{\text{SCAS}}$, $\overline{\text{DRAMW}}$, $\overline{\text{SDRAM\_CS}}$[1:0] and SCKE are dedicated SDRAM signals.

The SDRAM controller module provides glueless integration of the SDRAM with the MCF5282. The key features of the DRAM controller include the following:

- Support for two independent blocks of SDRAM
- Interface to standard SDRAM components
- Programmable SRAS, SCAS, and refresh timing
- Support for 8-, 16-, and 32-bit wide SDRAM blocks

2.3.3.5 Direct Memory Access Controller (DMAC)

The Direct Memory Access controller (DMAC) module provides an efficient way to move blocks of data with minimal processor interaction. The DMAC module provides four fully programmable channels (DMA0–DMA3) that allow byte, word, longword, or 16-byte burst line transfers. These transfers are triggered by software explicitly setting a DCR*n*[START] bit, by receiving data on a UART channel, or by the occurrence of a capture event or an output reference event in a DMA timer (DTIM0–DTIM3). The DMA controller supports dual address mode to on-chip devices. The transfers are dual-address and support 8-, 16- and 32-bit data transfer sizes as well as 16-byte (4 x 32-bit) burst transfers. The source and destination address pointers can be incremented, or they can remain constant upon each transfer; there is also one 24-bit byte-transfer counter per DMA channel. There is auto-alignment transfer support for efficient block movement, and bursting and cycle stealing is also supported. A crossbar switch in the DMA controller allows software-programmable connections from the 7 DMA requesters (the three UARTs and the four DMA timers) to the four DMA channels to trigger the transfer for each of the channels.

2.3.3.6 DMA Timers

There are four independent, DMA-transfer-generating 32-bit timers (DTIM0, DTIM1, DTIM2, DTIM3) on the MCF5282. Each timer module incorporates a 32-bit timer with a separate register set for configuration and control. The timers can be configured to operate from the system clock or from an external clock source using one of the DTIN*x* signals. If the system clock is selected, it can be divided by 16 or 1. The input clock is further divided by a user-programmable 8-bit prescaler which clocks the actual timer counter register (TCR*n*). The timer counter may be set as free running or restarting, and a timer resolution of 15-ns at 66.7 MHz can be achieved.

Each of these timers can be configured for input capture or reference compare mode. By setting the internal registers, each timer may be configured to assert an external signal, generate an interrupt (maskable) on a particular event or cause a DMA transfer. Input capture may be set to sense either edge transition on the input pin. Output compare supports programmable modes for the output pin toggling.

2.4 Operating Modes

Both the MMC2114 and the MCF5282 can be operated in either single-chip mode or master mode. Each of these modes have the same functionality and features for both devices. The mode of operation is selected through chip configuration at reset; for both devices, the default operating mode is single-chip mode. See Section 2.4.1, “Chip Configuration” for more information about chip configuration.

2.4.1 Chip Configuration

Both devices use the same method and similar pins when going through chip configuration. That is, when the $\overline{\text{RCON}}$ pin is asserted at reset, the state of several of the device’s pins determines the mode of operation for the device. This section discusses the common and different modes of operation for the two devices, and how each device is configured for the various modes.

For the MCF5282, there are five mode settings used to configure the device for a specific configuration. These mode settings are chip operating mode, boot device and size, output pad drive strength, chip clock mode, and chip select configuration. Each of these mode parameters have two to four different selections that are given below.

- Chip operating mode
  - Master mode
  - Single-chip mode (default)
  - Factory access slave test (FAST) mode for factory test only
  - Reserved
- Boot device/size
  - Internal Flash boot (default)
  - External device boot
    - 32-bit
    - 16-bit
    - 8-bit
- Output pad strength:
  - Partial drive strength (default)
  - Full drive strength
- Clock mode:
  - Normal PLL with external crystal (default)
  - Normal PLL with external clock
  - 1:1 PLL Mode
  - External oscillator mode (no PLL)
- Chip select configuration:
  - PF[7:5] configured as chip select(s) and/or address line(s)
    - PF[7:5] configured as A23-A21 (default)
    - PF7 configured as $\overline{\text{CS6}}$, PF[6:5] as A22-A21
    - PF[7:6] configured as $\overline{\text{CS}}$[6:5], PF5 as A21
    - PF[7:5] configured as $\overline{\text{CS}}$[6:4]

This set of chip configuration options is somewhat different from the set for the MMC2114. There are four chip operating mode selections for the MMC2114: the three offered for the MCF5282, plus an emulation mode. The boot device/size selection is also different from the MMC2114 as the MMC2114 has no option for an 8-bit boot device. The output drive strength and the clock mode selections are the same for the two devices, but the chip select configuration for the MCF5282 is not offered for the MMC2114. For the MMC2114, there are only four chip selects and there is no mode option to expand this number.

The two devices are also different in the pins used for chip configuration. Table 4 shows the data bus pins and the dedicated pins used by each device for chip configuration.

Table 4. MMC2114/MCF5282 Configuration Pin Descriptions

MMC2114 Pin	MCF5282 Pin	Chip Configuration Function	Pin State/Meaning	Comments
$\overline{\text{RCON}}$		Chip configuration enable	1 Disabled 0 Enabled	Active low: if asserted, then all configuration pins must be driven appropriately for desired operation
D26, D17, D16		Select chip operating mode	111 Master 110 Single-chip 101 Reserved (MCF5282)/FAST (MMC2114) 100 FAST 0xx Reserved (MCF5282)/ Emulation (MMC2114)	
D19, D18		Select boot device / data port size	00 internal (32-bit) 10 External (8-bit) (MCF5282) / internal (32-bit) (MMC2114) 01 External (16-bit) 11 External (32-bit)	Value read defaults to 32-bit if chip operating mode selected is single-chip
D21		Select output pad drive strength	1 Full 0 Partial	
PLLEN, D23, D22	CLKMOD1, CLKMOD0	Select clock mode	**MMC2114:** 111 Normal PLL with crystal reference 110 Normal PLL with external clock reference 10x 1:1 PLL mode 0xx External clock mode (no PLL) **MCF5282:** 11 Normal PLL with crystal reference 10 Normal PLL with external clock reference 01 1:1 PLL mode 00 External clock mode (no PLL)	V_{DD}PLL must be supplied if a PLL mode is selected
N/A	D25, D24	Select chip select / address line	00 PF[7:5] configured as A23-A21 (default) 10 PF7 configured as $\overline{\text{CS6}}$, PF[6:5] as A22-A21 01 PF[7:6] configured as $\overline{\text{CS}}$[6:5], PF5 as A21 11 PF[7:5] configured as $\overline{\text{CS}}$[6:4]	

Chip configuration is performed at reset, and reset out (RSTO) can be used to gate an external latch to provide the desired polarity on each of the pins during chip configuration. The recommended chip configuration circuit is shown in Figure 1.

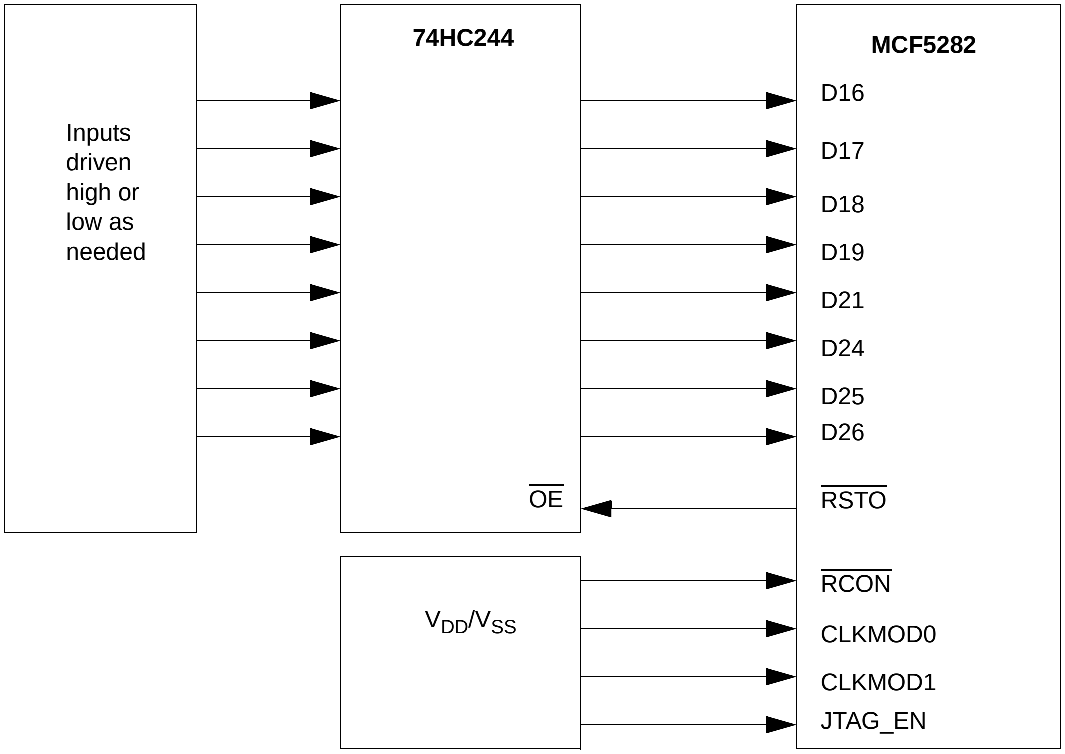

Figure 1. MCF5282 Recommended Reset Configuration Circuit

2.5 MMC2114 and MCF5282 Memory Maps

Memory maps for these two devices are completely different. Refer to the Memory Map section in the user's manual for each device for specific module and register locations. The primary memory mapping consideration when migrating to the MCF5282 is that the base address for the Flash memory, the SRAM system memory and the peripheral space can be defined programmatically. This is not so for the MMC2114, as all memories and peripheral addresses are fixed and cannot be changed.

The base address for the Flash array is set by programming the Flash base address register (FLASHBAR). The base address programmed into this register must be on a 32-Kbyte boundary within the 4-Gbyte address space (0x00–0xFFFF_FFFF).

Similarly, the base address for the internal SRAM module is set by programming the RAM base address register (RAMBAR). Its base address also must be programmed on a 32-Kbyte boundary within the 4-Gbyte address space.

Both of these memory base address registers must be initialized before using the particular memory array. After the base address has been set in these registers, then the valid (V) bit must then be set to enable the array. Note that if booting out of internal Flash, the FLASHBAR assumes the default value of 0x00 for the base address of Flash, and the Flash module is enabled by having the FLASHBAR's V bit automatically set at reset.

The final base address register is the internal peripheral system base address register (IPSBAR); it specifies the base address for the 1-Gbyte address space for all of the internal peripheral modules. Each on-chip peripheral has a unique offset from the IPSBAR for its control and status registers. The IPS base address must be programmed on a 1-Gbyte boundary within the 4-Gbyte address space. The default address for IPSBAR is 0x4000_0000; the other three addresses that the IPSBAR can assume are 0x00, 0x8000_0000, and 0xc000_0000. The IPS space is automatically enabled at reset by having the V bit set in IPSBAR by default.

Each of these three base address registers should be mapped such that its memory space is totally accessible and does not conflict with addresses of other memory spaces, including internal cache, SDRAM address space, and the programmed chip select spaces. Nothing prevents the overlapping of these memory spaces, but there is a built-in priority scheme to resolve addressing conflicts when an address hits multiple memory spaces. External memory space may be addressed by either the External Interface Module (EIM) or the SDRAM controller. The range that can be addressed by these modules is controlled by the respective chip selects for each module.

Another point about addressing the internal Flash is that writes to Flash during programming, or access to the Flash by a bus master other than the core, must use a “backdoor” access address. The base address for these backdoor accesses is the IPSBAR address plus an offset of 0x0400_0000.

3 Summary

As this document demonstrates, there is no lost functionality, memory resources, or performance when migrating from the MMC2114 to the MCF5282. Since code density is very comparable between these two families, any current MMC2114 application residing in the 256-Kbyte internal Flash and using the 32-Kbyte internal SRAM will easily fit into the 512-Kbyte Flash/64-Kbyte SRAM memory space of the MCF5282. With the capability of doubling the operating frequency in migrating from the MMC2114 to the MCF5282, there should not be any additional performance issues. Furthermore, any module functionality available on the MMC2114 is either duplicated or improved upon on the MCF5282, and the additional set of MCF5282 on-chip peripheral modules should enable many off-chip functions to be incorporated on chip, reducing hardware costs.

Because the two devices implement different architectures, there are a few things that the user has to consider when migrating. Differences in the two cores requires that any assembly code be converted. Other tasks related to the core, such as the program counter, stack pointer, and memory bar initialization, must be managed as part of system start-up. Vector tables and interrupt service routines will require special attention. There are also system-level changes, such as memory map differences, memory protection schemes, and low power mode operation, that will also need to be considered.

If the advantages gained from more functional integration and higher performance make this migration beneficial, then hopefully this document will serve as a valuable reference to make the transition as smooth as possible.

4 References

Table 5. References

Motorola Document Number	Title	Revision
MCF5282UM/D	MCF5282 Advance Information Manual	0
CFPRM/D	ColdFire Family Programmer's Reference Manual	2
CFPRODFACT/D	The ColdFire Family of 32-Bit Microprocessors Family Overview and Technology Roadmap	0
MCF5XXXWP	MCF5XXXWP WHITE PAPER: Motorola ColdFire VL RISC Processors	0
MCORERM/AD	M•CORE Reference Manual	0
MMC2114/D	MMC2114 Advance Information Manual	1

5 Document Revision History

Table 6 provides a revision history for this application note.

Table 6. Document Revision History

Rev. No.	Substantive Change(s)	Date of Release
0	Initial release.	November 2002

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK