Freescale Semiconductor
Application Note

AN3753
Rev. 1, 09/2008

MPC551x to MPC560xB/C, SPC560Bx/Cx Migration

by: Chris Platt
Applications Engineering
Microcontroller Solutions Group

1 Introduction

The MPC551x family has introduced features into a Power Architecture family of devices that support reduced current consumption, specifically to assist the power-sensitive body electronics system space. The power consumption can be tuned during dynamic operating modes, where code execution and actuation are supported, and also in static modes, where the minimum power consumption is necessary.

Freescale has recently introduced the MPC560x family in a joint development program with STMicroelectronics. This family shares many low-power features with the MPC551x family, but users migrating from one family to the other need to be aware of key differences between them.

This document outlines the differences between the MPC551x and the MPC560x familes.

Contents

1 Introduction 1
2 Overview of MPC560x. 2
2.1 Core. 2
2.2 Memory 2
2.3 Communications 2
2.4 Analog 2
2.5 Timed I/O. 2
2.6 Other Features 2
3 Main Software (Peripherals) Differences. 3
4 Peripheral Differences 5
4.1 Pinout 5
4.2 Memory Map 5
4.3 Core. 5
4.4 Interrupt Controller. 5
4.5 System Integration Unit 6
4.6 LIN Module 6
4.7 Analog-to-Digital Converter (ADC) 8
4.8 eMIOS Timer Module 9
4.9 Flash Module. 10
4.10 Watchdog Module 12
4.11 MPU (Memory Protection Unit) 13
5 Power Modes. 13
5.1 MPC551x Run, Stop, and SMS Modes 13
5.2 MPC551x Sleep Modes. 14
5.3 MPC560x Dynamic Power Modes. 15
5.4 MPC560x Static Power Modes 16
5.5 Power Mode Summary 16
5.6 External Wakeup Pin Comparison. 16
6 Non-Maskable Interrupts (NMI) 17
7 Clock Sources 19
8 Parametric Differences 21
9 Conclusion. 21

2 Overview of MPC560x

Figure 1 shows a block diagram of the MPC560x device.

Here are the key features:

2.1 Core

- PowerPC e200z0h core running 0–64 MHz
- VLE ISA instruction set for superior code density
- Vectored interrupt controller
- Memory protection unit with eight regions, 32-byte granularity

2.2 Memory

- 512 KB embedded program flash, 64 KB data flash
- 64 KB embedded data flash (for EE emulation)
- Up to 64 MHz nonsequential access with 3 WS
- ECC-enabled array with error detect/correct
- 48 KB SRAM (single-cycle access, ECC-enabled)

2.3 Communications

- Up to 6× enhanced FlexCAN with 64 message buffers each; full CAN 2.0 spec
- 4× LINFlex
- 3× DSPI, 8-16 bits wide and chip selects
- 1× IC

2.4 Analog

- 5 V 36-channel ADC 10-bit resolution (support down to 3 V with degraded performance)

2.5 Timed I/O

- 2×28 channel eMIOS module

2.6 Other Features

- Debug: Nexus 2 + (208MPABGA package only)
- I/O: 5 V or 3 V configurable I/O, high flexibility with selecting GPIO functionality
- Packages: 100LQFP, 144LQFP (208MAPBGA development only)
- Boot assist module for production and bench programming

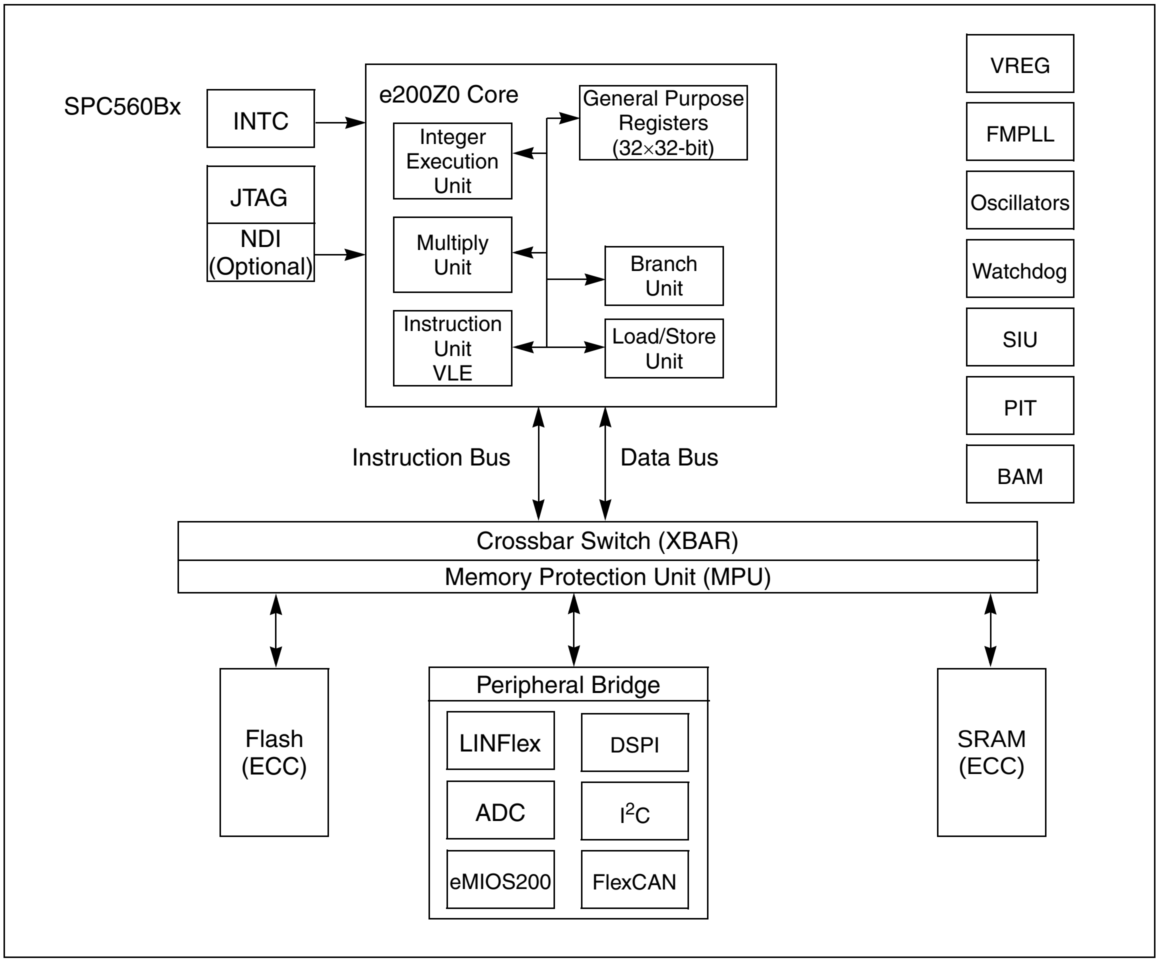

Figure 1. SPC560Bxx Block Diagram

3 Main Software (Peripherals) Differences

Most of the modules on the MPC560x family are shared with the MPC551x family. Even so, there are minor implementation differences which will be discussed later in this document. There are however three major peripheral differences between the families. Figure 2 shows a block diagram of the MPC551x family device with the main peripheral differences circled in red.

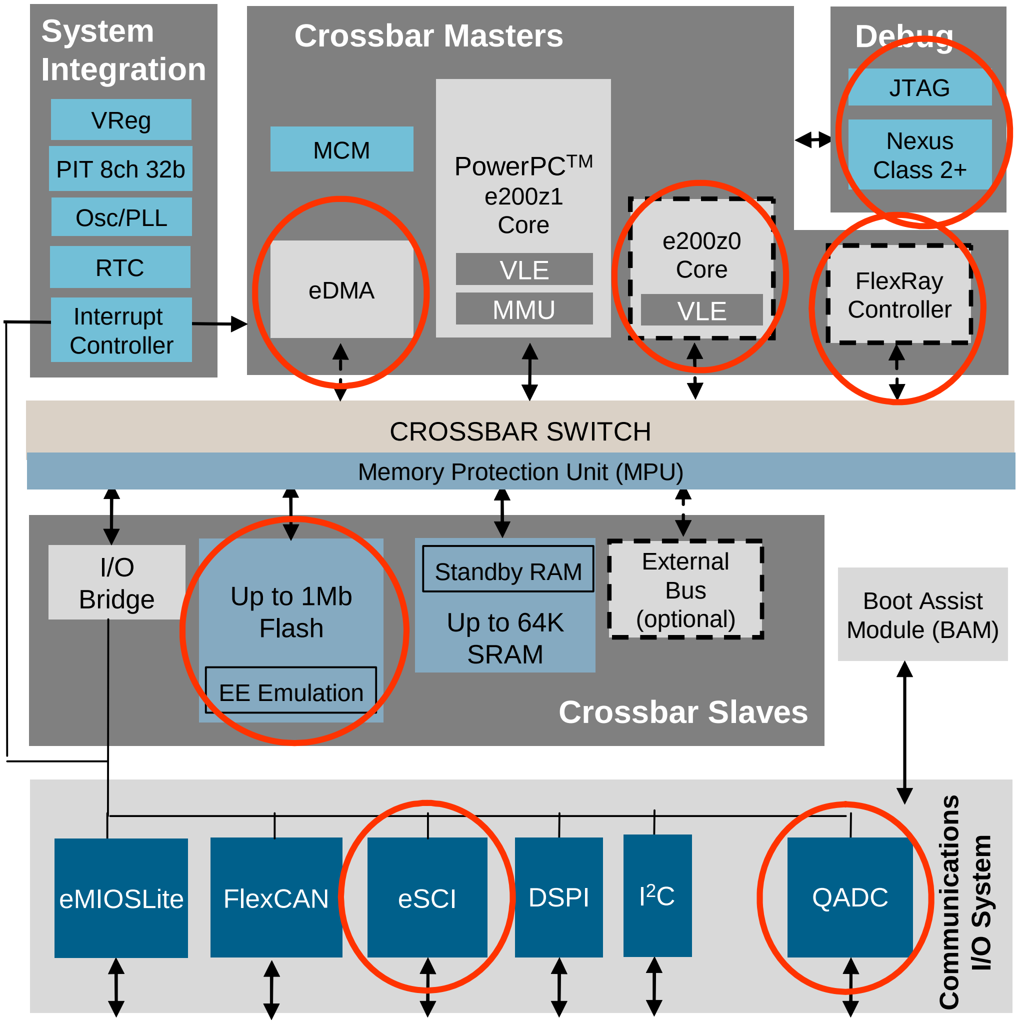

Figure 2. MPC551x with Main Peripheral Differences Highlighted

The main differences are shown in Table 1.

Table 1. Major Differences Between MPC560x and MPC551x Families

	MPC560x	MPC551x
Flash	ST 90nM	Freescale 0.13u
SCI	ST LINFlex controller	Freescale eSCI
ADC	ST ADC	Freescale QADC
eDMA	No	Yes
I/O Processor	No second core	Optional second core
Flexray Controller	Yes	No
Nexus	Nexus 2+	Nexus 2+

These differences will be discussed in more detail later in the document.

4 Peripheral Differences

4.1 Pinout

Since the two families do not use all of the same modules, and some of the modules which are common between them are configured differently, the pinouts of the two families are not identical. Therefore it will not be possible to simply replace an existing MPC551x part with a MPC560x device unless there is a PCB modification.

4.2 Memory Map

Wherever possible on the MPC560x family, the peripherals are kept in the same place in the memory map as in the MPC551x family. For example, RAM and FlexCAN occupy the same memory map space on both families.

However, there are some memory map inconsistencies between the families, although this should largely be transparent to the end user. This is because the compiler used to build the final application code will ensure that reads and writes are performed correctly.

4.3 Core

The MPC551x family employs a dual core implementation, namely a Z1 and a Z0. The Z0 on this family is essentially a stripped down Z1 — it runs only VLE code and has no MMU.

There is a single Z0 core on the MPC560x devices. However, it is not the same Z0 core as found on the MPC551x family. The Z0 core on the MPC551x has a unified address and data bus (Von Neumann architecture). The Z0 core on the MPC560x has separate address and data busses (Harvard architecture), providing improved performance over the Z0 core on the MPC551x, and more in line with the Z1 core found on the MPC551x.

This is another difference that should be somewhat transparent to the user.

4.4 Interrupt Controller

The interrupt controller found on the MPC560x family is the same as the one found on the MPC551x devices, but with some important implementation differences.

Sixteen programmable priority levels are supported, but there are around 70 fewer vectors than on the MPC551x family, mostly due to the different feature sets of the two families.

Some vectors are allocated differently, a difference which the user can handle in software. But when doing so, the user needs to be aware of priority level allocation. For example, the DSPI, PITs, IIC, and DMA interrupts are identical but exist at different locations.

Differences for the MPC560x family include:

- New vectors for new IP
- The addition of single-bit ECC error notification
- Missing vectors for unimplemented features

- LINFlex provides separate Rx, Tx, and error vectors for each module
- eMIOS interrupts are combined into groups of two channels
- SIU external interrupt support is combined differently
- FlexCAN interrupt arrangement efficiency is improved
- Since the ADC module is different (FIFO versus results register), the interrupt support is also different

4.5 System Integration Unit

The system integration unit (SIU) module on the MPC560x family is derived from the SIU found on MPC551x and is referred to as the SIUL (SIU Lite) on the MPC560x parts. The SIU is the module which is responsible for a variety of miscellaneous functions, including controlling the operation and characteristics of the input and output pads on the device.

The GPIO operation of the SIUL remains consistent with the SIU found on MPC551x, and pads remain controlled by the PCR register on a pad-by-pad basis. Figure 3 shows a pad configuration register for the MPC560x.

There is a new control feature added for analog pad control which was not present on MPC551x, as well as a bit for safe mode control (SMC) which can be used to set a particular pad to a known safe configuration.

R		SMC	APC		PA		OBE	IBE	DSC		ODE	HYS	SRC		WPE	WPS
W	▨			▨												
Reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Figure 3. Pad Configuration Register

On MPC560x, the function of the PA (pin assignment) bits is slightly different. These bits are used to select the functionality of multiplexed pads. On MPC551x the PA bits were used for both input and output functions. On MPC560x the PA bits are used only for output function selection. The input function selection is now handled by the new Muxed input register [PSMI(n)].

Also supported in the SIUL are external interrupts (16 possible in both families) as well as wake up pads, which are discussed later in the document. Input filtering is also supported on both families.

4.6 LIN Module

The LIN modules on the MPC551x and MPC560x are different. The former has the Freescale eSCI module to provide its LIN functionality whilst the latter has the ST Micro LINFlex module.

Here we compare the two LIN modules.

4.6.1 eSCI (MPC551x)

- LIN master mode and UART operating modes
- Compliant with LIN 1.3, LIN 2.0, and LIN 2.1 specifications

- Autonomous LIN frame handling when combined with DMA in master mode
- If required, LIN slave mode may be implemented in software
- LIN master mode state machine
- Supports generation of LIN message header (break, sync, ID)
- Detection and flagging of LIN errors
- Bit; Checksum, CRC, physical stuck-at, overrun, and timeout errors
- Classic or extended checksum calculation

4.6.2 LINFlex (MPCx)

- LIN master, LIN slave, and UART operating modes
- Compliant with LIN 1.3, LIN 2.0, and LIN 2.1 specifications
- Handles autonomous LIN frame transmission and reception (can discard data in master mode reception and slave mode)
- Autonomous LIN frame handling even in stop mode
- Message buffer to store identifier and up to eight data bytes
- Detection and flagging of LIN errors
  - Sync field
  - Delimiter
  - ID parity
  - Bit
  - Framing
  - Checksum
  - Overrun and timeout errors
- Classic or extended checksum calculation
- LIN slave mode features:
  - Autonomous LIN header handling
  - Autonomous LIN response handling
  - Automatic resynchronization for slave mode operation with internal RC source clock
  - Identifier filters for autonomous message handling
- Fractional baud rate prescaler

Obviously, since the two modules are totally different, the register structure contained within them is also totally different.

While both modules are fully compliant with the LIN 1.3, LIN 2.0, and LIN 2.1 specifications, the LINFlex module found on MPC560x contains some enhancements over the eSCI, namely:

- Classic or extended checksum calculation
- Slightly enhanced detection and flagging of LIN errors
- Hardware support for LIN slave mode (internal message and ID buffers)

- Identifier filters for autonomous message handling in slave mode

4.7 Analog-to-Digital Converter (ADC)

As with the LIN module, the ADC module on the MPC560x comes from ST Micro and is therefore different from the ADC on MPC551x.

Table 2 shows a comparison between the two modules.

Table 2. Analog-to-Digital Converter Comparison

	MPC551x	MPC560x
Resolution	12-bit/10-bit	10-bit
ADC Clock	Max 12 MHz	Max 20 MHz
Power sup	5.5 V	3 V to 5.5 V
Conv Time	1.25–11.83 µs	650 ns to 14.3 µs
TUE	< ±12 LSB (Prelim)	< ±2 LSB after offset canc.
IP Cap	0. 8 pF (ADC) 4 pF (PIN)	1.2 pF (ADC) 5 pF (PIN)
Current	4.5 mA	2 mA
Start time	10 µs	<1 µs

Figure 4 shows how the conversion principles differ between the two families.

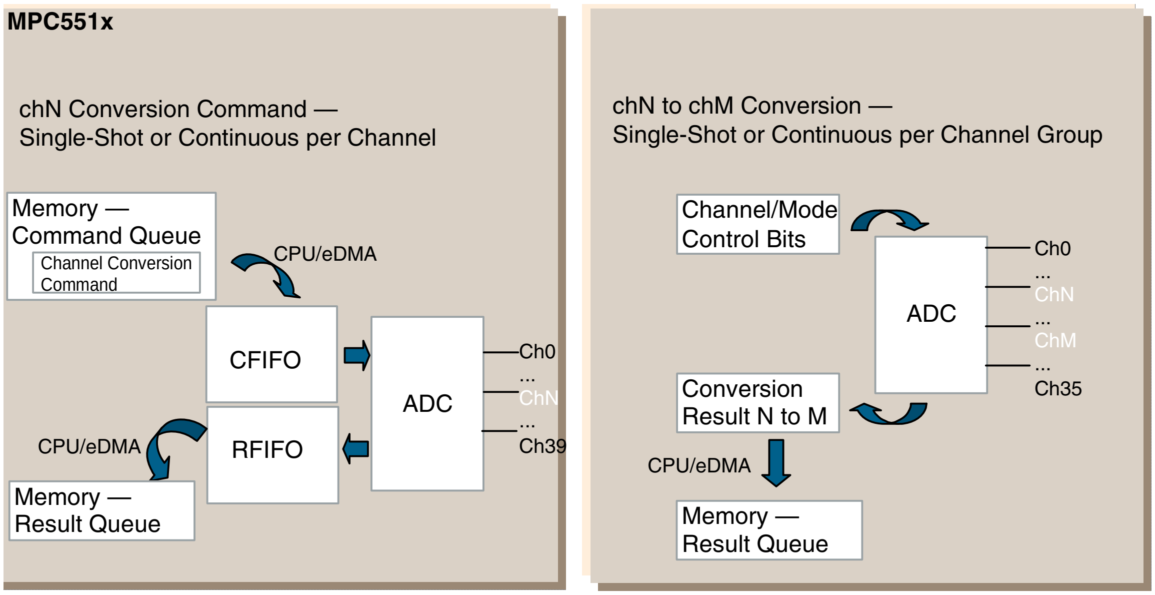

Figure 4. ADC Conversion Principle Comparison

Since the ADC converters are different (as discussed above), the pins required to operate the modules differ too. Figure 5 shows a comparison of the pins between the two families.

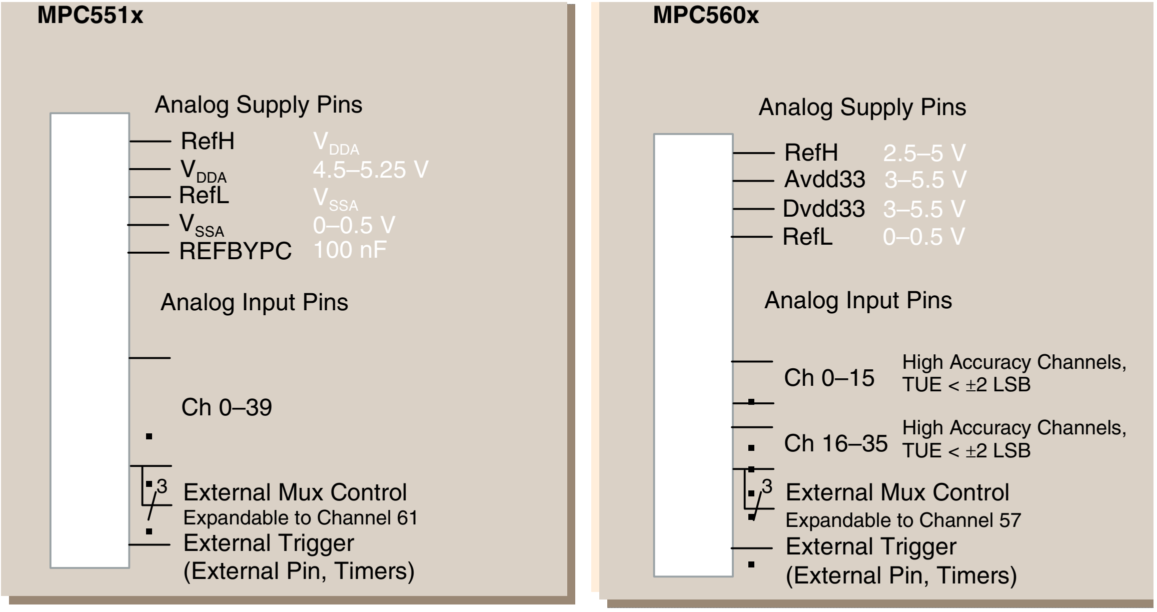

Figure 5. ADC Pin Comparison

Some new features available on the MPC560x ADC not available on the MPC551x module are:

- Injected conversion:
  — A continuing conversion sequence can be interrupted by a second one-shot sequence.
- Triggered injected conversion:
  — An external or internal trigger (eMIOS, PIT) can start an injected conversion.
- Analog Watchdog:
  — Allows continuous monitoring of four or eight analog input channels. An interrupt request is generated whenever the converted value of one of these inputs is outside the upper or lower programmed threshold values.

4.8 eMIOS Timer Module

The eMIOS timer system found on the MPC560x is based on the same module found on the MPC551x family but configured quite differently. It also has a new mode, OPWMT (OPWM with trigger generation), supporting shifted PWM and trigger generation to synchronize ADC conversion with the PWM signal.

Whilst the MPC551x family contains a single eMIOS module with 24 channels and 16 bit counter busses, the MPC560x features two separate eMIOS instantiations, each with 28 channels for a total of 56 available channels.

The start of both eMIOS modules can be synchronized, and all 50 output PWM channels are connected to the new cross triggering unit lite (CTUL) (not found on MPC551x), which allows the channels to be used to trigger ADC conversion synchronously to the PWM channels and without any CPU intervention.

The ADC trigger conversion can be configured for each OPWMT channel anywhere in the PWM period.

This is ideal for lighting applications, to implement synchronous analog diagnostic of power switches in T_{On} and T_{Off} states.

The MPC560x eMIOS configuration is shown in Figure 6.

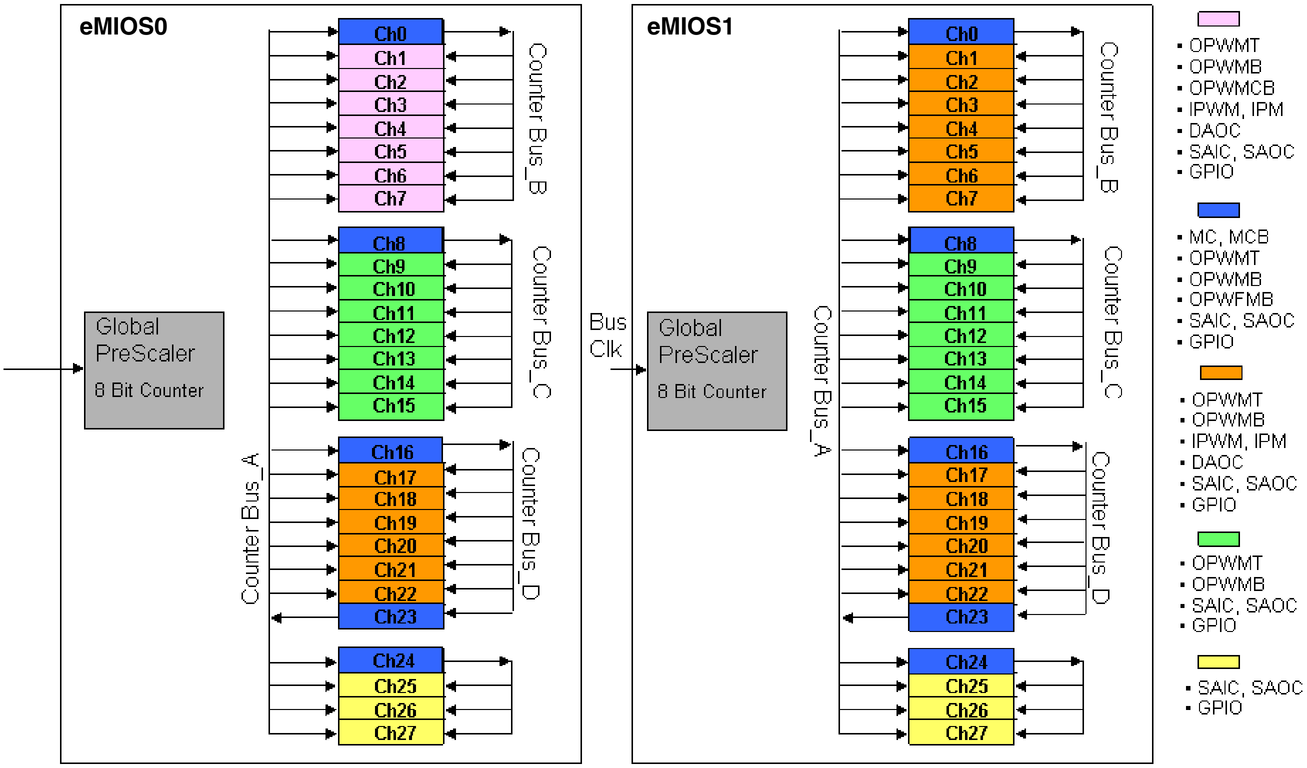

Figure 6. MPC560x eMIOS Configuration

4.9 Flash Module

Since the two families are manufactured by different technologies, the flash memory modules are a little different.

Both modules incorporate ECC (error correction code) on each 64 bits of flash memory, but the flash on the MPC560x now has the added feature of single bit error visibility.

Both families implement data flash with multiple 16K flash blocks using standard flash cell with ECC. Both also support program and erase operations with a state machine. The two families have different data flash configurations (shown below) and different positions in the memory map.

Read while write is supported slightly differently on MPC560x, through a separate flash array implementation. This restricts operation compared to MPC551x, where two partitions are supported in the data flash.

Figure 7 and Figure 8 show the relative flash configurations of the MPC551x and MPC560x families.

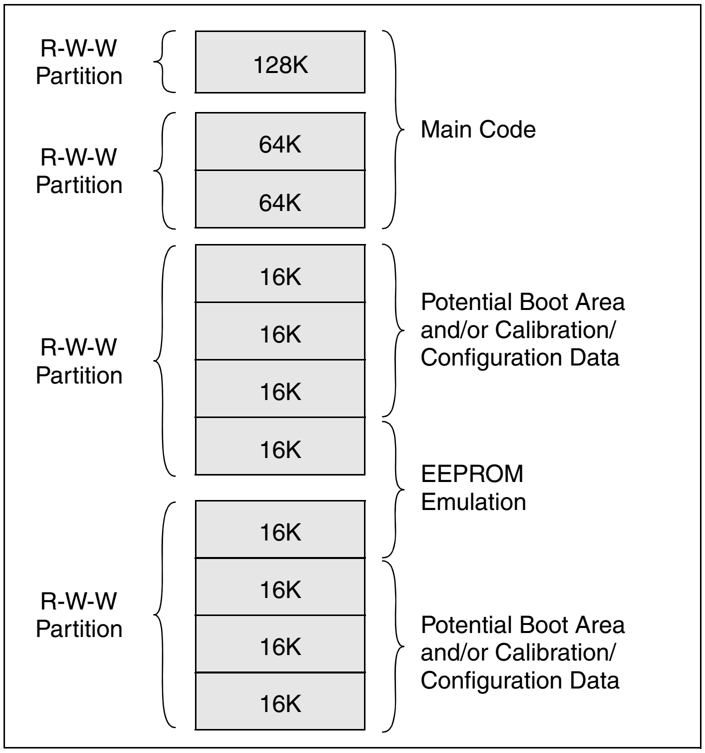

Figure 7. Flash Configuration on MPC551x

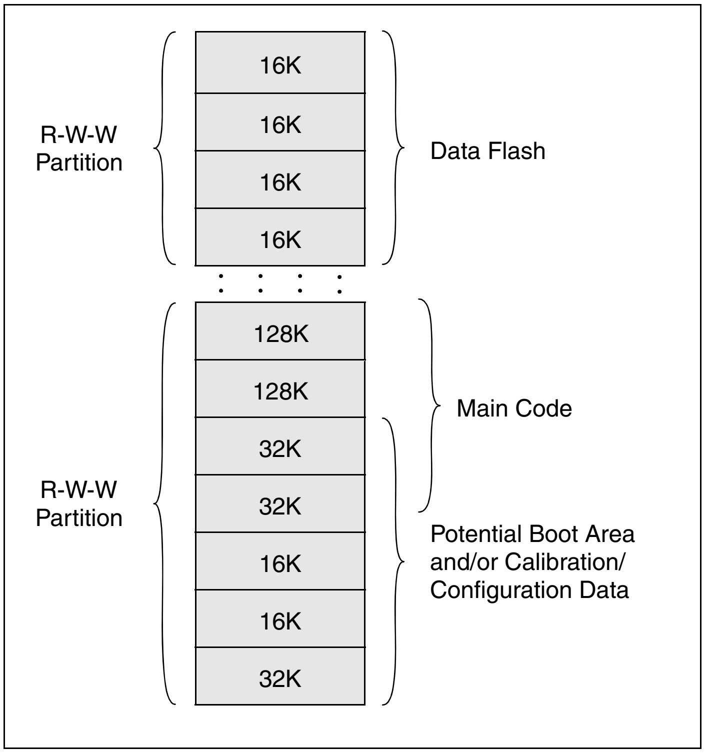

Figure 8. MPC560x Flash Configuration

Flash accesses are similar between the two families, with the MPC551x having two sets of 4×128-bit page buffers, compared to the MPC560x devices which have only a single 128-bit page buffer for accesses to the data flash.

The Z1 core on the MPC551x family could access the flash through the crossbar switch but also had a direct port to access the flash memory. On the MPC560x family all flash accesses must go through the crossbar switch (see Figure 9).

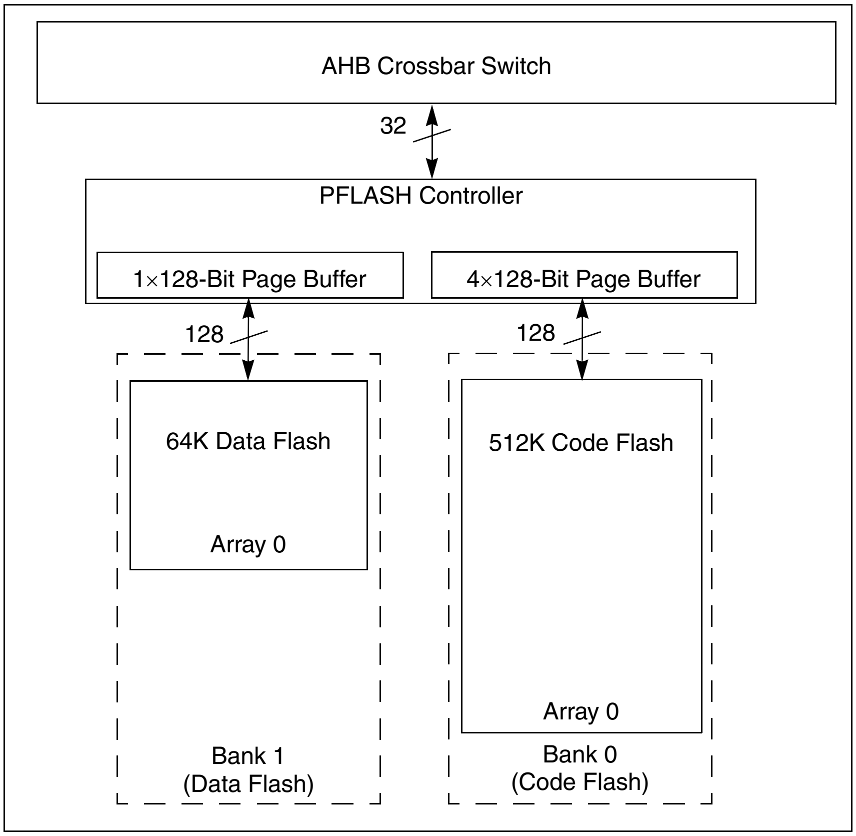

Figure 9. MPC560x — Flash Access Configuration

The write/erase cycles and data retention performance of both the MPC551x and the MPC560x flash blocks are planned to be comparable, but there is currently insufficient data for the MPC560x flash to print fully characterized data at this time.

4.10 Watchdog Module

The MPC560x parts have a new watchdog module on board which offers some improved safety features over the watchdog module on MPC551x.

The watchdog modules support standard and windowed operating modes. The watchdog module is enabled out of reset but can be disabled as required.

Writes to the service register hold off the watchdog.

MPC551X — 0x55 followed by 0xAA

MPC560x — 0xA602 followed by 0xB480

There is protection against inadvertent modification with optional soft locks and hard locks. Soft locks allow temporary locking of configuration; a hard lock, once it's enabled, prevents any changes until after a reset. There is also a configurable timeout time as well as response type on timeout, i.e., reset, interrupt, or interrupt followed by a reset.

4.11 MPU (Memory Protection Unit)

The MPU on the MPC551x devices contains 16×128-bit descriptors, whereas the MPU on MPC560x has 8×128-bit descriptors.

5 Power Modes

Since both families are specifically designed to be used in power sensitive applications, various low-power modes are available for selection by the user to minimize overall current consumption.

The power modes of both device families are shown in detail here.

5.1 MPC551x Run, Stop, and SMS Modes

5.1.1 Complete Microcontroller System

- Z1RUN
  - Traditional full run mode
  - All of the device is powered and RAM access is available
  - Z1 is allowed to execute
  - User can disable Z0 and any modules, hence inhibiting clock to specific modules
  - Power estimate: < 125 mA (all parameters are for indication only — committed values will be found in the current datasheet)
- Z1STOP
  - Traditional full stop mode
  - All of the device is powered and RAM is retained
  - I/O processor and Z1 held in halt mode
  - All peripherals clock-gated off
  - Power estimate: < 400 µA @ 25 °C

5.1.2 Small Microcontroller System (Achievable by Software)

- SMS Functionality
  - Use active clock gating to segment the device
  - Select either Z0 or Z1 as principal core
  - Clock the applicable core and a small peripheral set

  - Clock-gate off the remaining peripherals and other core
- SMSRUN
  - All RAM retained and access is available
  - Selected processor and peripheral configuration automatically restored from RAM
  - Selected processor allowed to execute
  - Power estimate: 10 mA based on 16 MHz internal RC
- SMSSTOP
  - All RAM retained
  - Selected processor and peripheral configuration automatically stored into RAM
  - Peripherals and processor powered down
  - Power estimate: <100 mA @ 25 °C (supported through software reconfiguration)

5.2 MPC551x Sleep Modes

- All power disabled to device except low power V_{reg}
- Uses aggressive power gating to minimize leakage
- Scalability of retained RAM for maximum customer flexibility
- Separate RAM power regulator to further reduce RAM leakage
- Pad keepers — output states retained (register values lost)
- Eight external wakeup sources
- Optional autonomous periodic interrupt (API), available in all sleep modes
  - Supports 1 ms to 1 hour wakeup capability
  - Incrementally consumes only 2 μA when enabled with on-chip 32 KHz IRC
- Optional RTC wakeup capability for even longer timeout
  - Supports 1 second to 24 hour wakeup periods
  - Incrementally consumes only 2 μA when enabled with on-chip 32 KHz IRC

Sleep mode configurations and estimates:

- All I/O states with full wakeup capability on selected I/O
- Power estimate: ~25 μA @ 25 °C + approximately 5 μA per 8K extra RAM
- 8K RAM retained and all I/O states
- 16K RAM retained and all I/O states
- 32K RAM retained and all I/O states
- 64K RAM retained and all I/O states

The MPC560x devices have similar modes but with slightly different functionality in each mode, as shown below.

5.3 MPC560x Dynamic Power Modes

5.3.1 Run Modes (0–3)

These are the main software running modes where most processing activity is done. These various run modes allow enablement of different clock and power configurations of the system, with respect to each other.

- Traditional full run mode
- RAM access available
- Z0 allowed to execute
- Multiple (four) run configurations with different clock and feature sets
- Offers rapid run mode change, to quickly change dynamic power and performance
- Four mode configuration registers hold the setup

5.3.2 DRUN (Default Run) Mode

- Entry mode for the embedded software — provides full system accessibility and enables system configuration; provides gate to enter user modes
- BAM (when present) is executed in DRUN mode
- Power estimate: < 100 mA

5.3.3 Wait Mode

CPU is stopped and clock is gated, with all peripherals remaining at normal full-speed operation

- Intended for short duration suspension of processing while allowing peripheral operations to continue
- Interrupt offers wakeup to core, to allow fast restart of CPU activity

5.3.4 Halt Mode

Reduced-activity low-power mode during which the core clock is disabled. Can be configured to switch off analog peripherals for efficient power management, at the cost of higher wakeup latency.

- Device continues to operate but with core clock gated, halting CPU processing
- Peripherals can continue to run at normal or reduced frequency
- Analog device components can be stopped, such as PLL, ADC, and main V_{reg}
- Flash can be turned off to save power
- PLL can be configured to be on or off (default is off)

5.4 MPC560x Static Power Modes

5.4.1 Stop Mode

Stop mode is an advanced low-power mode during which the clock to the core is disabled and main peripherals are all stopped.

- Use active clock gating to segment the device
- External oscillator stopped, but can be allowed to continue to run to support fast startup and the expense of added power
- All of device is powered and RAM is retained
- RTC and API can continue to run
- PLL always off
- Optional support of fast IRC and slow IRC
- Power estimate: < 180 μA @ 25 °C

5.4.2 Standby Mode

Provides the minimum power consumption mode for the MPC560x. In this mode power is cut off from most of device, with only a small power island retained. Wakeup requires recovery time for clocks and power supply.

- Recovery of data needed on exit from mode, with context saved prior to mode entry
- Selectable size of RAM supported: 8K or all RAM
- Optionally enabled low-speed IRC
- Wakeup from selected I/O or API/RTC
- Power estimate: All RAM retained < 50 μA @ 25 °C
- Power estimate: 8K RAM retained < 25 μA @ 25 °C

5.5 Power Mode Summary

In summary, both families have very similar regular run modes.

Both families have a halt mode which allows clock gating to the core and the peripherals. The operation of this mode is the same on both families.

Both families have an identical stop mode.

MPC551x has multiple sleep modes with varying amounts of RAM kept powered. MPC560x has standby mode which is essentially the same as sleep mode but using the ST naming convention. Only 0K or 8K RAM is selectable to be kept alive in this mode.

5.6 External Wakeup Pin Comparison

Both families have the ability to wake from low-power modes triggered by transitions on external pins.

On MPC551x, eight external wakeup pins can be configured using a multiplexer to select from a pool of 64 possible pins.

By comparison, the MPC560x devices allow 18 fixed external wakeup pins to be supported at any one time, as shown in Figure 10 below.

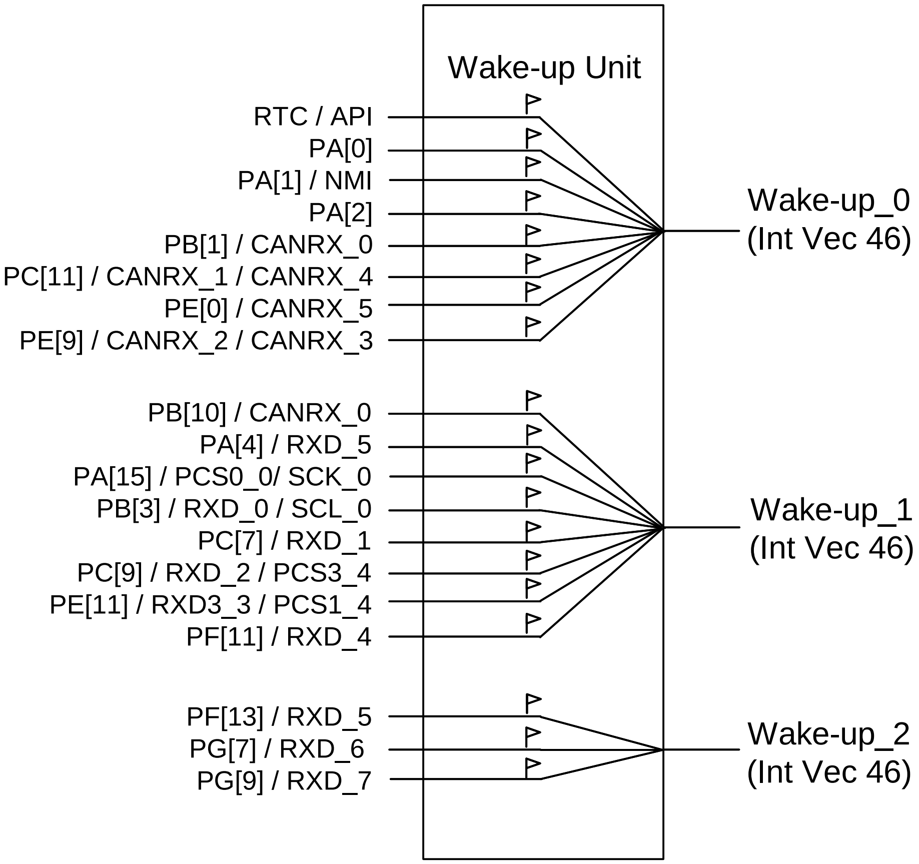

Figure 10. Wakeup Pin Assignment

6 Non-Maskable Interrupts (NMI)

Both families offer the ability to utilize a non-maskable interrupt function. However, the feature is achieved in slightly different ways on the two families.

Figure 11 shows how the function is achieved on MPC551x. The NMI bypasses the interrupt controller and is instead routed from external pins to the critical interrupts for the two cores (Z0 & Z1). Once enabled in the SIU, the NMI function cannot be blocked by re-configuring the pad in the SIU.

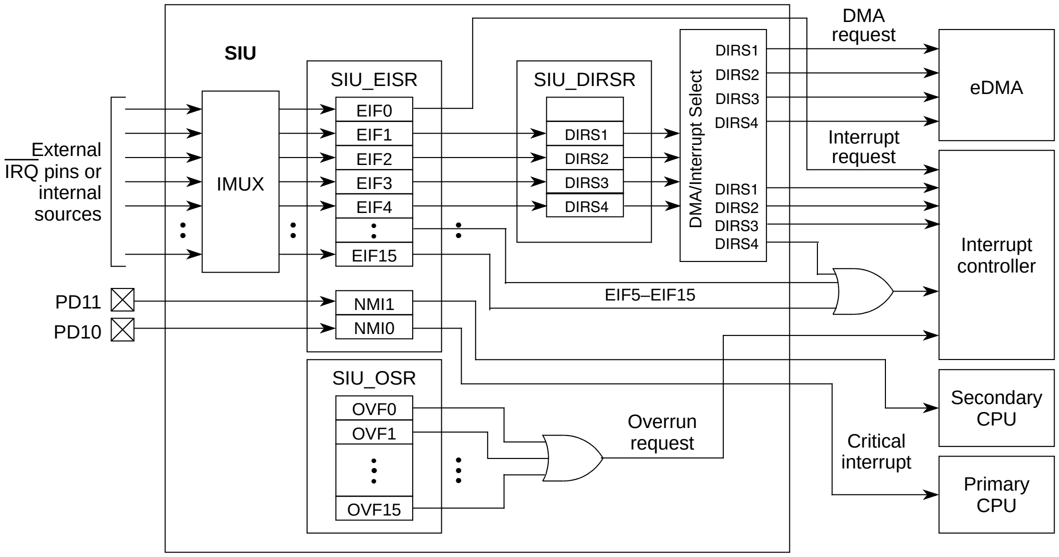

Figure 11. NMI Architecture on MPC551x

Figure 12 shows how the NMI is set up on the MPC560x parts. The NMI is configured in the NMI configuration register (NCR). Once enabled, the SIUL cannot reconfigure the allocation of this pin to inhibit the NMI.

There are three NMI types on MPC560x.

- Critical Interrupt (same as MPC551x)
  - Recoverable NMI, but can be blocked by CE bit in core MCR
- Machine Check Exception
  - Cannot be masked, but nonrecoverable
    - Other MCE can occur and overwrite the CSRR[x]
- Machine Check Exception
  - As above, but NMI held to prevent other critical exception from occurring and overwriting the save restore register values

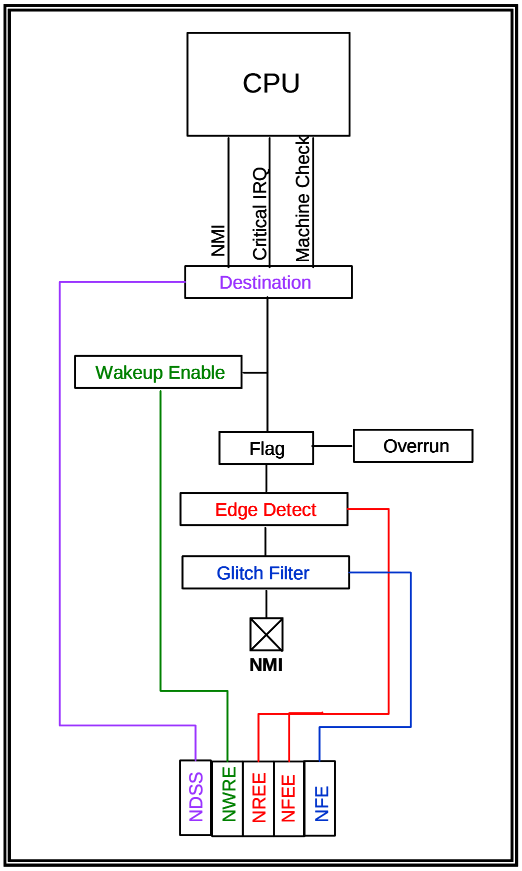

Figure 12. MPC560x NMI Architecture

7 Clock Sources

The two families have almost identical clock sources available to them.

Both have a main external oscillator input (4–40 MHz input on MPC551x, 4–16 MHz input on MPC560x).

Both also have an on-chip 16 MHz fast RC oscillator which can be used to drive the main system clock.

Both have an onboard slow oscillator which runs at 32 KHz on MPC551x and 128 KHz on MPC560x.

They also both have an external 32 KHz crystal source available to drive certain peripherals and onboard timers.

Figure 13 and Figure 14 show the two clocking schemes.

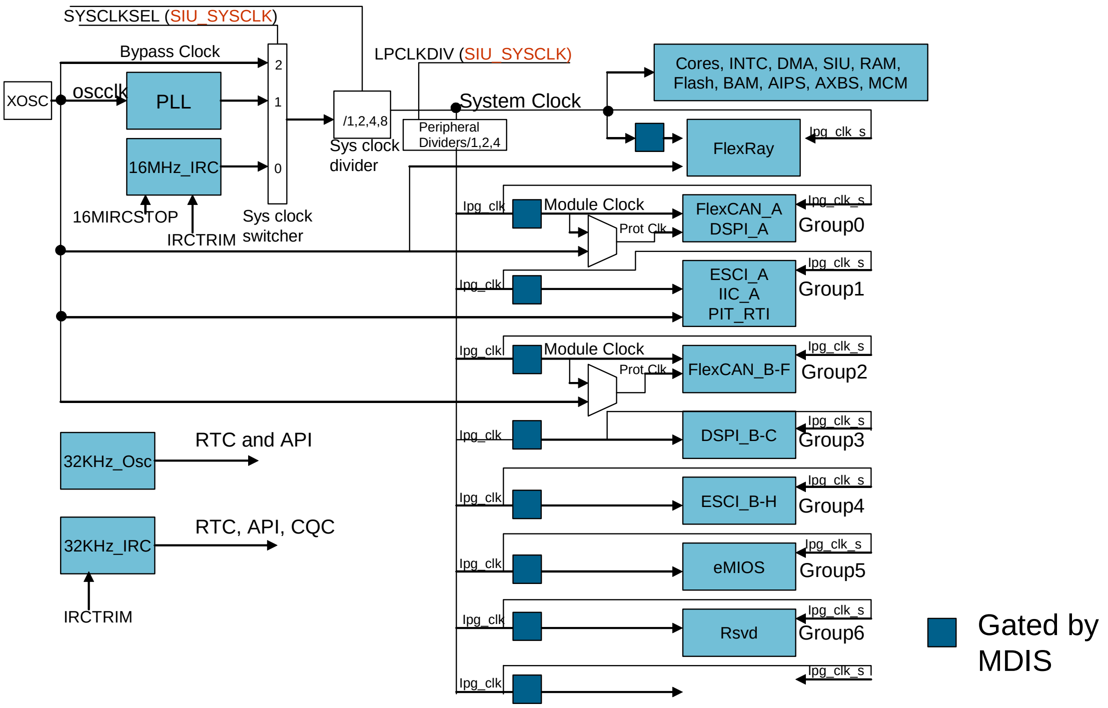

Figure 13. MPC551x Clocking Scheme

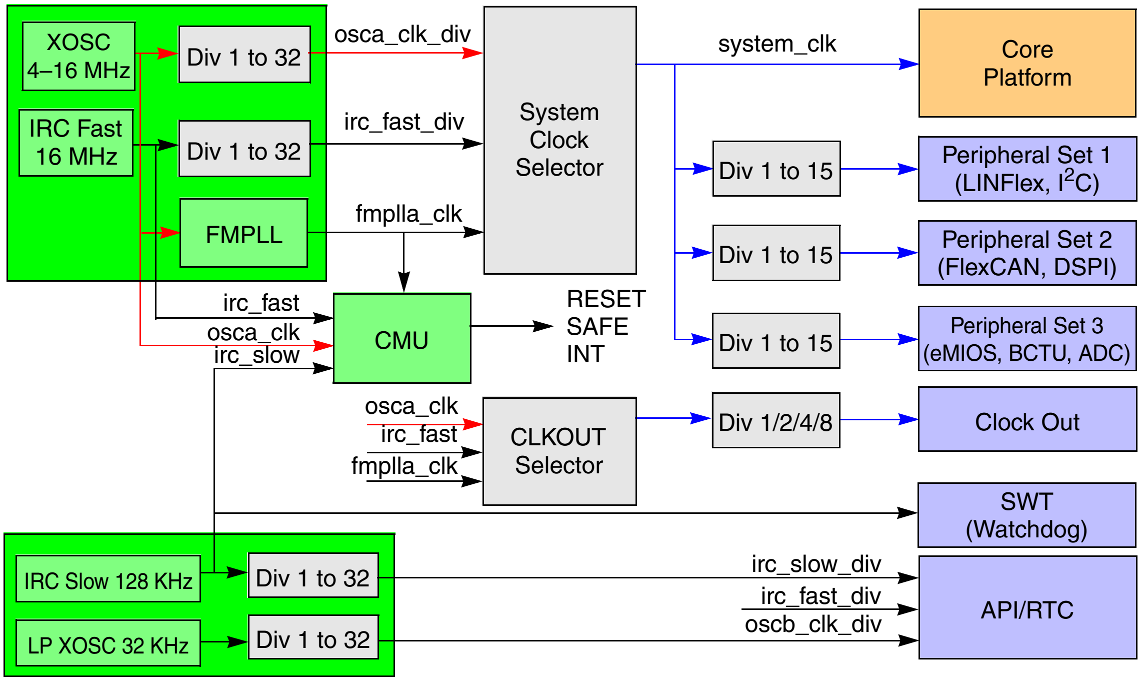

Figure 14. MPC560x Clocking Scheme

8 Parametric Differences

Since the two product families are built with different technologies, parametric differences will almost certainly be apparent in most areas.

All parameters quoted in this document are for indication only. Parametric data for MPC560x is available in the preliminary datasheet. Please refer to that document for committed values.

9 Conclusion

Despite being built with two different technologies, the MPC551x and MPC560x families share so many common peripherals, features, and tools that transitioning to the MPC560x family from the MPC551x family should prove a relatively simple task, if the user takes account of the differences outlined in this document.

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale’s Environmental Products program, go to http://www.freescale.com/epp.