



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

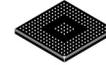




MPC5634M



144 LQFP
20 mm x 20 mm



208 MAPBGA
17 mm x 17 mm



176 LQFP
24 mm x 24 mm

MPC5634M Microcontroller Data Sheet

- Operating Parameters
 - Fully static operation, 0 MHz – 80 MHz (plus 2% frequency modulation - 82 MHz)
 - -40 °C to 150 °C junction temperature operating range
 - Low power design
 - Less than 400 mW power dissipation (nominal)
 - Designed for dynamic power management of core and peripherals
 - Software controlled clock gating of peripherals
 - Low power stop mode, with all clocks stopped
 - Fabricated in 90 nm process
 - 1.2 V internal logic
 - Single power supply with 5.0 V -10% / +5% (4.5 V to 5.25 V) with internal regulator to provide 3.3 V and 1.2 V for the core
 - Input and output pins with 5.0 V -10% / +5% (4.5 V to 5.25 V) range
 - 35%/65% V_{DDE} CMOS switch levels (with hysteresis)
 - Selectable hysteresis
 - Selectable slew rate control
 - Nexus pins powered by 3.3 V supply
 - Designed with EMI reduction techniques
 - Phase-locked loop
 - Frequency modulation of system clock frequency
 - On-chip bypass capacitance
 - Selectable slew rate and drive strength
- High performance e200z335 core processor
 - 32-bit *Power Architecture Book E* programmer's model
 - Variable Length Encoding Enhancements
 - Allows Power Architecture instruction set to be optionally encoded in a mixed 16 and 32-bit instructions
 - Results in smaller code size
 - Single issue, 32-bit *Power Architecture technology* compliant CPU
 - In-order execution and retirement
 - Precise exception handling
 - Branch processing unit
 - Dedicated branch address calculation adder
 - Branch acceleration using Branch Lookahead Instruction Buffer
 - Load/store unit
 - One-cycle load latency
 - Fully pipelined
 - Big and Little Endian support
 - Misaligned access support
 - Zero load-to-use pipeline bubbles
 - Thirty-two 64-bit general purpose registers (GPRs)
 - Memory management unit (MMU) with 16-entry fully-associative translation look-aside buffer (TLB)
 - Separate instruction bus and load/store bus
 - Vectored interrupt support
 - Interrupt latency < 120 ns @ 80 MHz (measured from interrupt request to execution of first instruction of interrupt exception handler)

This document contains information on a product under development. Freescale reserves the right to change or discontinue this product without notice.

© Freescale Semiconductor, Inc., 2008-2012. All rights reserved.

Table of Contents

| | | | | | |
|--------|------------------------------------|----|--------|-----------------------------------------------------------------------------------|-----|
| 1 | Introduction | 9 | 4.2 | Maximum ratings | 56 |
| 1.1 | Document overview | 9 | 4.3 | Thermal characteristics | 58 |
| 1.2 | Description | 9 | 4.3.1 | General notes for specifications at maximum junction temperature | 60 |
| 2 | Overview | 9 | 4.4 | Electromagnetic Interference (EMI) characteristics | 62 |
| 2.1 | Device comparison | 9 | 4.5 | Electromagnetic static discharge (ESD) characteristics | 62 |
| 2.2 | MPC5634M feature details | 9 | 4.6 | Power Management Control (PMC) and Power On Reset (POR) electrical specifications | 63 |
| 2.2.1 | e200z335 core | 9 | 4.6.1 | Regulator example | 67 |
| 2.2.2 | Crossbar | 11 | 4.6.2 | Recommended power transistors | 69 |
| 2.2.3 | eDMA | 11 | 4.7 | Power up/down sequencing | 69 |
| 2.2.4 | Interrupt controller | 12 | 4.8 | DC electrical specifications | 70 |
| 2.2.5 | FMPLL | 12 | 4.9 | I/O Pad current specifications | 77 |
| 2.2.6 | Calibration EBI | 13 | 4.9.1 | I/O pad VRC33 current specifications | 78 |
| 2.2.7 | SIU | 14 | 4.9.2 | LVDS pad specifications | 79 |
| 2.2.8 | ECSM | 14 | 4.10 | Oscillator and PLLMRFM electrical characteristics | 80 |
| 2.2.9 | Flash | 14 | 4.11 | Temperature sensor electrical characteristics | 82 |
| 2.2.10 | SRAM | 15 | 4.12 | eQADC electrical characteristics | 82 |
| 2.2.11 | BAM | 15 | 4.13 | Platform flash controller electrical characteristics | 85 |
| 2.2.12 | eMIOS | 16 | 4.14 | Flash memory electrical characteristics | 85 |
| 2.2.13 | eTPU2 | 16 | 4.15 | AC specifications | 87 |
| 2.2.14 | eQADC | 18 | 4.15.1 | Pad AC specifications | 87 |
| 2.2.15 | DSPI | 20 | 4.16 | AC timing | 90 |
| 2.2.16 | eSCI | 21 | 4.16.1 | IEEE 1149.1 interface timing | 90 |
| 2.2.17 | FlexCAN | 22 | 4.16.2 | Nexus timing | 93 |
| 2.2.18 | System timers | 23 | 4.16.3 | Calibration bus interface timing | 96 |
| 2.2.19 | Software Watchdog Timer (SWT) | 24 | 4.16.4 | eMIOS timing | 99 |
| 2.2.20 | Debug features | 24 | 4.16.5 | DSPI timing | 99 |
| 2.3 | MPC5634M series architecture | 26 | 4.16.6 | eQADC SSI timing | 105 |
| 2.3.1 | Block diagram | 26 | 5 | Packages | 106 |
| 2.3.2 | Block summary | 27 | 5.1 | Package mechanical data | 106 |
| 3 | Pinout and signal description | 29 | 5.1.1 | 144 LQFP | 106 |
| 3.1 | 144 LQFP pinout | 29 | 5.1.2 | 176 LQFP | 110 |
| 3.2 | 176 LQFP pinout (MPC5634M) | 30 | 5.1.3 | 208 MAPBGA | 113 |
| 3.3 | 176 LQFP pinout (MPC5633M) | 32 | 6 | Ordering information | 115 |
| 3.4 | 208 MAPBGA ballmap (MPC5634M) | 33 | 7 | Document revision history | 117 |
| 3.5 | 208 MAPBGA ballmap (MPC5633M only) | 34 | | | |
| 3.6 | Signal summary | 35 | | | |
| 3.7 | Signal details | 51 | | | |
| 4 | Electrical characteristics | 56 | | | |
| 4.1 | Parameter classification | 56 | | | |

- Non-maskable interrupt (NMI) input for handling external events that must produce an immediate response, e.g., power down detection. On this device, the NMI input is connected to the Critical Interrupt Input. (May not be recoverable)
- Critical Interrupt input. For external interrupt sources that are higher priority than provided by the Interrupt Controller. (Always recoverable)
- New ‘Wait for Interrupt’ instruction, to be used with new low power modes
- Reservation instructions for implementing read-modify-write accesses
- Signal processing extension (SPE) APU
 - Operating on all 32 GPRs that are all extended to 64 bits wide
 - Provides a full compliment of vector and scalar integer and floating point arithmetic operations (including integer vector MAC and MUL operations) (SIMD)
 - Provides rich array of extended 64-bit loads and stores to/from extended GPRs
 - Fully code compatible with e200z6 core
- Floating point (FPU)
 - IEEE 754 compatible with software wrapper
 - Scalar single precision in hardware, double precision with software library
 - Conversion instructions between single precision floating point and fixed point
 - Fully code compatible with e200z6 core
- Long cycle time instructions, except for guarded loads, do not increase interrupt latency
- Extensive system development support through Nexus debug port
- Advanced microcontroller bus architecture (AMBA) crossbar switch (XBAR)
 - Three master ports, four slave ports
 - Masters: CPU Instruction bus; CPU Load/store bus (Nexus); eDMA
 - Slave: Flash; SRAM; Peripheral Bridge; calibration EBI
 - 32-bit internal address bus, 64-bit internal data bus
- Enhanced direct memory access (eDMA) controller
 - 32 channels support independent 8-bit, 16-bit, or 32-bit single value or block transfers
 - Supports variable sized queues and circular queues
 - Source and destination address registers are independently configured to post-increment or remain constant
 - Each transfer is initiated by a peripheral, CPU, or eDMA channel request
 - Each eDMA channel can optionally send an interrupt request to the CPU on completion of a single value or block transfer
- Interrupt controller (INTC)
 - 191 peripheral interrupt request sources
 - 8 software settable interrupt request sources
 - 9-bit vector
 - Unique vector for each interrupt request source
 - Provided by hardware connection to processor or read from register
 - Each interrupt source can be programmed to one of 16 priorities
 - Preemption
 - Preemptive prioritized interrupt requests to processor
 - ISR at a higher priority preempts ISRs or tasks at lower priorities
 - Automatic pushing or popping of preempted priority to or from a LIFO
 - Ability to modify the ISR or task priority. Modifying the priority can be used to implement the Priority Ceiling Protocol for accessing shared resources.
 - Low latency—three clocks from receipt of interrupt request from peripheral to interrupt request to processor

- Frequency Modulating Phase-locked loop (FMPLL)
 - Reference clock pre-divider (PREDIV) for finer frequency synthesis resolution
 - Reduced frequency divider (RFD) for reducing the FMPLL output clock frequency without forcing the FMPLL to re-lock
 - System clock divider (SYSDIV) for reducing the system clock frequency in normal or bypass mode
 - Input clock frequency range from 4 MHz to 20 MHz before the pre-divider, and from 4 MHz to 16 MHz at the FMPLL input
 - Voltage controlled oscillator (VCO) range from 256 MHz to 512 MHz
 - VCO free-running frequency range from 25 MHz to 125 MHz
 - Four bypass modes: crystal or external reference with PLL on or off
 - Two normal modes: crystal or external reference
 - Programmable frequency modulation
 - Triangle wave modulation
 - Register programmable modulation frequency and depth
 - Lock detect circuitry reports when the FMPLL has achieved frequency lock and continuously monitors lock status to report loss of lock conditions
 - User-selectable ability to generate an interrupt request upon loss of lock
 - User-selectable ability to generate a system reset upon loss of lock
 - Clock quality monitor (CQM) module provides loss-of-clock detection for the FMPLL reference and output clocks
 - User-selectable ability to generate an interrupt request upon loss of clock
 - User-selectable ability to generate a system reset upon loss of clock
 - Backup clock (reference clock or FMPLL free-running) can be applied to the system in case of loss of clock
- Calibration bus interface (EBI)
 - Available only in the calibration package (496 CSP package)
 - 1.8 V to 3.3 V \pm 10% I/O (1.6 V to 3.6 V)
 - Memory controller with support for various memory types
 - 16-bit data bus, up to 22-bit address bus
 - Selectable drive strength
 - Configurable bus speed modes
 - Bus monitor
 - Configurable wait states
- System integration unit (SIU)
 - Centralized GPIO control of 80 I/O pins
 - Centralized pad control on a per-pin basis
 - Pin function selection
 - Configurable weak pull-up or pull-down
 - Drive strength
 - Slew rate
 - Hysteresis
 - System reset monitoring and generation
 - External interrupt inputs, filtering and control
 - Critical Interrupt control
 - Non-Maskable Interrupt control
 - Internal multiplexer subblock (IMUX)
 - Allows flexible selection of eQADC trigger inputs (eTPU, eMIOS and external signals)

- Allows selection of interrupt requests between external pins and DSPI
- Error correction status module (ECSM)
 - Configurable error-correcting codes (ECC) reporting
 - Single-bit error correction reporting
- On-chip flash memory
 - Up to 1.5 MB flash memory, accessed via a 64-bit wide bus interface
 - 16 KB shadow block
 - Fetch Accelerator
 - Provide single cycle flash access at 80 MHz
 - Quadruple 128-bit wide prefetch/burst buffers
 - Prefetch buffers can be configured to prefetch code or data or both
 - Censorship protection scheme to prevent flash content visibility
 - Flash divided into two independent arrays, allowing reading from one array while erasing/programming the other array (used for EEPROM emulation)
 - Memory block:
 - For MPC5634M: 18 blocks (4 × 16 KB, 2 × 32 KB, 2 × 64 KB, 10 × 128 KB)
 - For MPC5633M: 14 blocks (4 × 16 KB, 2 × 32 KB, 2 × 64 KB, 6 × 128 KB)
 - For MPC5632M: 12 blocks (4 × 16 KB, 2 × 32 KB, 2 × 64 KB, 4 × 128 KB)
 - Hardware programming state machine
- On-chip static RAM
 - For MPC5634M: 94 KB general purpose RAM of which 32 KB are on standby power supply
 - For MPC5633M: 64 KB general purpose RAM of which 32 KB are on standby power supply
 - For MPC5632M: 48 KB general purpose RAM of which 24 KB are on standby power supply
- Boot assist module (BAM)
 - Enables and manages the transition of MCU from reset to user code execution in the following configurations:
 - Execution from internal flash memory
 - Execution from external memory on the calibration bus
 - Download and execution of code via FlexCAN or eSCI
- Periodic interrupt timer (PIT)
 - 32-bit wide down counter with automatic reload
 - Four channels clocked by system clock
 - One channel clocked by crystal clock
 - Each channel can produce periodic software interrupt
 - Each channel can produce periodic triggers for eQADC queue triggering
 - One channel out of the five can be used as wake-up timer to wake device from low power stop mode
- System timer module (STM)
 - 32-bit up counter with 8-bit prescaler
 - Clocked from system clock
 - Four-channel timer compare hardware
 - Each channel can generate a unique interrupt request
 - Designed to address AUTOSAR task monitor function
- Software watchdog timer (SWT)
 - 32-bit timer
 - Clock by system clock or crystal clock
 - Can generate either system reset or non-maskable interrupt followed by system reset

- Enabled out of reset
- Enhanced modular I/O system (eMIOS)
 - 16 timer channels (up to 14 channels in 144 LQFP)
 - 24-bit timer resolution
 - Supports a subset of the timer modes found in eMIOS on MPC5554
 - 3 selectable time bases plus shared time or angle counter bus from eTPU2
 - DMA and interrupt request support
 - Motor control capability
- Second-generation enhanced time processor unit (eTPU2)
 - Object-code compatible with eTPU—no changes are required to hardware or software if only eTPU features are used
 - Intelligent co-processor designed for timing control
 - High level tools, assembler and compiler available
 - 32 channels (each channel has dedicated I/O pin in all packages)
 - 24-bit timer resolution
 - 14 KB code memory and 3 KB data memory
 - Double match and capture on all channels
 - Angle clock hardware support
 - Shared time or angle counter bus with eMIOS
 - DMA and interrupt request support
 - Nexus Class 1 debug support
 - eTPU2 enhancements
 - Counters and channels can run at full system clock speed
 - Software watchdog
 - Real-time performance monitor
 - Instruction set enhancements for smaller more flexible code generation
 - Programmable channel mode for customization of channel operation
- Enhanced queued A/D converter (eQADC)
 - Two independent on-chip redundant signed digit (RSD) cyclic ADCs
 - 8-, 10-, and 12-bit resolution
 - Differential conversions
 - Targets up to 10-bit accuracy at 500 KSample/s (ADC_CLK = 7.5 MHz) and 8-bit accuracy at 1 MSample/s (ADC_CLK = 15 MHz) for differential conversions
 - Differential channels include variable gain amplifier (VGA) for improved dynamic range (×1; ×2; ×4)
 - Differential channels include programmable pull-up and pull-down resistors for biasing and sensor diagnostics (200 kΩ; 100 kΩ; low value of 5 kΩ)
 - Single-ended signal range from 0 to 5 V
 - Sample times of 2 (default), 8, 64 or 128 ADC clock cycles
 - Provides time stamp information when requested
 - Parallel interface to eQADC command FIFOs (CFIFOs) and result FIFOs (RFIFOs)
 - Supports both right-justified unsigned and signed formats for conversion results
 - Temperature sensor to enable measurement of die temperature
 - Ability to measure all power supply pins directly
 - Automatic application of ADC calibration constants
 - Provision of reference voltages (25% VREF and 75% VREF) for ADC calibration purposes
 - Up to 34¹ input channels available to the two on-chip ADCs

- Four pairs of differential analog input channels
- Full duplex synchronous serial interface to an external device
 - Has a free-running clock for use by the external device
 - Supports a 26-bit message length
 - Transmits a null message when there are no triggered CFIFOs with commands bound for external CBuffers, or when there are triggered CFIFOs with commands bound for external CBuffers but the external CBuffers are full
- Parallel Side Interface to communicate with an on-chip companion module
- Zero jitter triggering for queue 0. (Queue 0 trigger causes current conversion to be aborted and the queued conversions in the CBUFFER to be bypassed. Delay from Trigger to start of conversion is 13 system clocks + 1 ADC clock.)
- eQADC Result Streaming. Generation of a continuous stream of ADC conversion results from a single eQADC command word. Controlled by two different trigger signals; one to define the rate at which results are generated and the other to define the beginning and ending of the stream. Used to digitize waveforms during specific time/angle windows, e.g., engine knock sensor sampling.
- Angular Decimation. The ability of the eQADC to sample an analog waveform in the time domain, perform Finite Impulse Response (FIR) or Infinite Impulse Response (IIR) filtering also in the time domain, but to down sample the results in the angle domain. Resulting in a time domain filtered result at a given engine angle.
- Priority Based CFIFOs
 - Supports six CFIFOs with fixed priority. The lower the CFIFO number, the higher its priority. When commands of distinct CFIFOs are bound for the same CBuffer, the higher priority CFIFO is always served first.
 - Supports software and several hardware trigger modes to arm a particular CFIFO
 - Generates interrupt when command coherency is not achieved
- External Hardware Triggers
 - Supports rising edge, falling edge, high level and low level triggers
 - Supports configurable digital filter
- Supports four external 8-to-1 muxes which can expand the input channel number from 34¹ to 59
- Two deserial serial peripheral interface modules (DSPI)
 - SPI
 - Full duplex communication ports with interrupt and DMA request support
 - Supports all functional modes from QSPI subblock of QSMCM (MPC5xx family)
 - Support for queues in RAM
 - 6 chip selects, expandable to 64 with external demultiplexers
 - Programmable frame size, baud rate, clock delay and clock phase on a per frame basis
 - Modified SPI mode for interfacing to peripherals with longer setup time requirements
 - LVDS option for output clock and data to allow higher speed communication
 - Deserial serial interface (DSI)
 - Pin reduction by hardware serialization and deserialization of eTPU, eMIOS channels and GPIO
 - 32 bits per DSPI module
 - Triggered transfer control and change in data transfer control (for reduced EMI)
 - Compatible with Microsecond Channel Version 1.0 downstream
- Two enhanced serial communication interface (eSCI) modules
 - UART mode provides NRZ format and half or full duplex interface
 - eSCI bit rate up to 1 Mbps

1. 176-pin and 208-pin packages have 34 input channels; 144-pin package has 32.
 1. 176-pin and 208-ball packages.

- Advanced error detection, and optional parity generation and detection
- Word length programmable as 8, 9, 12 or 13 bits
- Separately enabled transmitter and receiver
- LIN support
- DMA support
- Interrupt request support
- Programmable clock source: system clock or oscillator clock
- Support Microsecond Channel (Timed Serial Bus - TSB) upstream Version 1.0
- Two FlexCAN
 - One with 32 message buffers; the second with 64 message buffers
 - Full implementation of the CAN protocol specification, Version 2.0B
 - Based on and including all existing features of the Freescale TouCAN module
 - Programmable acceptance filters
 - Short latency time for high priority transmit messages
 - Arbitration scheme according to message ID or message buffer number
 - Listen only mode capabilities
 - Programmable clock source: system clock or oscillator clock
 - Message buffers may be configured as mailboxes or as FIFO
- Nexus port controller (NPC)
 - Per IEEE-ISTO 5001-2003
 - Real time development support for Power Architecture core and eTPU engine through Nexus class 2/1
 - Read and write access (Nexus class 3 feature that is supported on this device)
 - Run-time access of entire memory map
 - Calibration
 - Support for data value breakpoints / watchpoints
 - Run-time access of entire memory map
 - Calibration
 - Table constants calibrated using MMU and internal and external RAM
 - Scalar constants calibrated using cache line locking
 - Configured via the IEEE 1149.1 (JTAG) port
- IEEE 1149.1 JTAG controller (JTAGC)
 - IEEE 1149.1-2001 Test Access Port (TAP) interface
 - 5-bit instruction register that supports IEEE 1149.1-2001 defined instructions
 - 5-bit instruction register that supports additional public instructions
 - Three test data registers: a bypass register, a boundary scan register, and a device identification register
 - Censorship disable register. By writing the 64-bit serial boot password to this register, Censorship may be disabled until the next reset
 - TAP controller state machine that controls the operation of the data registers, instruction register and associated circuitry
- On-chip Voltage Regulator for single 5 V supply operation
 - On-chip regulator 5 V to 3.3 V for internal supplies
 - On-chip regulator controller 5 V to 1.2 V (with external bypass transistor) for core logic
- Low-power modes
 - SLOW Mode. Allows device to be run at very low speed (approximately 1 MHz), with modules (including the PLL) selectively disabled in software
 - STOP Mode. System clock stopped to all modules including the CPU. Wake-up timer used to restart the system clock after a predetermined time

1 Introduction

1.1 Document overview

This document provides an overview and describes the features of the MPC5634M series of microcontroller units (MCUs). For functional characteristics, refer to the device reference manual. Electrical specifications and package mechanical drawings are included in this device data sheet. Pin assignments can be found in both the reference manual and data sheet.

1.2 Description

These 32-bit automotive microcontrollers are a family of system-on-chip (SoC) devices that contain all the features of the MPC5500 family and many new features coupled with high performance 90 nm CMOS technology to provide substantial reduction of cost per feature and significant performance improvement. The advanced and cost-efficient host processor core of this automotive controller family is built on Power Architecture[®] technology. This family contains enhancements that improve the architecture's fit in embedded applications, includes additional instruction support for digital signal processing (DSP), integrates technologies—such as an enhanced time processor unit, enhanced queued analog-to-digital converter, Controller Area Network, and an enhanced modular input-output system—that are important for today's lower-end powertrain applications. This device family is a completely compatible extension to Freescale's MPC5500 family. The device has a single level of memory hierarchy consisting of up to 94 KB on-chip SRAM and up to 1.5 MB of internal flash memory. The device also has an external bus interface (EBI) for 'calibration'. This external bus interface has been designed to support most of the standard memories used with the MPC5xx and MPC55xx families.

2 Overview

This document provides electrical specifications, pin assignments, and package diagrams for the MPC5634M series of microcontroller units (MCUs). For functional characteristics, refer to the MPC5634M *Microcontroller Reference Manual*.

The MPC5634M series microcontrollers are system-on-chip devices that are built on Power Architecture[®] technology and:

- Are 100% user-mode compatible with the Power Architecture instruction set
- Contain enhancements that improve the architecture's fit in embedded applications
- Include additional instruction support for digital signal processing (DSP)
- Integrate technologies such as an enhanced time processor unit, enhanced queued analog-to-digital converter, Controller Area Network, and an enhanced modular input-output system

2.1 Device comparison

2.2 MPC5634M feature details

2.2.1 e200z335 core

The e200z335 processor utilizes a four stage pipeline for instruction execution. The Instruction Fetch (stage 1), Instruction Decode/Register file Read/Effective Address Calculation (stage 2), Execute/Memory Access (stage 3), and Register Writeback (stage 4) stages operate in an overlapped fashion, allowing single clock instruction execution for most instructions.

The integer execution unit consists of a 32-bit Arithmetic Unit (AU), a Logic Unit (LU), a 32-bit Barrel shifter (Shifter), a Mask-Insertion Unit (MIU), a Condition Register manipulation Unit (CRU), a Count-Leading-Zeros unit (CLZ), a 32×32 Hardware Multiplier array, result feed-forward hardware, and support hardware for division.

Most arithmetic and logical operations are executed in a single cycle with the exception of the divide instructions. A Count-Leading-Zeros unit operates in a single clock cycle. The Instruction Unit contains a PC incrementer and a dedicated

Overview

Branch Address adder to minimize delays during change of flow operations. Sequential prefetching is performed to ensure a supply of instructions into the execution pipeline. Branch target prefetching is performed to accelerate taken branches. Prefetched instructions are placed into an instruction buffer capable of holding six instructions.

Branches can also be decoded at the instruction buffer and branch target addresses calculated prior to the branch reaching the instruction decode stage, allowing the branch target to be prefetched early. When a branch is detected at the instruction buffer, a prediction may be made on whether the branch is taken or not. If the branch is predicted to be taken, a target fetch is initiated and its target instructions are placed in the instruction buffer following the branch instruction. Many branches take zero cycle to execute by using branch folding. Branches are folded out from the instruction execution pipe whenever possible. These include unconditional branches and conditional branches with condition codes that can be resolved early.

Conditional branches which are not taken and not folded execute in a single clock. Branches with successful target prefetching which are not folded have an effective execution time of one clock. All other taken branches have an execution time of two clocks. Memory load and store operations are provided for byte, halfword, and word (32-bit) data with automatic zero or sign extension of byte and halfword load data as well as optional byte reversal of data. These instructions can be pipelined to allow effective single cycle throughput. Load and store multiple word instructions allow low overhead context save and restore operations. The load/store unit contains a dedicated effective address adder to allow effective address generation to be optimized. Also, a load-to-use dependency does not incur any pipeline bubbles for most cases.

The Condition Register unit supports the condition register (CR) and condition register operations defined by the Power Architecture. The condition register consists of eight 4-bit fields that reflect the results of certain operations, such as move, integer and floating-point compare, arithmetic, and logical instructions, and provide a mechanism for testing and branching. Vectored and autovectored interrupts are supported by the CPU. Vectored interrupt support is provided to allow multiple interrupt sources to have unique interrupt handlers invoked with no software overhead.

The hardware floating-point unit utilizes the IEEE-754 single-precision floating-point format and supports single-precision floating-point operations in a pipelined fashion. The general purpose register file is used for source and destination operands, thus there is a unified storage model for single-precision floating-point data types of 32 bits and the normal integer type. Single-cycle floating-point add, subtract, multiply, compare, and conversion operations are provided. Divide instructions are multi-cycle and are not pipelined.

The Signal Processing Extension (SPE) Auxiliary Processing Unit (APU) provides hardware SIMD operations and supports a full complement of dual integer arithmetic operation including Multiply Accumulate (MAC) and dual integer multiply (MUL) in a pipelined fashion. The general purpose register file is enhanced such that all 32 of the GPRs are extended to 64 bits wide and are used for source and destination operands, thus there is a unified storage model for 32×32 MAC operations which generate greater than 32-bit results.

The majority of both scalar and vector operations (including MAC and MUL) are executed in a single clock cycle. Both scalar and vector divides take multiple clocks. The SPE APU also provides extended load and store operations to support the transfer of data to and from the extended 64-bit GPRs. This SPE APU is fully binary compatible with e200z6 SPE APU used in MPC5554 and MPC5553.

The CPU includes support for Variable Length Encoding (VLE) instruction enhancements. This enables the classic Power Architecture instruction set to be represented by a modified instruction set made up from a mixture of 16- and 32-bit instructions. This results in a significantly smaller code size footprint without noticeably affecting performance. The Power Architecture instruction set and VLE instruction set are available concurrently. Regions of the memory map are designated as PPC or VLE using an additional configuration bit in each of Table Look-aside Buffers (TLB) entries in the MMU.

The CPU core is enhanced by the addition of two additional interrupt sources; Non-Maskable Interrupt and Critical Interrupt. These two sources are routed directly from package pins, via edge detection logic in the SIU to the CPU, bypassing completely the Interrupt Controller. Once the edge detection logic is programmed, it cannot be disabled, except by reset. The non-maskable interrupt is, as the name suggests, completely un-maskable and when asserted will always result in the immediate execution of the respective interrupt service routine. The non-maskable interrupt is not guaranteed to be recoverable. The Critical Interrupt is very similar to the non-maskable interrupt, but it can be masked by other exceptional interrupts in the CPU and is guaranteed to be recoverable (code execution may be resumed from where it stopped).

The CPU core has an additional ‘Wait for Interrupt’ instruction that is used in conjunction with low power STOP mode. When Low Power Stop mode is selected, this instruction is executed to allow the system clock to be stopped. An external interrupt source or the system wake-up timer is used to restart the system clock and allow the CPU to service the interrupt.

2.2.2 Crossbar

The XBAR multi-port crossbar switch supports simultaneous connections between three master ports and four slave ports. The crossbar supports a 32-bit address bus width and a 64-bit data bus width.

The crossbar allows three concurrent transactions to occur from the master ports to any slave port; but each master must access a different slave. If a slave port is simultaneously requested by more than one master port, arbitration logic selects the higher priority master and grants it ownership of the slave port. All other masters requesting that slave port are stalled until the higher priority master completes its transactions. Requesting masters are treated with equal priority and are granted access to a slave port in round-robin fashion, based upon the ID of the last master to be granted access. The crossbar provides the following features:

- 3 master ports:
 - e200z335 core complex Instruction port
 - e200z335 core complex Load/Store port
 - eDMA
- 4 slave ports
 - FLASH
 - calibration bus
 - SRAM
 - Peripheral bridge A/B (eTPU2, eMIOS, SIU, DSPI, eSCI, FlexCAN, eQADC, BAM, decimation filter, PIT, STM and SWT)
- 32-bit internal address, 64-bit internal data paths

2.2.3 eDMA

The enhanced direct memory access (eDMA) controller is a second-generation module capable of performing complex data movements via 32 programmable channels, with minimal intervention from the host processor. The hardware micro architecture includes a DMA engine which performs source and destination address calculations, and the actual data movement operations, along with an SRAM-based memory containing the transfer control descriptors (TCD) for the channels. This implementation is utilized to minimize the overall block size. The eDMA module provides the following features:

- All data movement via dual-address transfers: read from source, write to destination
- Programmable source and destination addresses, transfer size, plus support for enhanced addressing modes
- Transfer control descriptor organized to support two-deep, nested transfer operations
- An inner data transfer loop defined by a “minor” byte transfer count
- An outer data transfer loop defined by a “major” iteration count
- Channel activation via one of three methods:
 - Explicit software initiation
 - Initiation via a channel-to-channel linking mechanism for continuous transfers
 - Peripheral-paced hardware requests (one per channel)
- Support for fixed-priority and round-robin channel arbitration
- Channel completion reported via optional interrupt requests
- 1 interrupt per channel, optionally asserted at completion of major iteration count
- Error termination interrupts are optionally enabled
- Support for scatter/gather DMA processing

- Channel transfers can be suspended by a higher priority channel

2.2.4 Interrupt controller

The INTC (interrupt controller) provides priority-based preemptive scheduling of interrupt requests, suitable for statically scheduled hard real-time systems. The INTC allows interrupt request servicing from up to 191 peripheral interrupt request sources, plus 165 sources reserved for compatibility with other family members).

For high priority interrupt requests, the time from the assertion of the interrupt request from the peripheral to when the processor is executing the interrupt service routine (ISR) has been minimized. The INTC provides a unique vector for each interrupt request source for quick determination of which ISR needs to be executed. It also provides an ample number of priorities so that lower priority ISRs do not delay the execution of higher priority ISRs. To allow the appropriate priorities for each source of interrupt request, the priority of each interrupt request is software configurable.

When multiple tasks share a resource, coherent accesses to that resource need to be supported. The INTC supports the priority ceiling protocol for coherent accesses. By providing a modifiable priority mask, the priority can be raised temporarily so that all tasks which share the resource can not preempt each other.

Multiple processors can assert interrupt requests to each other through software settable interrupt requests. These same software settable interrupt requests also can be used to break the work involved in servicing an interrupt request into a high priority portion and a low priority portion. The high priority portion is initiated by a peripheral interrupt request, but then the ISR asserts a software settable interrupt request to finish the servicing in a lower priority ISR. Therefore these software settable interrupt requests can be used instead of the peripheral ISR scheduling a task through the RTOS.

The INTC provides the following features:

- 356 peripheral interrupt request sources
- 8 software settable interrupt request sources
- 9-bit vector addresses
- Unique vector for each interrupt request source
- Hardware connection to processor or read from register
- Each interrupt source can be programmed to one of 16 priorities
- Preemptive prioritized interrupt requests to processor
- ISR at a higher priority preempts executing ISRs or tasks at lower priorities
- Automatic pushing or popping of preempted priority to or from a LIFO
- Ability to modify the ISR or task priority to implement the priority ceiling protocol for accessing shared resources
- Low latency—three clocks from receipt of interrupt request from peripheral to interrupt request to processor

This device also includes a non-maskable interrupt (NMI) pin that bypasses the INTC and multiplexing logic.

2.2.5 FMPLL

The FMPLL allows the user to generate high speed system clocks from a 4 MHz to 20 MHz crystal oscillator or external clock generator. Further, the FMPLL supports programmable frequency modulation of the system clock. The PLL multiplication factor, output clock divider ratio are all software configurable. The PLL has the following major features:

- Input clock frequency from 4 MHz to 20 MHz
- Voltage controlled oscillator (VCO) range from 256 MHz to 512 MHz, resulting in system clock frequencies from 16 MHz to 80 MHz with granularity of 4 MHz or better
- Reduced frequency divider (RFD) for reduced frequency operation without forcing the PLL to relock
- 3 modes of operation
 - Bypass mode with PLL off
 - Bypass mode with PLL running (default mode out of reset)
 - PLL normal mode

- Each of the three modes may be run with a crystal oscillator or an external clock reference
- Programmable frequency modulation
 - Modulation enabled/disabled through software
 - Triangle wave modulation up to 100 kHz modulation frequency
 - Programmable modulation depth (0% to 2% modulation depth)
 - Programmable modulation frequency dependent on reference frequency
- Lock detect circuitry reports when the PLL has achieved frequency lock and continuously monitors lock status to report loss of lock conditions
- Clock Quality Module
 - detects the quality of the crystal clock and cause interrupt request or system reset if error is detected
 - detects the quality of the PLL output clock. If an error is detected, causes a system reset or switches the system clock to the crystal clock and causes an interrupt request
- Programmable interrupt request or system reset on loss of lock

2.2.6 Calibration EBI

The Calibration EBI controls data transfer across the crossbar switch to/from memories or peripherals attached to the VertiCal connector in the calibration address space. The Calibration EBI is only available in the VertiCal Calibration System. The Calibration EBI includes a memory controller that generates interface signals to support a variety of external memories. The Calibration EBI memory controller supports legacy flash, SRAM, and asynchronous memories. In addition, the calibration EBI supports up to three regions via chip selects (two chip selects are multiplexed with two address bits), along with programmed region-specific attributes. The calibration EBI supports the following features:

- 22-bit address bus (two most significant signals multiplexed with two chip selects)
- 16-bit data bus
- Multiplexed mode with addresses and data signals present on the data lines

NOTE

The calibration EBI must be configured in multiplexed mode when the extended Nexus trace is used on the VertiCal Calibration System. This is because Nexus signals and address lines of the calibration bus share the same balls in the calibration package.

- Memory controller with support for various memory types:
 - Asynchronous/legacy flash and SRAM
 - Most standard memories used with the MPC5xx or MPC55xx family
- Bus monitor
 - User selectable
 - Programmable timeout period (with 8 external bus clock resolution)
- Configurable wait states (via chip selects)
- 3 chip-select (Cal_ \overline{CS} [0], Cal_ \overline{CS} [2:3]) signals (Multiplexed with 2 most significant address signals)
- 2 write/byte enable (WE[0:1]/BE[0:1]) signals
- Configurable bus speed modes
 - system frequency
 - 1/2 of system frequency
 - 1/4 of system frequency
- Optional automatic CLKOUT gating to save power and reduce EMI
- Compatible with MPC5xx external bus (with some limitations)
- Selectable drive strengths; 10 pF, 20 pF, 30 pF, 50 pF

2.2.7 SIU

The MPC5634MSIU controls MCU reset configuration, pad configuration, external interrupt, general purpose I/O (GPIO), internal peripheral multiplexing, and the system reset operation. The reset configuration block contains the external pin boot configuration logic. The pad configuration block controls the static electrical characteristics of I/O pins. The GPIO block provides uniform and discrete input/output control of the I/O pins of the MCU. The reset controller performs reset monitoring of internal and external reset sources, and drives the $\overline{\text{RSTOUT}}$ pin. Communication between the SIU and the e200z335 CPU core is via the crossbar switch. The SIU provides the following features:

- System configuration
 - MCU reset configuration via external pins
 - Pad configuration control for each pad
 - Pad configuration control for virtual I/O via DSPI serialization
- System reset monitoring and generation
 - Power-on reset support
 - Reset status register provides last reset source to software
 - Glitch detection on reset input
 - Software controlled reset assertion
- External interrupt
 - 11 interrupt requests
 - Rising or falling edge event detection
 - Programmable digital filter for glitch rejection
 - Critical Interrupt request
 - Non-Maskable Interrupt request
- GPIO
 - GPIO function on 80 I/O pins
 - Virtual GPIO on 64 I/O pins via DSPI serialization (requires external deserialization device)
 - Dedicated input and output registers for setting each GPIO and Virtual GPIO pin
- Internal multiplexing
 - Allows serial and parallel chaining of DSPIs
 - Allows flexible selection of eQADC trigger inputs
 - Allows selection of interrupt requests between external pins and DSPI

2.2.8 ECSM

The error correction status module provides status information regarding platform memory errors reported by error-correcting codes.

2.2.9 Flash

Devices in the MPC5634M family provide up to 1.5 MB of programmable, non-volatile, flash memory. The non-volatile memory (NVM) can be used for instruction and/or data storage. The flash module includes a Fetch Accelerator, that optimizes the performance of the flash array to match the CPU architecture and provides single cycle random access to the flash @ 80 MHz. The flash module interfaces the system bus to a dedicated flash memory array controller. For CPU 'loads', DMA transfers and CPU instruction fetch, it supports a 64-bit data bus width at the system bus port, and a 128-bit read data interface to flash memory. The module contains a four-entry, 128-bit prefetch buffer and a prefetch controller which prefetches sequential lines of data from the flash array into the buffer. Prefetch buffer hits allow no-wait responses. Normal flash array accesses are registered and are forwarded to the system bus on the following cycle, incurring three wait-states. Prefetch operations may be automatically controlled, and are restricted to instruction fetch.

The flash memory provides the following features:

- Supports a 64-bit data bus for instruction fetch, CPU loads and DMA access. Byte, halfword, word and doubleword reads are supported. Only aligned word and doubleword writes are supported.
- Fetch Accelerator
 - Architected to optimize the performance of the flash with the CPU to provide single cycle random access to the flash up to 80 MHz system clock speed
 - Configurable read buffering and line prefetch support
 - Four line read buffers (128 bits wide) and a prefetch controller
- Hardware and software configurable read and write access protections on a per-master basis
- Interface to the flash array controller is pipelined with a depth of one, allowing overlapped accesses to proceed in parallel for interleaved or pipelined flash array designs
- Configurable access timing allowing use in a wide range of system frequencies
- Multiple-mapping support and mapping-based block access timing (0-31 additional cycles) allowing use for emulation of other memory types
- Software programmable block program/erase restriction control
- Erase of selected block(s)
- Read page size of 128 bits (four words)
- ECC with single-bit correction, double-bit detection
- Program page size of 64 bits (two words)
- ECC single-bit error corrections are visible to software
- Minimum program size is two consecutive 32-bit words, aligned on a 0-modulo-8 byte address, due to ECC
- Embedded hardware program and erase algorithm
- Erase suspend
- Shadow information stored in non-volatile shadow block
- Independent program/erase of the shadow block

2.2.10 SRAM

The MPC5634M SRAM module provides a general-purpose up to 94 KB memory block. The SRAM controller includes these features:

- Supports read/write accesses mapped to the SRAM memory from any master
- 32 KB or 24 KB block powered by separate supply for standby operation
- Byte, halfword, word and doubleword addressable
- ECC performs single-bit correction, double-bit detection on 32-bit data element

2.2.11 BAM

The BAM (Boot Assist Module) is a block of read-only memory that is programmed once by Freescale and is identical for all MPC5634M MCUs. The BAM program is executed every time the MCU is powered-on or reset in normal mode. The BAM supports different modes of booting. They are:

- Booting from internal flash memory
- Serial boot loading (A program is downloaded into RAM via eSCI or the FlexCAN and then executed)
- Booting from external memory on calibration bus

The BAM also reads the reset configuration half word (RCHW) from internal flash memory and configures the MPC5634M hardware accordingly. The BAM provides the following features:

- Sets up MMU to cover all resources and mapping all physical address to logical addresses with minimum address translation

Overview

- Sets up the MMU to allow user boot code to execute as either Power Architecture code (default) or as Freescale VLE code
- Detection of user boot code
- Automatic switch to serial boot mode if internal flash is blank or invalid
- Supports user programmable 64-bit password protection for serial boot mode
- Supports serial bootloading via FlexCAN bus and eSCI using Freescale protocol
- Supports serial bootloading via FlexCAN bus and eSCI with auto baud rate sensing
- Supports serial bootloading of either Power Architecture code (default) or Freescale VLE code
- Supports booting from calibration bus interface
- Supports censorship protection for internal flash memory
- Provides an option to enable the core watchdog timer
- Provides an option to disable the software watchdog timer

2.2.12 eMIOS

The eMIOS (Enhanced Modular Input Output System) module provides the functionality to generate or measure time events. The channels on this module provide a range of operating modes including the capability to perform dual input capture or dual output compare as well as PWM output.

The eMIOS provides the following features:

- 16 channels (24-bit timer resolution)
- For compatibility with other family members selected channels and timebases are implemented:
 - Channels 0 to 6, 8 to 15, and 23
 - Timebases A, B and C
- Channels 1, 3, 5 and 6 support modes:
 - General Purpose Input/Output (GPIO)
 - Single Action Input Capture (SAIC)
 - Single Action Output Compare (SAOC)
- Channels 2, 4, 11 and 13 support all the modes above plus:
 - Output Pulse Width Modulation Buffered (OPWMB)
- Channels 0, 8, 9, 10, 12, 14, 15, 23 support all the modes above plus:
 - Input Period Measurement (IPM)
 - Input Pulse Width Measurement (IPWM)
 - Double Action Output Compare (set flag on both matches) (DAOC)
 - Modulus Counter Buffered (MCB)
 - Output Pulse Width and Frequency Modulation Buffered (OPWFMB)
- Three 24-bit wide counter buses
 - Counter bus A can be driven by channel 23 or by the eTPU2 and all channels can use it as a reference
 - Counter bus B is driven by channel 0 and channels 0 to 6 can use it as a reference
 - Counter bus C is driven by channel 8 and channels 8 to 15 can use it as a reference
- Shared time bases with the eTPU2 through the counter buses
- Synchronization among internal and external time bases

2.2.13 eTPU2

The eTPU2 is an enhanced co-processor designed for timing control. Operating in parallel with the host CPU, eTPU2 processes instructions and real-time input events, performs output waveform generation, and accesses shared data without host

intervention. Consequently, for each timer event, the host CPU setup and service times are minimized or eliminated. A powerful timer subsystem is formed by combining the eTPU2 with its own instruction and data RAM. High-level assembler/compiler and documentation allows customers to develop their own functions on the eTPU2.

The eTPU2 includes these distinctive features:

- The Timer Counter (TCR1), channel logic and digital filters (both channel and the external timer clock input [TCRCLK]) now have an option to run at full system clock speed or system clock / 2.
- Channels support unordered transitions: transition 2 can now be detected before transition 1. Related to this enhancement, the transition detection latches (TDL1 and TDL2) can now be independently negated by microcode.
- A new User Programmable Channel Mode has been added: the blocking, enabling, service request and capture characteristics of this channel mode can be programmed via microcode.
- Microinstructions now provide an option to issue Interrupt and Data Transfer requests selected by channel. They can also be requested simultaneously at the same instruction.
- Channel Flags 0 and 1 can now be tested for branching, in addition to selecting the entry point.
- Channel digital filters can be bypassed.
- The Timer Counter (TCR1), channel logic and digital filters (both channel and the external timer clock input [TCRCLK]) now have an option to run at full system clock speed or system clock / 2.
- Channels support unordered transitions: transition 2 can now be detected before transition 1. Related to this enhancement, the transition detection latches (TDL1 and TDL2) can now be independently negated by microcode.
- A new User Programmable Channel Mode has been added: the blocking, enabling, service request and capture characteristics of this channel mode can be programmed via microcode.
- Microinstructions now provide an option to issue Interrupt and Data Transfer requests selected by channel. They can also be requested simultaneously at the same instruction.
- Channel Flags 0 and 1 can now be tested for branching, in addition to selecting the entry point.
- Channel digital filters can be bypassed.
- 32 channels, each channel is associated with one input and one output signal
 - Enhanced input digital filters on the input pins for improved noise immunity.
 - Identical, orthogonal channels: each channel can perform any time function. Each time function can be assigned to more than one channel at a given time, so each signal can have any functionality.
 - Each channel has an event mechanism which supports single and double action functionality in various combinations. It includes two 24-bit capture registers, two 24-bit match registers, 24-bit greater-equal and equal-only comparators
 - Input and output signal states visible from the host
- 2 independent 24-bit time bases for channel synchronization:
 - First time base clocked by system clock with programmable prescale division from 2 to 512 (in steps of 2), or by output of second time base prescaler
 - Second time base counter can work as a continuous angle counter, enabling angle based applications to match angle instead of time
 - Both time bases can be exported to the eMIOS timer module
 - Both time bases visible from the host
- Event-triggered microengine:
 - Fixed-length instruction execution in two-system-clock microcycle
 - 14 KB of code memory (SCM)
 - 3 KB of parameter (data) RAM (SPRAM)
 - Parallel execution of data memory, ALU, channel control and flow control sub-instructions in selected combinations

Overview

- 32-bit microengine registers and 24-bit wide ALU, with 1 microcycle addition and subtraction, absolute value, bitwise logical operations on 24-bit, 16-bit, or byte operands, single-bit manipulation, shift operations, sign extension and conditional execution
- Additional 24-bit Multiply/MAC/Divide unit which supports all signed/unsigned Multiply/MAC combinations, and unsigned 24-bit divide. The MAC/Divide unit works in parallel with the regular microcode commands
- Resource sharing features support channel use of common channel registers, memory and microengine time:
 - Hardware scheduler works as a “task management” unit, dispatching event service routines by predefined, host-configured priority
 - Automatic channel context switch when a “task switch” occurs, i.e., one function thread ends and another begins to service a request from other channel: channel-specific registers, flags and parameter base address are automatically loaded for the next serviced channel
 - SPRAM shared between host CPU and eTPU2, supporting communication either between channels and host or inter-channel
 - Dual-parameter coherency hardware support allows atomic access to two parameters by host
- Test and development support features:
 - Nexus Class 1 debug, supporting single-step execution, arbitrary microinstruction execution, hardware breakpoints and watchpoints on several conditions
 - Software breakpoints
 - SCM continuous signature-check built-in self test (MISC — multiple input signature calculator), runs concurrently with eTPU2 normal operation
- System enhancements
 - Software watchdog with programmable timeout
 - Real-time performance information
- Channel enhancements
 - Channels 1 and 2 can optionally drive angle clock hardware
- Programming enhancements
 - Engine relative addressing mode

2.2.14 eQADC

The enhanced queued analog to digital converter (eQADC) block provides accurate and fast conversions for a wide range of applications. The eQADC provides a parallel interface to two on-chip analog to digital converters (ADC), and a single master to single slave serial interface to an off-chip external device. Both on-chip ADCs have access to all the analog channels.

The eQADC prioritizes and transfers commands from six command conversion command ‘queues’ to the on-chip ADCs or to the external device. The block can also receive data from the on-chip ADCs or from an off-chip external device into the six result queues, in parallel, independently of the command queues. The six command queues are prioritized with Queue_0 having the highest priority and Queue_5 the lowest. Queue_0 also has the added ability to bypass all buffering and queuing and abort a currently running conversion on either ADC and start a Queue_0 conversion. This means that Queue_0 will always have a deterministic time from trigger to start of conversion, irrespective of what tasks the ADCs were performing when the trigger occurred. The eQADC supports software and external hardware triggers from other blocks to initiate transfers of commands from the queues to the on-chip ADCs or to the external device. It also monitors the fullness of command queues and result queues, and accordingly generates DMA or interrupt requests to control data movement between the queues and the system memory, which is external to the eQADC.

The ADCs also support features designed to allow the direct connection of high impedance acoustic sensors that might be used in a system for detecting engine knock. These features include differential inputs; integrated variable gain amplifiers for increasing the dynamic range; programmable pull-up and pull-down resistors for biasing and sensor diagnostics.

The eQADC also integrates a programmable decimation filter capable of taking in ADC conversion results at a high rate, passing them through a hardware low pass filter, then down-sampling the output of the filter and feeding the lower sample rate

results to the result FIFOs. This allows the ADCs to sample the sensor at a rate high enough to avoid aliasing of out-of-band noise; while providing a reduced sample rate output to minimize the amount DSP processing bandwidth required to fully process the digitized waveform.

The eQADC provides the following features:

- Dual on-chip ADCs
 - $2 \times$ 12-bit ADC resolution
 - Programmable resolution for increased conversion speed (12 bit, 10 bit, 8 bit)
 - 12-bit conversion time – 1 μ s (1M sample/sec)
 - 10-bit conversion time – 867 ns (1.2M sample/second)
 - 8-bit conversion time – 733 ns (1.4M sample/second)
 - Up to 10-bit accuracy at 500 KSample/s and 9-bit accuracy at 1 MSample/s
 - Differential conversions
 - Single-ended signal range from 0 to 5 V
 - Variable gain amplifiers on differential inputs ($\times 1$, $\times 2$, $\times 4$)
 - Sample times of 2 (default), 8, 64 or 128 ADC clock cycles
 - Provides time stamp information when requested
 - Parallel interface to eQADC CFIFOs and RFIFOs
 - Supports both right-justified unsigned and signed formats for conversion results
- Up to 34¹ input channels (accessible by both ADCs)
- 23 additional internal channels for measuring control and monitoring voltages inside the device
 - Including Core voltage, I/O voltage, LVI voltages, etc.
- An internal bandgap reference to allow absolute voltage measurements
- 4 pairs of differential analog input channels
 - Programmable pull-up/pull-down resistors on each differential input for biasing and sensor diagnostic (200 k Ω , 100 k Ω , 5 k Ω)
- Silicon die temperature sensor
 - provides temperature of silicon as an analog value
 - read using an internal ADC analog channel
 - may be read with either ADC
- Decimation Filter
 - Programmable decimation factor (2 to 16)
 - Selectable IIR or FIR filter
 - Up to 4th order IIR or 8th order FIR
 - Programmable coefficients
 - Saturated or non-saturated modes
 - Programmable Rounding (Convergent; Two's Complement; Truncated)
 - Pre-fill mode to pre-condition the filter before the sample window opens
- Full duplex synchronous serial interface to an external device
 - Free-running clock for use by an external device
 - Supports a 26-bit message length
- Priority based Queues
 - Supports six Queues with fixed priority. When commands of distinct Queues are bound for the same ADC, the higher priority Queue is always served first

1. 176-pin and 208-pin packages have 34 input channels; 144-pin package has 32.

Overview

- Queue_0 can bypass all prioritization, buffering and abort current conversions to start a Queue_0 conversion a deterministic time after the queue trigger
- Streaming mode operation of Queue_0 to execute some commands several times
- Supports software and hardware trigger modes to arm a particular Queue
- Generates interrupt when command coherency is not achieved
- External hardware triggers
 - Supports rising edge, falling edge, high level and low level triggers
 - Supports configurable digital filter
- Supports four external 8-to-1 muxes which can expand the input channels to 56 channels total

2.2.15 DSPI

The deserial serial peripheral interface (DSPI) block provides a synchronous serial interface for communication between the MPC5634M MCU and external devices. The DSPI supports pin count reduction through serialization and deserialization of eTPU and eMIOS channels and memory-mapped registers. The channels and register content are transmitted using a SPI-like protocol. This SPI-like protocol is completely configurable for baud rate, polarity and phase, frame length, chip select assertion, etc. Each bit in the frame may be configured to serialize either eTPU channels, eMIOS channels or GPIO signals. The DSPI can be configured to serialize data to an external device that supports the Microsecond Channel protocol. There are two identical DSPI blocks on the MPC5634M MCU. The DSPI output pins support 5 V logic levels or Low Voltage Differential Signalling (LVDS) according to the Microsecond Channel specification.

The DSPIs have three configurations:

- Serial peripheral interface (SPI) configuration where the DSPI operates as an up to 16-bit SPI with support for queues
- Enhanced deserial serial interface (DSI) configuration where DSPI serializes up to 32 bits with three possible sources per bit
 - eTPU, eMIOS, new virtual GPIO registers as possible bit source
 - Programmable inter-frame gap in continuous mode
 - Bit source selection allows microsecond channel downstream with command or data frames up to 32 bits
 - Microsecond channel dual receiver mode
- Combined serial interface (CSI) configuration where the DSPI operates in both SPI and DSI configurations interleaving DSI frames with SPI frames, giving priority to SPI frames

For queued operations, the SPI queues reside in system memory external to the DSPI. Data transfers between the memory and the DSPI FIFOs are accomplished through the use of the eDMA controller or through host software.

The DSPI supports these SPI features:

- Full-duplex, synchronous transfers
- Selectable LVDS Pads working at 40 MHz for SOUT and SCK pins
- Master and Slave Mode
- Buffered transmit operation using the TX FIFO with parameterized depth of 4 entries
- Buffered receive operation using the RX FIFO with parameterized depth of 4 entries
- TX and RX FIFOs can be disabled individually for low-latency updates to SPI queues
- Visibility into the TX and RX FIFOs for ease of debugging
- FIFO Bypass Mode for low-latency updates to SPI queues
- Programmable transfer attributes on a per-frame basis:
 - Parameterized number of transfer attribute registers (from two to eight)
 - Serial clock with programmable polarity and phase
 - Various programmable delays:
 - PCS to SCK delay

- SCK to PCS delay
 - Delay between frames
- Programmable serial frame size of 4 to 16 bits, expandable with software control
- Continuously held chip select capability
- 6 Peripheral Chip Selects, expandable to 64 with external demultiplexer
- Deglitching support for up to 32 Peripheral Chip Selects with external demultiplexer
- DMA support for adding entries to TX FIFO and removing entries from RX FIFO:
 - TX FIFO is not full (TFFF)
 - RX FIFO is not empty (RFDF)
- 6 Interrupt conditions:
 - End of queue reached (EOQF)
 - TX FIFO is not full (TFFF)
 - Transfer of current frame complete (TCF)
 - Attempt to transmit with an empty Transmit FIFO (TFUF)
 - RX FIFO is not empty (RFDF)
 - FIFO Underrun (slave only and SPI mode, the slave is asked to transfer data when the TxFIFO is empty)
 - FIFO Overrun (serial frame received while RX FIFO is full)
- Modified transfer formats for communication with slower peripheral devices
- Continuous Serial Communications Clock (SCK)
- Power savings via support for Stop Mode
- Enhanced DSI logic to implement a 32-bit Timed Serial Bus (TSB) configuration, supporting the Microsecond Channel downstream frame format

The DSPIs also support these features unique to the DSI and CSI configurations:

- 2 sources of the serialized data:
 - eTPU_A and eMIOS output channels
 - Memory-mapped register in the DSPI
- Destinations for the deserialized data:
 - eTPU_A and eMIOS input channels
 - SIU External Interrupt Request inputs
 - Memory-mapped register in the DSPI
- Deserialized data is provided as Parallel Output signals and as bits in a memory-mapped register
- Transfer initiation conditions:
 - Continuous
 - Edge sensitive hardware trigger
 - Change in data
- Pin serialization/deserialization with interleaved SPI frames for control and diagnostics
- Continuous serial communications clock
- Support for parallel and serial chaining of up to four DSPI blocks

2.2.16 eSCI

The enhanced serial communications interface (eSCI) allows asynchronous serial communications with peripheral devices and other MCUs. It includes special support to interface to Local Interconnect Network (LIN) slave devices. The eSCI block provides the following features:

- Full-duplex operation
- Standard mark/space non-return-to-zero (NRZ) format

Overview

- 13-bit baud rate selection
- Programmable 8-bit or 9-bit, data format
- Programmable 12-bit or 13-bit data format for Timed Serial Bus (TSB) configuration to support the Microsecond Channel upstream
- Automatic parity generation
- LIN support
 - Autonomous transmission of entire frames
 - Configurable to support all revisions of the LIN standard
 - Automatic parity bit generation
 - Double stop bit after bit error
 - 10- or 13-bit break support
- Separately enabled transmitter and receiver
- Programmable transmitter output parity
- 2 receiver wake up methods:
 - Idle line wake-up
 - Address mark wake-up
- Interrupt-driven operation with flags
- Receiver framing error detection
- Hardware parity checking
- 1/16 bit-time noise detection
- DMA support for both transmit and receive data
 - Global error bit stored with receive data in system RAM to allow post processing of errors

2.2.17 FlexCAN

The MPC5634M MCU contains two controller area network (FlexCAN) blocks. The FlexCAN module is a communication controller implementing the CAN protocol according to Bosch Specification version 2.0B. The CAN protocol was designed to be used primarily as a vehicle serial data bus, meeting the specific requirements of this field: real-time processing, reliable operation in the EMI environment of a vehicle, cost-effectiveness and required bandwidth. FlexCAN module 'A' contains 64 message buffers (MB); FlexCAN module 'C' contains 32 message buffers.

The FlexCAN module provides the following features:

- Based on and including all existing features of the Freescale TouCAN module
- Full Implementation of the CAN protocol specification, Version 2.0B
 - Standard data and remote frames
 - Extended data and remote frames
 - Zero to eight bytes data length
 - Programmable bit rate up to 1 Mbit/s
- Content-related addressing
- 64 / 32 message buffers of zero to eight bytes data length
- Individual Rx Mask Register per message buffer
- Each message buffer configurable as Rx or Tx, all supporting standard and extended messages
- Includes 1056 / 544 bytes of embedded memory for message buffer storage
- Includes a 256-byte and a 128-byte memories for storing individual Rx mask registers
- Full featured Rx FIFO with storage capacity for six frames and internal pointer handling
- Powerful Rx FIFO ID filtering, capable of matching incoming IDs against 8 extended, 16 standard or 32 partial (8 bits) IDs, with individual masking capability

- Selectable backwards compatibility with previous FlexCAN versions
- Programmable clock source to the CAN Protocol Interface, either system clock or oscillator clock
- Listen only mode capability
- Programmable loop-back mode supporting self-test operation
- 3 programmable Mask Registers
- Programmable transmit-first scheme: lowest ID, lowest buffer number or highest priority
- Time Stamp based on 16-bit free-running timer
- Global network time, synchronized by a specific message
- Maskable interrupts
- Warning interrupts when the Rx and Tx Error Counters reach 96
- Independent of the transmission medium (an external transceiver is assumed)
- Multi master concept
- High immunity to EMI
- Short latency time due to an arbitration scheme for high-priority messages
- Low power mode, with programmable wake-up on bus activity

2.2.18 System timers

The system timers provide two distinct types of system timer:

- Periodic interrupts/triggers using the Periodic Interrupt Timer (PIT)
- Operating system task monitors using the System Timer Module (STM)

2.2.18.1 Periodic Interrupt Timer (PIT)

The PIT provides five independent timer channels, capable of producing periodic interrupts and periodic triggers. The PIT has no external input or output pins and is intended to be used to provide system ‘tick’ signals to the operating system, as well as periodic triggers for eQADC queues. Of the five channels in the PIT, four are clocked by the system clock, one is clocked by the crystal clock. This one channel is also referred to as Real Time Interrupt (RTI) and is used to wakeup the device from low power stop mode.

The following features are implemented in the PIT:

- 5 independent timer channels
- Each channel includes 32-bit wide down counter with automatic reload
- 4 channels clocked from system clock
- 1 channel clocked from crystal clock (wake-up timer)
- Wake-up timer remains active when System STOP mode is entered. Used to restart system clock after predefined timeout period
- Each channel can optionally generate an interrupt request or a trigger event (to trigger eQADC queues) when the timer reaches zero

2.2.18.2 System Timer Module (STM)

The System Timer Module (STM) is designed to implement the software task monitor as defined by AUTOSAR (see <http://www.autosar.org>). It consists of a single 32-bit counter, clocked by the system clock, and four independent timer comparators. These comparators produce a CPU interrupt when the timer exceeds the programmed value.

The following features are implemented in the STM:

- One 32-bit up counter with 8-bit prescaler
- Four 32-bit compare channels

Overview

- Independent interrupt source for each channel
- Counter can be stopped in debug mode

2.2.19 Software Watchdog Timer (SWT)

The Software Watchdog Timer (SWT) is a second watchdog module to complement the standard Power Architecture watchdog integrated in the CPU core. The SWT is a 32-bit modulus counter, clocked by the system clock or the crystal clock, that can provide a system reset or interrupt request when the correct software key is not written within the required time window.

The following features are implemented:

- 32-bit modulus counter
- Clocked by system clock or crystal clock
- Optional programmable watchdog window mode
- Can optionally cause system reset or interrupt request on timeout
- Reset by writing a software key to memory mapped register
- Enabled out of reset
- Configuration is protected by a software key or a write-once register

2.2.20 Debug features

2.2.20.1 Nexus port controller

The NPC (Nexus Port Controller) block provides real-time development support capabilities for the MPC5634M Power Architecture-based MCU in compliance with the IEEE-ISTO 5001-2003 standard. This development support is supplied for MCUs without requiring external address and data pins for internal visibility. The NPC block is an integration of several individual Nexus blocks that are selected to provide the development support interface for MPC5634M. The NPC block interfaces to the host processor (e200z335), eTPU, and internal buses to provide development support as per the IEEE-ISTO 5001-2003 standard. The development support provided includes program trace and run-time access to the MCUs internal memory map and access to the Power Architecture and eTPU internal registers during halt. The Nexus interface also supports a JTAG only mode using only the JTAG pins. MPC5634M in the production 144 LQFP supports a 3.3 V reduced (4-bit wide) Auxiliary port. These Nexus port pins can also be used as 5 V I/O signals to increase usable I/O count of the device. When using this Nexus port as IO, Nexus trace is still possible using VertiCal calibration. In the VertiCal calibration package, the full 12-bit Auxiliary port is available.

NOTE

In the VertiCal package, the full Nexus Auxiliary port shares balls with the addresses of the calibration bus. Therefore multiplexed address/data bus mode must be used for the calibration bus when using full width Nexus trace in VertiCal assembly.

The following features are implemented:

- 5-pin JTAG port (JCOMP, TDI, TDO, TMS, and TCK)
 - Always available in production package
 - Supports both JTAG Boundary Scan and debug modes
 - 3.3 V interface
 - Supports Nexus class 1 features
 - Supports Nexus class 3 read/write feature
- 9-pin Reduced Port interface in 144 LQFP production package
 - Alternate function as IO
 - 5 V (in GPIO or alternate function mode), 3.3 V (in Nexus mode) interface

- Auxiliary Output port
 - 1 MCKO (message clock out) pin
 - 4 MDO (message data out) pins
 - 2 $\overline{\text{MSEO}}$ (message start/end out) pins
 - 1 $\overline{\text{EVT0}}$ (event out) pin
- Auxiliary input port
 - 1 $\overline{\text{EVTI}}$ (event in) pin
- 17-pin Full Port interface in calibration package used on VertiCal boards
 - 3.3 V interface
 - Auxiliary Output port
 - 1 MCKO (message clock out) pin
 - 4 (reduced port mode) or 12 (full port mode) MDO (message data out) pins; 8 extra full port pins shared with calibration bus
 - 2 $\overline{\text{MSEO}}$ (message start/end out) pins
 - 1 $\overline{\text{EVT0}}$ (event out) pin
 - Auxiliary input port
 - 1 $\overline{\text{EVTI}}$ (event in) pin
- Host processor (e200) development support features
 - IEEE-ISTO 5001-2003 standard class 2 compliant
 - Program trace via branch trace messaging (BTM). Branch trace messaging displays program flow discontinuities (direct branches, indirect branches, exceptions, etc.), allowing the development tool to interpolate what transpires between the discontinuities. Thus, static code may be traced.
 - Watchpoint trigger enable of program trace messaging
 - Data Value Breakpoints (JTAG feature of the e200z335 core): allows CPU to be halted when the CPU writes a specific value to a memory location
 - 4 data value breakpoints
 - CPU only
 - Detects ‘equal’ and ‘not equal’
 - Byte, half word, word (naturally aligned)

NOTE

This feature is imprecise due to CPU pipelining.

- Subset of Power Architecture software debug facilities with OnCE block (Nexus class 1 features)
- eTPU development support features
 - IEEE-ISTO 5001-2003 standard class 1 compliant for the eTPU
 - Nexus based breakpoint configuration and single step support (JTAG feature of the eTPU)
- Run-time access to the on-chip memory map via the Nexus read/write access protocol. This feature supports accesses for run-time internal visibility, calibration variable acquisition, calibration constant tuning, and external rapid prototyping for powertrain automotive development systems.
- All features are independently configurable and controllable via the IEEE 1149.1 I/O port
- Power-on-reset status indication during reset via MDO[0] in disabled and reset modes

2.2.20.2 JTAG

The JTAGC (JTAG Controller) block provides the means to test chip functionality and connectivity while remaining transparent to system logic when not in test mode. Testing is performed via a boundary scan technique, as defined in the IEEE 1149.1-2001