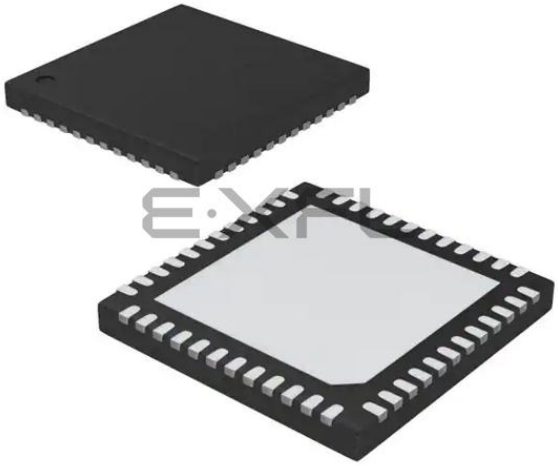
Analog Devices Inc./Maxim Integrated - MAXQ7665BATM+T Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	-
Core Size	16-Bit
Speed	8MHz
Connectivity	CANbus, LINbus, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	8
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 16
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.25V
Data Converters	A/D 8x12b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-SQFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/maxq7665batm-t

ELECTRICAL CHARACTERISTICS (continued)

(AV_{DD} = DV_{DDIO} = +5.0V, DV_{DD} = +3.3V, f_{SYSCLK} = 8MHz, V_{REFDAC} = V_{REFADC} = +5V, T_A = T_{MIN} to T_{MAX}, unless otherwise noted. Typical values are at T_A = +25°C.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
MEMORY SECTION						
Flash Memory Size		MAXQ7665A		128		KB
		MAXQ7665B		64		
		MAXQ7665C		48		
		MAXQ7665D		32		
Flash Erase/Write Endurance		DV_{DD} = +3V, at +25°C		1		MCycles
		DV_{DD} = +3V, at +85°C		100		kCycles
		DV_{DD} = +3V, at +125°C		100		kCycles
Flash Erase Timing		One sector		0.7	15	s
Flash Program Timing		Single word		11	360	µs
		Entire flash		1.5	4.5	s
Flash Data Retention Time		T_A = +125°C, single write	20			Years
		First 100,000 cycles at +25°C, then retention tested at T_A = +125°C	10			
RAM Memory Size				512		Bytes
Utility ROM Size				4096		Words
ANALOG SENSE PATH						
Resolution	N_{ADC}	No missing codes	12			Bits
Integral Nonlinearity	INL_{ADC}	Gain = 1, bipolar mode, V_{IN} = ±2500mV, 500ksps		±0.5	±4.0	LSB
		Gain = 8, unipolar mode, V_{IN} = +400mV, 142ksps		±2.0		
		Gain = 16, bipolar mode, V_{IN} = ±156mV, 142ksps		±2.0	±4.0	
		Gain = 32, bipolar mode, V_{IN} = ±50mV, 142ksps		±2.0		
Differential Nonlinearity	DNL_{ADC}	Gain = 1, bipolar, V_{IN} = ±2500mV, 500ksps			±1.0	LSB
		Gain = 16, bipolar, V_{IN} = ±156mV, 142ksps			±1.0	
		All other gain settings		±0.6		
Offset Error		Input referred		±2.5	±5	mV
Offset-Error Temperature Coefficient				±8		µV/°C
Zero-Code Error		Bipolar, differential measurement of error for ideal ADC output of 0x000		±2.5		mV
Gain Error		Exclude offset and reference error	-1.0		+1.0	%
Gain-Error Temperature Coefficient				±8.5		ppm/°C
Signal-to-Noise Plus Distortion	SINAD	PGA gain = 1V/V		-71		dB
Total Harmonic Distortion	THD	PGA gain = 1V/V		-85		dB

ELECTRICAL CHARACTERISTICS (continued)

(AV_{DD} = DV_{DDIO} = +5.0V, DV_{DD} = +3.3V, f_{SYSCLK} = 8MHz, V_{REFDAC} = V_{REFADC} = +5V, T_A = T_{MIN} to T_{MAX}, unless otherwise noted. Typical values are at T_A = +25°C.) (Note 1)

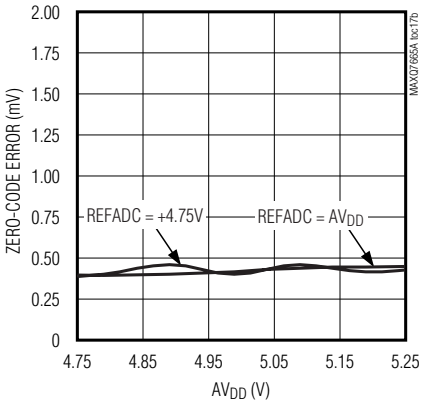
PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Spurious-Free Dynamic Range	SFDR	PGA gain = 1V/V			-91		dB
Conversion Clock Frequency	f_{ADCCLK}	f_{SYSCLK} = 8MHz		0.5		8.0	MHz
Sample Rate	f_{SAMPLE}	PGA gain = 1V/V, $R_{SOURCE} \le 1k\Omega$				500	ksps
		Any PGA gain setting > 1V/V, $R_{SOURCE} \le 5k\Omega$				142	
Conversion Time	t_{CONV}	t_{ACQ} plus 13 ADCCLK cycles at 8MHz				t_{ACQ} + 1.625	µs
Channel/Gain Select Plus Conversion Time		PGA gain = 1V/V, $R_{SOURCE} \le 1k\Omega$				2	µs
		Any PGA gain setting, $R_{SOURCE} \le 5k\Omega$				7	
Track-and-Hold Acquisition Time	t_{ACQ}	PGA gain = 1V/V, $R_{SOURCE} \le 1k\Omega$				375	ns
		Any PGA gain setting > 1V/V, $R_{SOURCE} \le 5k\Omega$				5	µs
Turn-On Time	t_{RECOV}				5		µs
Aperture Delay					30		ns
Aperture Jitter					50		ps_{P-P}
Input-Voltage Range		Unipolar mode	PGA gain = 1	0		AV_{DD}	V
			PGA gain = 2	0		1.6	
			PGA gain = 4	0		0.8	
			PGA gain = 8	0		0.4	
			PGA gain = 16	0		0.2	
			PGA gain = 32	0		0.1	
		Bipolar mode, AIN+ to AIN-	PGA gain = 1	$-V_{REFADC}/2$		$+V_{REFADC}/2$	
			PGA gain = 2	$-V_{REFADC}/4$		$+V_{REFADC}/4$	
			PGA gain = 4	$-V_{REFADC}/8$		$+V_{REFADC}/8$	
			PGA gain = 8	$-V_{REFADC}/16$		$+V_{REFADC}/16$	
			PGA gain = 16	$-V_{REFADC}/32$		$+V_{REFADC}/32$	
			PGA gain = 32	$-V_{REFADC}/64$		$+V_{REFADC}/64$	
Absolute Input-Voltage Range				AGND		AV_{DD}	V
Input Leakage Current		AIN15–AIN0			±20		nA
Small-Signal Bandwidth (-3dB)		V_{IN} x gain = $100mV_{P-P}$	PGA gain = 1		180		MHz
			PGA gain = 2		140		
			PGA gain = 4		120		
			PGA gain = 8		100		
			PGA gain = 16		82		
			PGA gain = 32		80		

16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

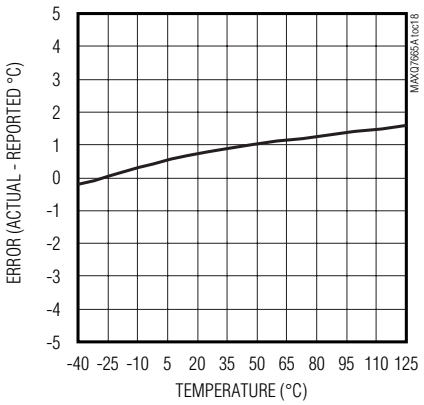
Typical Operating Characteristics (continued)

(AV_{DD} = DV_{DDIO} = +5.0V, DV_{DD} = +3.3V, f_{ADCCLK} = 8MHz, f_{ADC} = 500kHz, T_A = +25°C, unless otherwise noted.)

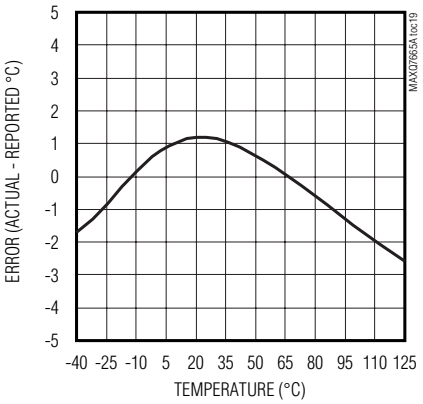
ADC/PGA ZERO-CODE ERROR (GAIN = 16) vs. AV_{DD} SUPPLY VOLTAGE



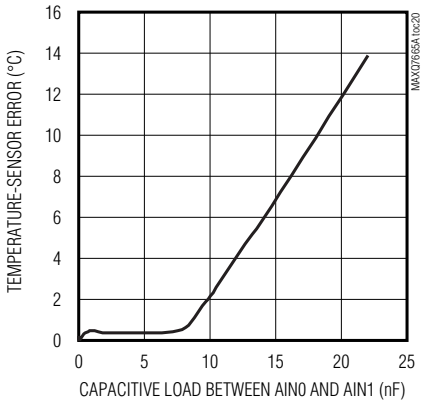
INTERNAL DIODE TEMPERATURE-SENSOR ERROR vs. TEMPERATURE



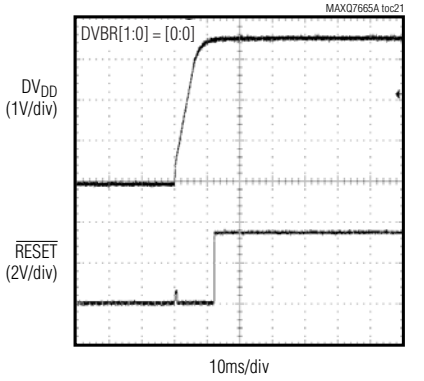
EXTERNAL DIODE TEMPERATURE-SENSOR ERROR vs. TEMPERATURE



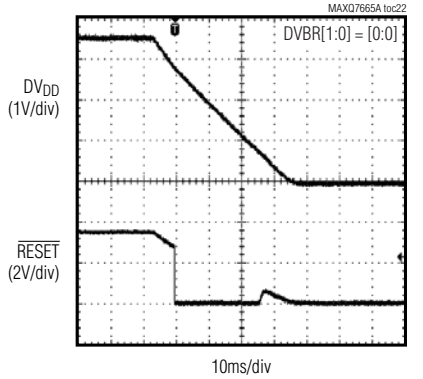
REMOTE TEMPERATURE-SENSOR ERROR DUE TO CAPACITIVE LOADING



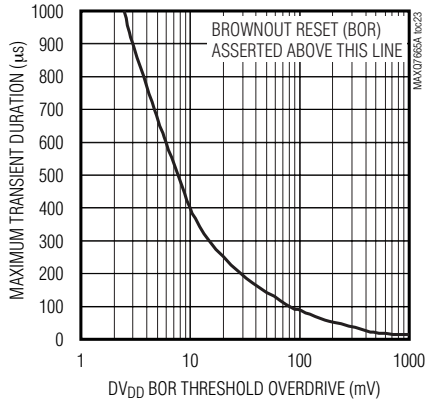
DV_{DD}, $\overline{RESET}$ POWER-UP CHARACTERISTICS



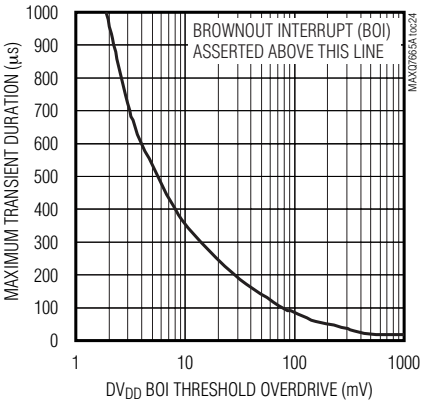
DV_{DD}, $\overline{RESET}$ POWER-DOWN CHARACTERISTICS



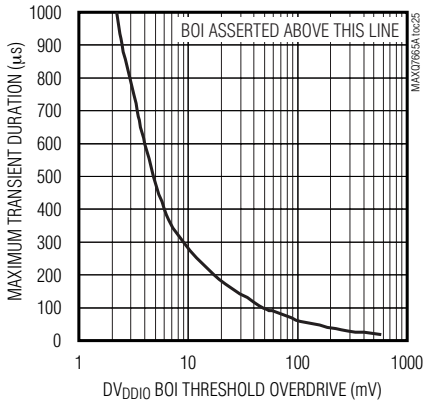
MAXIMUM DV_{DD} TRANSIENT DURATION vs. BOR THRESHOLD OVERDRIVE



MAXIMUM DV_{DD} TRANSIENT DURATION vs. BOI THRESHOLD OVERDRIVE



MAXIMUM DV_{DDIO} TRANSIENT DURATION vs. BOI THRESHOLD OVERDRIVE



16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

Pin Description

PIN	NAME	FUNCTION
1	AIN11	Analog Input Channel 11. AIN11 is multiplexed to the PGA as a differential input with AIN10.
2	AIN10	Analog Input Channel 10. AIN10 is multiplexed to the PGA as a differential input with AIN11.
3	AIN9	Analog Input Channel 9. AIN9 is multiplexed to the PGA as a differential input with AIN8.
4	AIN8	Analog Input Channel 8. AIN8 is multiplexed to the PGA as a differential input with AIN9.
5, 8	AGND	Analog Ground
6	REFADC	ADC External Reference Input. Connect an external reference voltage between 1V and AV_{DD} to REFADC.
7	REFDAC	DAC External Reference Input. Connect an external reference voltage between 0V and AV_{DD} to REFDAC.
9	AIN7	Analog Input Channel 7. AIN7 is multiplexed to the PGA as a differential input with AIN6.
10	AIN6	Analog Input Channel 6. AIN6 is multiplexed to the PGA as a differential input with AIN7.
11	AIN5	Analog Input Channel 5. AIN5 is multiplexed to the PGA as a differential input with AIN4.
12	AIN4	Analog Input Channel 4. AIN4 is multiplexed to the PGA as a differential input with AIN5.
13	AIN3	Analog Input Channel 3. AIN3 is multiplexed to the PGA as a differential input with AIN2. AIN3–AIN0 have remote temperature sensor capability.
14	AIN2	Analog Input Channel 2. AIN2 is multiplexed to the PGA as a differential input with AIN3. AIN3–AIN0 have remote temperature sensor capability.
15	AIN1	Analog Input Channel 1. AIN1 is multiplexed to the PGA as a differential input with AIN0. AIN3–AIN0 have remote temperature sensor capability.
16	AIN0	Analog Input Channel 0. AIN0 is multiplexed to the PGA as a differential input with AIN1. AIN3–AIN0 have remote temperature sensor capability.
17	DACOUT	DAC Buffer Output. DACOUT is the DAC voltage buffer output.
18, 19, 31	DGND	Digital Ground for the Digital Core and Flash
20	CANRXD	CAN Bus Receiver Input. Control area network receiver input.
21	CANTXD	CAN Bus Transmitter Output. Control area network transmitter output.
22	UTX	UART Transmitter Output
23	URX	UART Receiver Input
24	P0.6/T0	Port 0 Bit 6/Timer 0. P0.6 is a general-purpose digital I/O with interrupt/wake-up input capability. T0 is a primary timer/PWM input or output.
25	P0.7/T1	Port 0 Bit 7/Timer 1. P0.7 is a general-purpose digital I/O with interrupt/wake-up input capability. T1 is a primary timer/PWM input or output.
26, 39	DV_{DDIO}	Digital I/O Supply Voltage. Supplies all digital I/O except for XIN, XOUT, and $\overline{RESET}$. Bypass DV_{DDIO} to GNDIO with a 0.1µF capacitor placed as close as possible to the device. DV_{DDIO} is also connected to the input of the linear regulator.
27	GNDIO	Digital I/O Ground
28, 29	I.C.	Internal Connection. Connect I.C. to GNDIO or DV_{DDIO}.
30	N.C.	No Connection. No internal connection. Leave N.C. unconnected.
32	P0.0/TDO	Port 0 Data 0/JTAG Serial Test Data Output. P0.0 is a general-purpose digital I/O with interrupt/wake-up capability. TDO is the JTAG serial test, data output.

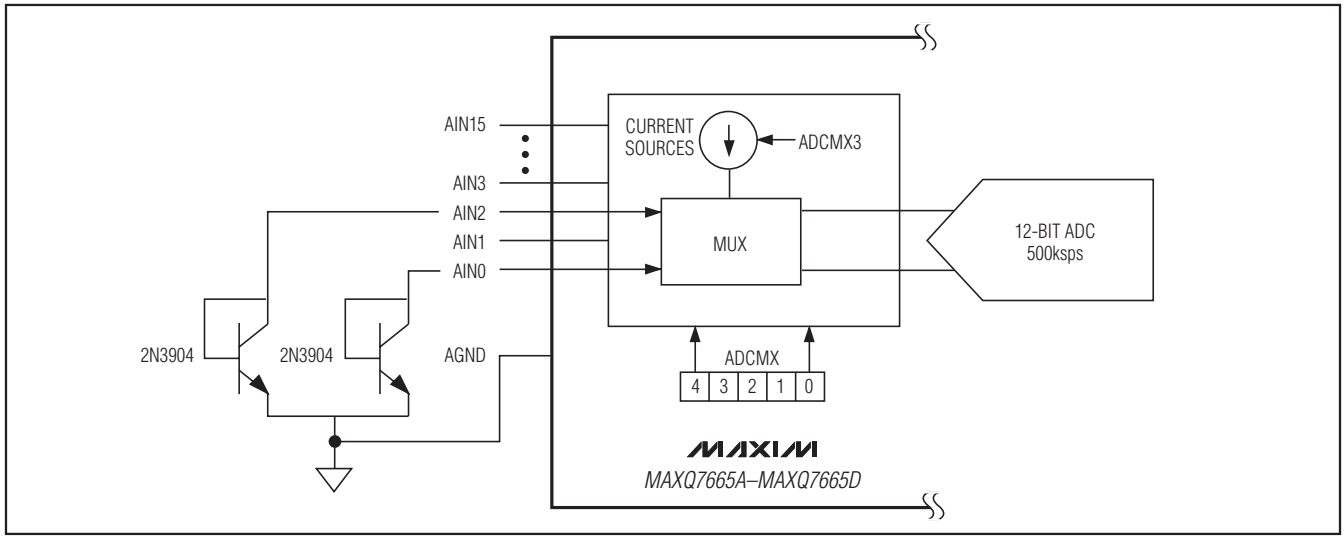


Figure 3. Temperature-Sensor Application Circuit—Single-Ended Configuration

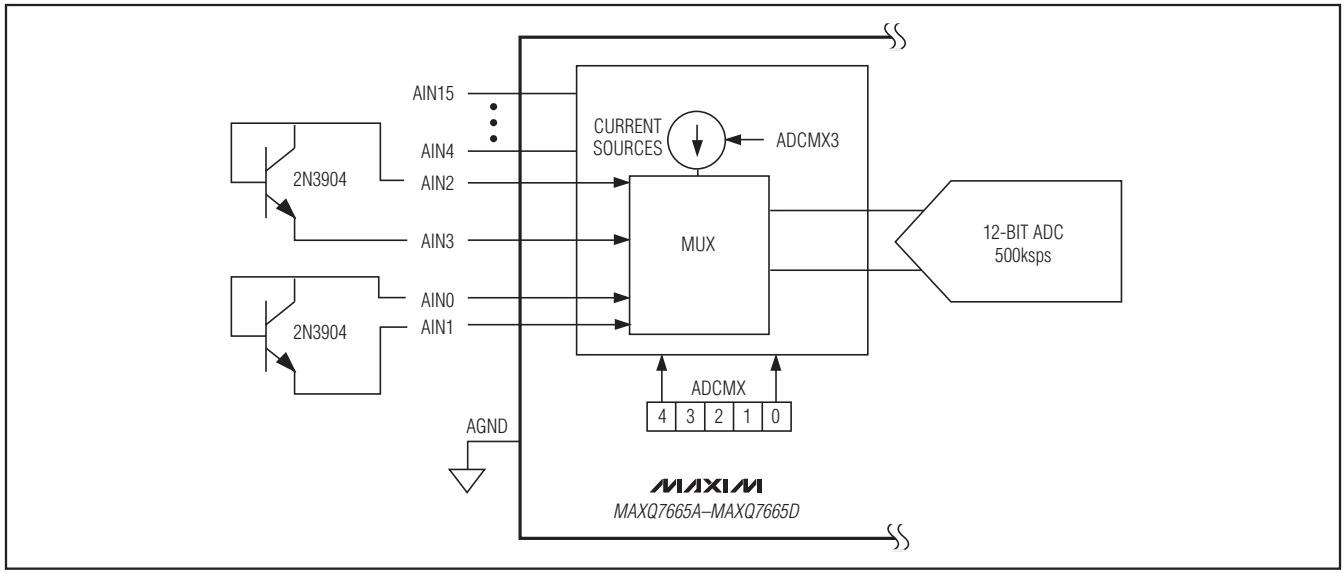


Figure 4. Temperature-Sensor Application Circuit—Differential Configuration

Power-On Reset and Brownout

Power supplies DV_{DD} and DV_{DDIO} each include a brownout monitor that alerts the µC through interrupt when their corresponding supply voltages drop below a selectable threshold. This condition is generally referred to as brownout interrupt (BOI), and these thresholds are set by the VDBI and VIOBI bits for DV_{DD} and DV_{DDIO}, respectively. Continuous monitoring ensures that a valid supply is present at all times while the µC is executing code. For example, the brownout monitors check that DV_{DDIO} does not drop during a CAN bus transfer, or DV_{DD} is not disrupted while the µC core is executing. The DV_{DDIO} brownout monitor also covers the analog peripherals if AV_{DD} and DV_{DDIO} are directly connected.

The DV_{DD} supply (internal core logic) also includes a voltage supervisor that controls the µC reset during power-up (DV_{DD} rising) and brownout (DV_{DD} falling) conditions (see Figure 5 for a POR and brownout timing example).

Note: The MAXQ7665A–MAXQ7665D do not have secondary timer I/O pins (such as T0B and T1B) that are present in some other MAXQ products.

16-Bit x 16-Bit Hardware Multiplier

A hardware multiplier supports high-speed multiplications. The multiplier is capable of completing a 16-bit x 16-bit multiply in a single cycle and contains a 48-bit accumulator that requires one more cycle. The multiplier is not part of the MAXQ core function but a peripheral that performs seven different multiply operations without interfering with the normal core functions:

- Unsigned 16-bit multiplication (one cycle)
- Unsigned 16-bit multiplication and accumulation (two cycles)
- Unsigned 16-bit multiplication and subtraction (two cycles)
- Signed 16-bit multiplication (one cycle)
- Signed 16-bit multiplication and negate (one cycle)
- Signed 16-bit multiplication and accumulation (two cycles)
- Signed 16-bit multiplication and subtraction (two cycles)

Figure 9 illustrates the simplified hardware multiplier circuitry. Two 16-bit parallel-load registers and a 48-bit accumulator are used: operand A (MA), operand B (MB), and accumulator (MC). The accumulator is formed by three 16-bit parallel registers (MC2, MC1, and MC0). The overflow bit is organized in the MCNT status/control register. The multiplicand and the multiplier are initially loaded into the MA and MB registers, respectively. Loading the required operands triggers the respective multiply, multiply-accumulate/subtract or multiply-negate operation. The multiply operation completes in a single cycle with the results in the read-only MC1R/MC0R register. The multiply-accumulate/subtract operation requires one extra wait cycle for the results to be stable in the MC2, MC1, and MC0 registers.

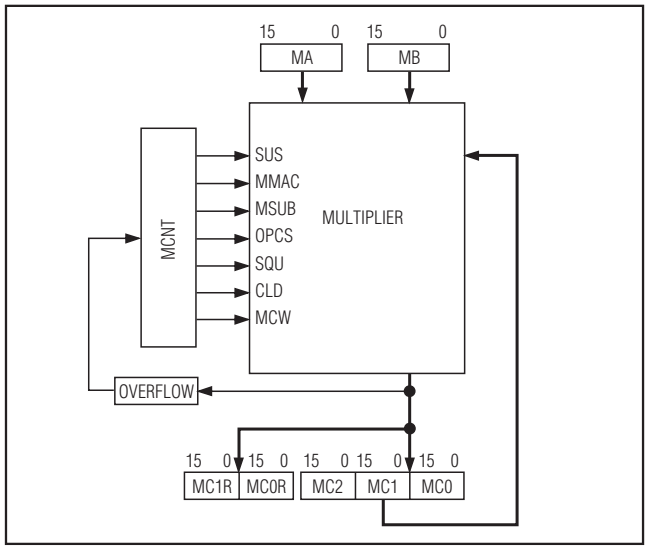


Figure 9. 16-Bit Hardware Multiplier Functional Diagram

The main arithmetic unit is the 16-bit x 16-bit multiplier, which processes operands feeding from the MA and MB registers and generates a 32-bit final product. The product value goes through the 32-bit adder to perform final accumulation with zeroes for multiply operation or with the contents from the MC1 and MC0 registers for multiply-accumulation. The final sum is accessible directly from the accumulator.

To support negate operations including signed multiply-negate and signed and unsigned multiply-subtract, the operand in MA is negated by 1's complement operation before being supplied to the arithmetic unit and the partial product terms are sign corrected. Refer to the *MAXQ7665/MAXQ7666 User's Guide* for more detailed information.

CAN Interface Bus

The MAXQ7665A–MAXQ7665D incorporate a CAN controller that is fully compliant with the CAN 2.0B specification.

The µC interface to the CAN controller is broken into two groups of registers. To simplify the software associated with the operation of the CAN controllers, most of the global CAN status and controls as well as the individual message center control/status registers are located in the peripheral register map. The remaining registers associated with the data identification, identification masks, format, and data are located in a dual port memory to allow the CAN controller and the processor access to the required functions. The CAN controller can directly access the dual port memory. A dedicated interface is incorporated to support dual port memory accessing by the processor through the CAN 0 data pointer (C0DP) and the CAN 0 data buffer (C0DB) special function registers.

16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

CAN Functional Description

The basic functions covered by the CAN controller include the use of 11-bit standard or 29-bit extended acceptance identifiers, as programmed by the µC for each message center, as shown in Figure 10. The CAN unit provides storage for up to 15 messages, with the standard 8-byte data field, in each message.

Each of the first 14 message centers is programmable in either transmit or receive mode. Message center 15 is designed as a receive-only message center with a buffer FIFO arrangement to help prevent the inadvertent loss of data when the µC is busy and is not allowed time to retrieve the incoming message prior to the acceptance of a second message into message center 15. Message center 15 also utilizes an independent set of mask registers and identification registers, which are only applied once an incoming message has not been accepted by any of the first 14 message centers. A second filter test is also supported for all message centers (1–15) to allow the CAN controller to use two separate 8-bit media masks and media arbitration fields to verify the contents of the first 2 bytes of data of

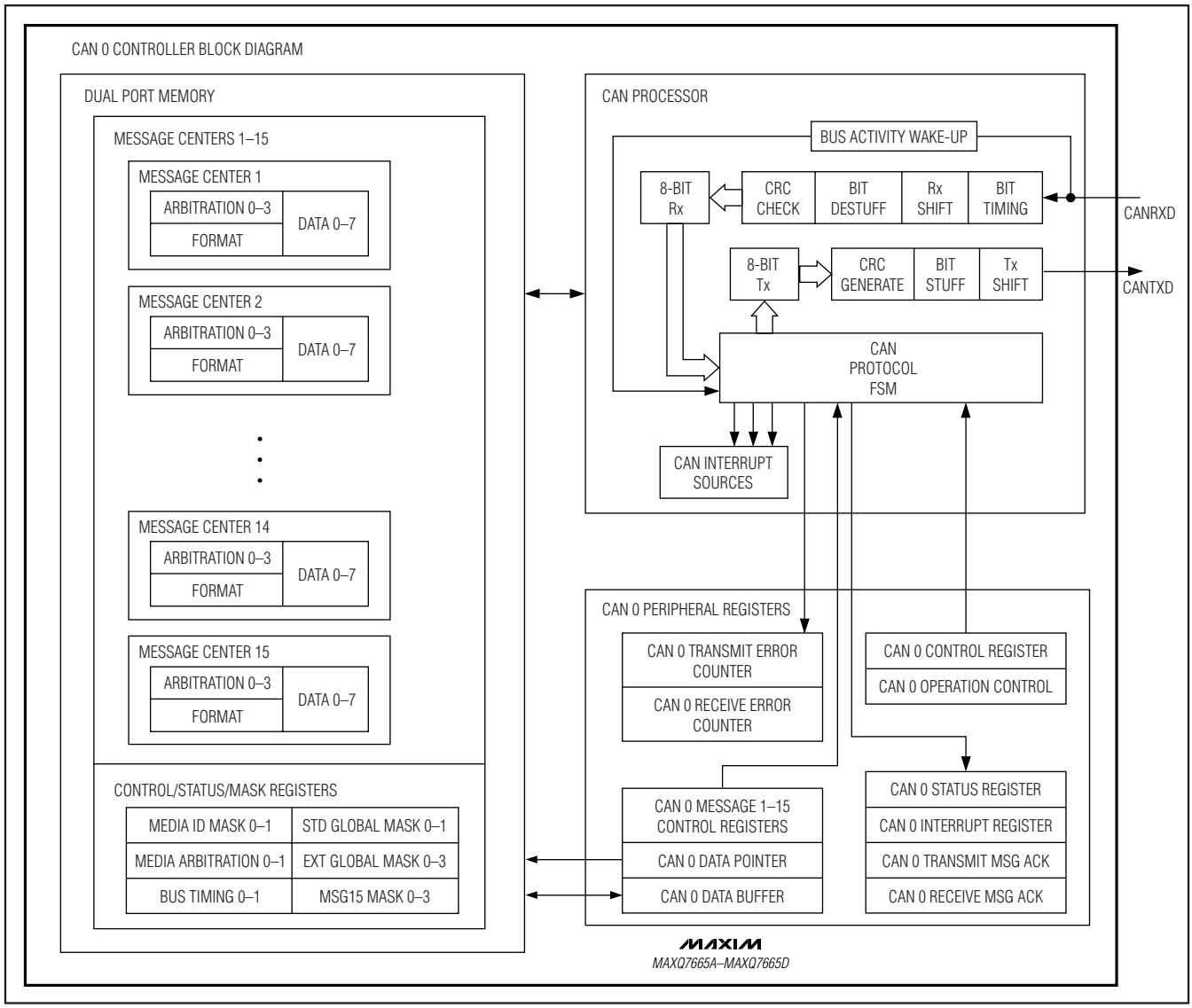


Figure 10. CAN 0 Controller Block Diagram

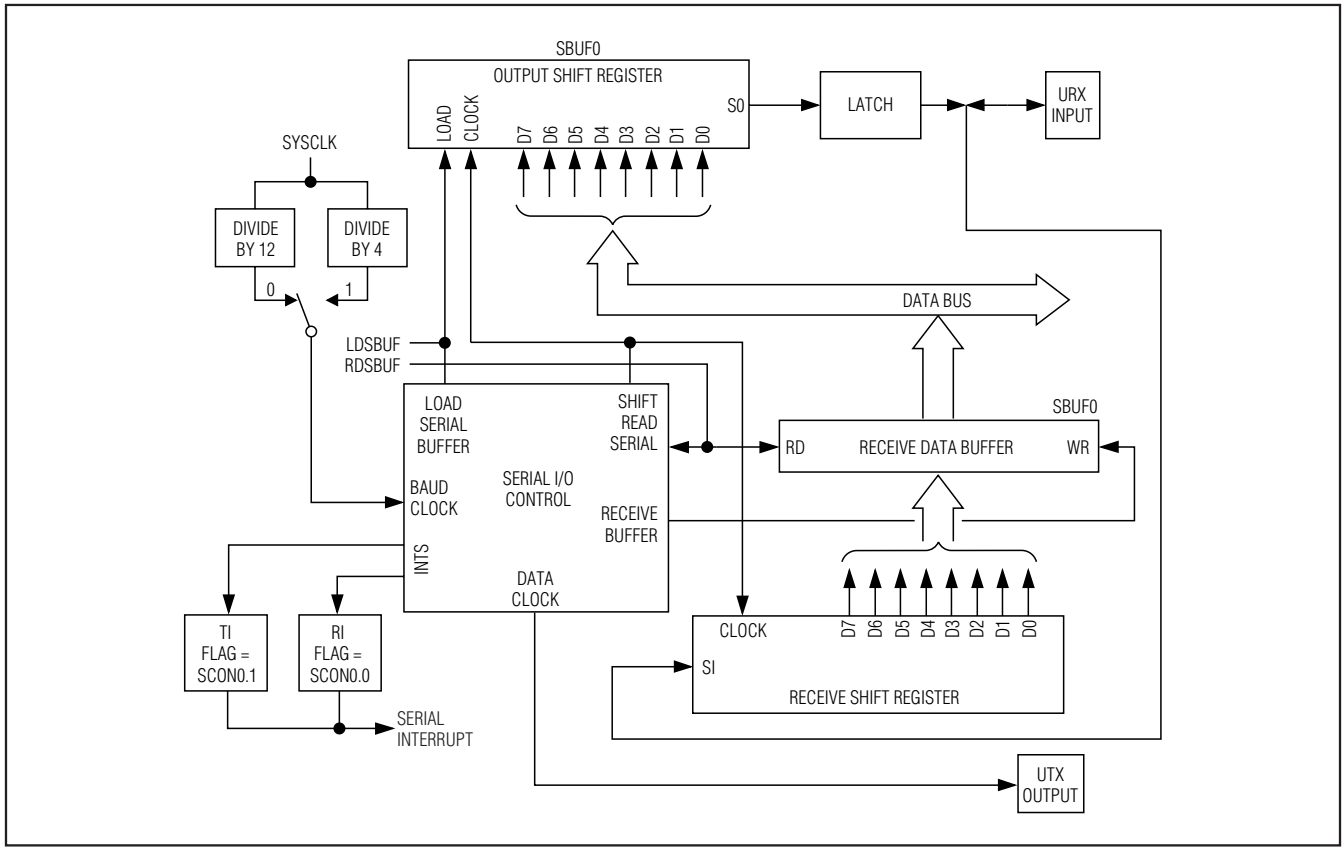


Figure 11a. UART Synchronous Mode (Mode 0)

each incoming message, before accepting an incoming message. This feature allows the CAN unit to directly support the use of higher CAN protocols, which make use of the first and/or second byte of data as a part of the acceptance layer for storing incoming messages. Each message center can also be programmed independently to perform testing of the incoming data with or without the use of the global masks.

Global controls and status registers in the CAN unit allow the µC to evaluate error messages, validate new data and the location of such data, establish the bus timing for the CAN bus, establish the identification mask bits, and verify the source of individual messages. In addition, each message center is individually equipped with the necessary status and controls to establish directions, interrupt generation, identification mode (standard or extended), data field size, data status, automatic remote frame request and acknowledgment, and masked or nonmasked identification acceptance testing.

UART Interface

Serial interfacing is provided through one (UTX/URX) 8051-style universal synchronous/asynchronous receiver/transmitter (UART) capable of interfacing with a LIN transceiver. Figure 11a shows the UART block diagram in synchronous mode and Figure 11b shows asynchronous mode. The UART allows the device to conveniently communicate with other RS-232 interface-enabled devices, as well as PCs and serial modems when paired with an external RS-232 line driver/receiver. The UART can detect framing errors and indicate the condition through a user-accessible software bit. The time base of the serial port is derived from either a division of the system clock or the dedicated baud clock generator. The UART is capable of supporting LIN protocol implementation in software when using one of the timers for autobaud

16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

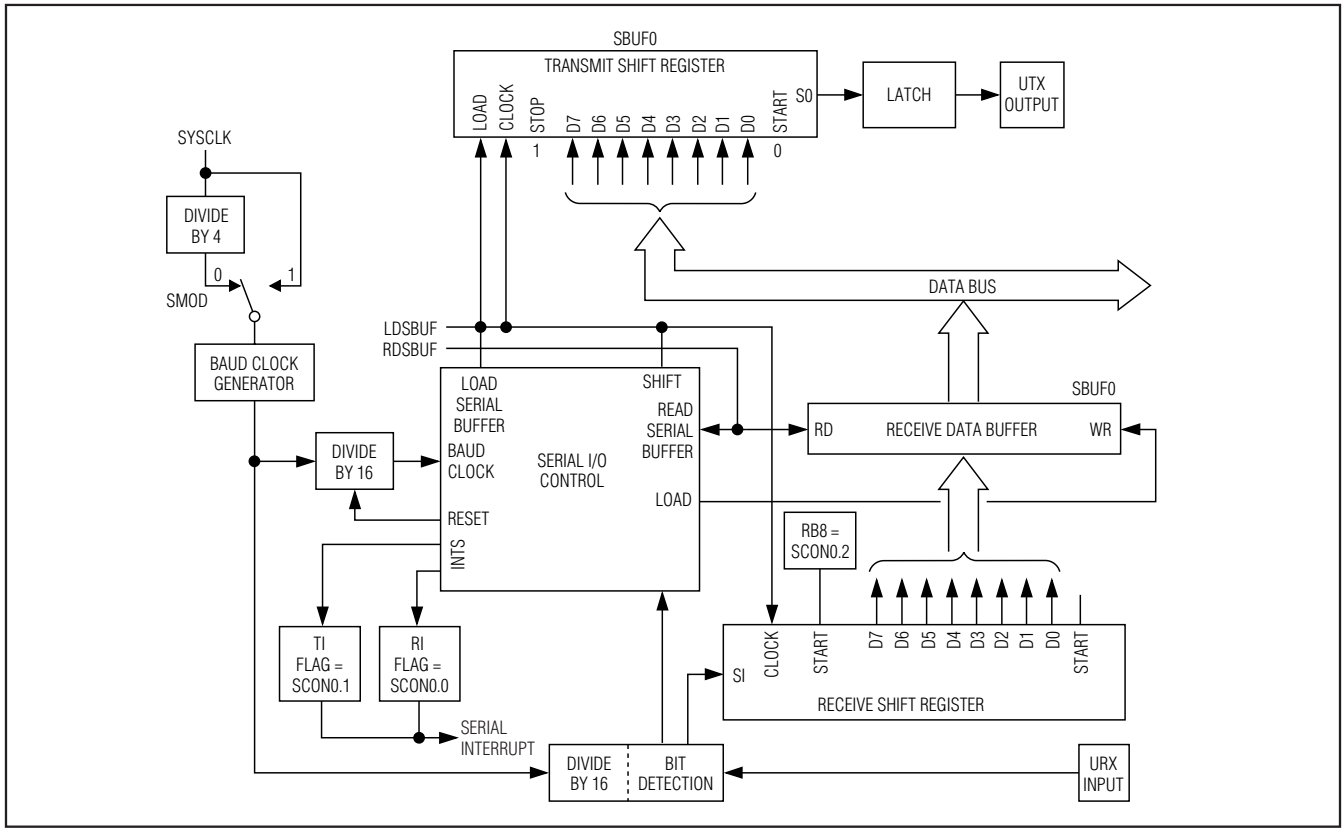


Figure 11b. UART Asynchronous Mode (Mode 1)

detection. Table 1 summarizes the operating characteristics as well as the maximum baud rate of each mode.

JTAG Interface Bus

The joint test action group (JTAG) IEEE 1149.1 standard defines a unique method for in-circuit testing and programming. The MAXQ7665A–MAXQ7665D conform to this standard, implementing an external test access port (TAP) and internal TAP controller for communication with a JTAG bus master, such as an automatic test equipment (ATE) system. For detailed information on the TAP and TAP controller, refer to *IEEE Standard 1149.1* on the IEEE website at http://standards.ieee.org. The JTAG on the MAXQ7665A–MAXQ7665D is used for in-circuit emulation and debug support, but does not support boundary scan test capability.

The TAP controller communicates synchronously with the host system (bus master) through four digital I/O pins: test mode select (TMS), test clock (TCK), test data input (TDI), and test data output (TDO). The internal TAP module consists of several shift registers and a

Table 1. Operating Characteristics and Mode Baud Rate

MODE	TYPE	BAUD CLOCK	START BITS	DATA BITS	STOP BITS	MAX BAUD RATE AT 8MHz
Mode 0	Synchronous	4 or 12 clock	N/A	8	N/A	2Mbps
Mode 1	Asynchronous	Baud generation	1	8	1	250kbps
Mode 2	Asynchronous	32 or 64 clock	1	8 + 1	1	250kbps
Mode 3	Asynchronous	Baud generation	1	8 + 1	1	250kbps

the host to the internal TAP module shift registers. Data is transferred LSB first.

- TCK—Serial clock for the test logic.
- TMS—Test mode selection. Test signals received at TMS are sampled at the rising edge of TCK and decoded by the TAP controller to control the test operation.

General-Purpose Digital I/Os

The MAXQ7665A–MAXQ7665D provide eight general-purpose digital I/Os (GPIOs). All GPIOs have an additional special function (SF), such as a timer input/output, or TAP signal for JTAG communication. For example, the state of pin P0.6/T0 can be programmed to depend on timer channel 0 logic. When programmed as a port, each I/O is configurable for high-impedance or weak pullup to DV_{DDIO}. At power-up, each GPIO is configured as an input with pullups to DV_{DDIO}. Note that at power-up, the JTAG function is enabled and should be turned off before normal operation. In addition, each GPIO can be programmed to cause an interrupt (on falling or rising edges). In stop mode, any interrupt can be used to wake up the device.

The data input/output direction in a port is independently controlled by the port direction register (PD). Each I/O within the port can be individually set as an output or input. The port output register (PO) contains the current state of the logic output buffers. When an I/O is configured as an output, writing to the PO register controls the output logic state. Reading the PO register shows the current state of the output buffers, independent of the data direction. The port input register (PI) is a read-only register that always reflects the logic state of the I/Os. When an I/O is configured as an input, writing to the PO register enables/disables the pull-up resistor. Refer to the *MAXQ7665/MAXQ7666 User's Guide* for more detailed information.

Port Characteristics

The MAXQ7665A–MAXQ7665D contain only one port (P0). It is a bidirectional 8-bit I/O port, which contains the following features:

- Schmitt trigger input circuitry with software-selectable high-impedance or weak pullup to DV_{DDIO}
- Software-selectable push-pull CMOS output drivers capable of sinking and sourcing 1.6mA
- Software-selectable open-drain output drivers capable of sinking 1.6mA
- Falling or rising edge interrupt capability
- All I/Os contain an additional special function, such as a logic input/output for a timer channel. Selecting an I/O for a special function alters the port characteristics of that I/O (refer to the *MAXQ7665/MAXQ7666 User's Guide* for more details). Figure 13 illustrates the functional blocks of an I/O.

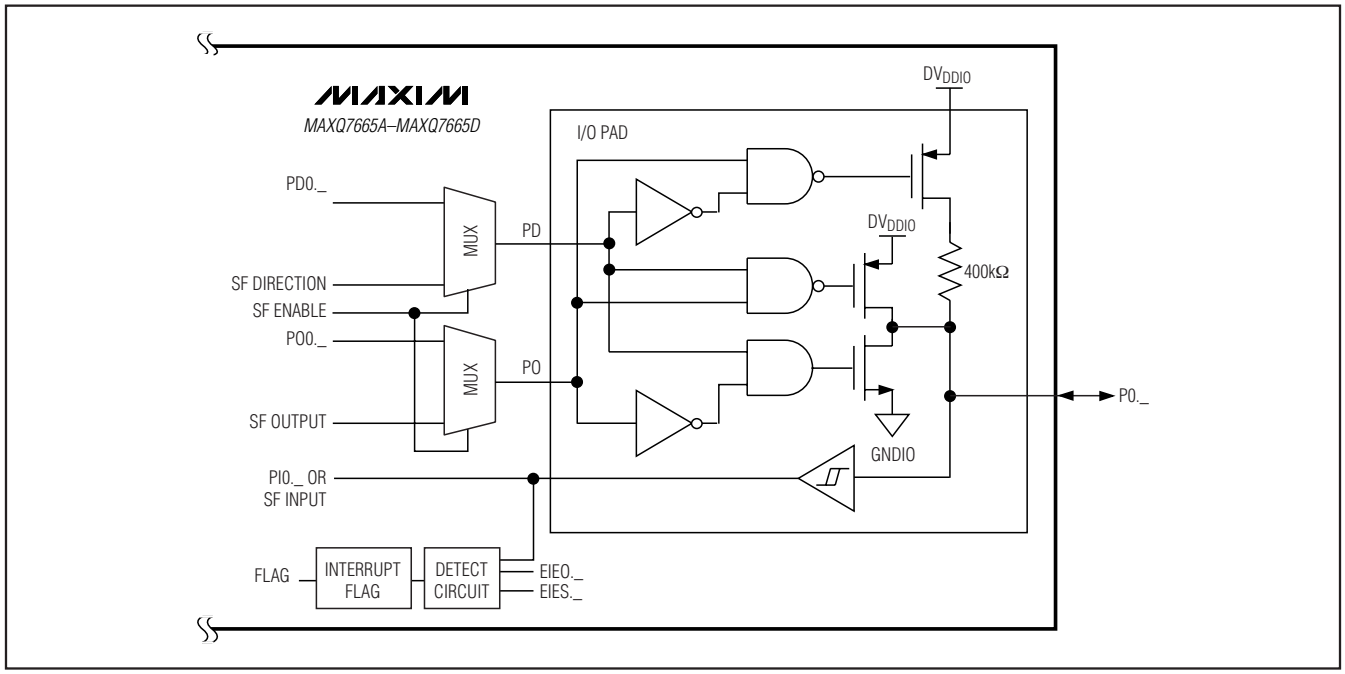


Figure 13. Digital I/O Circuitry

MAXQ Core Architecture

The MAXQ7665A–MAXQ7665D are low-cost, high-performance, CMOS, fully static, 16-bit µCs with flash memory and are members of the MAXQ family of µCs. The MAXQ7665A–MAXQ7665D are structured on a highly advanced, accumulator-based, 16-bit RISC architecture. Fetch and execution operations are completed in one cycle without pipelining, because the instruction contains both the operation code and data. The result is a streamlined 8 million instructions-per-second (MIPS) µC.

The highly efficient core is supported by a 16-level hardware stack, enabling fast subroutine calling and task switching. Data can be quickly and efficiently manipulated with three internal data pointers. Multiple data pointers allow more than one function to access data memory without having to save and restore data pointers each time. The data pointers can automatically increment or decrement following an operation, eliminating the need for software intervention. As a result, application speed is greatly increased.

Instruction Set

The instruction set is composed of fixed-length, 16-bit instructions that operate on registers and memory locations. The instruction set is highly orthogonal, allowing arithmetic and logical operations to use any register along with the accumulator. Special-function registers (also called peripheral registers) control the peripherals and are subdivided into register modules. The family architecture is modular, so that new devices and modules can reuse code developed for existing products.

The architecture is transport-triggered. This means that writes or reads from certain register locations can also cause side effects to occur. These side effects form the basis for the higher level operation codes defined by the assembler, such as ADDC, OR, JUMP, etc. The operation codes are actually implemented as MOVE instructions between certain register locations, while the assembler handles the encoding, which need not be a concern to the programmer.

The 16-bit instruction word is designed for efficient execution. Bit 15 indicates the format for the source field of the instruction. Bits 0 to 7 of the instruction represent the source for the transfer. Depending on the value of the format field, this can either be an immediate value or a source register. If this field represents a register, the lower 4 bits contain the module specifier and the upper 4 bits contain the register index in that module.

Bits 8 to 14 represent the destination for the transfer. This value always represents a destination register, with the lower 4 bits containing the module specifier and the upper 3 bits containing the register subindex within that module. Any time that it is necessary to directly select one of the upper 24 registers as a destination, the prefix register, PFX, is needed to supply the extra destination bits. This prefix register write is inserted automatically by the assembler and requires only one additional execution cycle.

Memory Organization

The MAXQ7665A–MAXQ7665D incorporate several memory areas:

- 8KB (4K x 16) utility ROM
- Up to 128KB (64K x 16) of flash memory for program storage
- 512 bytes (256 x 16) of SRAM for storage of temporary variables
- 16-level stack memory for storage of program return addresses and general-purpose use

The memory is arranged by default in a Harvard architecture, with separate address spaces for program and data memory (see Figure 14). A special mode allows data memory to be mapped into program space, permitting code execution from data memory. In addition, another mode allows program memory to be mapped into data space, permitting code constants to be accessed as data memory.

The incorporation of flash memory allows the devices to be reprogrammed, eliminating the expense of throwing away one-time programmable devices during development and field upgrades (see Figure 15 for the flash memory sector maps). Flash memory can be password protected with a 16-word key, denying access to program memory by unauthorized individuals.

Following any reset, execution begins in the utility ROM. The ROM software determines whether the program execution should immediately jump to location 0000h, the start of user-application code, or to one of the special routines mentioned. Routines within the utility ROM are user-accessible and can be called as subroutines by the application software. More information on the utility ROM contents is contained in the *MAXQ7665/MAXQ7666 User's Guide*.

Some applications require protection against unauthorized viewing of program code memory. For these applications, access to in-system programming, in-application programming, or in-circuit debugging functions is prohibited until a password has been supplied. The password is defined as the 16 words of physical program memory at addresses 0010h to 001Fh.

A single password lock (PWL) bit is implemented in the SC register. When the PWL is set to one (POR default), the password is required to access the utility ROM, including in-circuit debug and in-system programming routines that allow reading or writing of internal memory. When PWL is cleared to zero, these utilities are fully accessible without the password. The password is automatically set to all ones following a mass erase.

Programming

The flash memory of the µC can be programmed by two different methods: in-system programming and in-application programming. Both methods afford great flexibility in system design as well as reduce the lifecycle cost of the embedded system. These features can be password protected to prevent unauthorized access to program memory.

In-System Programming

An internal bootstrap loader allows the device to be reloaded over a simple JTAG interface. As a result, software can be upgraded in-system, eliminating the need for a costly hardware retrofit when updates are required. Remote software uploads are possible that enable physically inaccessible applications to be frequently updated. The interface hardware can be a JTAG connection to another µC, or a connection to a PC serial port using a serial-to-JTAG converter such as the MAXQJTAG-001, available from Maxim Integrated Products, Inc. If in-system programmability is not required, a commercial gang programmer can be used for mass programming.

After a power-up or reset, the JTAG interface is active and loading the TAP with the system programming instruction invokes the bootstrap loader. Setting the SPE bit to 1 during reset through the JTAG interface executes the bootstrap-loader-mode program that resides in the utility ROM. When programming is complete, the bootstrap loader can clear the SPE bit and reset the device, allowing the device to bypass the utility ROM and begin execution of the application software.

The following bootstrap loader functions are supported:

- Load
- Dump
- CRC
- Verify
- Erase

In-Application Programming

The in-application programming feature allows the µC to modify its own flash program memory while simultaneously executing its application software. This allows on-the-fly software updates in mission-critical applications that cannot afford downtime. Alternatively, it allows the application to develop custom loader software that can operate under the control of the application software. The utility ROM contains user-accessible flash programming functions that erase and program flash memory. These functions are described in detail in the *MAXQ7665/MAXQ7666 User's Guide* for these devices.

Register Set

Most functions of these devices are controlled by sets of registers. These registers provide a working space for memory operations as well as configuring and addressing peripheral registers on the device. Registers are divided into two major types: system registers and peripheral registers. The common register set, also known as the system registers, includes the ALU, accumulator registers, data pointers, interrupt vectors and control, and stack pointer. The peripheral registers define additional functionality that may be included by different products based on the MAXQ architecture. This functionality is broken up into discrete modules so that only the features required for a given product need to be included. Tables 2 and 4 show the MAXQ7665A–MAXQ7665D register set. Tables 3 and 5 show the bit functions and reset values.

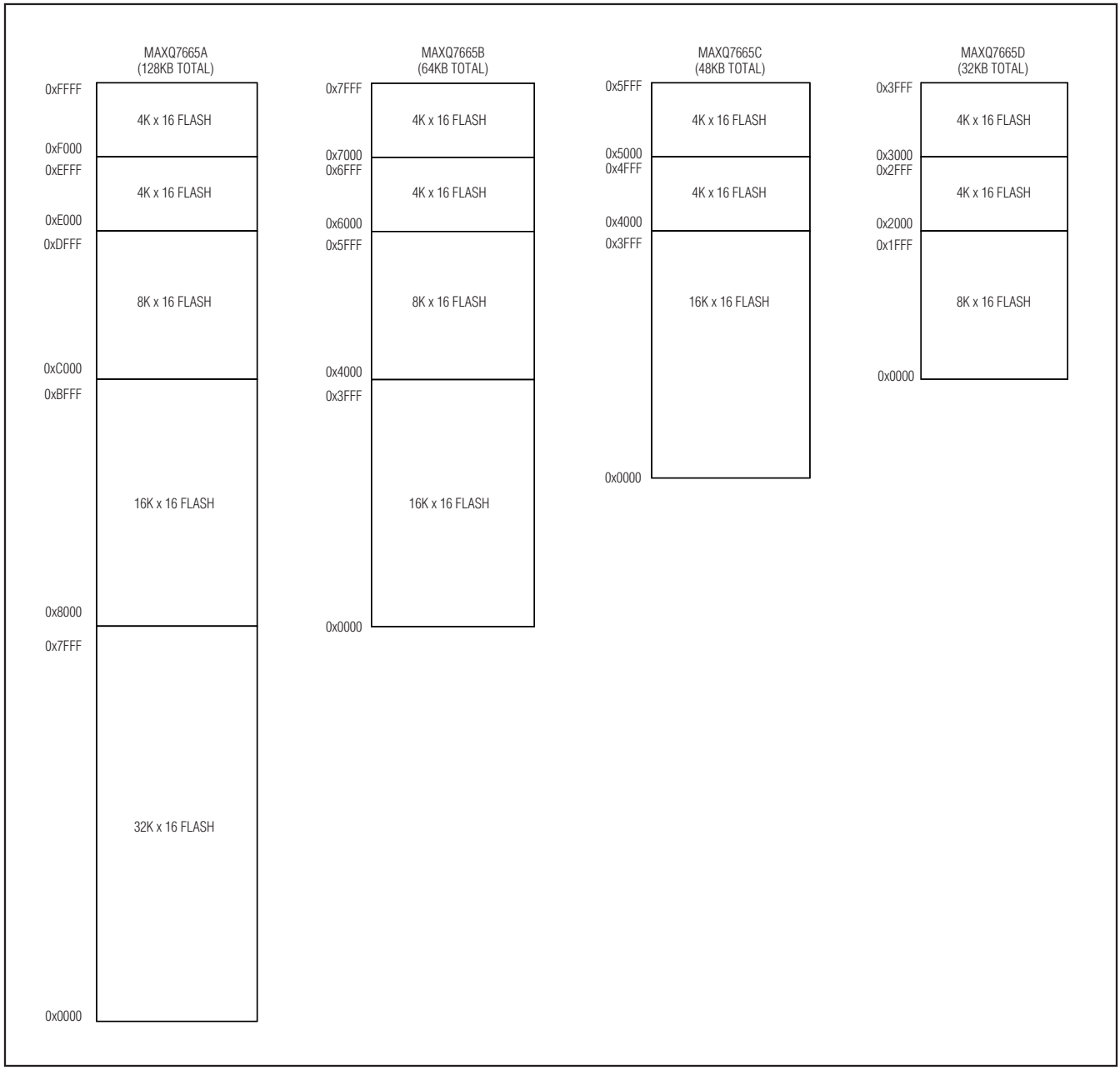


Figure 15. Flash Memory Sector Maps

External System Reset

Asserting the external $\overline{\text{RESET}}$ input low causes the device to enter the reset state. The external reset functions as described in the *MAXQ7665/MAXQ7666 User's Guide*. Execution resumes at location 8000h after $\overline{\text{RESET}}$ is released.

Crystal Selection

The MAXQ7665A–MAXQ7665D require a crystal with the following specifications:

Frequency: 8MHz

C_{LOAD}: 6pF (min)

Drive level: 5µW

Series resonance resistance: 30Ω max

Note: Series resonance resistance is the resistance observed when the resonator is in the series resonant condition. This is a parameter often stated by quartz crystal vendors and is called R1. When a resonator is used in the parallel resonant mode with an external load capacitance, as is the case with the MAXQ7665A–MAXQ7665D oscillator circuit, the effective resistance is sometimes stated. This effective resistance at the loaded frequency of oscillation is:

$$R1 \times (1 + (C_O/C_{LOAD}))^2$$

For typical C_O and C_{LOAD} values, the effective resistance can be greater than R1 by a factor of 2.

Development and Technical Support

A variety of highly versatile, affordably priced development tools for this µC are available from Maxim and third-party suppliers, including:

- Compilers
- Evaluation kits
- Integrated development environments (IDEs)
- JTAG-to-serial converters for programming and debugging

A list of some development-tool vendors can be found at **www.maxim-ic.com/microcontrollers**.

Technical support is available through email at maxq.support@maxim-ic.com.

16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

Table 3. System Register Bit Functions and Reset Values

REGISTER	REGISTER BIT															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AP									—	—	—	—	AP (4 Bits)			
									0	0	0	0	0	0	0	0
APC									CLR	IDS	—	—	—	MOD2	MOD1	MOD0
									0	0	0	0	0	0	0	0
PSF									Z	S	—	GPF1	GPF0	OV	C	E
									1	0	0	0	0	0	0	0
IC									—	—	CGDS	—	—	—	INS	IGE
									0	0	0	0	0	0	0	0
IMR									IMS	—	IM5	IM4	IM3	IM2	IM1	IM0
									0	0	0	0	0	0	0	0
SC									TAP	—	CDA1	CDA0	UPA	ROD	PWL	—
									1	0	0	0	0	0	s*	0
IIR									IIS	—	II5	II4	II3	II2	II1	II0
									0	0	0	0	0	0	0	0
CKCN									XT	—	RGMD	STOP	SWB	—	—	CD0
									s*	0	s*	0	0	0	0	1
WDCN									POR	EWDI	WD1	WD0	WDIF	WTRF	EWT	RWT
									s*	s*	0	0	0	s*	s*	0
A[n] (0..15)	A[n] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PFX[n] (0..15)	PFX[n] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
IP	IP (16 Bits)															
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SP	—	—	—	—	—	—	—	—	—	—	—	—	SP (4 Bits)			
	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
IV	IV (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LC[0]	LC[0] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LC[1]	LC[1] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
OFFS									OFFS (8 Bits)							
									0	0	0	0	0	0	0	0
DPC	—	—	—	—	—	—	—	—	—	—	—	WBS2	WBS1	WBS0	SDPS1	SDPS0
	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
GR	GR.15	GR.14	GR.13	GR.12	GR.11	GR.10	GR.9	GR.8	GR.7	GR.6	GR.5	GR.4	GR.3	GR.2	GR.1	GR.0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRL									GR.7	GR.6	GR.5	GR.4	GR.3	GR.2	GR.1	GR.0
									0	0	0	0	0	0	0	0
BP	BP (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRS	GR.7	GR.6	GR.5	GR.4	GR.3	GR.2	GR.1	GR.0	GR.15	GR.14	GR.13	GR.12	GR.11	GR.10	GR.9	GR.8
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRH									GR.15	GR.14	GR.13	GR.12	GR.11	GR.10	GR.9	GR.8
									0	0	0	0	0	0	0	0
GRXL	GR.7	GR.7	GR.7	GR.7	GR.7	GR.7	GR.7	GR.7	GR.7	GR.6	GR.5	GR.4	GR.3	GR.2	GR.1	GR.0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FP	FP (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DP[0]	DP[0] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DP[1]	DP[1] (16 Bits)															
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Bits indicated by an "s" are only affected by a POR and not by other forms of reset. These bits are set to 0 after a POR. Refer to the MAXQ7665/MAXQ7666 User's Guide for more information.*

Table 4. Peripheral Register Map

REGISTER INDEX	MODULE NAME (BASE SPECIFIER)					
	M0 (0h)	**M1 (1h)**	**M2 (2h)**	**M3 (3h)**	**M4 (4h)**	**M5 (5h)**
0h	PO0	MCNT	T2CNA0	T2CNA2	C0C	VMC
1h	—	MA	T2H0	T2H2	C0S	APE
2h	—	MB	T2RH0	T2RH2	**C0IR**	ACNT
3h	EIF0	MC2	T2CH0	T2CH2	C0TE	DCNT
4h	—	MC1	T2CNA1	—	C0RE	DACI
5h	—	MC0	T2H1	—	COR	—
6h	—	—	T2RH1	—	C0DP	DACO
7h	SBUF0	—	T2CH1	—	C0DB	—
8h	**PI0**	—	T2BNB0	T2CNB2	**C0RMS**	**ADCD**
9h	—	—	T2V0	T2V2	**C0TMA**	**TSO**
Ah	—	FCNTL	T2R0	T2R2	—	AIE
Bh	EIE0	FDATA	*T2C0*	T2C2	—	**ASR**
Ch	—	**MC1R**	T2CNB1	—	—	OSCC
Dh	—	**MC0R**	T2V1	—	—	—
Eh	—	—	T2R1	—	—	—
Fh	—	—	T2C1	—	—	—
10h	PD0	—	T2CFG0	T2CFG2	—	—
11h	—	—	T2CFG1	—	C0M1C	—
12h	—	—	—	—	C0M2C	—
13h	EIES0	—	—	—	C0M3C	—
14h	—	—	—	—	C0M4C	—
15h	—	—	—	—	C0M5C	—
16h	—	—	—	—	C0M6C	—
17h	—	—	—	—	C0M7C	—
18h	—	—	ICDT0	—	C0M8C	—
19h	—	—	ICDT1	—	C0M9C	—
1Ah	—	—	ICDC	—	C0M10C	—
1Bh	—	—	ICDF	—	C0M11C	—
1Ch	—	Reserved	ICDB	—	C0M12C	—
1Dh	SCON0	—	ICDA	—	C0M13C	—
1Eh	SMD0	—	ICDD	—	C0M14C	—
1Fh	PR0	—	—	—	C0M15C	—

Note: *Names that appear in bold indicate that the register is read-only.*

Table 5. Peripheral Register Bit Functions and Reset Values

REGISTER	REGISTER BIT															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PO0	—	—	—	—	—	—	—	—	PO0.7	PO0.6	PO0.5	PO0.4	PO0.3	PO0.2	PO0.1	PO0.0
(M0, 0h)	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
EIF0	—	—	—	—	—	—	—	—	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
(M0, 3h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SBUF0	—	—	—	—	—	—	—	—	SBUF0.7	SBUF0.6	SBUF0.5	SBUF0.4	SBUF0.3	SBUF0.2	SBUF0.1	SBUF0.0
(M0, 7h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PI0	—	—	—	—	—	—	—	—	PI0.7	PI0.6	PI0.5	PI0.4	PI0.3	PI0.2	PI0.1	PI0.0
(M0, 8h)	0	0	0	0	0	0	0	0	ST	ST	ST	ST	ST	ST	ST	ST
EIE0	—	—	—	—	—	—	—	—	EX7	EX6	EX5	EX4	EX3	EX2	EX1	EX0
(M0, Bh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PD0	—	—	—	—	—	—	—	—	PD0.7	PD0.6	PD0.5	PD0.4	PD0.3	PD0.2	PD0.1	PD0.0
(M0, 10h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
EIES0	—	—	—	—	—	—	—	—	IT7	IT6	IT5	IT4	IT3	IT2	IT1	IT0
(M0, 13h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCON0	—	—	—	—	—	—	—	—	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI
(M0, 1Dh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SMD0	—	—	—	—	—	—	—	—	—	—	—	—	—	ESI	SMOD	FEDE
(M0, 1Eh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PR0	PR0.15	PR0.14	PR0.13	PR0.12	PR0.11	PR0.10	PR0.9	PR0.8	PR0.7	PR0.6	PR0.5	PR0.4	PR0.3	PR0.2	PR0.1	PR0.0
(M0, 1Fh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MCNT	—	—	—	—	—	—	—	—	OF	MCW	CLD	SQU	OPCS	MSUB	MMAC	SUS
(M1, 0h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MA	MA.15	MA.14	MA.13	MA.12	MA.11	MA.10	MA.9	MA.8	MA.7	MA.6	MA.5	MA.4	MA.3	MA.2	MA.1	MA.0
(M1, 1h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MB	MB.15	MB.14	MB.13	MB.12	MB.11	MB.10	MB.9	MB.8	MB.7	MB.6	MB.5	MB.4	MB.3	MB.2	MB.1	MB.0
(M1, 2h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MC2	MC2.15	MC2.14	MC2.13	MC2.12	MC2.11	MC2.10	MC2.9	MC2.8	MC2.7	MC2.6	MC2.5	MC2.4	MC2.3	MC2.2	MC2.1	MC2.0
(M1, 3h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MC1	MC1.15	MC1.14	MC1.13	MC1.12	MC1.11	MC1.10	MC1.9	MC1.8	MC1.7	MC1.6	MC1.5	MC1.4	MC1.3	MC1.2	MC1.1	MC1.0
(M1, 4h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MC0	MC0.15	MC0.14	MC0.13	MC0.12	MC0.11	MC0.10	MC0.9	MC0.8	MC0.7	MC0.6	MC0.5	MC0.4	MC0.3	MC0.2	MC0.1	MC0.0
(M1, 5h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FCNTL	—	—	—	—	—	—	—	—	FBUSY	FERR	FINE	FBYP	DQ5	FC2	FC1	—
(M1, Ah)	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
FDATA	FDATA.15	FDATA.14	FDATA.13	FDATA.12	FDATA.11	FDATA.10	FDATA.9	FDATA.8	FDATA.7	FDATA.6	FDATA.5	FDATA.4	FDATA.3	FDATA.2	FDATA.1	FDATA.0
(M1, Bh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MC1R	MC1R.15	MC1R.14	MC1R.13	MC1R.12	MC1R.11	MC1R.10	MC1R.9	MC1R.8	MC1R.7	MC1R.6	MC1R.5	MC1R.4	MC1R.3	MC1R.2	MC1R.1	MC1R.0
(M1, Ch)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MC0R	MC0R.15	MC0R.14	MC0R.13	MC0R.12	MC0R.11	MC0R.10	MC0R.9	MC0R.8	MC0R.7	MC0R.6	MC0R.5	MC0R.4	MC0R.3	MC0R.2	MC0R.1	MC0R.0
(M1, Dh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T2CNA0	—	—	—	—	—	—	—	—	ET2	T2OE0	T2POL0	TR2L	TR2	CPRL2	SS2	G2EN
(M2, 0h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T2H0	—	—	—	—	—	—	—	—	T2H0.7	T2H0.6	T2H0.5	T2H0.4	T2H0.3	T2H0.2	T2H0.1	T2H0.0
(M2, 1h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5. Peripheral Register Bit Functions and Reset Values (continued)

REGISTER	REGISTER BIT															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
C0M14C	—	—	—	—	—	—	—	—	MSRDY	ETI	ERI	INTRQ	EXTRQ	MTRQ	ROW/TIH	DTUP
(M4, 1Eh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C0M15C	—	—	—	—	—	—	—	—	MSRDY	ETI	ERI	INTRQ	EXTRQ	MTRQ	ROW/TIH	DTUP
(M4, 1Fh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VMC	—	—	—	—	—	—	—	—	—	—	VIOBI1	VIOBI0	VDBI1	VDBI0	VDBR1	VDBR0
(M5, 0h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	S	S
APE	—	—	—	VIBE	VDBE	VDPE	—	—	PGG2	PGG1	PGG0	TSE	PGAE	—	DACE	ADCE
(M5, 1h)	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
ACNT	ADCMX4	ADCMX3	ADCMX2	ADCMX1	ADCMX0	ADCDIF	ADCBIP	—	—	ADCDUL	—	ADCASD	ADCBY	ADCS2	ADCS1	ADCS0
(M5, 2h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DCNT	—	—	—	—	—	—	—	—	—	DACLD2	DACLD1	DACLD0	—	—	—	—
(M5, 3h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DACI	—	—	—	—	DACI.11	DACI.10	DACI.9	DACI.8	DACI.7	DACI.6	DACI.5	DACI.4	DACI.3	DACI.2	DACI.1	DACI.0
(M5, 4h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DACO	—	—	—	—	DACO.11	DACO.10	DACO.9	DACO.8	DACO.7	DACO.6	DACO.5	DACO.4	DACO.3	DACO.2	DACO.1	DACO.0
(M5, 6h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ADCD	—	—	—	—	ADCD.11	ADCD.10	ADCD.9	ADCD.8	ADCD.7	ADCD.6	ADCD.5	ADCD.4	ADCD.3	ADCD.2	ADCD.1	ADCD.0
(M5, 8h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TSO	TSO.15	TSO.14	TSO.13	TSO.12	TSO.11	TSO.10	TSO.9	TSO.8	TSO.7	TSO.6	TSO.5	TSO.4	TSO.3	TSO.2	TSO.1	TSO.0
(M5, 9h)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AIE	—	—	—	—	—	—	—	—	—	HFFIE	VIOBIE	DVBIE	—	AORIE	ADCIE	—
(M5, Ah)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ASR	VIOLVL	DVLVL	—	—	XHFRY	—	—	—	—	HFFINT	VIOBI	DVBI	—	ADCOV	ADCRY	—
(M5, Bh)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
OSCC	—	—	—	—	HFOC1	HFOC0	HFIC1	HFIC0	ADCCD2	ADCCD1	ADCCD0	—	—	EXTHF	RCE	HFE
(M5, Ch)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bits indicated by "—" are unused.

Bits indicated by "ST" reflect the input signal state.

Bits indicated by "S" are only affected by a POR and not by other forms of reset. These bits are set to 0 after a POR.

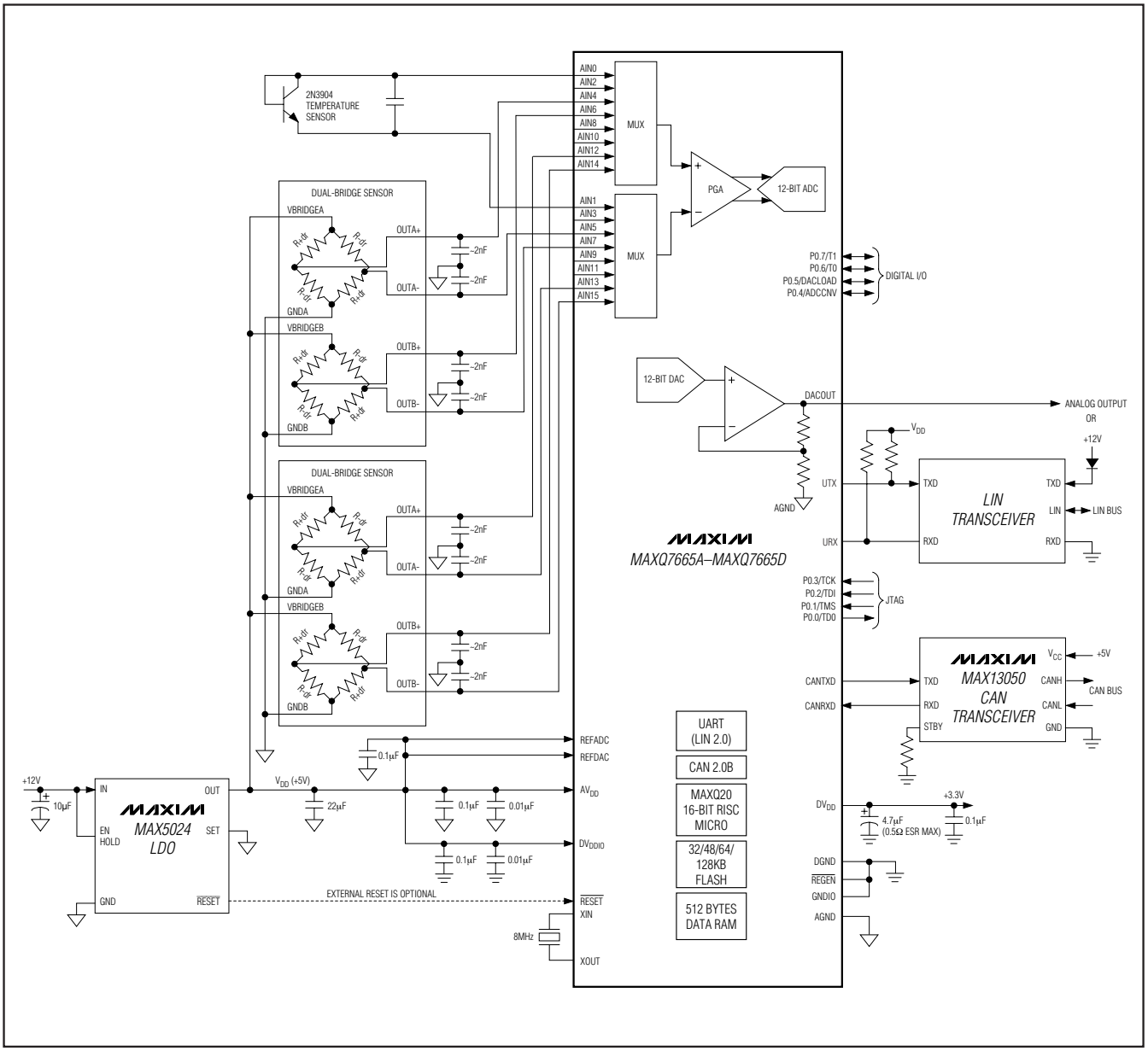
Bits indicated by "DB" have read/write access only in background or debug mode. These bits are cleared after a POR.

Bits indicated by "DW" are only written to in debug mode. These bits are cleared after a POR.

The OSCC register is cleared to 0002h after a POR and is not affected by other forms of reset.

16-Bit RISC Microcontroller-Based Smart Data-Acquisition Systems

Typical Operating Circuit



Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	3/08	Initial release	—
1	10/08	Restricted minimum clock speed	1, 2, 5, 6, 7, 9, 14, 23, 35–38, 40, 44