

# XC886/888CLM

8-Bit Single-Chip Microcontroller

Preliminary

8bit

Microcontrollers



Never stop thinking

**Edition 2006-02**

**Published by Infineon Technologies AG,  
81726 München, Germany**

**© Infineon Technologies AG 2006.  
All Rights Reserved.**

**Attention please!**

The information herein is given to describe certain components and shall not be considered as a guarantee of characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

**Information**

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office ([www.infineon.com](http://www.infineon.com)).

**Warnings**

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

# XC886/888CLM

8-Bit Single-Chip Microcontroller

Preliminary

Microcontrollers



Never stop thinking

Previous Version:

Page	Subjects (major changes since last revision)

**We Listen to Your Comments**

Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document.

Please send your proposal (including a reference to this document) to:

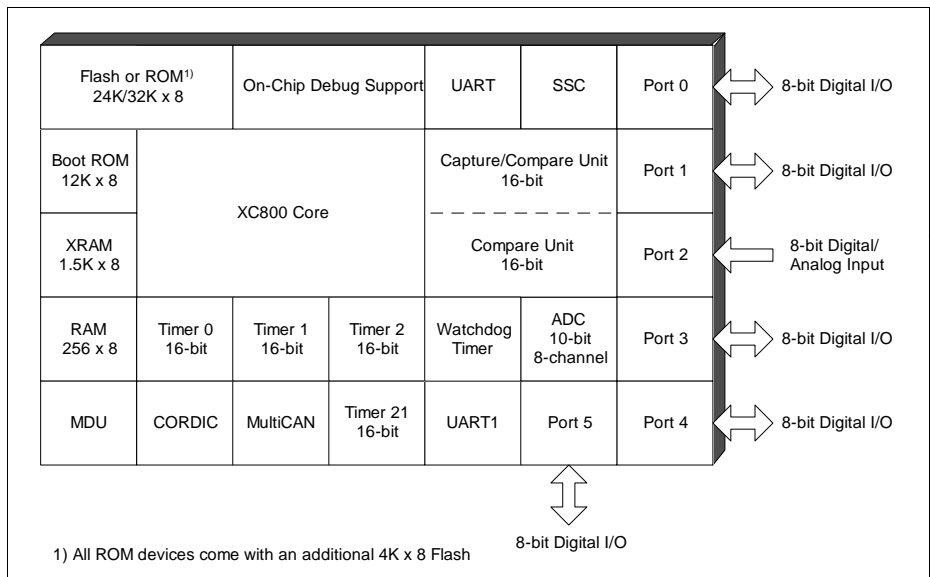
**[mcdocu.comments@infineon.com](mailto:mcdocu.comments@infineon.com)**



# 1 Summary of Features

- High-performance XC800 Core
  - compatible with standard 8051 processor
  - two clocks per machine cycle architecture (for memory access without wait state)
  - two data pointers
- On-chip memory
  - 12 Kbytes of Boot ROM
  - 256 bytes of RAM
  - 1.5 Kbytes of XRAM
  - 24/32 Kbytes of Flash; or  
24/32 Kbytes of ROM, with additional 4 Kbytes of Flash  
(includes memory protection strategy)
- I/O port supply at 3.3 V or 5.0 V and core logic supply at 2.5 V (generated by embedded voltage regulator)

(more features on next page)



**Figure 1 XC886/888 Functional Units**

---

**Summary of Features**

Features (continued):

- Power-on reset generation
- Brownout detection for core logic supply
- On-chip OSC and PLL for clock generation
  - PLL loss-of-lock detection
- Power saving modes
  - slow-down mode
  - idle mode
  - power-down mode with wake-up capability via RXD or EXINT0
  - clock gating control to each peripheral
- Programmable 16-bit Watchdog Timer (WDT)
- Six ports
  - 34/48 pins as digital I/O
  - 8 pins as digital/analog input
- 8-channel, 10-bit ADC
- Four 16-bit timers
  - Timer 0 and Timer 1 (T0 and T1)
  - Timer 2 and Timer 21
- Multiplication/Division Unit for arithmetic operations (MDU)
- CORDIC Coprocessor for computation of trigonometric, hyperbolic and linear functions
- MultiCAN with 2 nodes, 32 message objects (MCAN)
- Capture/compare unit for PWM signal generation (CCU6)
- Two full-duplex serial interfaces (UART and UART1)
- Synchronous serial channel (SSC)
- On-chip debug support
  - 1 Kbyte of monitor ROM (part of the 12-Kbyte Boot ROM)
  - 64 bytes of monitor RAM
- Packages:
  - PG-TQFP-48
  - PG-TQFP-64
- Temperature range  $T_A$ :
  - SAF (-40 to 85 °C)
  - SAK (-40 to 125 °C)

**Summary of Features**
**XC886/888 Variant Devices**

The XC886/888 product family features devices with different configurations, program memory sizes, package options, temperature and quality profiles (Automotive or Industrial), to offer cost-effective solutions for different application requirements.

The list of XC886/888 device configurations are summarized in [Table 1](#). For each configuration, 2 types of packages are available:

- PG-TQFP-48, which is denoted by XC886 and;
- PG-TQFP-64, which is denoted by XC888.

**Table 1 Device Configuration**

Device Name	CAN Module	LIN BSL Support	MDU Module
XC886/888	No	No	No
XC886/888C	Yes	No	No
XC886/888CM	Yes	No	Yes
XC886/888LM	No	Yes	Yes
XC886/888CLM	Yes	Yes	Yes

From these 10 different combinations of configuration and package type, each are further made available in 6 sales types, which are grouped according to program memory sizes, temperature and quality profiles (Automotive or Industrial), as shown in [Table 2](#).

**Table 2 Device Profile**

Sales Type	Device Type	Program Memory Size (Kbytes)	Temperature Profile (°C)	Quality Profile
SAK-XC886*/888*-8FFA	Flash	32	-40 to 125	Automotive
SAK-XC886*/888*-6FFA	Flash	24	-40 to 125	Automotive
SAF-XC886*/888*-8FFA	Flash	32	-40 to 85	Automotive
SAF-XC886*/888*-6FFA	Flash	24	-40 to 85	Automotive
SAF-XC886*/888*-8FFI	Flash	32	-40 to 85	Industrial
SAF-XC886*/888*-6FFI	Flash	24	-40 to 85	Industrial

*Note: The asterisk (\*) above denotes the device configuration letters from [Table 1](#). Corresponding ROM derivatives will be available on request.*

**Ordering Information**

The ordering code for Infineon Technologies microcontrollers provides an exact reference to the required product. This ordering code identifies:

- The derivative itself, i.e. its function set
- the specified temperature range
- the package and the type of delivery

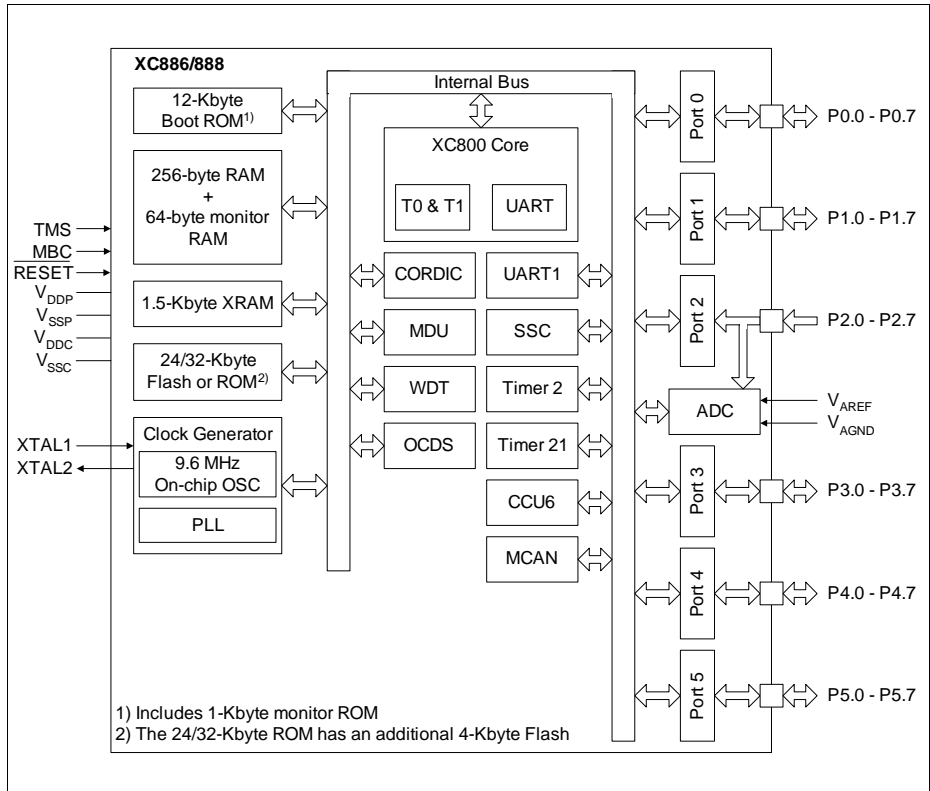
For the available ordering codes for the XC886/888, please refer to the “**Product Catalog Microcontrollers**” which summarizes all available microcontroller variants.

*Note: The ordering codes for the Mask-ROM versions are defined for each product after verification of the respective ROM code.*



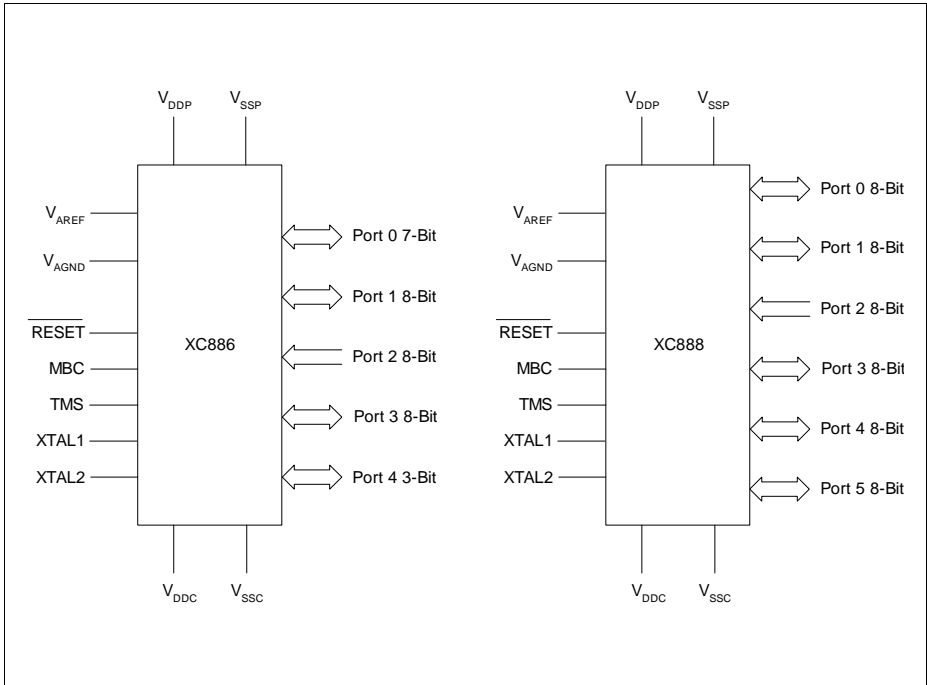
## 2 General Device Information

### 2.1 Block Diagram



**Figure 2 XC886/888 Block Diagram**

## 2.2 Logic Symbol



**Figure 3 XC886/888 Logic Symbol**

### 2.3 Pin Configuration

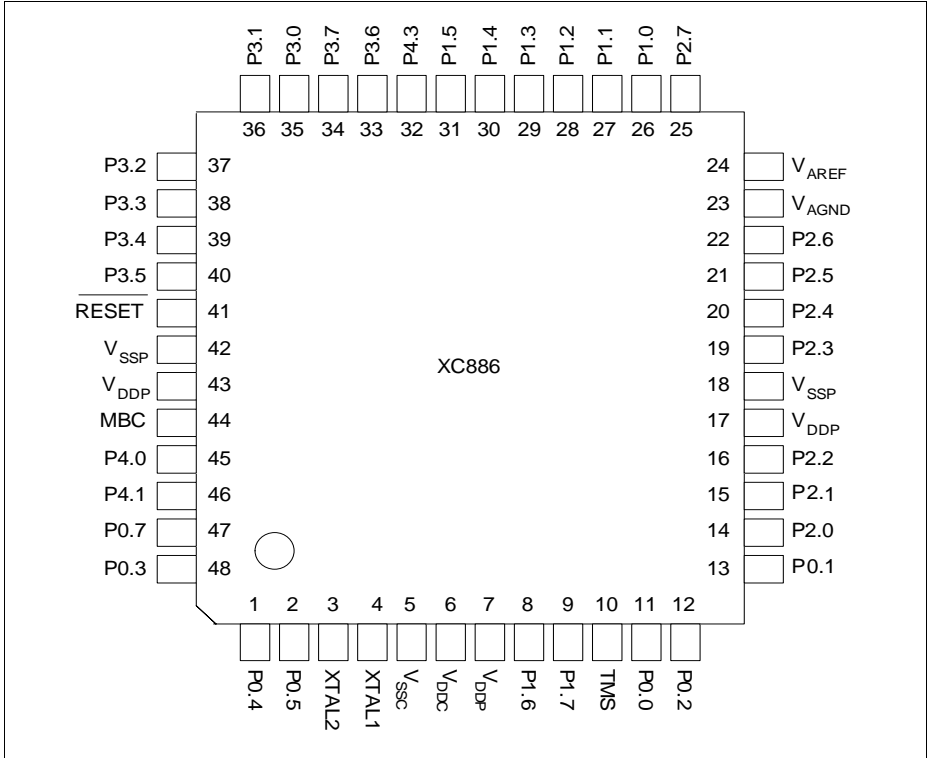
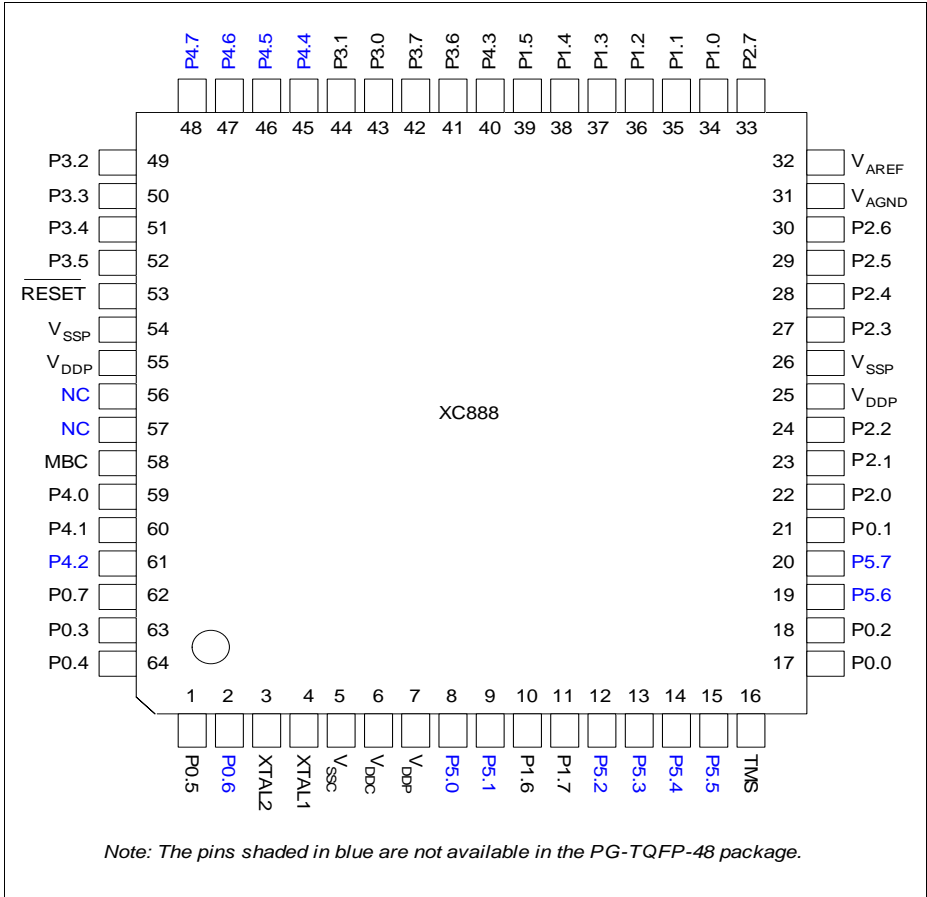


Figure 4 XC886 Pin Configuration, PG-TQFP-48 Package (top view)



**Figure 5** XC888 Pin Configuration, PG-TQFP-64 Package (top view)

**2.4 Pin Definitions and Functions**
**Table 3 Pin Definitions and Functions**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
<b>P0</b>		I/O		<b>Port 0</b> Port 0 is an 8-bit bidirectional general purpose I/O port. It can be used as alternate functions for the JTAG, CCU6, UART, UART1, Timer 2, Timer 21, MCAN and SSC.
P0.0	11/17		Hi-Z	TCK_0 JTAG Clock Input T12HR_1 CCU6 Timer 12 Hardware Run Input CC61_1 Input/Output of Capture/Compare channel 1 CLKOUT_0 Clock Output
P0.1	13/21		Hi-Z	RXDO_1 UART Transmit Data Output TDI_0 JTAG Serial Data Input T13HR_1 CCU6 Timer 13 Hardware Run Input RXD_1 UART Receive Data Input RXDC1_0 MCAN Node 1 Receiver Input COUT61_1 Output of Capture/Compare channel 1 EXF2_1 Timer 2 External Flag Output
P0.2	12/18		PU	CTRAP_2 CCU6 Trap Input TDO_0 JTAG Serial Data Output TXD_1 UART Transmit Data Output/ Clock Output TXDC1_0 MCAN Node 1 Transmitter Output
P0.3	48/63		Hi-Z	SCK_1 SSC Clock Input/Output COUT63_1 Output of Capture/Compare channel 3 RXDO1_0 UART1 Transmit Data Output

**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
P0.4	1/64		Hi-Z	MTSR_1 SSC Master Transmit Output/ Slave Receive Input CC62_1 Input/Output of Capture/ Compare channel 2 TXD1_0 UART1 Transmit Data Output/ Clock Output
P0.5	2/1		Hi-Z	MRST_1 SSC Master Receive Input/ Slave Transmit Output EXINT0_0 External Interrupt Input 0 T2EX1_1 Timer 21 External Trigger Input RXD1_0 UART1 Receive Data Input COUT62_1 Output of Capture/Compare channel 2
P0.6	-/2		PU	GPIO
P0.7	47/62		PU	CLKOUT_1 Clock Output

**General Device Information**
**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
<b>P1</b>		I/O		<b>Port 1</b> Port 1 is an 8-bit bidirectional general purpose I/O port. It can be used as alternate functions for the JTAG, CCU6, UART, Timer 0, Timer 1, Timer 2, Timer 21, MCAN and SSC.
P1.0	26/34		PU	RXD_0     UART Receive Data Input T2EX       Timer 2 External Trigger Input RXDC0_0   MCAN Node 0 Receiver Input
P1.1	27/35		PU	EXINT3     External Interrupt Input 3 T0_1       Timer 0 Input TDO_1      JTAG Serial Data Output TXD_0      UART Transmit Data Output/ Clock Output TXDC0_0    MCAN Node 0 Transmitter Output
P1.2	28/36		PU	SCK_0      SSC Clock Input/Output
P1.3	29/37		PU	MTRS_0     SSC Master Transmit Output/ Slave Receive Input TXDC1_3    MCAN Node 1 Transmitter Output
P1.4	30/38		PU	MRST_0     SSC Master Receive Input/ Slave Transmit Output EXINT0_1   External Interrupt Input 6 RXDC1_3    MCAN Node 1 Receiver Input
P1.5	31/39		PU	CCPOS0_1   CCU6 Hall Input 0 EXINT5     External Interrupt Input 5 T1_1       Timer 1 Input EXF2_0     Timer 2 External Flag Output RXDO_0     UART Transmit Data Output

General Device Information

**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
P1.6	8/10		PU	CCPOS1_1 CCU6 Hall Input 1 T12HR_0 CCU6 Timer 12 Hardware Run Input EXINT6_0 External Interrupt Input 6 RXDC0_2 MCAN Node 0 Receiver Input T21_1 Timer 21 Input
P1.7	9/11		PU	CCPOS2_1 CCU6 Hall Input 2 T13HR_0 CCU6 Timer 13 Hardware Run Input T2_1 Timer 2 Input TXDC0_2 MCAN Node 0 Transmitter Output  P1.5 and P1.6 can be used as a software chip select output for the SSC.



**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
<b>P2</b>		I		<b>Port 2</b> Port 2 is an 8-bit general purpose input-only port. It can be used as alternate functions for the digital inputs of the JTAG and CCU6. It is also used as the analog inputs for the ADC.
P2.0	14/22		Hi-Z	CCPOS0_0 CCU6 Hall Input 0 EXINT1_0 External Interrupt Input 1 T12HR_2 CCU6 Timer 12 Hardware Run Input TCK_1 JTAG Clock Input CC61_3 Input of Capture/Compare channel 1 AN0 Analog Input 0
P2.1	15/23		Hi-Z	CCPOS1_0 CCU6 Hall Input 1 EXINT2_0 External Interrupt Input 2 T13HR_2 CCU6 Timer 13 Hardware Run Input TDI_1 JTAG Serial Data Input CC62_3 Input of Capture/Compare channel 2 AN1 Analog Input 1
P2.2	16/24		Hi-Z	CCPOS2_0 CCU6 Hall Input 2 CTRAP_1 CCU6 Trap Input CC60_3 Input of Capture/Compare channel 0 AN2 Analog Input 2
P2.3	19/27		Hi-Z	AN3 Analog Input 3
P2.4	20/28		Hi-Z	AN4 Analog Input 4
P2.5	21/29		Hi-Z	AN5 Analog Input 5
P2.6	22/30		Hi-Z	AN6 Analog Input 6
P2.7	25/33		Hi-Z	AN7 Analog Input 7

**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
<b>P3</b>		I/O		<b>Port 3</b> Port 3 is an 8-bit bidirectional general purpose I/O port. It can be used as alternate functions for CCU6, UART1, Timer 21 and MCAN.
P3.0	35/43		Hi-Z	CCPOS1_2 CCU6 Hall Input 1 CC60_0 Input/Output of Capture/Compare channel 0
P3.1	36/44		Hi-Z	RXDO1_1 UART1 Transmit Data Output CCPOS0_2 CCU6 Hall Input 0 CC61_2 Input/Output of Capture/Compare channel 1 COUT60_0 Output of Capture/Compare channel 0 TXD1_1 UART1 Transmit Data Output/Clock Output
P3.2	37/49		Hi-Z	CCPOS2_2 CCU6 Hall Input 2 RXDC1_1 MCAN Node 0 Receiver Input RXD1_1 UART1 Receive Data Input CC61_0 Input/Output of Capture/Compare channel 1
P3.3	38/50		Hi-Z	COUT61_0 Output of Capture/Compare channel 1 TXDC1_1 MCAN Node 1 Transmitter Output
P3.4	39/51		Hi-Z	CC62_0 Input/Output of Capture/Compare channel 2 RXDC0_1 MCAN Node 0 Receiver Input T2EX1_0 Timer 21 External Trigger Input
P3.5	40/52		Hi-Z	COUT62_0 Output of Capture/Compare channel 2 EXF21_0 Timer 21 External Flag Output TXDC0_1 MCAN Node 0 Transmitter Output
P3.6	33/41		PD	CTRAP_0 CCU6 Trap Input

**General Device Information**
**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
P3.7	34/42		Hi-Z	EXINT4 External Interrupt Input 4 COU63_0 Output of Capture/Compare channel 3
<b>P4</b>		I/O		<b>Port 4</b> Port 4 is an 8-bit bidirectional general purpose I/O port. It can be used as alternate functions for CCU6, Timer 0, Timer 1, Timer 21 and MCAN.
P4.0	45/59		Hi-Z	RXDC0_3 MCAN Node 0 Receiver Input CC60_1 Output of Capture/Compare channel 0
P4.1	46/60		Hi-Z	TXDC0_3 MCAN Node 0 Transmitter Output COU60_1 Output of Capture/Compare channel 0
P4.2	-/61		PU	EXINT6_1 External Interrupt Input 6 T21_0 Timer 21 Input
P4.3	32/40		Hi-Z	EXF21_1 Timer 21 External Flag Output COU63_2 Output of Capture/Compare channel 3
P4.4	-/45		Hi-Z	CCPOS0_3 CCU6 Hall Input 0 T0_0 Timer 0 Input CC61_4 Output of Capture/Compare channel 1
P4.5	-/46		Hi-Z	CCPOS1_3 CCU6 Hall Input 1 T1_0 Timer 1 Input COU61_2 Output of Capture/Compare channel 1
P4.6	-/47		Hi-Z	CCPOS2_3 CCU6 Hall Input 2 T2_0 Timer 2 Input CC62_2 Output of Capture/Compare channel 2
P4.7	-/48		Hi-Z	CTRAP_3 CCU6 Trap Input COU62_2 Output of Capture/Compare channel 2

**Table 3 Pin Definitions and Functions (cont'd)**

Symbol	Pin Number (TQFP-48/64)	Type	Reset State	Function
<b>P5</b>		I/O		<b>Port 5</b> Port 5 is an 8-bit bidirectional general purpose I/O port. It can be used as alternate functions for UART, UART1 and JTAG.
P5.0	-/8		PU	EXINT1_1 External Interrupt Input 1
P5.1	-/9		PU	EXINT2_1 External Interrupt Input 2
P5.2	-/12		PU	RXD_2 UART Receive Data Input
P5.3	-/13		PU	TXD_2 UART Transmit Data Output/ Clock Output
P5.4	-/14		PU	RXDO_2 UART Transmit Data Output
P5.5	-/15		PU	TDO_2 JTAG Serial Data Output TXD1_2 UART1 Transmit Data Output/ Clock Output
P5.6	-/19		PU	TCK_2 JTAG Clock Input RXDO1_2 UART1 Transmit Data Output
P5.7	-/20		PU	TDI_2 JTAG Serial Data Input RXD1_2 UART1 Receive Data Input
<b>V<sub>DDP</sub></b>	7, 17, 43/ 7, 25, 55	–	–	<b>I/O Port Supply (3.3 or 5.0 V)</b>
<b>V<sub>SSP</sub></b>	18, 42/26, 54	–	–	<b>I/O Port Ground</b>
<b>V<sub>DDC</sub></b>	6/6	–	–	<b>Core Supply Monitor (2.5 V)</b>
<b>V<sub>SSC</sub></b>	5/5	–	–	<b>Core Supply Ground</b>
<b>V<sub>AREF</sub></b>	24/32	–	–	<b>ADC Reference Voltage</b>
<b>V<sub>AGND</sub></b>	23/31	–	–	<b>ADC Reference Ground</b>
<b>XTAL1</b>	4/4	I	Hi-Z	<b>External Oscillator Input (backup for on-chip OSC, normally NC)</b>
<b>XTAL2</b>	3/3	O	Hi-Z	<b>External Oscillator Output (backup for on-chip OSC, normally NC)</b>
<b>TMS</b>	10/16	I	PD	<b>Test Mode Select</b>
<b>RESET</b>	41/53	I	PU	<b>Reset Input</b>
<b>MBC</b>	44/58	I	PU	<b>Monitor &amp; BootStrap Loader Control</b>
<b>NC</b>	-/21, 59, 60	–	–	<b>No Connection</b>

## 3 Functional Description

### 3.1 Processor Architecture

The XC886/888 is based on a high-performance 8-bit Central Processing Unit (CPU) that is compatible with the standard 8051 processor. While the standard 8051 processor is designed around a 12-clock machine cycle, the XC886/888 CPU uses a 2-clock machine cycle. This allows fast access to ROM or RAM memories without wait state. Access to the Flash memory, however, requires an additional wait state (one machine cycle). The instruction set consists of 45% one-byte, 41% two-byte and 14% three-byte instructions.

The XC886/888 CPU provides a range of debugging features, including basic stop/start, single-step execution, breakpoint support and read/write access to the data memory, program memory and SFRs.

**Figure 6** shows the CPU functional blocks.

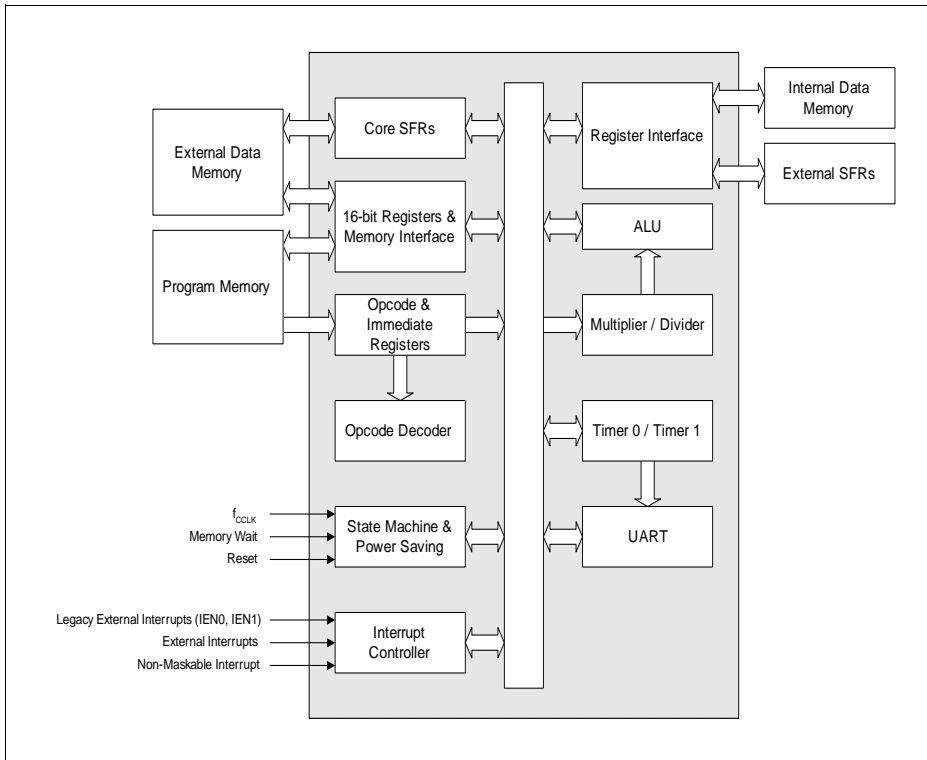


Figure 6 CPU Block Diagram

### 3.2 Memory Organization

The XC886/888 CPU operates in the following five address spaces:

- 12 Kbytes of Boot ROM program memory
- 256 bytes of internal RAM data memory
- 1.5 Kbytes of XRAM memory  
(XRAM can be read/written as program memory or external data memory)
- a 128-byte Special Function Register area
- 24/32 Kbytes of Flash program memory (Flash devices); or  
24/32 Kbytes of ROM program memory, with additional 4 Kbytes of Flash  
(ROM devices)

Figure 7 illustrates the memory address spaces of the 32-Kbyte Flash devices. For the 24-Kbyte Flash devices, the shaded banks are not available.

Functional Description

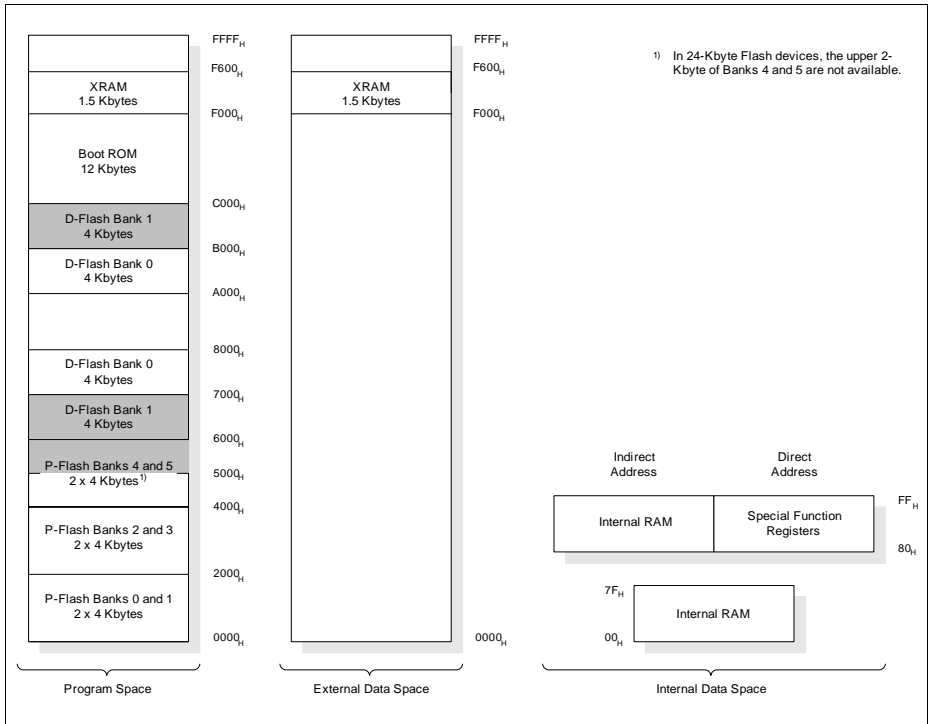


Figure 7 Memory Map of XC886/888 Flash Device

### 3.2.1 Memory Protection Strategy

The XC886/888 memory protection strategy includes:

- Read-out protection: The user is able to protect the contents in the Flash (for Flash devices) and ROM (for ROM devices) memory from being read
- Flash program and erase protection (for Flash devices only)

Flash memory protection modes are available only for Flash devices:

- Mode 0: Only the P-Flash is protected; the D-Flash is unprotected
- Mode 1: Both the P-Flash and D-Flash are protected

The selection of each protection mode and the restrictions imposed are summarized in [Table 4](#).

**Table 4 Flash Protection Modes**

Mode	0	1
<b>Activation</b>	Program a valid password via BSL mode 6	
<b>Selection</b>	MSB of password = 0	MSB of password = 1
<b>P-Flash contents can be read by</b>	Read instructions in the P-Flash	Read instructions in the P-Flash or D-Flash
<b>P-Flash program and erase</b>	Not possible	Not possible
<b>D-Flash contents can be read by</b>	Read instructions in any program memory	Read instructions in the P-Flash or D-Flash
<b>D-Flash program</b>	Possible	Not possible
<b>D-Flash erase</b>	Possible, on the condition that bit DFLASHEN in register MISC_CON is set to 1 prior to each erase operation	Not possible

BSL mode 6, which is used for enabling Flash protection, can also be used for disabling Flash protection. Here, the programmed password must be provided by the user. A password match triggers an automatic erase of the protected P-Flash and D-Flash contents, including the programmed password. The Flash protection is then disabled upon the next reset.

Although no protection scheme can be considered infallible, the XC886/888 memory protection strategy provides a very high level of protection for a general purpose microcontroller.

*Note: If ROM read-out protection is enabled, only read instructions in the ROM memory can target the ROM contents.*



### 3.2.2 Special Function Register

The Special Function Registers (SFRs) occupy direct internal data memory space in the range 80<sub>H</sub> to FF<sub>H</sub>. All registers, except the program counter, reside in the SFR area. The SFRs include pointers and registers that provide an interface between the CPU and the on-chip peripherals. As the 128-SFR range is less than the total number of registers required, address extension mechanisms are required to increase the number of addressable SFRs. The address extension mechanisms include:

- Mapping
- Paging

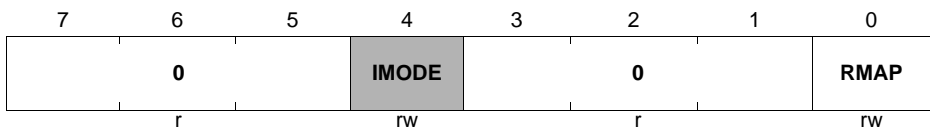
#### 3.2.2.1 Address Extension by Mapping

Address extension is performed at the system level by mapping. The SFR area is extended into two portions: the standard (non-mapped) SFR area and the mapped SFR area. Each portion supports the same address range 80<sub>H</sub> to FF<sub>H</sub>, bringing the number of addressable SFRs to 256. The extended address range is not directly controlled by the CPU instruction itself, but is derived from bit RMAP in the system control register SYSCON0 at address 8F<sub>H</sub>. To access SFRs in the mapped area, bit RMAP in SFR SYSCON0 must be set. Alternatively, the SFRs in the standard area can be accessed by clearing bit RMAP. The SFR area can be selected as shown in [Figure 8](#).

#### SYSCON0

##### System Control Register 0

Reset Value: 00<sub>H</sub>



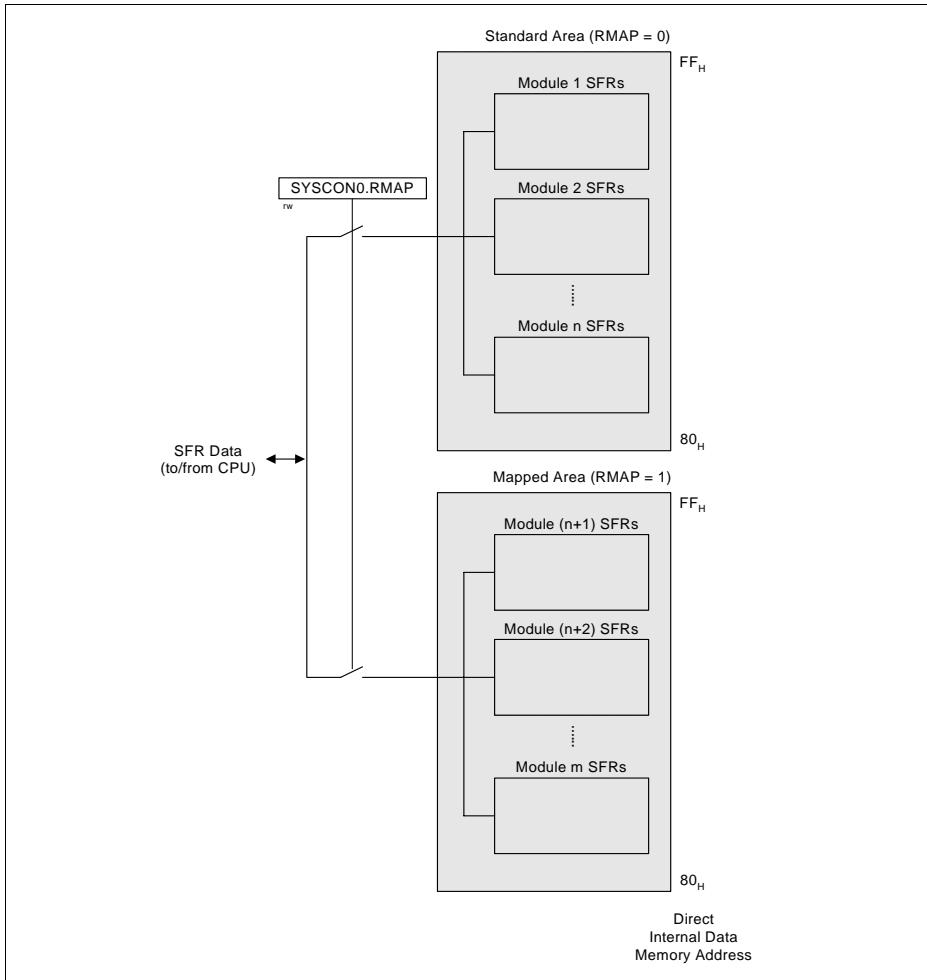
The functions of the shaded bits are not described here

Field	Bits	Type	Description
<b>RMAP</b>	0	rw	<b>Special Function Register Map Control</b> 0 The access to the standard SFR area is enabled. 1 The access to the mapped SFR area is enabled.
<b>0</b>	[7:5], [3:1]	r	<b>Reserved</b> Returns 0 if read; should be written with 0.

**Functional Description**

*Note: The RMAP bit must be cleared/set by ANL or ORL instructions. The rest bits of SYSCON0 should not be modified.*

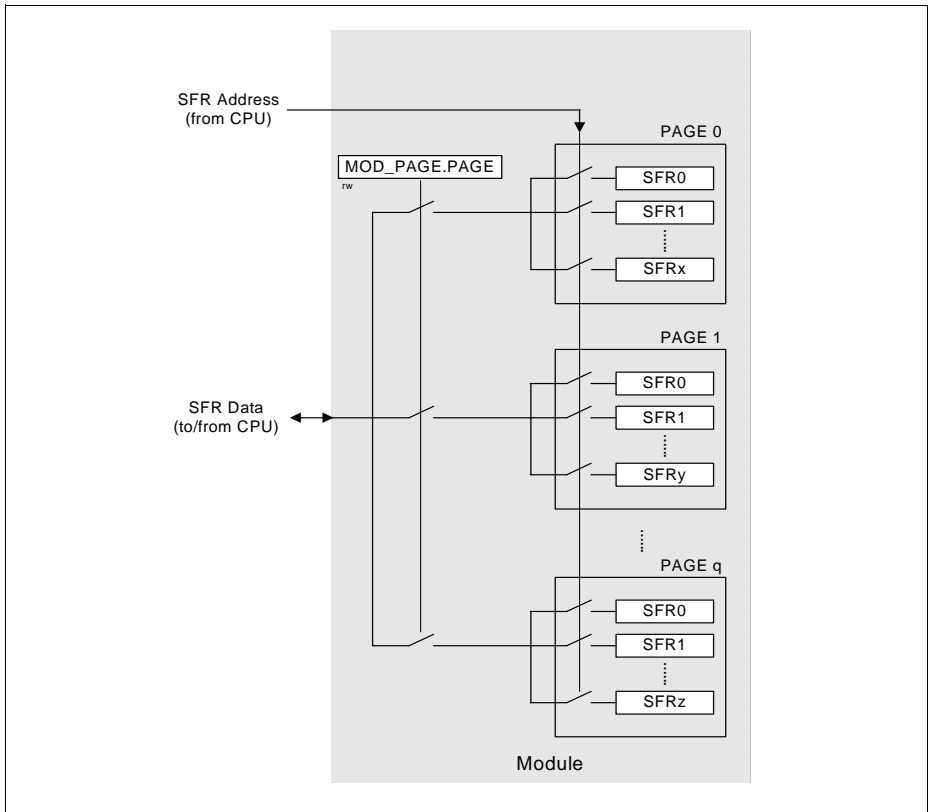
As long as bit RMAP is set, the mapped SFR area can be accessed. This bit is not cleared automatically by hardware. Thus, before standard/mapped registers are accessed, bit RMAP must be cleared/set, respectively, by software.



**Figure 8 Address Extension by Mapping**

### 3.2.2.2 Address Extension by Paging

Address extension is further performed at the module level by paging. With the address extension by mapping, the XC886/888 has a 256-SFR address range. However, this is still less than the total number of SFRs needed by the on-chip peripherals. To meet this requirement, some peripherals have a built-in local address extension mechanism for increasing the number of addressable SFRs. The extended address range is not directly controlled by the CPU instruction itself, but is derived from bit field PAGE in the module page register MOD\_PAGE. Hence, the bit field PAGE must be programmed before accessing the SFR of the target module. Each module may contain a different number of pages and a different number of SFRs per page, depending on the specific requirement. Besides setting the correct RMAP bit value to select the SFR area, the user must also ensure that a valid PAGE is selected to target the desired SFR. A page inside the extended address range can be selected as shown in **Figure 9**.



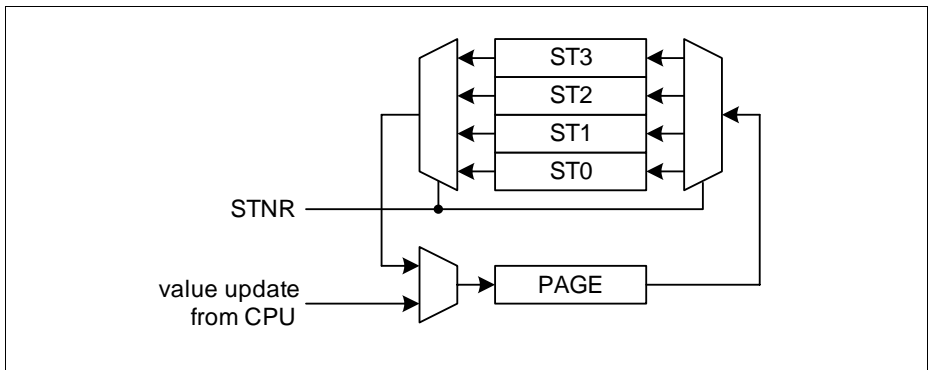
**Figure 9** Address Extension by Paging

**Functional Description**

In order to access a register located in a page different from the actual one, the current page must be left. This is done by reprogramming the bit field PAGE in the page register. Only then can the desired access be performed.

If an interrupt routine is initiated between the page register access and the module register access, and the interrupt needs to access a register located in another page, the current page setting can be saved, the new one programmed and finally, the old page setting restored. This is possible with the storage fields STx (x = 0 - 3) for the save and restore action of the current page setting. By indicating which storage bit field should be used in parallel with the new page value, a single write operation can:

- Save the contents of PAGE in STx before overwriting with the new value (this is done in the beginning of the interrupt routine to save the current page setting and program the new page number); or
- Overwrite the contents of PAGE with the contents of STx, ignoring the value written to the bit positions of PAGE (this is done at the end of the interrupt routine to restore the previous page setting before the interrupt occurred)



**Figure 10 Storage Elements for Paging**

With this mechanism, a certain number of interrupt routines (or other routines) can perform page changes without reading and storing the previously used page information. The use of only write operations makes the system simpler and faster. Consequently, this mechanism significantly improves the performance of short interrupt routines.

The XC886/888 supports local address extension for:

- Parallel Ports
- Analog-to-Digital Converter (ADC)
- Capture/Compare Unit 6 (CCU6)
- System Control Registers

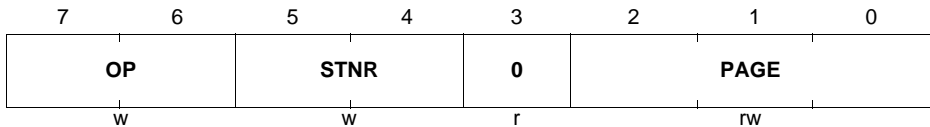
Functional Description

The page register has the following definition:

**MOD\_PAGE**

**Page Register for module MOD**

**Reset Value: 00<sub>H</sub>**



Field	Bits	Type	Description
<b>PAGE</b>	[2:0]	rw	<p><b>Page Bits</b></p> <p>When written, the value indicates the new page. When read, the value indicates the currently active page.</p>
<b>STNR</b>	[5:4]	w	<p><b>Storage Number</b></p> <p>This number indicates which storage bit field is the target of the operation defined by bit field OP. If OP = 10<sub>B</sub>, the contents of PAGE are saved in STx before being overwritten with the new value. If OP = 11<sub>B</sub>, the contents of PAGE are overwritten by the contents of STx. The value written to the bit positions of PAGE is ignored.</p> <p>00 ST0 is selected. 01 ST1 is selected. 10 ST2 is selected. 11 ST3 is selected.</p>

**Functional Description**

Field	Bits	Type	Description
<b>OP</b>	[7:6]	w	<p><b>Operation</b></p> <p>0X Manual page mode. The value of STNR is ignored and PAGE is directly written.</p> <p>10 New page programming with automatic page saving. The value written to the bit positions of PAGE is stored. In parallel, the previous contents of PAGE are saved in the storage bit field STx indicated by STNR.</p> <p>11 Automatic restore page action. The value written to the bit positions PAGE is ignored and instead, PAGE is overwritten by the contents of the storage bit field STx indicated by STNR.</p>
<b>0</b>	3	r	<p><b>Reserved</b></p> <p>Returns 0 if read; should be written with 0.</p>

**Functional Description**
**3.2.3 Bit Protection Scheme**

The bit protection scheme prevents direct software writing of selected bits (i.e., protected bits) using the PASSWD register. When the bit field MODE is 11<sub>B</sub>, writing 10011<sub>B</sub> to the bit field PASS opens access to writing of all protected bits, and writing 10101<sub>B</sub> to the bit field PASS closes access to writing of all protected bits. Note that access is opened for maximum 32 CCLKs if the “close access” password is not written. If “open access” password is written again before the end of 32 CCLK cycles, there will be a recount of 32 CCLK cycles. The protected bits include the N- and K-Divider bits, NDIV and KDIV; the Watchdog Timer enable bit, WDTEN; and the power-down and slow-down enable bits, PD and SD.

**PASSWD**
**Password Register**
**Reset Value: 07<sub>H</sub>**

	7	6	5	4	3	2	1	0
<b>PASS</b>						<b>PROTECT _S</b>	<b>MODE</b>	
wh						rh	rw	

Field	Bits	Type	Description
<b>MODE</b>	[1:0]	rw	<b>Bit Protection Scheme Control bits</b> 00 Scheme Disabled 11 Scheme Enabled (default) Others: Scheme Enabled These two bits cannot be written directly. To change the value between 11 <sub>B</sub> and 00 <sub>B</sub> , the bit field PASS must be written with 11000 <sub>B</sub> ; only then, will the MODE[1:0] be registered.
<b>PROTECT_S</b>	2	rh	<b>Bit Protection Signal Status bit</b> This bit shows the status of the protection. 0 Software is able to write to all protected bits. 1 Software is unable to write to any protected bits.
<b>PASS</b>	[7:3]	wh	<b>Password bits</b> The Bit Protection Scheme only recognizes three patterns. 11000 <sub>B</sub> Enables writing of the bit field MODE. 10011 <sub>B</sub> Opens access to writing of all protected bits. 10101 <sub>B</sub> Closes access to writing of all protected bits.

**Functional Description**
**3.2.4 XC886/888 Register Overview**

The SFRs of the XC886/888 are organized into groups according to their functional units. The contents (bits) of the SFRs are summarized in **Table 5** to **Table 18**, with the addresses of the bitaddressable SFRs appearing in bold typeface.

The CPU SFRs can be accessed in both the standard and mapped memory areas (RMAP = 0 or 1).

**Table 5 CPU Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0		
RMAP = 0 or 1												
81 <sub>H</sub>	<b>SP</b> Stack Pointer Register	Reset: 07 <sub>H</sub>	Bit Field	SP								
			Type	rw								
82 <sub>H</sub>	<b>DPL</b> Data Pointer Register Low	Reset: 00 <sub>H</sub>	Bit Field	DPL7	DPL6	DPL5	DPL4	DPL3	DPL2	DPL1	DPL0	
			Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
83 <sub>H</sub>	<b>DPH</b> Data Pointer Register High	Reset: 00 <sub>H</sub>	Bit Field	DPH7	DPH6	DPH5	DPH4	DPH3	DPH2	DPH1	DPH0	
			Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
87 <sub>H</sub>	<b>PCON</b> Power Control Register	Reset: 00 <sub>H</sub>	Bit Field	SMOD	0		GF1	GF0	0	IDLE		
			Type	rw	r		rw	rw	r	rw		
88 <sub>H</sub>	<b>TCON</b> Timer Control Register	Reset: 00 <sub>H</sub>	Bit Field	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
			Type	rwh	rw	rwh	rw	rwh	rw	rwh	rw	
89 <sub>H</sub>	<b>TMOD</b> Timer Mode Register	Reset: 00 <sub>H</sub>	Bit Field	GATE1	0	T1M	GATE0	0	TOM			
			Type	rw	r	rw	rw	r	rw			
8A <sub>H</sub>	<b>TL0</b> Timer 0 Register Low	Reset: 00 <sub>H</sub>	Bit Field	VAL								
			Type	rwh								
8B <sub>H</sub>	<b>TL1</b> Timer 1 Register Low	Reset: 00 <sub>H</sub>	Bit Field	VAL								
			Type	rwh								
8C <sub>H</sub>	<b>TH0</b> Timer 0 Register High	Reset: 00 <sub>H</sub>	Bit Field	VAL								
			Type	rwh								
8D <sub>H</sub>	<b>TH1</b> Timer 1 Register High	Reset: 00 <sub>H</sub>	Bit Field	VAL								
			Type	rwh								
98 <sub>H</sub>	<b>SCON</b> Serial Channel Control Register	Reset: 00 <sub>H</sub>	Bit Field	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
			Type	rw	rw	rw	rw	rw	rwh	rwh	rwh	
99 <sub>H</sub>	<b>SBUF</b> Serial Data Buffer Register	Reset: 00 <sub>H</sub>	Bit Field	VAL								
			Type	rwh								
A2 <sub>H</sub>	<b>EO</b> Extended Operation Register	Reset: 00 <sub>H</sub>	Bit Field	0		TRAP_EN	0			DPSEL_0		
			Type	r		rw	r			rw		
A8 <sub>H</sub>	<b>IEN0</b> Interrupt Enable Register 0	Reset: 00 <sub>H</sub>	Bit Field	EA	0	ET2	ES	ET1	EX1	ET0	EX0	
			Type	rw	r	rw	rw	rw	rw	rw	rw	
B8 <sub>H</sub>	<b>IP</b> Interrupt Priority Register	Reset: 00 <sub>H</sub>	Bit Field	0		PT2	PS	PT1	PX1	PT0	PX0	
			Type	r		rw	rw	rw	rw	rw	rw	
B9 <sub>H</sub>	<b>IPH</b> Interrupt Priority Register High	Reset: 00 <sub>H</sub>	Bit Field	0		PT2H	PSH	PT1H	PX1H	PT0H	PX0H	
			Type	r		rw	rw	rw	rw	rw	rw	
D0 <sub>H</sub>	<b>PSW</b> Program Status Word Register	Reset: 00 <sub>H</sub>	Bit Field	CY	AC	F0	RS1	RS0	OV	F1	P	
			Type	rwh	rwh	rw	rw	rw	rwh	rw	rh	
E0 <sub>H</sub>	<b>ACC</b> Accumulator Register	Reset: 00 <sub>H</sub>	Bit Field	ACC7	ACC6	ACC5	ACC4	ACC3	ACC2	ACC1	ACC0	
			Type	rw	rw	rw	rw	rw	rw	rw	rw	
E8 <sub>H</sub>	<b>IEN1</b> Interrupt Enable Register 1	Reset: 00 <sub>H</sub>	Bit Field	ECCIP_3	ECCIP_2	ECCIP_1	ECCIP_0	EXM	EX2	ESSC	EADC	
			Type	rw	rw	rw	rw	rw	rw	rw	rw	



**Functional Description**
**Table 5 CPU Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
F0 <sub>H</sub>	<b>B</b> B Register Reset: 00 <sub>H</sub>	Bit Field	B7	B6	B5	B4	B3	B2	B1	B0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
F8 <sub>H</sub>	<b>IP1</b> Interrupt Priority Register 1 Reset: 00 <sub>H</sub>	Bit Field	PCCIP 3	PCCIP 2	PCCIP 1	PCCIP 0	PXM	PX2	PSSC	PADC
		Type	rw	rw	rw	rw	rw	rw	rw	rw
F9 <sub>H</sub>	<b>IPH1</b> Interrupt Priority Register 1 High Reset: 00 <sub>H</sub>	Bit Field	PCCIP 3H	PCCIP 2H	PCCIP 1H	PCCIP 0H	PXMH	PX2H	PSSCH	PADC H
		Type	rw	rw	rw	rw	rw	rw	rw	rw

The MDU SFRs can be accessed in the mapped memory area (RMAP = 1).

**Table 6 MDU Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 1											
B0 <sub>H</sub>	<b>MDUSTAT</b> MDU Status Register Reset: 00 <sub>H</sub>	Bit Field	0					BSY	IERR	IRDY	
		Type	r					rh	rwh	rwh	
B1 <sub>H</sub>	<b>MDUCON</b> MDU Control Register Reset: 00 <sub>H</sub>	Bit Field	IE	IR	RSEL	START	OPCODE				
		Type	rw	rw	rw	rwh	rw				
B2 <sub>H</sub>	<b>MD0</b> MDU Data Register 0 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
	<b>MR0</b> MDU Data Register 0 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								
B3 <sub>H</sub>	<b>MD1</b> MDU Data Register 1 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
	<b>MR1</b> MDU Data Register 1 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								
B4 <sub>H</sub>	<b>MD2</b> MDU Data Register 2 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
	<b>MR2</b> MDU Data Register 2 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								
B5 <sub>H</sub>	<b>MD3</b> MDU Data Register 3 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
	<b>MR3</b> MDU Data Register 3 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								
B6 <sub>H</sub>	<b>MD4</b> MDU Data Register 4 <i>Multiplication/Division</i> <i>Shift/Normalization</i> Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
			0	SLR	SCTR						
		rw	rw	rw							
	<b>MR4</b> MDU Data Register 4 <i>Multiplication/Division</i> <i>Shift/Normalization</i> Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								
			0	SCTR							
		rh		rh							
B7 <sub>H</sub>	<b>MD5</b> MDU Data Register 5 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rw								
	<b>MR5</b> MDU Data Register 5 Reset: 00 <sub>H</sub>	Bit Field	DATA								
		Type	rh								

**Functional Description**

The CORDIC SFRs can be accessed in the mapped memory area (RMAP = 1).

**Table 7 CORDIC Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 1											
9A <sub>H</sub>	<b>CD_CORDXL</b> CORDIC X Data Low Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAL						
			Type		rw						
9B <sub>H</sub>	<b>CD_CORDXH</b> CORDIC X Data High Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAH						
			Type		rw						
9C <sub>H</sub>	<b>CD_CORDYL</b> CORDIC Y Data Low Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAL						
			Type		rw						
9D <sub>H</sub>	<b>CD_CORDYH</b> CORDIC Y Data High Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAH						
			Type		rw						
9E <sub>H</sub>	<b>CD_CORDZL</b> CORDIC Z Data Low Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAL						
			Type		rw						
9F <sub>H</sub>	<b>CD_CORDZH</b> CORDIC Z Data High Byte	Reset: 00 <sub>H</sub>	Bit Field		DATAH						
			Type		rw						
A0 <sub>H</sub>	<b>CD_STATC</b> CORDIC Status and Data Control Register	Reset: 00 <sub>H</sub>	Bit Field	KEEPZ	KEEPY	KEEPX	DMAP	INT_E N	EOC	ERRO R	BSY
			Type	rw	rw	rw	rw	rw	rw	rh	rh
A1 <sub>H</sub>	<b>CD_CON</b> CORDIC Control Register	Reset: 00 <sub>H</sub>	Bit Field	MPS	X_USI GN	ST_MO DE	ROTV C	MODE		ST	
			Type	rw	w	rw	rw	rw	rw	rw	

The system control SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 8 System Control Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 0 or 1											
8F <sub>H</sub>	<b>SYSCON0</b> System Control Register 0	Reset: 00 <sub>H</sub>	Bit Field		0		IMODE	0		RMAP	
			Type		r		rw	r		rw	
RMAP = 0											
BF <sub>H</sub>	<b>SCU_PAGE</b> Page Register	Reset: 00 <sub>H</sub>	Bit Field	OP		STNR		0	PAGE		
			Type	w		w		r	rw		
RMAP = 0, PAGE 0											
B3 <sub>H</sub>	<b>MODPISEL</b> Peripheral Input Select Register	Reset: 00 <sub>H</sub>	Bit Field	0	URRIS H	JTAGT DIS	JTAGT CKS	EXINT 2IS	EXINT 1IS	EXINT 0IS	URRIS
			Type	r	rw	rw	rw	rw	rw	rw	rw
B4 <sub>H</sub>	<b>IRCON0</b> Interrupt Request Register 0	Reset: 00 <sub>H</sub>	Bit Field	0	EXINT 6	EXINT 5	EXINT 4	EXINT 3	EXINT 2	EXINT 1	EXINT 0
			Type	r	rwh	rwh	rwh	rwh	rwh	rwh	rwh
B5 <sub>H</sub>	<b>IRCON1</b> Interrupt Request Register 1	Reset: 00 <sub>H</sub>	Bit Field	0	CANS RC2	CANS RC1	ADCS RC1	ADCS RC0	RIR	TIR	EIR
			Type	r	rwh	rwh	rwh	rwh	rwh	rwh	rwh
B6 <sub>H</sub>	<b>IRCON2</b> Interrupt Request Register 2	Reset: 00 <sub>H</sub>	Bit Field	0			CANS RC3	0		CANS RC0	
			Type	r			rwh	r		rwh	
B7 <sub>H</sub>	<b>EXICON0</b> External Interrupt Control Register 0	Reset: F0 <sub>H</sub>	Bit Field	EXINT3		EXINT2		EXINT1		EXINT0	
			Type	rw		rw		rw		rw	
BA <sub>H</sub>	<b>EXICON1</b> External Interrupt Control Register 1	Reset: 3F <sub>H</sub>	Bit Field	0		EXINT6		EXINT5		EXINT4	
			Type	r		rw		rw		rw	

**Functional Description**
**Table 8 System Control Register Overview (cont'd)**

Addr	Register Name	Reset:	Bit	7	6	5	4	3	2	1	0	
BB <sub>H</sub>	<b>NMICON</b> NMI Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	0	NMI ECC	NMI VDDP	NMI VDD	NMI OCDS	NMI FLASH	NMI PLL	NMI WDT	
			Type	r	rw	rw	rw	rw	rw	rw		
BC <sub>H</sub>	<b>NMISR</b> NMI Status Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	0	FNMI ECC	FNMI VDDP	FNMI VDD	FNMI OCDS	FNMI FLASH	FNMI PLL	FNMI WDT	
			Type	r	rwh	rwh	rwh	rwh	rwh	rwh		
BD <sub>H</sub>	<b>BCON</b> Baud Rate Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	BGSEL		0	BRDIS	BRPRE		R		
			Type	rw		r	rw	rw		rw		
BE <sub>H</sub>	<b>BG</b> Baud Rate Timer/Reload Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	BR_VALUE								
			Type	rwh								
E9 <sub>H</sub>	<b>FDCON</b> Fractional Divider Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	BGS	SYNEN	ERRSY N	EOFSY N	BRK	NDOV	FDM	FDEN	
			Type	rw	rw	rwh	rwh	rwh	rwh	rw	rw	
EA <sub>H</sub>	<b>FDSTEP</b> Fractional Divider Reload Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	STEP								
			Type	rw								
EB <sub>H</sub>	<b>FDRES</b> Fractional Divider Result Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	RESULT								
			Type	rh								
RMAP = 0, PAGE 1												
B3 <sub>H</sub>	<b>ID</b> Identity Register	<b>Reset: 09<sub>H</sub></b>	Bit Field	PRODID					VERID			
			Type	r					r			
B4 <sub>H</sub>	<b>PMCON0</b> Power Mode Control Register 0	<b>Reset: 00<sub>H</sub></b>	Bit Field	0	WDT RST	WKRS	WK SEL	SD	PD	WS		
			Type	r	rwh	rwh	rw	rw	rwh	rw		
B5 <sub>H</sub>	<b>PMCON1</b> Power Mode Control Register 1	<b>Reset: 00<sub>H</sub></b>	Bit Field	0	CDC_D IS	CAN_D IS	MDU_ DIS	T2_DIS	CCU_ DIS	SSC_ DIS	ADC_ DIS	
			Type	r	rw	rw	rw	rw	rw	rw	rw	
B6 <sub>H</sub>	<b>OSC_CON</b> OSC Control Register	<b>Reset: 08<sub>H</sub></b>	Bit Field	0			OSC PD	XPD	OSC SS	ORDR ES	OSCR	
			Type	r			rw	rw	rw	rwh	rh	
B7 <sub>H</sub>	<b>PLL_CON</b> PLL Control Register	<b>Reset: 90<sub>H</sub></b>	Bit Field	NDIV				VCOB YP	OSC DISC	RESLD	LOCK	
			Type	rw				rw	rw	rwh	rh	
BA <sub>H</sub>	<b>CMCON</b> Clock Control Register	<b>Reset: 10<sub>H</sub></b>	Bit Field	VCO SEL	KDIV	0	FCCFG	CLKREL				
			Type	rw	rw	r	rw	rw				
BB <sub>H</sub>	<b>PASSWD</b> Password Register	<b>Reset: 07<sub>H</sub></b>	Bit Field	PASS					PROTE CT_S	MODE		
			Type	wh					rh	rw		
BC <sub>H</sub>	<b>FEAL</b> Flash Error Address Register Low	<b>Reset: 00<sub>H</sub></b>	Bit Field	ECCERRADDR								
			Type	rh								
BD <sub>H</sub>	<b>FEAH</b> Flash Error Address Register High	<b>Reset: 00<sub>H</sub></b>	Bit Field	ECCERRADDR								
			Type	rh								
BE <sub>H</sub>	<b>COCON</b> Clock Output Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	0	TLEN	COUT S	COREL					
			Type	r	rw	rw	rw					
E9 <sub>H</sub>	<b>MISC_CON</b> Miscellaneous Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	0								DFLAS HEN
			Type	r								rwh
RMAP = 0, PAGE 3												
B3 <sub>H</sub>	<b>XADDRH</b> On-chip XRAM Address Higher Order	<b>Reset: F0<sub>H</sub></b>	Bit Field	ADDRH								
			Type	rw								

**Functional Description**
**Table 8 System Control Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
B4 <sub>H</sub>	<b>IRCON3</b> Interrupt Request Register 3 Reset: 00 <sub>H</sub>	Bit Field	0		CANS RC5	CCU6S R1	0		CANS RC4	CCU6S R0
		Type	r		rwh	rwh	r		rwh	rwh
B5 <sub>H</sub>	<b>IRCON4</b> Interrupt Request Register 4 Reset: 00 <sub>H</sub>	Bit Field	0		CANS RC7	CCU6S R3	0		CANS RC6	CCU6S R2
		Type	r		rwh	rwh	r		rwh	rwh
B7 <sub>H</sub>	<b>MODPISEL1</b> Peripheral Input Select Register 1 Reset: 00 <sub>H</sub>	Bit Field	EXINT 6IS	0		UR1RIS	T21EXI S	JTAGT DIS1	JTAGT CKS1	
		Type	rw	r		rw	rw	rw	rw	rw
BA <sub>H</sub>	<b>MODPISEL2</b> Peripheral Input Select Register 2 Reset: 00 <sub>H</sub>	Bit Field	0			T21IS	T2IS	T1IS	TOIS	
		Type		r		rw	rw	rw	rw	rw
BB <sub>H</sub>	<b>PMCON2</b> Power Mode Control Register 2 Reset: 00 <sub>H</sub>	Bit Field	0						UART1 _DIS	T21 _DIS
		Type		r					rw	rw
BD <sub>H</sub>	<b>MODSUSP</b> Module Suspend Control Register Reset: 00 <sub>H</sub>	Bit Field	0			T21SU SP	T2SUS P	T13SU SP	T12SU SP	WDT USP
		Type	r			rw	rw	rw	rw	rw

The WDT SFRs can be accessed in the mapped memory area (RMAP = 1).

**Table 9 WDT Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 1										
BB <sub>H</sub>	<b>WDTCON</b> Watchdog Timer Control Register Reset: 00 <sub>H</sub>	Bit Field	0		WINB EN	WDT PR	0	WDT EN	WDT RS	WDT IN
		Type	r		rw	rh	r	rw	rwh	rw
BC <sub>H</sub>	<b>WDTREL</b> Watchdog Timer Reload Register Reset: 00 <sub>H</sub>	Bit Field	WDTREL							
		Type	rw							
BD <sub>H</sub>	<b>WDTWINB</b> Watchdog Window-Boundary Count Register Reset: 00 <sub>H</sub>	Bit Field	WDTWINB							
		Type	rw							
BE <sub>H</sub>	<b>WDTL</b> Watchdog Timer Register Low Reset: 00 <sub>H</sub>	Bit Field	WDT[7:0]							
		Type	rh							
BF <sub>H</sub>	<b>WDTH</b> Watchdog Timer Register High Reset: 00 <sub>H</sub>	Bit Field	WDT[15:8]							
		Type	rh							

The Port SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 10 Port Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 0										
B2 <sub>H</sub>	<b>PORT_PAGE</b> Page Register for PORT Reset: 00 <sub>H</sub>	Bit Field	OP		STNR	0		PAGE		
		Type	w		w	r		rw		
RMAP = 0, Page 0										
80 <sub>H</sub>	<b>P0_DATA</b> P0 Data Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
86 <sub>H</sub>	<b>P0_DIR</b> P0 Direction Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
90 <sub>H</sub>	<b>P1_DATA</b> P1 Data Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw

**Functional Description**
**Table 10 Port Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
91 <sub>H</sub>	<b>P1_DIR</b> P1 Direction Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
92 <sub>H</sub>	<b>P5_DATA</b> P5 Data Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
93 <sub>H</sub>	<b>P5_DIR</b> P5 Direction Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
A0 <sub>H</sub>	<b>P2_DATA</b> P2 Data Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
A1 <sub>H</sub>	<b>P2_DIR</b> P2 Direction Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
B0 <sub>H</sub>	<b>P3_DATA</b> P3 Data Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
B1 <sub>H</sub>	<b>P3_DIR</b> P3 Direction Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
C8 <sub>H</sub>	<b>P4_DATA</b> P4 Data Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
C9 <sub>H</sub>	<b>P4_DIR</b> P4 Direction Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
RMAP = 0, Page 1											
80 <sub>H</sub>	<b>P0_PUDSEL</b> P0 Pull-Up/Pull-Down Select Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
86 <sub>H</sub>	<b>P0_PUDEN</b> P0 Pull-Up/Pull-Down Enable Register	Reset: C4 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
90 <sub>H</sub>	<b>P1_PUDSEL</b> P1 Pull-Up/Pull-Down Select Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
91 <sub>H</sub>	<b>P1_PUDEN</b> P1 Pull-Up/Pull-Down Enable Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
92 <sub>H</sub>	<b>P5_PUDSEL</b> P5 Pull-Up/Pull-Down Select Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
93 <sub>H</sub>	<b>P5_PUDEN</b> P5 Pull-Up/Pull-Down Enable Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
A0 <sub>H</sub>	<b>P2_PUDSEL</b> P2 Pull-Up/Pull-Down Select Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
A1 <sub>H</sub>	<b>P2_PUDEN</b> P2 Pull-Up/Pull-Down Enable Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
B0 <sub>H</sub>	<b>P3_PUDSEL</b> P3 Pull-Up/Pull-Down Select Register	Reset: BF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
B1 <sub>H</sub>	<b>P3_PUDEN</b> P3 Pull-Up/Pull-Down Enable Register	Reset: 40 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
C8 <sub>H</sub>	<b>P4_PUDSEL</b> P4 Pull-Up/Pull-Down Select Register	Reset: FF <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
C9 <sub>H</sub>	<b>P4_PUDEN</b> P4 Pull-Up/Pull-Down Enable Register	Reset: 04 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
RMAP = 0, Page 2											
80 <sub>H</sub>	<b>P0_ALTSEL0</b> P0 Alternate Select 0 Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
86 <sub>H</sub>	<b>P0_ALTSEL1</b> P0 Alternate Select 1 Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw
90 <sub>H</sub>	<b>P1_ALTSEL0</b> P1 Alternate Select 0 Register	Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw	rw

**Functional Description**
**Table 10 Port Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
91 <sub>H</sub>	<b>P1_ALTSEL1</b> P1 Alternate Select 1 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
92 <sub>H</sub>	<b>P5_ALTSEL0</b> P5 Alternate Select 0 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
93 <sub>H</sub>	<b>P5_ALTSEL1</b> P5 Alternate Select 1 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
B0 <sub>H</sub>	<b>P3_ALTSEL0</b> P3 Alternate Select 0 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
B1 <sub>H</sub>	<b>P3_ALTSEL1</b> P3 Alternate Select 1 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
C8 <sub>H</sub>	<b>P4_ALTSEL0</b> P4 Alternate Select 0 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
C9 <sub>H</sub>	<b>P4_ALTSEL1</b> P4 Alternate Select 1 Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
RMAP = 0, Page 3										
80 <sub>H</sub>	<b>P0_OD</b> P0 Open Drain Control Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
90 <sub>H</sub>	<b>P1_OD</b> P1 Open Drain Control Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
92 <sub>H</sub>	<b>P5_OD</b> P5 Open Drain Control Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
B0 <sub>H</sub>	<b>P3_OD</b> P3 Open Drain Control Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw
C8 <sub>H</sub>	<b>P4_OD</b> P4 Open Drain Control Register Reset: 00 <sub>H</sub>	Bit Field	P7	P6	P5	P4	P3	P2	P1	P0
		Type	rw	rw	rw	rw	rw	rw	rw	rw

The ADC SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 11 ADC Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 0										
D1 <sub>H</sub>	<b>ADC_PAGE</b> Page Register for ADC Reset: 00 <sub>H</sub>	Bit Field	OP		STNR		0	PAGE		
		Type	w		w		r	rw		
RMAP = 0, Page 0										
CA <sub>H</sub>	<b>ADC_GLOBCTR</b> Global Control Register Reset: 30 <sub>H</sub>	Bit Field	ANON	DW	CTC		0			
		Type	rw	rw	rw		r			
CB <sub>H</sub>	<b>ADC_GLOBSTR</b> Global Status Register Reset: 00 <sub>H</sub>	Bit Field	0		CHNR			0	SAM PLE	BUSY
		Type	r		rh			r	rh	rh
CC <sub>H</sub>	<b>ADC_PRAR</b> Priority and Arbitration Register Reset: 00 <sub>H</sub>	Bit Field	ASEN1	ASEN0	0	ARBM	CSM1	PRI01	CSM0	PRI00
		Type	rw	rw	r	rw	rw	rw	rw	rw
CD <sub>H</sub>	<b>ADC_LCBR</b> Limit Check Boundary Register Reset: B7 <sub>H</sub>	Bit Field	BOUND1				BOUND0			
		Type	rw				rw			
CE <sub>H</sub>	<b>ADC_INPCR0</b> Input Class Register 0 Reset: 00 <sub>H</sub>	Bit Field	STC							
		Type	rw							
CF <sub>H</sub>	<b>ADC_ETRCR</b> External Trigger Control Register Reset: 00 <sub>H</sub>	Bit Field	SYNEN 1	SYNEN 0	ETRSEL1			ETRSEL0		
		Type	rw	rw	rw			rw		
RMAP = 0, Page 1										

**Functional Description**
**Table 11 ADC Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
CA <sub>H</sub>	<b>ADC_CHCTR0</b> Channel Control Register 0 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
CB <sub>H</sub>	<b>ADC_CHCTR1</b> Channel Control Register 1 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
CC <sub>H</sub>	<b>ADC_CHCTR2</b> Channel Control Register 2 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
CD <sub>H</sub>	<b>ADC_CHCTR3</b> Channel Control Register 3 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
CE <sub>H</sub>	<b>ADC_CHCTR4</b> Channel Control Register 4 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
CF <sub>H</sub>	<b>ADC_CHCTR5</b> Channel Control Register 5 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
D2 <sub>H</sub>	<b>ADC_CHCTR6</b> Channel Control Register 6 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
D3 <sub>H</sub>	<b>ADC_CHCTR7</b> Channel Control Register 7 Reset: 00 <sub>H</sub>	Bit Field	0		LCC		0		RESRSEL	
		Type	r		rw		r		rw	
RMAP = 0, Page 2										
CA <sub>H</sub>	<b>ADC_RESR0L</b> Result Register 0 Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[1:0]		0	VF	DRC	CHNR		
		Type	rh		r	rh	rh	rh		
CB <sub>H</sub>	<b>ADC_RESR0H</b> Result Register 0 High Reset: 00 <sub>H</sub>	Bit Field	RESULT[9:2]							
		Type	rh							
CC <sub>H</sub>	<b>ADC_RESR1L</b> Result Register 1 Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[1:0]		0	VF	DRC	CHNR		
		Type	rh		r	rh	rh	rh		
CD <sub>H</sub>	<b>ADC_RESR1H</b> Result Register 1 High Reset: 00 <sub>H</sub>	Bit Field	RESULT[9:2]							
		Type	rh							
CE <sub>H</sub>	<b>ADC_RESR2L</b> Result Register 2 Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[1:0]		0	VF	DRC	CHNR		
		Type	rh		r	rh	rh	rh		
CF <sub>H</sub>	<b>ADC_RESR2H</b> Result Register 2 High Reset: 00 <sub>H</sub>	Bit Field	RESULT[9:2]							
		Type	rh							
D2 <sub>H</sub>	<b>ADC_RESR3L</b> Result Register 3 Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[1:0]		0	VF	DRC	CHNR		
		Type	rh		r	rh	rh	rh		
D3 <sub>H</sub>	<b>ADC_RESR3H</b> Result Register 3 High Reset: 00 <sub>H</sub>	Bit Field	RESULT[9:2]							
		Type	rh							
RMAP = 0, Page 3										
CA <sub>H</sub>	<b>ADC_RESRA0L</b> Result Register 0, View A Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[2:0]			VF	DRC	CHNR		
		Type	rh			rh	rh	rh		
CB <sub>H</sub>	<b>ADC_RESRA0H</b> Result Register 0, View A High Reset: 00 <sub>H</sub>	Bit Field	RESULT[10:3]							
		Type	rh							
CC <sub>H</sub>	<b>ADC_RESRA1L</b> Result Register 1, View A Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[2:0]			VF	DRC	CHNR		
		Type	rh			rh	rh	rh		
CD <sub>H</sub>	<b>ADC_RESRA1H</b> Result Register 1, View A High Reset: 00 <sub>H</sub>	Bit Field	RESULT[10:3]							
		Type	rh							
CE <sub>H</sub>	<b>ADC_RESRA2L</b> Result Register 2, View A Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[2:0]			VF	DRC	CHNR		
		Type	rh			rh	rh	rh		
CF <sub>H</sub>	<b>ADC_RESRA2H</b> Result Register 2, View A High Reset: 00 <sub>H</sub>	Bit Field	RESULT[10:3]							
		Type	rh							
D2 <sub>H</sub>	<b>ADC_RESRA3L</b> Result Register 3, View A Low Reset: 00 <sub>H</sub>	Bit Field	RESULT[2:0]			VF	DRC	CHNR		
		Type	rh			rh	rh	rh		
D3 <sub>H</sub>	<b>ADC_RESRA3H</b> Result Register 3, View A High Reset: 00 <sub>H</sub>	Bit Field	RESULT[10:3]							
		Type	rh							

**Functional Description**
**Table 11 ADC Register Overview (cont'd)**

Addr	Register Name	Reset:	Bit	7	6	5	4	3	2	1	0
RMAP = 0, Page 4											
CA <sub>H</sub>	<b>ADC_RCR0</b> Result Control Register 0	Reset: 00 <sub>H</sub>	Bit Field	VFCTR	WFR	FEN	IEN	0			DRCT R
			Type	rw	rw	rw	rw	r			rw
CB <sub>H</sub>	<b>ADC_RCR1</b> Result Control Register 1	Reset: 00 <sub>H</sub>	Bit Field	VFCTR	WFR	FEN	IEN	0			DRCT R
			Type	rw	rw	rw	rw	r			rw
CC <sub>H</sub>	<b>ADC_RCR2</b> Result Control Register 2	Reset: 00 <sub>H</sub>	Bit Field	VFCTR	WFR	FEN	IEN	0			DRCT R
			Type	rw	rw	rw	rw	r			rw
CD <sub>H</sub>	<b>ADC_RCR3</b> Result Control Register 3	Reset: 00 <sub>H</sub>	Bit Field	VFCTR	WFR	FEN	IEN	0			DRCT R
			Type	rw	rw	rw	rw	r			rw
CE <sub>H</sub>	<b>ADC_VFCR</b> Valid Flag Clear Register	Reset: 00 <sub>H</sub>	Bit Field	0			VFC3	VFC2	VFC1	VFC0	
			Type	r			w	w	w	w	
RMAP = 0, Page 5											
CA <sub>H</sub>	<b>ADC_CHINFR</b> Channel Interrupt Flag Register	Reset: 00 <sub>H</sub>	Bit Field	CHINF 7	CHINF 6	CHINF 5	CHINF 4	CHINF 3	CHINF 2	CHINF 1	CHINF 0
			Type	rh	rh	rh	rh	rh	rh	rh	rh
CB <sub>H</sub>	<b>ADC_CHINCR</b> Channel Interrupt Clear Register	Reset: 00 <sub>H</sub>	Bit Field	CHINC 7	CHINC 6	CHINC 5	CHINC 4	CHINC 3	CHINC 2	CHINC 1	CHINC 0
			Type	w	w	w	w	w	w	w	w
CC <sub>H</sub>	<b>ADC_CHINSR</b> Channel Interrupt Set Register	Reset: 00 <sub>H</sub>	Bit Field	CHINS 7	CHINS 6	CHINS 5	CHINS 4	CHINS 3	CHINS 2	CHINS 1	CHINS 0
			Type	w	w	w	w	w	w	w	w
CD <sub>H</sub>	<b>ADC_CHINPR</b> Channel Interrupt Node Pointer Register	Reset: 00 <sub>H</sub>	Bit Field	CHINP 7	CHINP 6	CHINP 5	CHINP 4	CHINP 3	CHINP 2	CHINP 1	CHINP 0
			Type	rw	rw	rw	rw	rw	rw	rw	rw
CE <sub>H</sub>	<b>ADC_EVINFR</b> Event Interrupt Flag Register	Reset: 00 <sub>H</sub>	Bit Field	EVINF 7	EVINF 6	EVINF 5	EVINF 4	0		EVINF 1	EVINF 0
			Type	rh	rh	rh	rh	r		rh	rh
CF <sub>H</sub>	<b>ADC_EVINCR</b> Event Interrupt Clear Flag Register	Reset: 00 <sub>H</sub>	Bit Field	EVINC 7	EVINC 6	EVINC 5	EVINC 4	0		EVINC 1	EVINC 0
			Type	w	w	w	w	r		w	w
D2 <sub>H</sub>	<b>ADC_EVINSR</b> Event Interrupt Set Flag Register	Reset: 00 <sub>H</sub>	Bit Field	EVINS 7	EVINS 6	EVINS 5	EVINS 4	0		EVINS 1	EVINS 0
			Type	w	w	w	w	r		w	w
D3 <sub>H</sub>	<b>ADC_EVINPR</b> Event Interrupt Node Pointer Register	Reset: 00 <sub>H</sub>	Bit Field	EVINP 7	EVINP 6	EVINP 5	EVINP 4	0		EVINP 1	EVINP 0
			Type	rw	rw	rw	rw	r		rw	rw
RMAP = 0, Page 6											
CA <sub>H</sub>	<b>ADC_CRCR1</b> Conversion Request Control Register 1	Reset: 00 <sub>H</sub>	Bit Field	CH7	CH6	CH5	CH4	0			
			Type	rwh	rwh	rwh	rwh	r			
CB <sub>H</sub>	<b>ADC_CRPR1</b> Conversion Request Pending Register 1	Reset: 00 <sub>H</sub>	Bit Field	CHP7	CHP6	CHP5	CHP4	0			
			Type	rwh	rwh	rwh	rwh	r			
CC <sub>H</sub>	<b>ADC_CRRM1</b> Conversion Request Mode Register 1	Reset: 00 <sub>H</sub>	Bit Field	Rsv	LDEV	CLR PND	SCAN	ENSI	ENTR	ENGT	
			Type	r	w	w	rw	rw	rw	rw	
CD <sub>H</sub>	<b>ADC_QMR0</b> Queue Mode Register 0	Reset: 00 <sub>H</sub>	Bit Field	CEV	TREV	FLUSH	CLRV	0	ENTR	ENGT	
			Type	w	w	w	w	r	rw	rw	



**Functional Description**
**Table 11 ADC Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
CE <sub>H</sub>	<b>ADC_QSR0</b> Queue Status Register 0 Reset: 20 <sub>H</sub>	Bit Field	Rsv	0	EMPTY	EV	0		FILL	
		Type	r	r	rh	rh	r		rh	
CF <sub>H</sub>	<b>ADC_Q0R0</b> Queue 0 Register 0 Reset: 00 <sub>H</sub>	Bit Field	EXTR	ENSI	RF	V	0	REQCHNR		
		Type	rh	rh	rh	rh	r	rh		
D2 <sub>H</sub>	<b>ADC_QBUR0</b> Queue Backup Register 0 Reset: 00 <sub>H</sub>	Bit Field	EXTR	ENSI	RF	V	0	REQCHNR		
		Type	rh	rh	rh	rh	r	rh		
D2 <sub>H</sub>	<b>ADC_QINR0</b> Queue Input Register 0 Reset: 00 <sub>H</sub>	Bit Field	EXTR	ENSI	RF	0		REQCHNR		
		Type	w	w	w	r		w		

The Timer 2 SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 12 Timer 2 Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
C0 <sub>H</sub>	<b>T2_T2CON</b> Timer 2 Control Register Reset: 00 <sub>H</sub>	Bit Field	TF2	EXF2	0		EXEN2	TR2	0	CP/ RL2
		Type	rwh	rwh	r		rw	rwh	r	rw
C1 <sub>H</sub>	<b>T2_T2MOD</b> Timer 2 Mode Register Reset: 00 <sub>H</sub>	Bit Field	T2 REGS	T2 RHEN	EDGE SEL	PREN	T2PRE			DCEN
		Type	rw	rw	rw	rw	rw			rw
C2 <sub>H</sub>	<b>T2_RC2L</b> Timer 2 Reload/Capture Register Low Reset: 00 <sub>H</sub>	Bit Field	RC2							
		Type	rwh							
C3 <sub>H</sub>	<b>T2_RC2H</b> Timer 2 Reload/Capture Register High Reset: 00 <sub>H</sub>	Bit Field	RC2							
		Type	rwh							
C4 <sub>H</sub>	<b>T2_T2L</b> Timer 2 Register Low Reset: 00 <sub>H</sub>	Bit Field	THL2							
		Type	rwh							
C5 <sub>H</sub>	<b>T2_T2H</b> Timer 2 Register High Reset: 00 <sub>H</sub>	Bit Field	THL2							
		Type	rwh							

The Timer 21 SFRs can be accessed in the standard memory area (RMAP = 1).

**Table 13 T21 Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 1										
C0 <sub>H</sub>	<b>T2CON</b> Timer 2 Control Register Reset: 00 <sub>H</sub>	Bit Field	TF2	EXF2	0	0	EXEN2	TR2	C/T2	CP/ RL2
		Type	rwh	rwh	r	r	rw	rwh	rw	rw
C1 <sub>H</sub>	<b>T2MOD</b> Timer 2 Mode Register Reset: 00 <sub>H</sub>	Bit Field	T2 REGS	T2 RHEN	EDGE SEL	PREN	T2PRE			DCEN
		Type	rw	rw	rw	rw	rw			rw
C2 <sub>H</sub>	<b>RC2L</b> Timer 2 Reload/Capture Register Low Reset: 00 <sub>H</sub>	Bit Field	RC2							
		Type	rwh							
C3 <sub>H</sub>	<b>RC2H</b> Timer 2 Reload/Capture Register High Reset: 00 <sub>H</sub>	Bit Field	RC2							
		Type	rwh							
C4 <sub>H</sub>	<b>T2L</b> Timer 2 Register Low Reset: 00 <sub>H</sub>	Bit Field	THL2							
		Type	rwh							
C5 <sub>H</sub>	<b>T2H</b> Timer 2 Register High Reset: 00 <sub>H</sub>	Bit Field	THL2							
		Type	rwh							

**Functional Description**

The CCU6 SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 14 CCU6 Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 0											
A3 <sub>H</sub>	<b>CCU6_PAGE</b> Page Register for CCU6 Reset: 00 <sub>H</sub>	Bit Field	OP		STNR		0		PAGE		
		Type	w		w		r		rw		
RMAP = 0, Page 0											
9A <sub>H</sub>	<b>CCU6_CC63SRL</b> Capture/Compare Shadow Register for Channel CC63 Low Reset: 00 <sub>H</sub>	Bit Field	CC63SL								
		Type	rw								
9B <sub>H</sub>	<b>CCU6_CC63SRH</b> Capture/Compare Shadow Register for Channel CC63 High Reset: 00 <sub>H</sub>	Bit Field	CC63SH								
		Type	rw								
9C <sub>H</sub>	<b>CCU6_TCTR4L</b> Timer Control Register 4 Low Reset: 00 <sub>H</sub>	Bit Field	T12 STD	T12 STR	0		DTRES	T12 RES	T12RS	T12RR	
		Type	w	w	r		w	w	w	w	
9D <sub>H</sub>	<b>CCU6_TCTR4H</b> Timer Control Register 4 High Reset: 00 <sub>H</sub>	Bit Field	T13 STD	T13 STR	0			T13 RES	T13RS	T13RR	
		Type	w	w	r			w	w	w	
9E <sub>H</sub>	<b>CCU6_MCMOUTSL</b> Multi-Channel Mode Output Shadow Register Low Reset: 00 <sub>H</sub>	Bit Field	STRM CM	0		MCMPS					
		Type	w	r		rw					
9F <sub>H</sub>	<b>CCU6_MCMOUTSH</b> Multi-Channel Mode Output Shadow Register High Reset: 00 <sub>H</sub>	Bit Field	STRHP	0		CURHS			EXPHS		
		Type	w	r		rw			rw		
A4 <sub>H</sub>	<b>CCU6_ISRL</b> Capture/Compare Interrupt Status Reset Register Low Reset: 00 <sub>H</sub>	Bit Field	RT12P M	RT12O M	RCC62 F	RCC62 R	RCC61 F	RCC61 R	RCC60 F	RCC60 R	
		Type	w	w	w	w	w	w	w	w	
A5 <sub>H</sub>	<b>CCU6_ISRH</b> Capture/Compare Interrupt Status Reset Register High Reset: 00 <sub>H</sub>	Bit Field	RSTR	RIDLE	RWHE	RCHE	0		RTRPF	RT13 PM	RT13 CM
		Type	w	w	w	w	r		w	w	w
A6 <sub>H</sub>	<b>CCU6_CMPMODIFL</b> Compare State Modification Register Low Reset: 00 <sub>H</sub>	Bit Field	0	MCC63 S	0			MCC62 S	MCC61 S	MCC60 S	
		Type	r	w	r			w	w	w	
A7 <sub>H</sub>	<b>CCU6_CMPMODIFH</b> Compare State Modification Register High Reset: 00 <sub>H</sub>	Bit Field	0	MCC63 R	0			MCC62 R	MCC61 R	MCC60 R	
		Type	r	w	r			w	w	w	
FA <sub>H</sub>	<b>CCU6_CC60SRL</b> Capture/Compare Shadow Register for Channel CC60 Low Reset: 00 <sub>H</sub>	Bit Field	CC60SL								
		Type	rwh								
FB <sub>H</sub>	<b>CCU6_CC60SRH</b> Capture/Compare Shadow Register for Channel CC60 High Reset: 00 <sub>H</sub>	Bit Field	CC60SH								
		Type	rwh								
FC <sub>H</sub>	<b>CCU6_CC61SRL</b> Capture/Compare Shadow Register for Channel CC61 Low Reset: 00 <sub>H</sub>	Bit Field	CC61SL								
		Type	rwh								
FD <sub>H</sub>	<b>CCU6_CC61SRH</b> Capture/Compare Shadow Register for Channel CC61 High Reset: 00 <sub>H</sub>	Bit Field	CC61SH								
		Type	rwh								
FE <sub>H</sub>	<b>CCU6_CC62SRL</b> Capture/Compare Shadow Register for Channel CC62 Low Reset: 00 <sub>H</sub>	Bit Field	CC62SL								
		Type	rwh								
FF <sub>H</sub>	<b>CCU6_CC62SRH</b> Capture/Compare Shadow Register for Channel CC62 High Reset: 00 <sub>H</sub>	Bit Field	CC62SH								
		Type	rwh								

**Functional Description**
**Table 14 CCU6 Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 0, Page 1										
9A <sub>H</sub>	<b>CCU6_CC63RL</b> Capture/Compare Register for Channel CC63 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	CC63VL							
		Type	rh							
9B <sub>H</sub>	<b>CCU6_CC63RH</b> Capture/Compare Register for Channel CC63 High <b>Reset: 00<sub>H</sub></b>	Bit Field	CC63VH							
		Type	rh							
9C <sub>H</sub>	<b>CCU6_T12PRL</b> Timer T12 Period Register Low <b>Reset: 00<sub>H</sub></b>	Bit Field	T12PVL							
		Type	rwh							
9D <sub>H</sub>	<b>CCU6_T12PRH</b> Timer T12 Period Register High <b>Reset: 00<sub>H</sub></b>	Bit Field	T12PVH							
		Type	rwh							
9E <sub>H</sub>	<b>CCU6_T13PRL</b> Timer T13 Period Register Low <b>Reset: 00<sub>H</sub></b>	Bit Field	T13PVL							
		Type	rwh							
9F <sub>H</sub>	<b>CCU6_T13PRH</b> Timer T13 Period Register High <b>Reset: 00<sub>H</sub></b>	Bit Field	T13PVH							
		Type	rwh							
A4 <sub>H</sub>	<b>CCU6_T12DTCL</b> Dead-Time Control Register for Timer T12 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	DTM							
		Type	rw							
A5 <sub>H</sub>	<b>CCU6_T12DTCH</b> Dead-Time Control Register for Timer T12 High <b>Reset: 00<sub>H</sub></b>	Bit Field	0	DTR2	DTR1	DTR0	0	DTE2	DTE1	DTE0
		Type	r	rh	rh	rh	r	rw	rw	rw
A6 <sub>H</sub>	<b>CCU6_TCTR0L</b> Timer Control Register 0 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	CTM	CDIR	STE12	T12R	T12 PRE	T12CLK		
		Type	rw	rh	rh	rh	rw	rw		
A7 <sub>H</sub>	<b>CCU6_TCTR0H</b> Timer Control Register 0 High <b>Reset: 00<sub>H</sub></b>	Bit Field	0	STE13	T13R	T13 PRE	T13CLK			
		Type	r	rh	rh	rw	rw			
FA <sub>H</sub>	<b>CCU6_CC60RL</b> Capture/Compare Register for Channel CC60 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	CC60VL							
		Type	rh							
FB <sub>H</sub>	<b>CCU6_CC60RH</b> Capture/Compare Register for Channel CC60 High <b>Reset: 00<sub>H</sub></b>	Bit Field	CC60VH							
		Type	rh							
FC <sub>H</sub>	<b>CCU6_CC61RL</b> Capture/Compare Register for Channel CC61 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	CC61VL							
		Type	rh							
FD <sub>H</sub>	<b>CCU6_CC61RH</b> Capture/Compare Register for Channel CC61 High <b>Reset: 00<sub>H</sub></b>	Bit Field	CC61VH							
		Type	rh							
FE <sub>H</sub>	<b>CCU6_CC62RL</b> Capture/Compare Register for Channel CC62 Low <b>Reset: 00<sub>H</sub></b>	Bit Field	CC62VL							
		Type	rh							
FF <sub>H</sub>	<b>CCU6_CC62RH</b> Capture/Compare Register for Channel CC62 High <b>Reset: 00<sub>H</sub></b>	Bit Field	CC62VH							
		Type	rh							
RMAP = 0, Page 2										
9A <sub>H</sub>	<b>CCU6_T12MSELL</b> T12 Capture/Compare Mode Select Register Low <b>Reset: 00<sub>H</sub></b>	Bit Field	MSEL61				MSEL60			
		Type	rw				rw			
9B <sub>H</sub>	<b>CCU6_T12MSELH</b> T12 Capture/Compare Mode Select Register High <b>Reset: 00<sub>H</sub></b>	Bit Field	DBYP	HSYNC			MSEL62			
		Type	rw	rw			rw			

**Functional Description**
**Table 14 CCU6 Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
9C <sub>H</sub>	<b>CCU6_IENL</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Enable Register Low	Bit Field	ENT12 PM	ENT12 OM	ENCC 62F	ENCC 62R	ENCC 61F	ENCC 61R	ENCC 60F	ENCC 60R	
		Type	rw	rw	rw	rw	rw	rw	rw	rw	
9D <sub>H</sub>	<b>CCU6_IENH</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Enable Register High	Bit Field	ENSTR	EN IDLE	EN WHE	EN CHE	0	EN TRPF	ENT13 PM	ENT13 CM	
		Type	rw	rw	rw	rw	r	rw	rw	rw	
9E <sub>H</sub>	<b>CCU6_INPL</b> <b>Reset: 40<sub>H</sub></b> Capture/Compare Interrupt Node Pointer Register Low	Bit Field	INPCHE		INPCC62		INPCC61		INPCC60		
		Type	rw		rw		rw		rw		
9F <sub>H</sub>	<b>CCU6_INPH</b> <b>Reset: 39<sub>H</sub></b> Capture/Compare Interrupt Node Pointer Register High	Bit Field	0		INPT13		INPT12		INPERR		
		Type	r		rw		rw		rw		
A4 <sub>H</sub>	<b>CCU6_ISSL</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Status Set Register Low	Bit Field	ST12P M	ST12O M	SCC62 F	SCC62 R	SCC61 F	SCC61 R	SCC60 F	SCC60 R	
		Type	w	w	w	w	w	w	w	w	
A5 <sub>H</sub>	<b>CCU6_ISSH</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Status Set Register High	Bit Field	SSTR	SIDLE	SWHE	SCHE	SWHC	STRPF	ST13 PM	ST13 CM	
		Type	w	w	w	w	w	w	w	w	
A6 <sub>H</sub>	<b>CCU6_PSLR</b> <b>Reset: 00<sub>H</sub></b> Passive State Level Register	Bit Field	PSL63	0	PSL						
		Type	rwh	r	rwh						
A7 <sub>H</sub>	<b>CCU6_MCMCTR</b> <b>Reset: 00<sub>H</sub></b> Multi-Channel Mode Control Register	Bit Field	0		SWSYN		0	SWSEL			
		Type	r		rw		r	rw			
FA <sub>H</sub>	<b>CCU6_TCTR2L</b> <b>Reset: 00<sub>H</sub></b> Timer Control Register 2 Low	Bit Field	0	T13TED		T13TEC			T13 SSC	T12 SSC	
		Type	r	rw		rw			rw	rw	
FB <sub>H</sub>	<b>CCU6_TCTR2H</b> <b>Reset: 00<sub>H</sub></b> Timer Control Register 2 High	Bit Field	0			T13RSEL		T12RSEL			
		Type	r			rw		rw			
FC <sub>H</sub>	<b>CCU6_MODCTRL</b> <b>Reset: 00<sub>H</sub></b> Modulation Control Register Low	Bit Field	MC MEN	0	T12MODEN						
		Type	rw	r	rw						
FD <sub>H</sub>	<b>CCU6_MODCTRH</b> <b>Reset: 00<sub>H</sub></b> Modulation Control Register High	Bit Field	ECT13 O	0	T13MODEN						
		Type	rw	r	rw						
FE <sub>H</sub>	<b>CCU6_TRPCTRL</b> <b>Reset: 00<sub>H</sub></b> Trap Control Register Low	Bit Field	0				TRPM2	TRPM1	TRPM0		
		Type	r				rw	rw	rw		
FF <sub>H</sub>	<b>CCU6_TRPCTRH</b> <b>Reset: 00<sub>H</sub></b> Trap Control Register High	Bit Field	TRPPE N	TRPEN 13	TRPEN						
		Type	rw	rw	rw						
RMAP = 0, Page 3											
9A <sub>H</sub>	<b>CCU6_MCMOUTL</b> <b>Reset: 00<sub>H</sub></b> Multi-Channel Mode Output Register Low	Bit Field	0	R	MCMP						
		Type	r	rh	rh						
9B <sub>H</sub>	<b>CCU6_MCMOUTH</b> <b>Reset: 00<sub>H</sub></b> Multi-Channel Mode Output Register High	Bit Field	0		CURH			EXPH			
		Type	r		rh			rh			
9C <sub>H</sub>	<b>CCU6_ISL</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Status Register Low	Bit Field	T12PM	T12OM	ICC62F	ICC62 R	ICC61F	ICC61 R	ICC60F	ICC60 R	
		Type	rh	rh	rh	rh	rh	rh	rh	rh	
9D <sub>H</sub>	<b>CCU6_ISH</b> <b>Reset: 00<sub>H</sub></b> Capture/Compare Interrupt Status Register High	Bit Field	STR	IDLE	WHE	CHE	TRPS	TRPF	T13PM	T13CM	
		Type	rh	rh	rh	rh	rh	rh	rh	rh	
9E <sub>H</sub>	<b>CCU6_PISEL0L</b> <b>Reset: 00<sub>H</sub></b> Port Input Select Register 0 Low	Bit Field	ISTRP		ISCC62		ISCC61		ISCC60		
		Type	rw		rw		rw		rw		

**Functional Description**
**Table 14 CCU6 Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
9F <sub>H</sub>	<b>CCU6_PISEL0H</b> Reset: 00 <sub>H</sub> Port Input Select Register 0 High	Bit Field	IST12HR		ISPOS2		ISPOS1		ISPOS0	
		Type	rw		rw		rw		rw	
A4 <sub>H</sub>	<b>CCU6_PISEL2</b> Reset: 00 <sub>H</sub> Port Input Select Register 2	Bit Field	0						IST13HR	
		Type	r						rw	
FA <sub>H</sub>	<b>CCU6_T12L</b> Reset: 00 <sub>H</sub> Timer T12 Counter Register Low	Bit Field	T12CVL							
		Type	rwh							
FB <sub>H</sub>	<b>CCU6_T12H</b> Reset: 00 <sub>H</sub> Timer T12 Counter Register High	Bit Field	T12CVH							
		Type	rwh							
FC <sub>H</sub>	<b>CCU6_T13L</b> Reset: 00 <sub>H</sub> Timer T13 Counter Register Low	Bit Field	T13CVL							
		Type	rwh							
FD <sub>H</sub>	<b>CCU6_T13H</b> Reset: 00 <sub>H</sub> Timer T13 Counter Register High	Bit Field	T13CVH							
		Type	rwh							
FE <sub>H</sub>	<b>CCU6_CMPSTATL</b> Reset: 00 <sub>H</sub> Compare State Register Low	Bit Field	0	CC63 ST	CCPO S2	CCPO S1	CCPO S0	CC62 ST	CC61 ST	CC60 ST
		Type	r	rh	rh	rh	rh	rh	rh	rh
FF <sub>H</sub>	<b>CCU6_CMPSTATH</b> Reset: 00 <sub>H</sub> Compare State Register High	Bit Field	T13IM	COUT 63PS	COUT 62PS	CC62 PS	COUT 61PS	CC61 PS	COUT 60PS	CC60 PS
		Type	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh

The UART1 SFRs can be accessed in the mapped memory area (RMAP = 1).

**Table 15 UART1 Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 1										
C8 <sub>H</sub>	<b>SCON</b> Reset: 00 <sub>H</sub> Serial Channel Control Register	Bit Field	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
		Type	rw	rw	rw	rw	rw	rwh	rwh	rwh
C9 <sub>H</sub>	<b>SBUF</b> Reset: 00 <sub>H</sub> Serial Data Buffer Register	Bit Field	VAL							
		Type	rwh							
CA <sub>H</sub>	<b>BCON</b> Reset: 00 <sub>H</sub> Baud Rate Control Register	Bit Field	0				BRPRE			R
		Type	r				rw			rw
CB <sub>H</sub>	<b>BG</b> Reset: 00 <sub>H</sub> Baud Rate Timer/Reload Register	Bit Field	BR_VALUE							
		Type	rwh							
CC <sub>H</sub>	<b>FDCON</b> Reset: 00 <sub>H</sub> Fractional Divider Control Register	Bit Field	0				NDOV	FDM	F DEN	
		Type	r				rwh	rw	rw	
CD <sub>H</sub>	<b>FDSTEP</b> Reset: 00 <sub>H</sub> Fractional Divider Reload Register	Bit Field	STEP							
		Type	rw							
CE <sub>H</sub>	<b>FDRES</b> Reset: 00 <sub>H</sub> Fractional Divider Result Register	Bit Field	RESULT							
		Type	rh							

The SSC SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 16 SSC Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 0											
A9 <sub>H</sub>	<b>SSC_PISEL</b> Reset: 00 <sub>H</sub> Port Input Select Register	Bit Field	0					CIS	SIS	MIS	
		Type	r					rw	rw	rw	

**Functional Description**
**Table 16 SSC Register Overview**

AA <sub>H</sub>	<b>SSC_CONL</b> Control Register Low <i>Programming Mode</i>	<b>Reset: 00<sub>H</sub></b>	Bit Field	LB	PO	PH	HB	BM			
			Type	rw	rw	rw	rw	rw			
	<i>Operating Mode</i>			Bit Field	0				BC		
			Type	r				rh			
AB <sub>H</sub>	<b>SSC_CONH</b> Control Register High <i>Programming Mode</i>	<b>Reset: 00<sub>H</sub></b>	Bit Field	EN	MS	0	AREN	BEN	PEN	REN	TEN
			Type	rw	rw	r	rw	rw	rw	rw	rw
	<i>Operating Mode</i>			Bit Field	EN	MS	0	BSY	BE	PE	RE
			Type	rw	rw	r	rh	rwh	rwh	rwh	rwh
AC <sub>H</sub>	<b>SSC_TBL</b> Transmitter Buffer Register Low	<b>Reset: 00<sub>H</sub></b>	Bit Field	TB_VALUE							
			Type	rw							
AD <sub>H</sub>	<b>SSC_RBL</b> Receiver Buffer Register Low	<b>Reset: 00<sub>H</sub></b>	Bit Field	RB_VALUE							
			Type	rh							
AE <sub>H</sub>	<b>SSC_BRL</b> Baudrate Timer Reload Register Low	<b>Reset: 00<sub>H</sub></b>	Bit Field	BR_VALUE							
			Type	rw							
AF <sub>H</sub>	<b>SSC_BRH</b> Baudrate Timer Reload Register High	<b>Reset: 00<sub>H</sub></b>	Bit Field	BR_VALUE							
			Type	rw							

The MultiCAN SFRs can be accessed in the standard memory area (RMAP = 0).

**Table 17 MultiCAN Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 0											
D8 <sub>H</sub>	<b>ADCON</b> CAN Address/Data Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	V3	V2	V1	V0	AUAD		BSY	RWEN
			Type	rw	rw	rw	rw	rw		rh	rw
D9 <sub>H</sub>	<b>ADL</b> CAN Address Low Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	CA9	CA8	CA7	CA6	CA5	CA4	CA3	CA2
			Type	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh
DA <sub>H</sub>	<b>ADH</b> CAN Address High Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	0				CA13	CA12	CA11	CA10
			Type	r				rwh	rwh	rwh	rwh
DB <sub>H</sub>	<b>DATA0</b> CAN Data Register 0	<b>Reset: 00<sub>H</sub></b>	Bit Field	CD							
			Type	rwh							
DC <sub>H</sub>	<b>DATA1</b> CAN Data Register 1	<b>Reset: 00<sub>H</sub></b>	Bit Field	CD							
			Type	rwh							
DD <sub>H</sub>	<b>DATA2</b> CAN Data Register 2	<b>Reset: 00<sub>H</sub></b>	Bit Field	CD							
			Type	rwh							
DE <sub>H</sub>	<b>DATA3</b> CAN Data Register 3	<b>Reset: 00<sub>H</sub></b>	Bit Field	CD							
			Type	rwh							

The OCDS SFRs can be accessed in the mapped memory area (RMAP = 1).

**Table 18 OCDS Register Overview**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
RMAP = 1											
E9 <sub>H</sub>	<b>MMCR2</b> Monitor Mode Control 2 Register	<b>Reset: 1U<sub>H</sub></b>	Bit Field	STMO DE	EXBC	DSUSP	MBCO N	ALTDI	MMEP	MMOD E	JENA
			Type	rw	rw	rw	rwh	rw	rwh	rh	rh
F1 <sub>H</sub>	<b>MMCR</b> Monitor Mode Control Register	<b>Reset: 00<sub>H</sub></b>	Bit Field	MEXIT _P	MEXIT	0	MSTEP	MRAM S_P	MRAM S	TRF	RRF
			Type	w	hw	r	rw	w	rwh	rh	rh

**Functional Description**
**Table 18 OCDS Register Overview (cont'd)**

Addr	Register Name	Bit	7	6	5	4	3	2	1	0	
F2 <sub>H</sub>	<b>MMSR</b> Monitor Mode Status Register <b>Reset: 00<sub>H</sub></b>	Bit Field	MBCA M	MBCIN	EXBF	SWBF	HWB3 F	HWB2 F	HWB1 F	HWB0 F	
		Type	rw	rwh	rwh	rwh	rwh	rwh	rwh	rwh	
F3 <sub>H</sub>	<b>MMBPCR</b> BreakPoints Control Register <b>Reset: 00<sub>H</sub></b>	Bit Field	SWBC	HWB3C		HWB2C		HWB1 C	HWB0C		
		Type	rw	rw		rw		rw	rw		
F4 <sub>H</sub>	<b>MMICR</b> Monitor Mode Interrupt Control Register <b>Reset: 00<sub>H</sub></b>	Bit Field	DVECT	DRETR	COM RST	MST SEL	MMUIE _P	MMUIE	RRIE_ P	RRIE	
		Type	rwh	rwh	rwh	rh	w	rw	w	rw	
F5 <sub>H</sub>	<b>MMDR</b> Monitor Mode Data Transfer Register <b>Reset: 00<sub>H</sub></b> <i>Receive</i>	Bit Field	MMRR								
		Type	rh								
		<i>Transmit</i>	MMTR								
		Type	w								
F6 <sub>H</sub>	<b>HWBPSR</b> Hardware Breakpoints Select Register <b>Reset: 00<sub>H</sub></b>	Bit Field	0			BPSEL _P	BPSEL				
		Type	r			w	rw				
F7 <sub>H</sub>	<b>HWBPD</b> Hardware Breakpoints Data Register <b>Reset: 00<sub>H</sub></b>	Bit Field	HWBPxx								
		Type	rw								
EB <sub>H</sub>	<b>MMWR1</b> Monitor Work Register 1 <b>Reset: 00<sub>H</sub></b>	Bit Field	MMWR1								
		Type	rw								
EC <sub>H</sub>	<b>MMWR2</b> Monitor Work Register 2 <b>Reset: 00<sub>H</sub></b>	Bit Field	MMWR2								
		Type	rw								

### 3.3 Flash Memory

The Flash memory provides an embedded user-programmable non-volatile memory, allowing fast and reliable storage of user code and data. It is operated from a single 2.5 V supply from the Embedded Voltage Regulator (EVR) and does not require additional programming or erasing voltage. The sectorization of the Flash memory allows each sector to be erased independently.

#### Features:

- In-System Programming (ISP) via UART
- In-Application Programming (IAP)
- Error Correction Code (ECC) for dynamic correction of single-bit errors
- Background program and erase operations for CPU load minimization
- Support for aborting erase operation
- Minimum program width<sup>1)</sup> of 32-byte for D-Flash and 64-byte for P-Flash
- 1-sector minimum erase width
- 1-byte read access
- 135.1 ns minimum read access time ( $3 \times t_{\text{CCLK}} @ f_{\text{CCLK}} = 24 \text{ MHz} \pm 7.5 \%^{2)}$ )
- Operating supply voltage:  $2.5 \text{ V} \pm 7.5 \%$
- Program time:  $2.3 \text{ ms}^{3)}$
- Erase time:  $120 \text{ ms}^{3)}$

**Table 19** shows the Flash data retention and endurance targets<sup>4)</sup>.

**Table 19 Flash Data Retention and Endurance Targets**

Retention up to	Endurance up to	Programming Temperature	Size
20 years	1,000 cycles	0 – 100°C	15 Kbytes
5 years	10,000 cycles	-40 – 125°C	896 bytes
2 years	70,000 cycles	-40 – 125°C	512 bytes
2 years	100,000 cycles	-40 – 125°C	128 bytes

<sup>1)</sup> P-Flash: 64-byte wordline can only be programmed once, i.e., one gate disturb allowed.

D-Flash: 32-byte wordline can be programmed twice, i.e., two gate disturbs allowed.

<sup>2)</sup>  $f_{\text{sys}} = 96 \text{ MHz} \pm 7.5\%$  ( $f_{\text{CCLK}} = 24 \text{ MHz} \pm 7.5\%$ ) is the maximum frequency range for Flash read access.

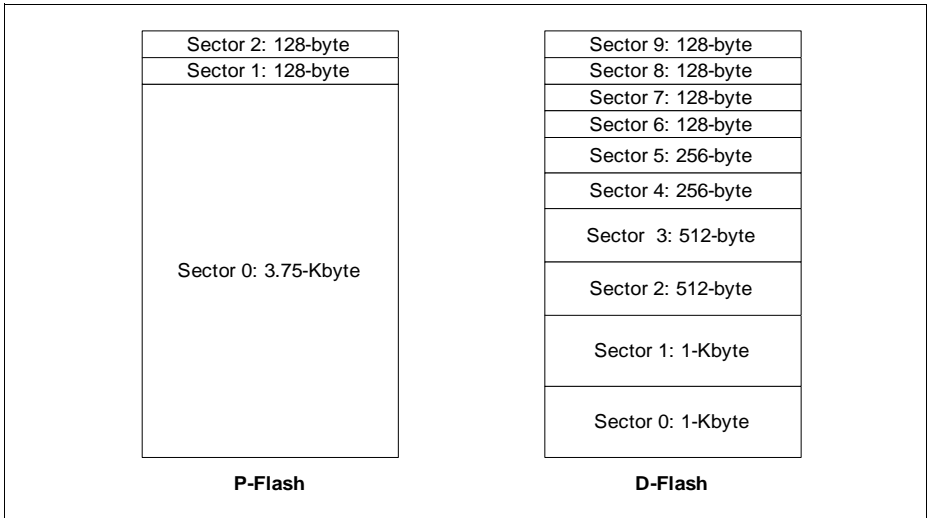
<sup>3)</sup>  $f_{\text{sys}} = 96 \text{ MHz} \pm 7.5\%$  is the only frequency range for Flash programming and erasing.  $f_{\text{sysmin}}$  is used for obtaining the worst case timing.

<sup>4)</sup> Specification according to operating temperature profile with 0.2ppm error rate.



### 3.3.1 Flash Bank Sectorization

The XC886/888 product family offers Flash devices with either 24 Kbytes or 32 Kbytes of embedded Flash memory. Each Flash device consists of Program Flash (P-Flash) bank(s) and a single Data Flash (D-Flash) bank with different sectorization shown in **Figure 11**. Both types can be used for code and data storage. The label “Data” neither implies that the D-Flash is mapped to the data memory region, nor that it can only be used for data storage. It is used to distinguish the different Flash bank sectorizations. The XC886/888 ROM devices offer a single 4-Kbyte D-Flash bank.



**Figure 11 Flash Bank Sectorization**

The internal structure of each Flash bank represents a sector architecture for flexible erase capability. The minimum erase width is always a complete sector, and sectors can be erased separately or in parallel. Contrary to standard EPROMs, erased Flash memory cells contain 0s.

The D-Flash bank is divided into more physical sectors for extended erasing and reprogramming capability; even numbers for each sector size are provided to allow greater flexibility and the ability to adapt to a wide range of application requirements.

### 3.3.2 Flash Programming Width

For the P-Flash banks, a programmed wordline (WL) must be erased before it can be reprogrammed as the Flash cells can only withstand one gate disturb. This means that the entire sector containing the WL must be erased since it is impossible to erase a single WL.

For the D-Flash bank, the same WL can be programmed twice before erasing is required as the Flash cells are able to withstand two gate disturbs. Hence, it is possible to program the same WL, for example, with 16 bytes of data in two times (see [Figure 12](#)).

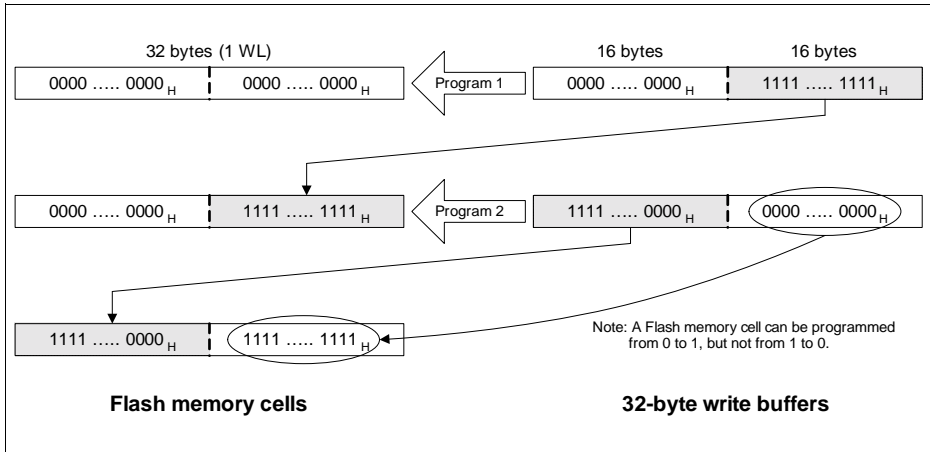


Figure 12 D-Flash Programming

*Note: When programming a D-Flash WL the second time, the previously programmed Flash memory cells (whether 0s or 1s) should be reprogrammed with 0s to retain its original contents and to prevent “over-programming”.*

### 3.4 Interrupt System

The XC800 Core supports one non-maskable interrupt (NMI) and 14 maskable interrupt requests. In addition to the standard interrupt functions supported by the core, e.g., configurable interrupt priority and interrupt masking, the XC886/888 interrupt system provides extended interrupt support capabilities such as the mapping of each interrupt vector to several interrupt sources to increase the number of interrupt sources supported, and additional status registers for detecting and determining the interrupt source.

#### 3.4.1 Interrupt Source

Figure 13 to Figure 17 give a general overview of the interrupt sources and illustrates the request and control flags.

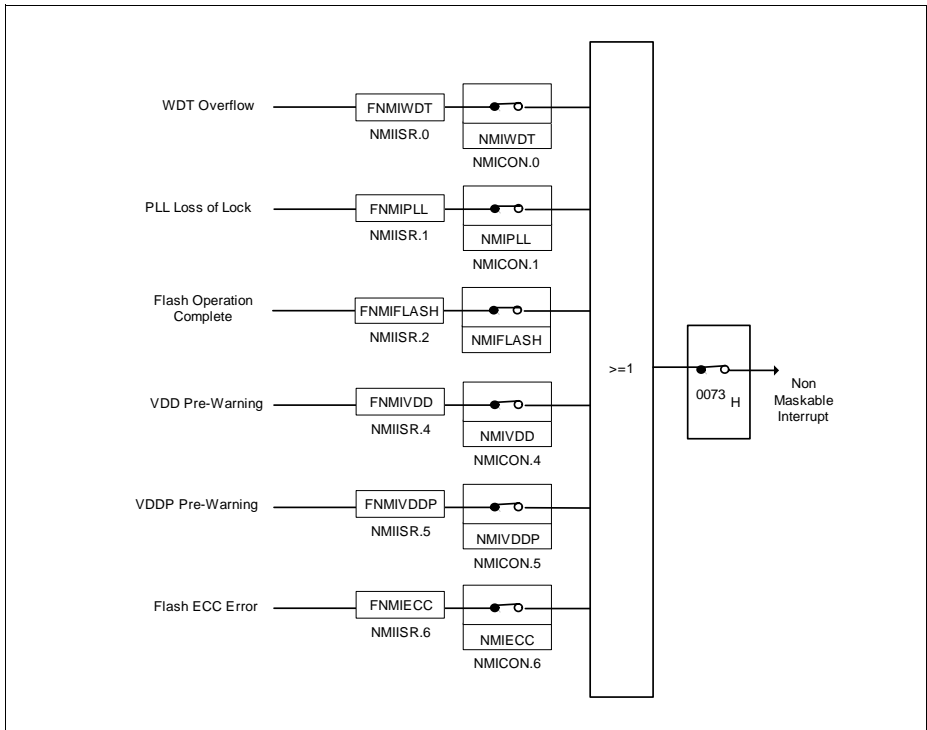


Figure 13 Non-Maskable Interrupt Request Sources

Functional Description

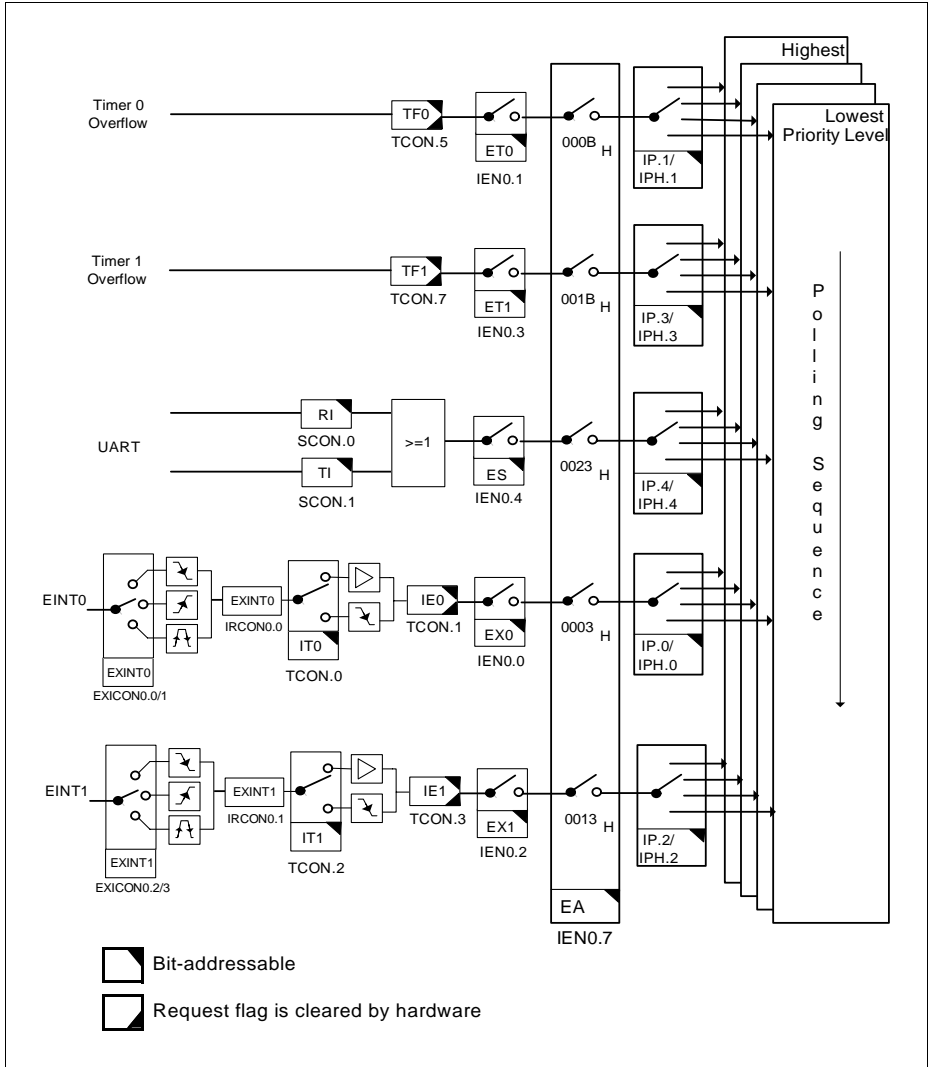


Figure 14 Interrupt Request Sources (Part 1)

Functional Description

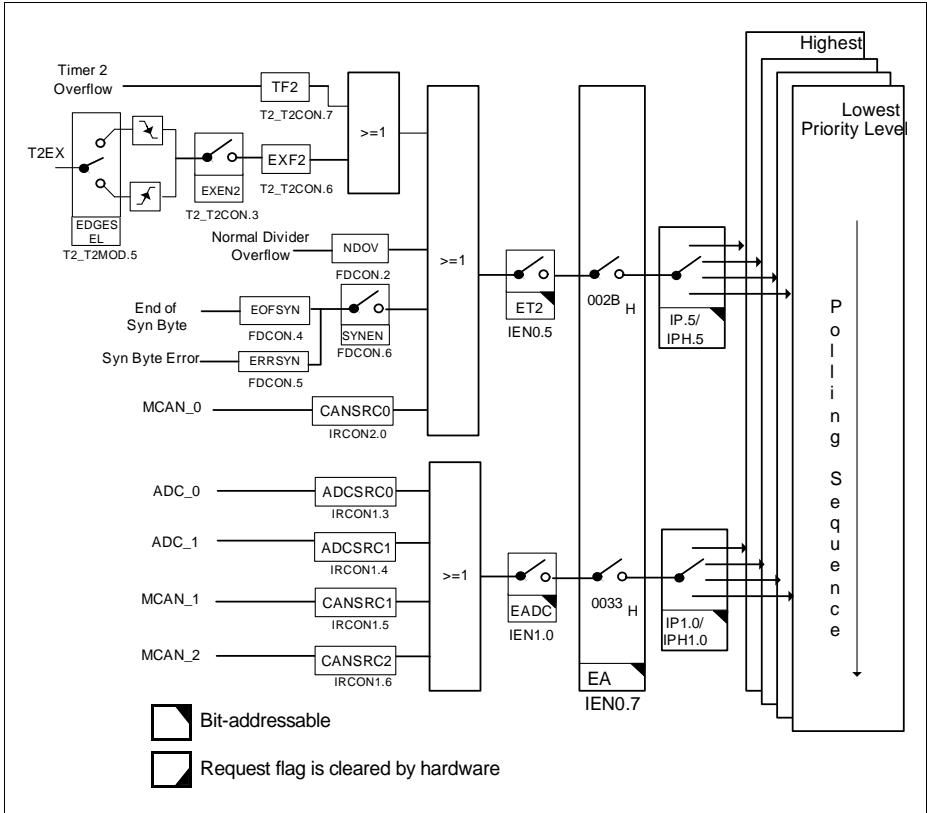


Figure 15 Interrupt Request Sources (Part 2)

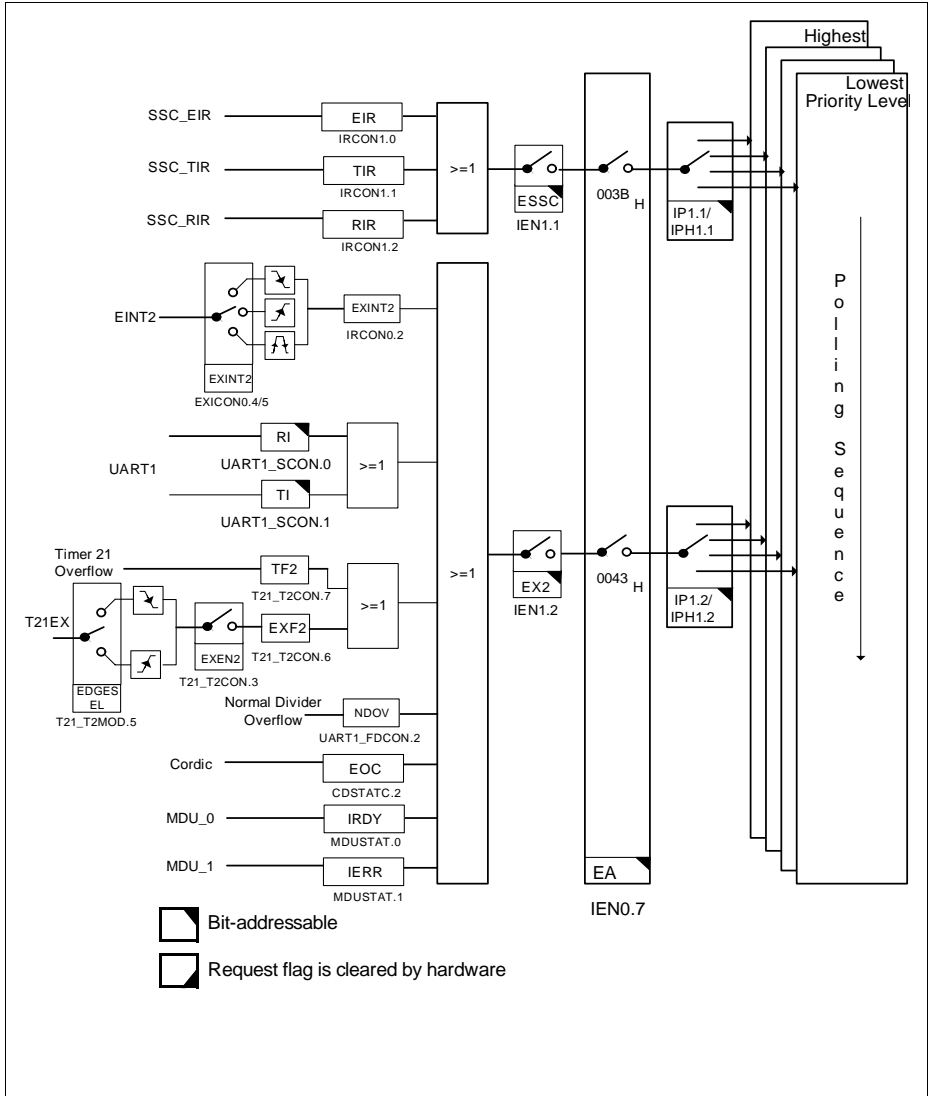


Figure 16 Interrupt Request Sources (Part 3)

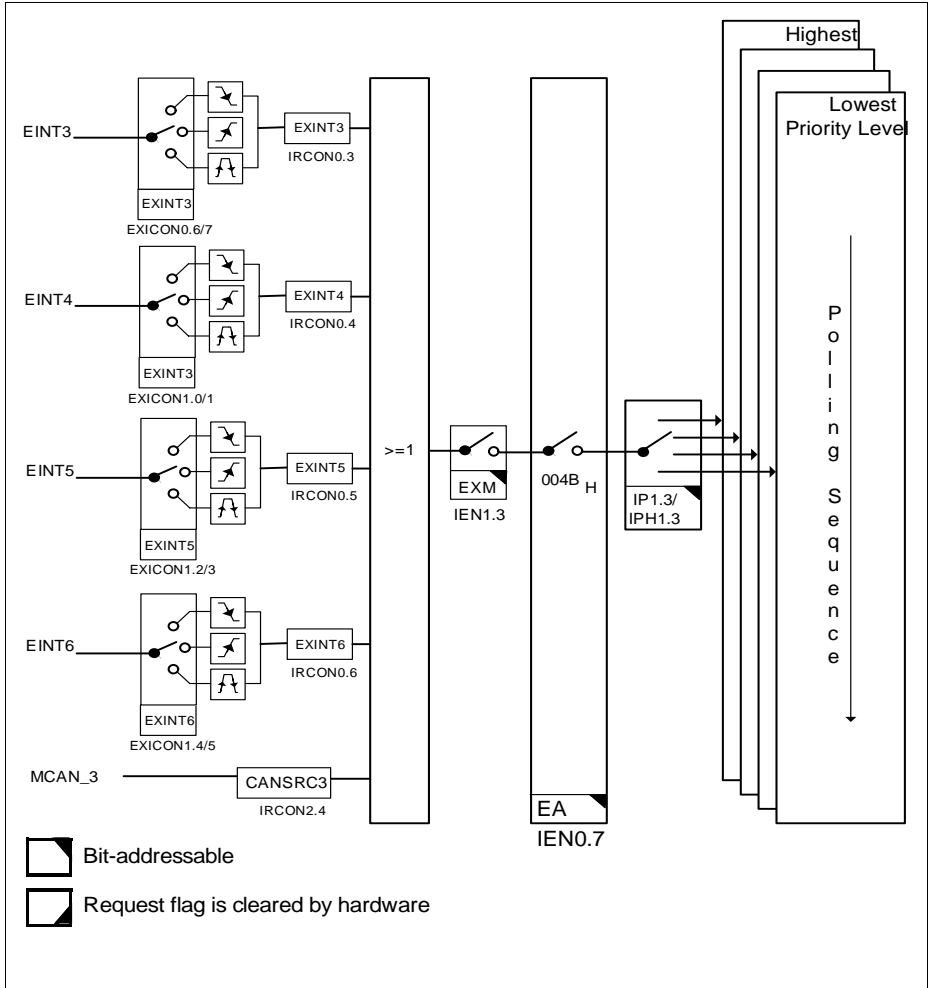


Figure 17 Interrupt Request Sources (Part 4)

Functional Description

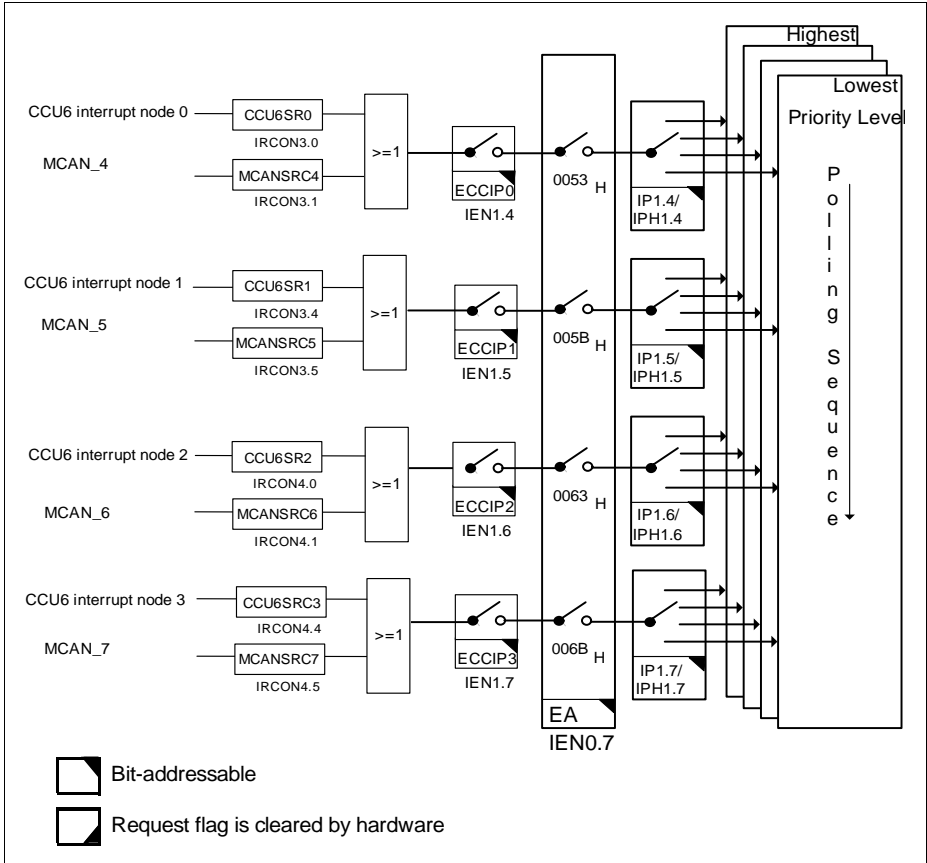


Figure 18 Interrupt Request Sources (Part 5)



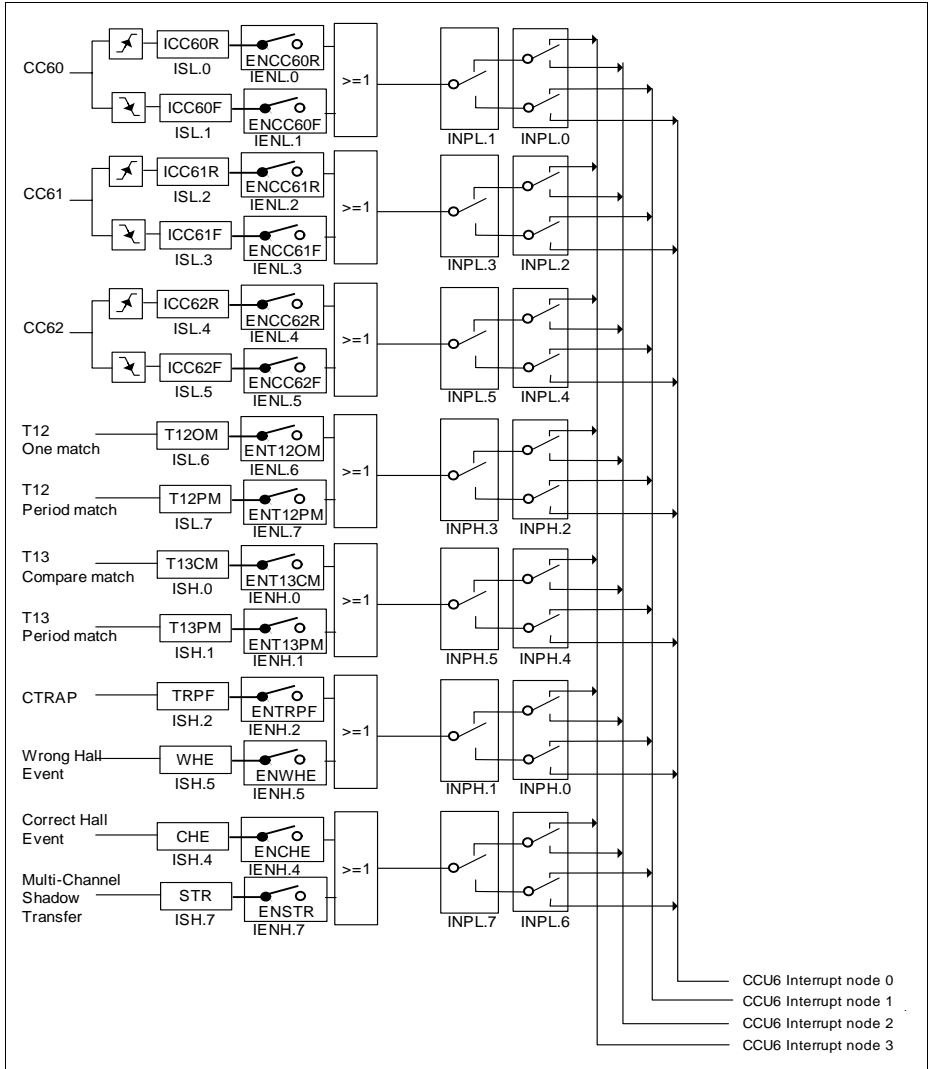


Figure 19 Interrupt Request Sources (Part 6)

### 3.4.2 Interrupt Source and Vector

Each interrupt source has an associated interrupt vector address. This vector is accessed to service the corresponding interrupt source request. The interrupt service of each interrupt source can be individually enabled or disabled via an enable bit. The assignment of the XC886/888 interrupt sources to the interrupt vector addresses and the corresponding interrupt source enable bits are summarized in [Table 20](#).

**Table 20 Interrupt Vector Addresses**

Interrupt Source	Vector Address	Assignment for XC886/888	Enable Bit	SFR
NMI	0073 <sub>H</sub>	Watchdog Timer NMI	NMIWDT	NMICON
		PLL NMI	NMIPLL	
		Flash NMI	NMIFLASH	
		VDDC Prewarning NMI	NMIVDD	
		VDDP Prewarning NMI	NMIVDDP	
		Flash ECC NMI	NMIECC	
XINTR0	0003 <sub>H</sub>	External Interrupt 0	EX0	IEN0
XINTR1	000B <sub>H</sub>	Timer 0	ET0	
XINTR2	0013 <sub>H</sub>	External Interrupt 1	EX1	
XINTR3	001B <sub>H</sub>	Timer 1	ET1	
XINTR4	0023 <sub>H</sub>	UART	ES	
XINTR5	002B <sub>H</sub>	T2	ET2	
		UART Fractional Divider (Normal Divider Overflow)		
		MultiCAN Node 0		
		LIN		

Functional Description

**Table 20** Interrupt Vector Addresses (cont'd)

XINTR6	0033 <sub>H</sub>	MultiCAN Nodes 1 and 2	EADC	IEN1
		ADC[1:0]		
XINTR7	003B <sub>H</sub>	SSC	ESSC	
XINTR8	0043 <sub>H</sub>	External Interrupt 2	EX2	
		T21		
		CORDIC		
		UART1		
		UART1 Fractional Divider (Normal Divider Overflow)		
		MDU[1:0]		
XINTR9	004B <sub>H</sub>	External Interrupt 3	EXM	
		External Interrupt 4		
		External Interrupt 5		
		External Interrupt 6		
		MultiCAN Node 3		
XINTR10	0053 <sub>H</sub>	CCU6 INP0	ECCIP0	
		MultiCAN Node 4		
XINTR11	005B <sub>H</sub>	CCU6 INP1	ECCIP1	
		MultiCAN Node 5		
XINTR12	0063 <sub>H</sub>	CCU6 INP2	ECCIP2	
		MultiCAN Node 6		
XINTR13	006B <sub>H</sub>	CCU6 INP3	ECCIP3	
		MultiCAN Node 7		

### 3.4.3 Interrupt Priority

Each interrupt source, except for NMI, can be individually programmed to one of the four possible priority levels. The NMI has the highest priority and supersedes all other interrupts. Two pairs of interrupt priority registers (IP and IPH, IP1 and IPH1) are available to program the priority level of each non-NMI interrupt vector.

A low-priority interrupt can be interrupted by a high-priority interrupt, but not by another interrupt of the same or lower priority. Further, an interrupt of the highest priority cannot be interrupted by any other interrupt source.

If two or more requests of different priority levels are received simultaneously, the request of the highest priority is serviced first. If requests of the same priority are received simultaneously, then an internal polling sequence determines which request is serviced first. Thus, within each priority level, there is a second priority structure determined by the polling sequence shown in [Table 21](#).

**Table 21 Priority Structure within Interrupt Level**

Source	Level
Non-Maskable Interrupt (NMI)	(highest)
External Interrupt 0	1
Timer 0 Interrupt	2
External Interrupt 1	3
Timer 1 Interrupt	4
UART Interrupt	5
Timer 2, UART Fractional Divider, MCAN, LIN Interrupt	6
ADC, MCAN Interrupt	7
SSC Interrupt	8
External Interrupt 2, Timer 21, UART1, UART1 Fractional Divider, MDU, CORDIC Interrupt	9
External Interrupt [6:3], MCAN Interrupt	10
CCU6 Interrupt Node Pointer 0, MCAN interrupt	11
CCU6 Interrupt Node Pointer 1, MCAN Interrupt	12
CCU6 Interrupt Node Pointer 2, MCAN Interrupt	13
CCU6 Interrupt Node Pointer 3, MCAN Interrupt	14

### **3.5 Parallel Ports**

The XC886 has 34 port pins organized into five parallel ports, Port 0 (P0) to Port 4 (P4), while the XC888 has 48 port pins organized into six parallel ports, Port 0 (P0) to Port 6 (P6). Each pin has a pair of internal pull-up and pull-down devices that can be individually enabled or disabled. Ports P0, P1, P3, P4 and P5 are bidirectional and can be used as general purpose input/output (GPIO) or to perform alternate input/output functions for the on-chip peripherals. When configured as an output, the open drain mode can be selected. Port P2 is an input-only port, providing general purpose input functions, alternate input functions for the on-chip peripherals, and also analog inputs for the Analog-to-Digital Converter (ADC).

#### **Bidirectional Port Features:**

- Configurable pin direction
- Configurable pull-up/pull-down devices
- Configurable open drain mode
- Transfer of data through digital inputs and outputs (general purpose I/O)
- Alternate input/output for on-chip peripherals

#### **Input Port Features:**

- Configurable input driver
- Configurable pull-up/pull-down devices
- Receive of data through digital input (general purpose input)
- Alternate input for on-chip peripherals
- Analog input for ADC module

Functional Description

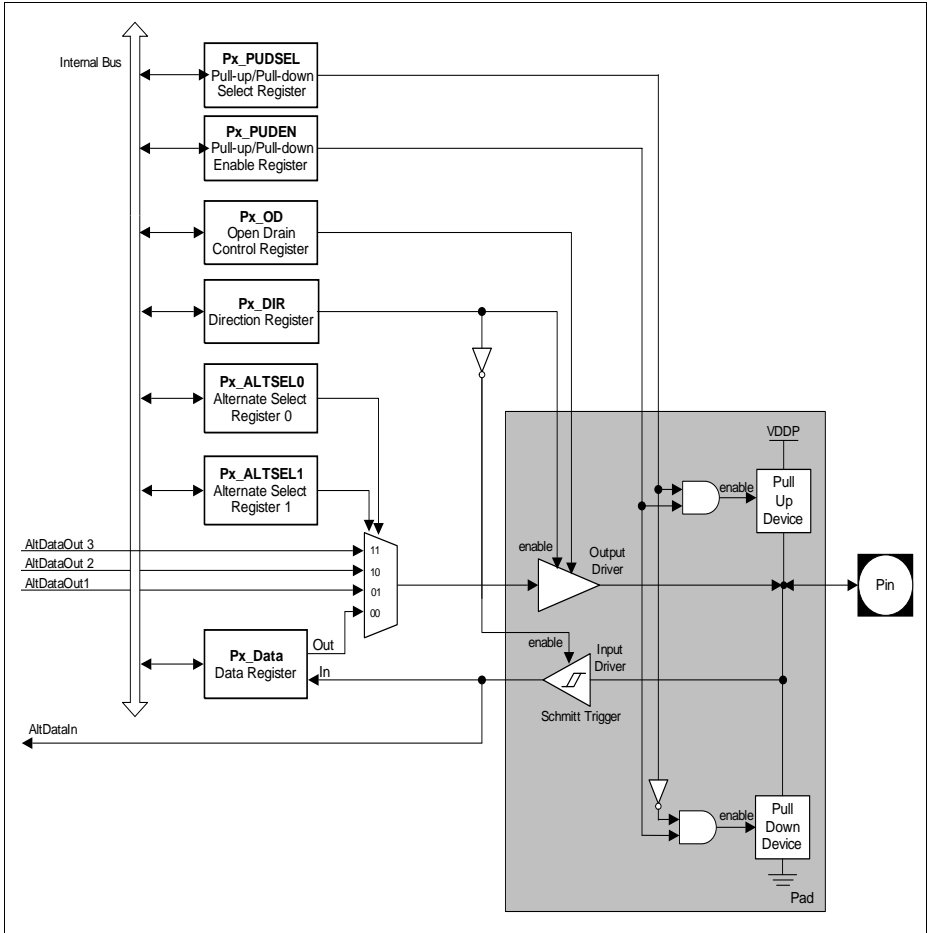


Figure 20 General Structure of Bidirectional Port

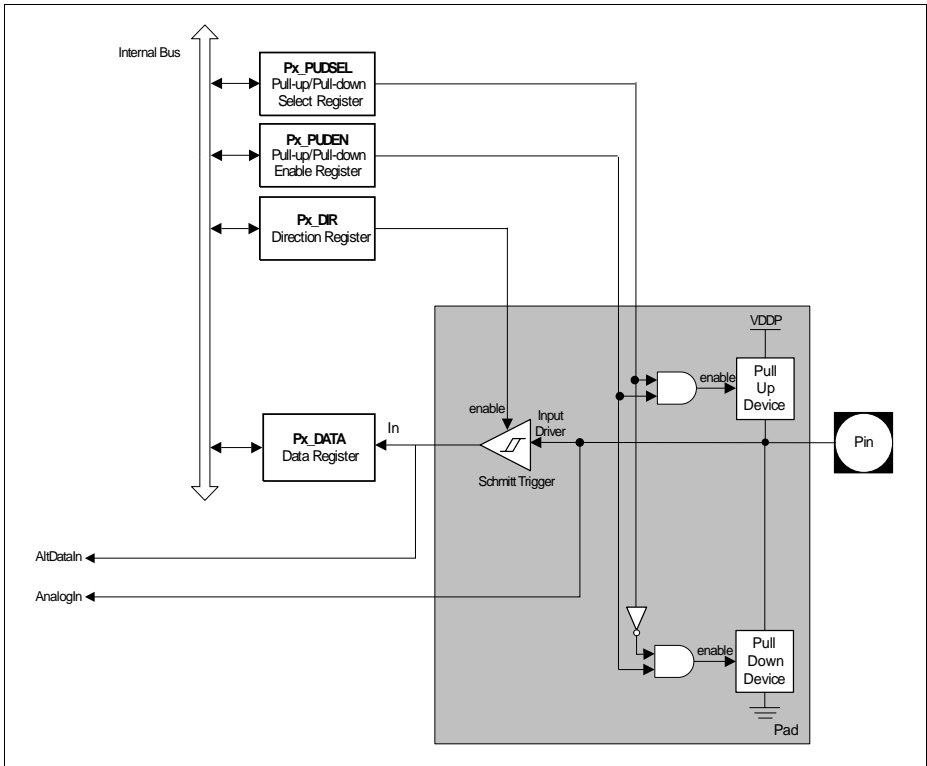


Figure 21 General Structure of Input Port

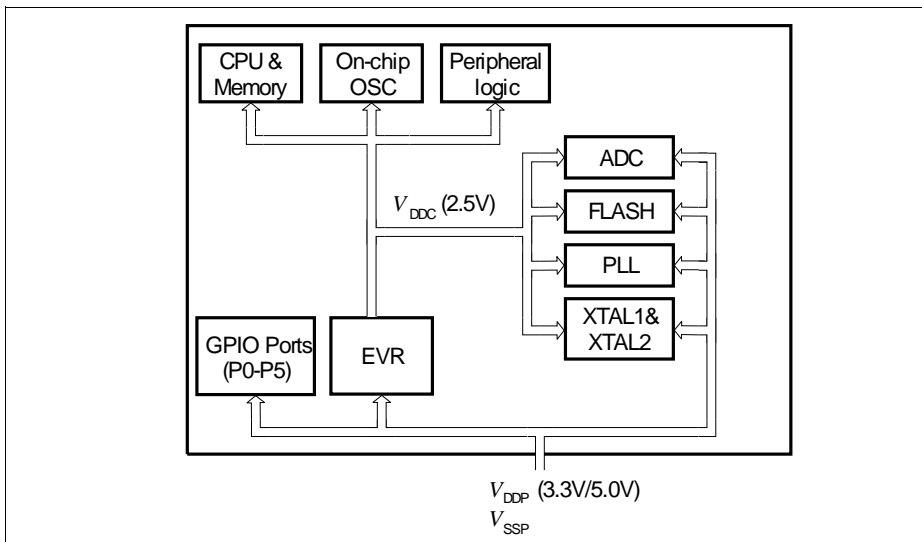
### 3.6 Power Supply System with Embedded Voltage Regulator

The XC886/888 microcontroller requires two different levels of power supply:

- 3.3 V or 5.0 V for the Embedded Voltage Regulator (EVR) and Ports
- 2.5 V for the core, memory, on-chip oscillator, and peripherals

**Figure 22** shows the XC886/888 power supply system. A power supply of 3.3 V or 5.0 V must be provided from the external power supply pin. The 2.5 V power supply for the logic is generated by the EVR. The EVR helps to reduce the power consumption of the whole chip and the complexity of the application board design.

The EVR consists of a main voltage regulator and a low power voltage regulator. In active mode, both voltage regulators are enabled. In power-down mode, the main voltage regulator is switched off, while the low power voltage regulator continues to function and provide power supply to the system with low power consumption.



**Figure 22 XC886/888 Power Supply System**

#### **EVR Features:**

- Input voltage ( $V_{DDP}$ ): 3.3 V/5.0 V
- Output voltage ( $V_{DDC}$ ): 2.5 V  $\pm$  7.5%
- Low power voltage regulator provided in power-down mode
- $V_{DDC}$  and  $V_{DDP}$  prewarning detection
- $V_{DDC}$  brownout detection



### 3.7 Reset Control

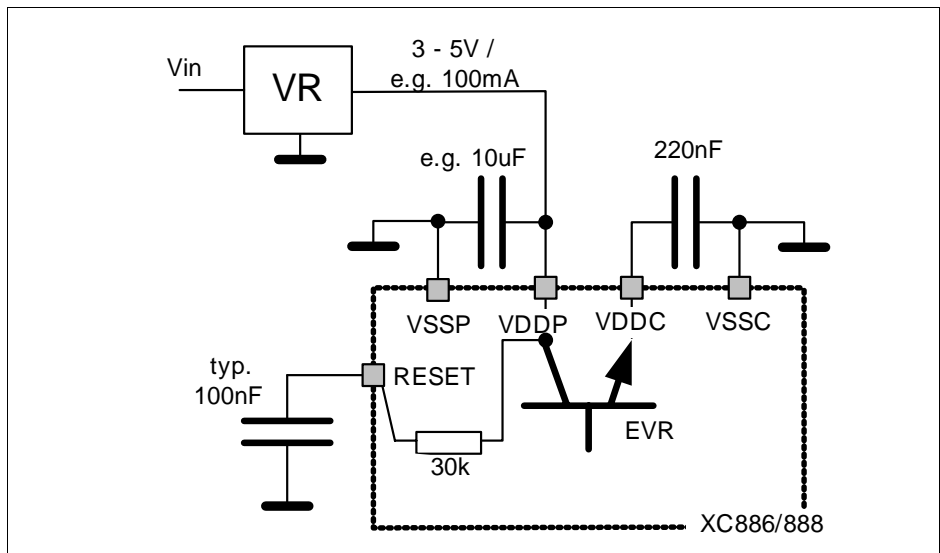
The XC886/888 has five types of reset: power-on reset, hardware reset, watchdog timer reset, power-down wake-up reset, and brownout reset.

When the XC886/888 is first powered up, the status of certain pins (see [Table 23](#)) must be defined to ensure proper start operation of the device. At the end of a reset sequence, the sampled values are latched to select the desired boot option, which cannot be modified until the next power-on reset or hardware reset. This guarantees stable conditions during the normal operation of the device.

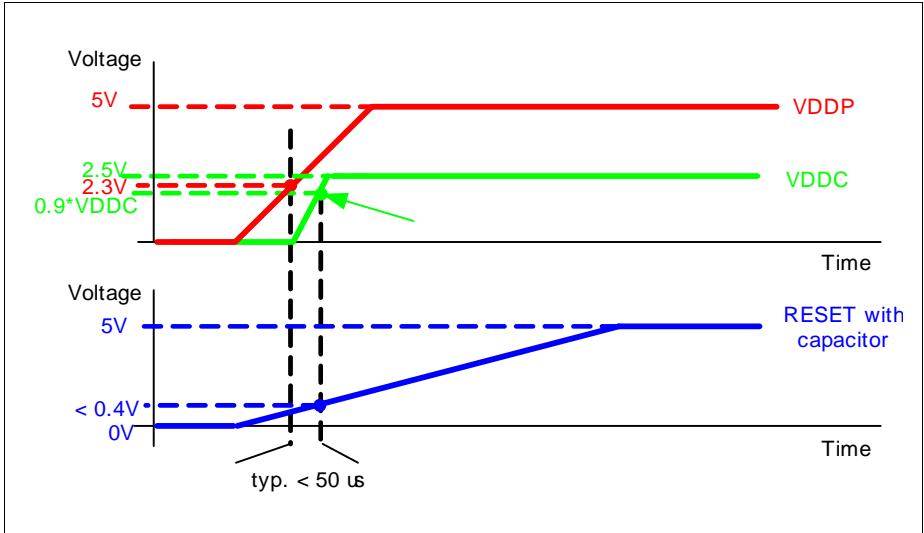
In order to power up the system properly, the external reset pin  $\overline{\text{RESET}}$  must be asserted until  $V_{\text{DDC}}$  reaches  $0.9 \cdot V_{\text{DDC}}$ . The delay of external reset can be realized by an external capacitor at  $\overline{\text{RESET}}$  pin. This capacitor value must be selected so that  $V_{\text{RESET}}$  reaches 0.4 V, but not before  $V_{\text{DDC}}$  reaches  $0.9 \cdot V_{\text{DDC}}$ .

A typical application example is shown in [Figure 23](#). For a voltage regulator with  $I_{\text{DDmax}} = 100 \text{ mA}$ , the  $V_{\text{DDP}}$  capacitor value is 10  $\mu\text{F}$ .  $V_{\text{DDC}}$  capacitor value is 220 nF. The capacitor connected to  $\overline{\text{RESET}}$  pin is 100 nF.

Typically, the time taken for  $V_{\text{DDC}}$  to reach  $0.9 \cdot V_{\text{DDC}}$  is less than 50  $\mu\text{s}$  once  $V_{\text{DDP}}$  reaches 2.3V. Hence, based on the condition that 10% to 90%  $V_{\text{DDP}}$  (slew rate) is less than 500  $\mu\text{s}$ , the  $\overline{\text{RESET}}$  pin should be held low for 500  $\mu\text{s}$  typically. See [Figure 24](#).



**Figure 23** Reset Circuitry



**Figure 24**  $V_{DDP}$ ,  $V_{DDC}$  and  $V_{RESET}$  during Power-on Reset

The second type of reset in XC886/888 is the hardware reset. This reset function can be used during normal operation or when the chip is in power-down mode. A reset input pin RESET is provided for the hardware reset.

The Watchdog Timer (WDT) module is also capable of resetting the device if it detects a malfunction in the system.

Another type of reset that needs to be detected is a reset while the device is in power-down mode (wake-up reset). While the contents of the static RAM are undefined after a power-on reset, they are well defined after a wake-up reset from power-down mode.

### 3.7.1 Module Reset Behavior

**Table 22** shows how the functions of the XC886/888 are affected by the various reset types. A “■” means that this function is reset to its default state.

**Table 22 Effect of Reset on Device Functions**

Module/ Function	Wake-Up Reset	Watchdog Reset	Hardware Reset	Power-On Reset	Brownout Reset
<b>CPU Core</b>	■	■	■	■	■
<b>Peripherals</b>	■	■	■	■	■
<b>On-Chip Static RAM</b>	Not affected, reliable	Not affected, reliable	Not affected, reliable	Affected, un- reliable	Affected, un- reliable
<b>Oscillator, PLL</b>	■	Not affected	■	■	■
<b>Port Pins</b>	■	■	■	■	■
<b>EVR</b>	The voltage regulator is switched on	Not affected	■	■	■
<b>FLASH</b>	■	■	■	■	■
<b>NMI</b>	Disabled	Disabled	■	■	■

### 3.7.2 Booting Scheme

When the XC886/888 is reset, it must identify the type of configuration with which to start the different modes once the reset sequence is complete. Thus, boot configuration information that is required for activation of special modes and conditions needs to be applied by the external world through input pins. After power-on reset or hardware reset, the pins MBC, TMS and P0.0 collectively select the different boot options. **Table 23** shows the available boot options in the XC886/888.

**Table 23 XC886/888 Boot Selection**

MBC	TMS	P0.0	Type of Mode	PC Start Value
1	0	x	User Mode; on-chip OSC/PLL non-bypassed	0000 <sub>H</sub>
0	0	x	BSL Mode; on-chip OSC/PLL non-bypassed	0000 <sub>H</sub>
0	1	0	OCDS Mode; on-chip OSC/PLL non-bypassed	0000 <sub>H</sub>
1	1	0	User (JTAG) Mode <sup>1)</sup> ; on-chip OSC/PLL non-bypassed (normal)	0000 <sub>H</sub>

<sup>1)</sup> Normal user mode with standard JTAG (TCK, TDI, TDO) pins for hot-attach purpose.

### 3.8 Clock Generation Unit

The Clock Generation Unit (CGU) allows great flexibility in the clock generation for the XC886/888. The power consumption is indirectly proportional to the frequency, whereas the performance of the microcontroller is directly proportional to the frequency. During user program execution, the frequency can be programmed for an optimal ratio between performance and power consumption. Therefore the power consumption can be adapted to the actual application state.

#### Features:

- Phase-Locked Loop (PLL) for multiplying clock source by different factors
- PLL Base Mode
- Prescaler Mode
- PLL Mode
- Power-down mode support

The CGU consists of an oscillator circuit and a PLL. In the XC886/888, the oscillator can be from either of these two sources: the on-chip oscillator (9.6 MHz) or the external oscillator (3 MHz to 12 MHz). The term "oscillator" is used to refer to both on-chip oscillator and external oscillator, unless otherwise stated. After the reset, the on-chip oscillator will be used by default. The external oscillator can be selected via software. In addition, the PLL provides a fail-safe logic to perform oscillator run and loss-of-lock detection. This allows emergency routines to be executed for system recovery or to perform system shut down.

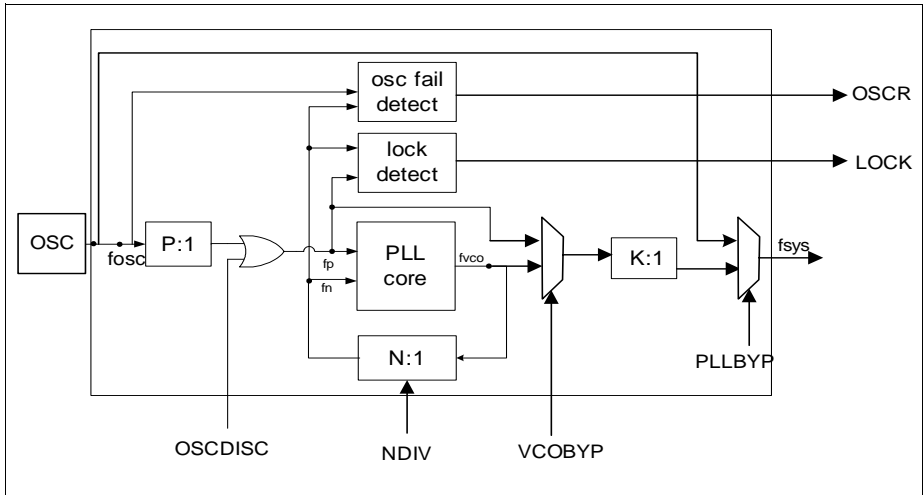


Figure 25 CGU Block Diagram

**Direct Drive (PLL Bypass Operation)**

During PLL bypass operation, the system clock has the same frequency as the external clock source. For the XC886/888, the PLL bypass cannot be set active. Hence, the direct drive mode is not available for use.

$$f_{SYS} = f_{OSC}$$

**PLL Base Mode**

The system clock is derived from the VCO base frequency clock divided by the K factor. Both VCO bypass and PLL bypass must be inactive for this PLL mode.

$$f_{SYS} = f_{VCObase} \times \frac{1}{K}$$

**Prescaler Mode (VCO Bypass Operation)**

In VCO bypass operation, the system clock is derived from the oscillator clock, divided by the P and K factors.

$$f_{SYS} = f_{OSC} \times \frac{1}{P \times K}$$

**PLL Mode**

The system clock is derived from the oscillator clock, multiplied by the N factor, and divided by the P and K factors. Both VCO bypass and PLL bypass must be inactive for this PLL mode. The PLL mode is used during normal system operation. .

$$f_{SYS} = f_{OSC} \times \frac{N}{P \times K}$$

**System Frequency Selection**

For the XC886/888, the value of P is fixed to 1. In order to obtain the required fsys, the value of N and K can be selected by bits NDIV and KDIV respectively for different oscillator inputs. The output frequency must always be configured for 96 MHz. **Table 24** provides examples on how  $f_{sys} = 96$  MHz can be obtained for the different oscillator sources.

**Table 24 System frequency ( $f_{sys} = 96$  MHz)**

Oscillator	fosc	N	P	K	fsys
On-chip	9.6 MHz	20	1	2	96 MHz
External	8 MHz	24	1	2	96 MHz
	6 MHz	32	1	2	96 MHz
	4 MHz	48	1	2	96 MHz

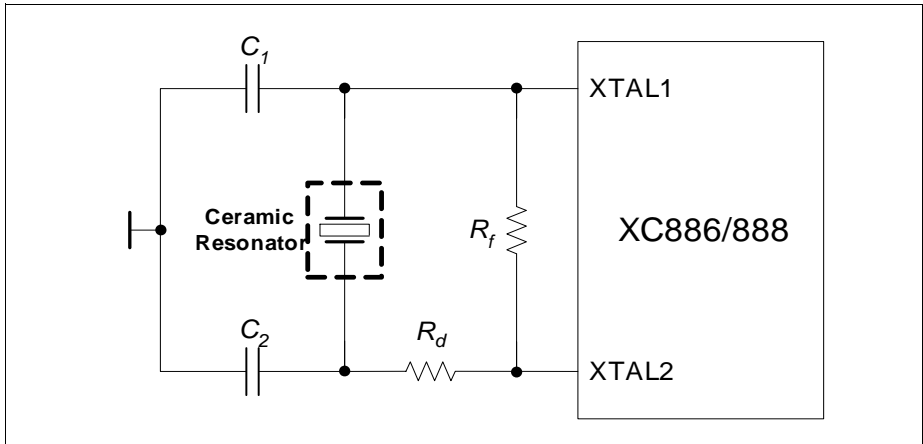
**Table 25** shows the VCO range for the XC886/888.

**Table 25 VCO Range**

fVCOmin	fVCOmax	fVCOFREEmin	fVCOFREEmax	Unit
150	200	20	80	MHz
100	150	10	80	MHz

**3.8.1 Resonator Circuitry**

**Figure 26** shows the recommended ceramic resonator circuitry. When using an external resonator, its frequency can be within the range of 3 MHz to 12 MHz. A resonator load circuitry must be used, connected to both pins, XTAL1 and XTAL2. It normally consists of two load capacitances  $C_1$  and  $C_2$ , and in some cases, a feedback ( $R_f$ ) and/or damp ( $R_d$ ) resistor might be necessary.



**Figure 26 External Ceramic Resonator Circuitry**

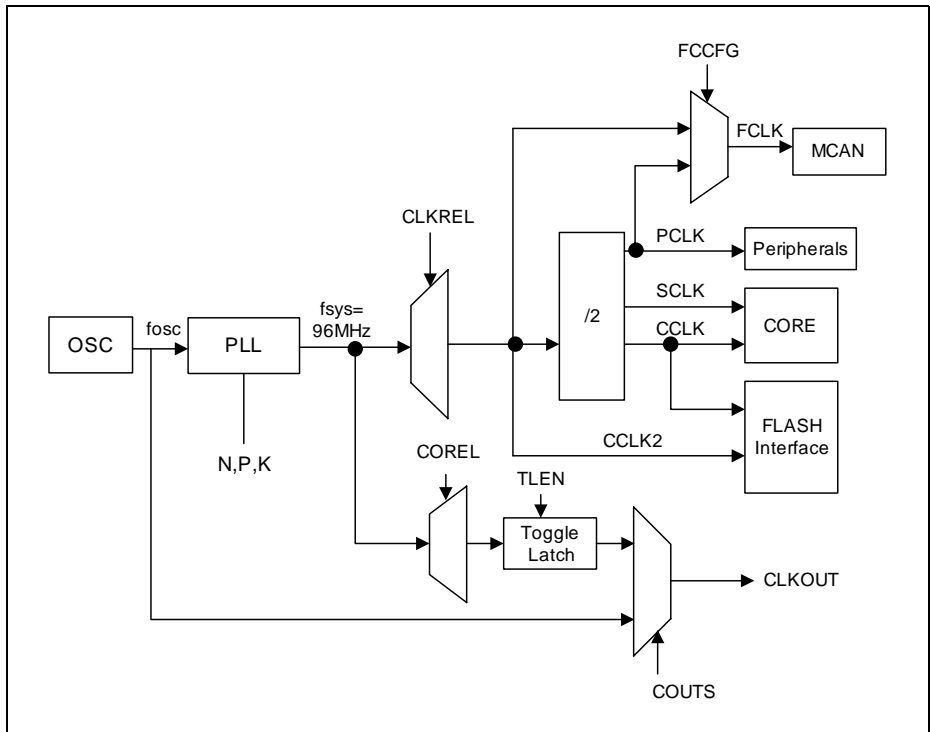
*Note: The manufacturer of the ceramic resonator should check the resonator circuitry and make recommendations for the  $C_1$ ,  $C_2$ ,  $R_f$  and  $R_d$  values to be used for stable start-up behavior.*

### 3.8.2 Clock Management

The CGU generates all clock signals required within the microcontroller from a single clock,  $f_{sys}$ . During normal system operation, the typical frequencies of the different modules are as follow:

- CPU clock: CCLK, SCLK = 24 MHz
- Fast clock (used by MCAN): FCLK = 24 or 48 MHz
- Peripheral clock: PCLK = 24 MHz
- Flash Interface clock: CCLK2 = 96 MHz and CCLK = 24 MHz

In addition, different clock frequency can output to pin CLKOUT(P0.0 or P0.7). The clock output frequency can further be divided by 2 using toggle latch (bit TLEN is set to 1), the resulting output frequency has 50% duty cycle. **Figure 27** shows the clock distribution of the XC886/888.



**Figure 27** Clock Generation from  $f_{sys}$



---

**Functional Description**

For power saving purposes, the clocks may be disabled or slowed down according to [Table 26](#).

**Table 26** System frequency ( $f_{\text{sys}} = 96 \text{ MHz}$ )

<b>Power Saving Mode</b>	<b>Action</b>
Idle	Clock to the CPU is disabled.
Slow-down	Clocks to the CPU and all the peripherals are divided by a common programmable factor defined by bit field CMCON.CLKREL.
Power-down	Oscillator and PLL are switched off.

### 3.9 Power Saving Modes

The power saving modes of the XC886/888 provide flexible power consumption through a combination of techniques, including:

- Stopping the CPU clock
- Stopping the clocks of individual system components
- Reducing clock speed of some peripheral components
- Power-down of the entire system with fast restart capability

After a reset, the active mode (normal operating mode) is selected by default (see [Figure 28](#)) and the system runs in the main system clock frequency. From active mode, different power saving modes can be selected by software. They are:

- Idle mode
- Slow-down mode
- Power-down mode

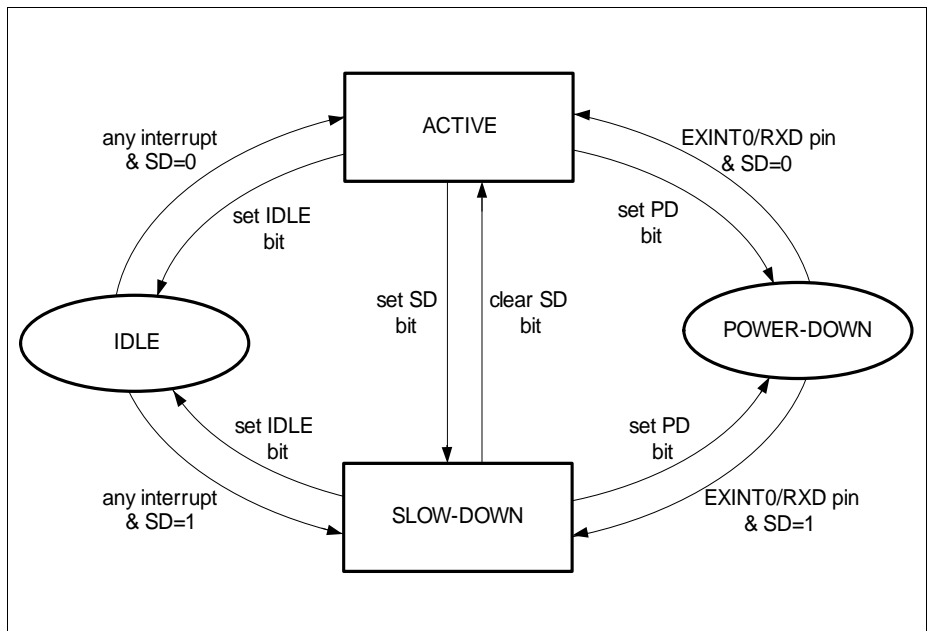


Figure 28 Transition between Power Saving Modes

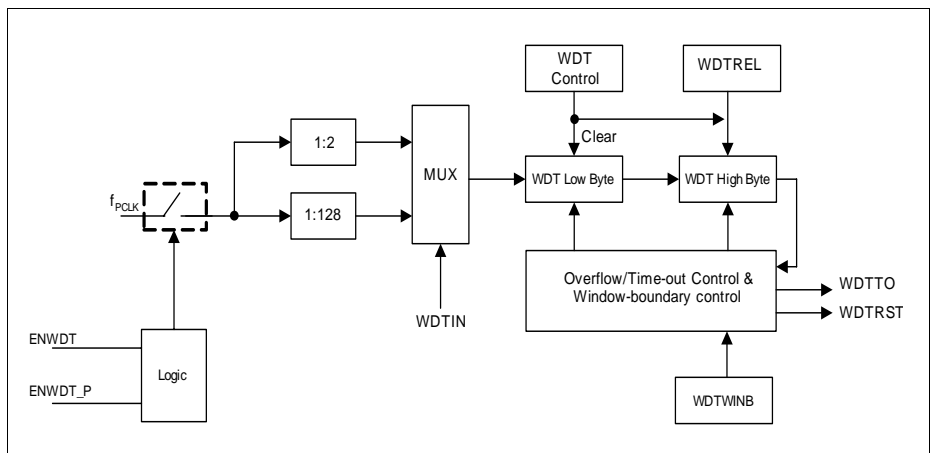
### 3.10 Watchdog Timer

The Watchdog Timer (WDT) provides a highly reliable and secure way to detect and recover from software or hardware failures. The WDT is reset at a regular interval that is predefined by the user. The CPU must service the WDT within this interval to prevent the WDT from causing an XC886/888 system reset. Hence, routine service of the WDT confirms that the system is functioning properly. This ensures that an accidental malfunction of the XC886/888 will be aborted in a user-specified time period. In debug mode, the WDT is suspended and stops counting. Therefore, there is no need to refresh the WDT during debugging.

**Features:**

- 16-bit Watchdog Timer
- Programmable reload value for upper 8 bits of timer
- Programmable window boundary
- Selectable input frequency of  $f_{PCLK}/2$  or  $f_{PCLK}/128$
- Time-out detection with NMI generation and reset prewarning activation (after which a system reset will be performed)

The WDT is a 16-bit timer incremented by a count rate of  $f_{PCLK}/2$  or  $f_{PCLK}/128$ . This 16-bit timer is realized as two concatenated 8-bit timers. The upper 8 bits of the WDT can be preset to a user-programmable value via a watchdog service access in order to modify the watchdog expire time period. The lower 8 bits are reset on each service access. **Figure 29** shows the block diagram of the WDT unit.



**Figure 29 WDT Block Diagram**

**Functional Description**

If the WDT is not serviced before the timer overflow, a system malfunction is assumed. As a result, the WDT NMI is triggered (assert WDTTO) and the reset prewarning is entered. The prewarning period lasts for 30<sub>H</sub> count, after which the system is reset (assert WDTRST).

The WDT has a “programmable window boundary” which disallows any refresh during the WDT’s count-up. A refresh during this window boundary constitutes an invalid access to the WDT, causing the reset prewarning to be entered but without triggering the WDT NMI. The system will still be reset after the prewarning period is over. The window boundary is from 0000<sub>H</sub> to the value obtained from the concatenation of WDTWINB and 00<sub>H</sub>.

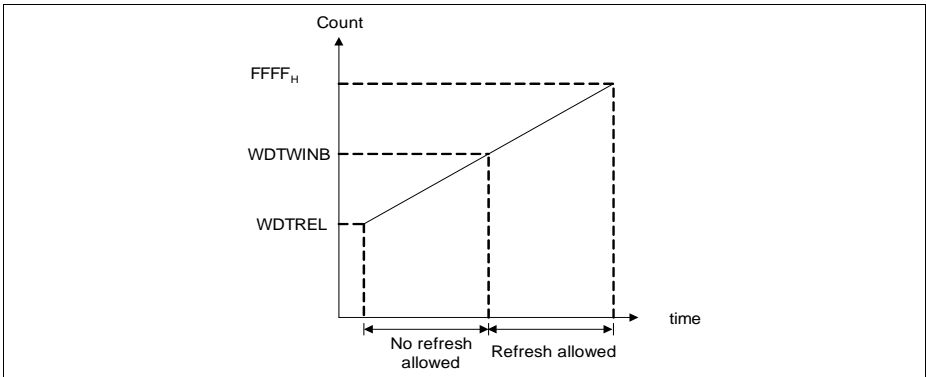
After being serviced, the WDT continues counting up from the value (<WDTREL> \* 2<sup>8</sup>). The time period for an overflow of the WDT is programmable in two ways:

- the input frequency to the WDT can be selected to be either f<sub>PCLK</sub>/2 or f<sub>PCLK</sub>/128
- the reload value WDTREL for the high byte of WDT can be programmed in register WDTREL

The period, P<sub>WDT</sub>, between servicing the WDT and the next overflow can be determined by the following formula:

$$P_{WDT} = \frac{2^{(1 + WDTIN \times 6)} \times (2^{16} - WDTREL \times 2^8)}{f_{PCLK}}$$

If the Window-Boundary Refresh feature of the WDT is enabled, the period P<sub>WDT</sub> between servicing the WDT and the next overflow is shortened if WDTWINB is greater than WDTREL, see [Figure 30](#). This period can be calculated using the same formula by replacing WDTREL with WDTWINB. For this feature to be useful, WDTWINB should not be smaller than WDTREL.



**Figure 30 WDT Timing Diagram**

**Functional Description**

**Table 27** lists the possible watchdog time range that can be achieved for different module clock frequencies . Some numbers are rounded to 3 significant digits.

**Table 27 Watchdog Time Ranges**

Reload value in WDTREL	Prescaler for $f_{PCLK}$	
	2 (WDTIN = 0)	128 (WDTIN = 1)
	24 MHz	24 MHz
FF <sub>H</sub>	21.3 $\mu$ s	1.37 ms
7F <sub>H</sub>	2.75 ms	176 ms
00 <sub>H</sub>	5.46 ms	350 ms

### 3.11 UART and UART1

The XC886/888 provides two Universal Asynchronous Receiver/Transmitter (UART and UART1) modules for full-duplex asynchronous reception/transmission. Both are also receive-buffered, i.e., they can commence reception of a second byte before a previously received byte has been read from the receive register. However, if the first byte still has not been read by the time reception of the second byte is complete, one of the bytes will be lost.

#### Features:

- Full-duplex asynchronous modes
  - 8-bit or 9-bit data frames, LSB first
  - fixed or variable baud rate
- Receive buffered
- Multiprocessor communication
- Interrupt generation on the completion of a data transmission or reception

The UART modules can operate in four asynchronous modes as shown in [Table 28](#). Data is transmitted on TXD and received on RXD.

**Table 28 UART Modes**

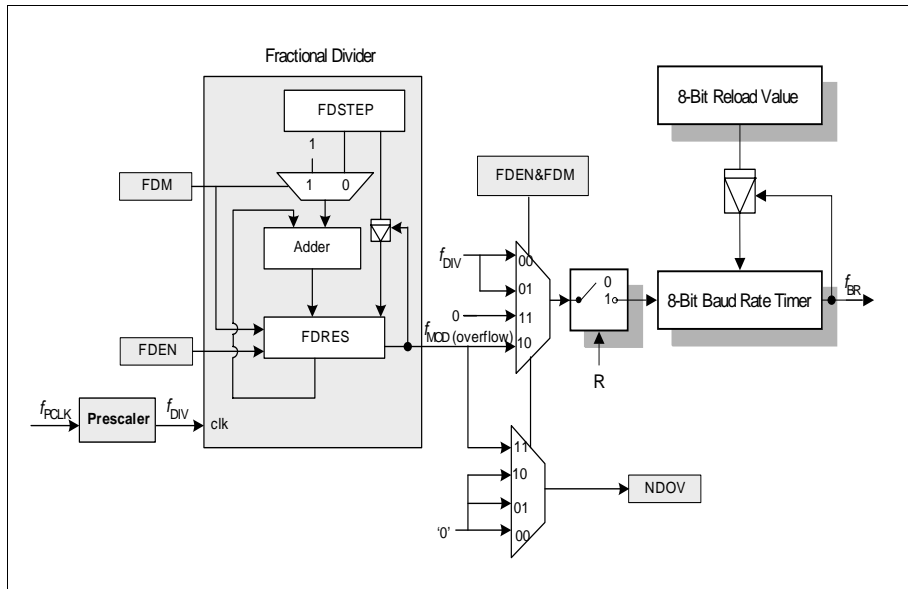
Operating Mode	Baud Rate
Mode 0: 8-bit shift register	$f_{PCLK}/2$
Mode 1: 8-bit shift UART	Variable
Mode 2: 9-bit shift UART	$f_{PCLK}/32$ or $f_{PCLK}/64$ <sup>1)</sup>
Mode 3: 9-bit shift UART	Variable

<sup>1)</sup> For UART1 module, the baud rate is fixed at  $f_{PCLK}/64$ .

There are several ways to generate the baud rate clock for the serial port, depending on the mode in which it is operating. In mode 0, the baud rate for the transfer is fixed at  $f_{PCLK}/2$ . In mode 2, the baud rate is generated internally based on the UART input clock and can be configured to either  $f_{PCLK}/32$  or  $f_{PCLK}/64$ . For UART1 module, only  $f_{PCLK}/64$  is available. The variable baud rate is set by the underflow rate on the dedicated baud-rate generator. For UART module, the variable baud rate alternatively can be set by the overflow rate on Timer 1.

### 3.11.1 Baud-Rate Generator

Both UART modules have their own dedicated baud-rate generator, which is based on a programmable 8-bit reload value, and includes divider stages (i.e., prescaler and fractional divider) for generating a wide range of baud rates based on its input clock  $f_{PCLK}$ , see [Figure 31](#).



**Figure 31 Baud-rate Generator Circuitry**

The baud rate timer is a count-down timer and is clocked by either the output of the fractional divider ( $f_{MOD}$ ) if the fractional divider is enabled ( $FDCON.FDEN = 1$ ), or the output of the prescaler ( $f_{DIV}$ ) if the fractional divider is disabled ( $FDEN = 0$ ). For baud rate generation, the fractional divider must be configured to fractional divider mode ( $FDCON.FDM = 0$ ). This allows the baud rate control run bit  $BCON.R$  to be used to start or stop the baud rate timer. At each timer underflow, the timer is reloaded with the 8-bit reload value in register BG and one clock pulse is generated for the serial channel.

Enabling the fractional divider in normal divider mode ( $FDEN = 1$  and  $FDM = 1$ ) stops the baud rate timer and nullifies the effect of bit  $BCON.R$ . See [Section 3.12](#).

The baud rate ( $f_{BR}$ ) value is dependent on the following parameters:

- Input clock  $f_{PCLK}$
- Prescaling factor ( $2^{BRPRE}$ ) defined by bit field BRPRE in register BCON

**Functional Description**

- Fractional divider (STEP/256) defined by register FDSTEP (to be considered only if fractional divider is enabled and operating in fractional divider mode)
- 8-bit reload value (BR\_VALUE) for the baud rate timer defined by register BG

The following formulas calculate the final baud rate without and with the fractional divider respectively:

$$\text{baud rate} = \frac{f_{\text{PCLK}}}{16 \times 2^{\text{BRPRE}} \times (\text{BR\_VALUE} + 1)} \quad \text{where } 2^{\text{BRPRE}} \times (\text{BR\_VALUE} + 1) > 1$$

$$\text{baud rate} = \frac{f_{\text{PCLK}}}{16 \times 2^{\text{BRPRE}} \times (\text{BR\_VALUE} + 1)} \times \frac{\text{STEP}}{256}$$

The maximum baud rate that can be generated is limited to  $f_{\text{PCLK}}/32$ . Hence, for a module clock of 24 MHz, the maximum achievable baud rate is 0.75 MBaud.

Standard LIN protocol can support a maximum baud rate of 20kHz, the baud rate accuracy is not critical and the fractional divider can be disabled. Only the prescaler is used for auto baud rate calculation. For LIN fast mode, which supports the baud rate of 20kHz to 115.2kHz, the higher baud rates require the use of the fractional divider for greater accuracy.

**Table 29** lists the various commonly used baud rates with their corresponding parameter settings and deviation errors. The fractional divider is disabled and a module clock of 24 MHz is used.

**Table 29 Typical Baud rates for UART with Fractional Divider disabled**

Baud rate	Prescaling Factor ( $2^{\text{BRPRE}}$ )	Reload Value (BR_VALUE + 1)	Deviation Error
19.2 kBaud	1 (BRPRE=000 <sub>B</sub> )	78 (4E <sub>H</sub> )	0.17 %
9600 Baud	1 (BRPRE=000 <sub>B</sub> )	156 (9C <sub>H</sub> )	0.17 %
4800 Baud	2 (BRPRE=001 <sub>B</sub> )	156 (9C <sub>H</sub> )	0.17 %
2400 Baud	4 (BRPRE=010 <sub>B</sub> )	156 (9C <sub>H</sub> )	0.17 %

The fractional divider allows baud rates of higher accuracy (lower deviation error) to be generated. **Table 30** lists the resulting deviation errors from generating a baud rate of 115.2 kHz, using different module clock frequencies. The fractional divider is enabled (fractional divider mode) and the corresponding parameter settings are shown.



**Table 30      Deviation Error for UART with Fractional Divider enabled**

<b>f<sub>PCLK</sub></b>	<b>Prescaling Factor (2<sup>BRPRE</sup>)</b>	<b>Reload Value (BR_VALUE + 1)</b>	<b>STEP</b>	<b>Deviation Error</b>
24 MHz	1	10 (A <sub>H</sub> )	197 (C5 <sub>H</sub> )	+0.20 %
12 MHz	1	6 (6 <sub>H</sub> )	236 (EC <sub>H</sub> )	+0.03 %
6.67 MHz	1	3 (3 <sub>H</sub> )	236 (EC <sub>H</sub> )	+0.03 %

### 3.11.2 Baud Rate Generation using Timer 1

In UART modes 1 and 3 of UART module, Timer 1 can be used for generating the variable baud rates. In theory, this timer could be used in any of its modes. But in practice, it should be set into auto-reload mode (Timer 1 mode 2), with its high byte set to the appropriate value for the required baud rate. The baud rate is determined by the Timer 1 overflow rate and the value of SMOD as follows:

$$\text{Mode 1, 3 baud rate} = \frac{2^{\text{SMOD}} \times f_{\text{PCLK}}}{32 \times 2 \times (256 - \text{TH1})} \quad [3.1]$$

### 3.12 Normal Divider Mode (8-bit Auto-reload Timer)

Setting bit FDM in register FDCON to 1 configures the fractional divider to normal divider mode, while at the same time disables baud rate generation (see [Figure 31](#)). Once the fractional divider is enabled (FDEN = 1), it functions as an 8-bit auto-reload timer (with no relation to baud rate generation) and counts up from the reload value with each input clock pulse. Bit field RESULT in register FDRES represents the timer value, while bit field STEP in register FDSTEP defines the reload value. At each timer overflow, an overflow flag (FDCON.NDOV) will be set and an interrupt request generated. This gives an output clock  $f_{\text{MOD}}$  that is 1/n of the input clock  $f_{\text{DIV}}$ , where n is defined by 256 - STEP. The output frequency in normal divider mode is derived as follows:

$$f_{\text{MOD}} = f_{\text{DIV}} \times \frac{1}{256 - \text{STEP}} \quad [3.2]$$

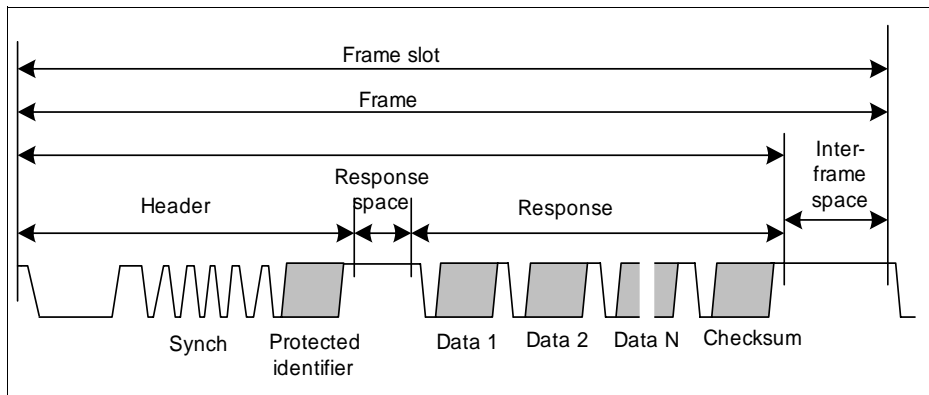
### 3.13 LIN Protocol

The UART module can be used to support the Local Interconnect Network (LIN) protocol for both master and slave operations. This option is not available with UART1 module. The LIN baud rate detection feature provides the capability to detect the baud rate within LIN protocol using Timer 2. This allows the UART to be synchronized to the LIN baud rate for data transmission and reception.

LIN is a holistic communication concept for local interconnected networks in vehicles. The communication is based on the SCI (UART) data format, a single-master/multiple-slave concept, a clock synchronization for nodes without stabilized time base. An attractive feature of LIN is self-synchronization of the slave nodes without a crystal or ceramic resonator, which significantly reduces the cost of hardware platform. Hence, the baud rate must be calculated and returned with every message frame.

The structure of a LIN frame is shown in **Figure 32**. The frame consists of the:

- header, which comprises a Break (13-bit time low), Synch Byte (55<sub>H</sub>), and ID field
- response time
- data bytes (according to UART protocol)
- checksum



**Figure 32 Structure of LIN Frame**

#### 3.13.1 LIN Header Transmission

LIN header transmission is only applicable in master mode. In the LIN communication, a master task decides when and which frame is to be transferred on the bus. It also identifies a slave task to provide the data transported by each frame. The information needed for the handshaking between the master and slave tasks is provided by the master task through the header portion of the frame.

---

**Functional Description**

The header consists of a break and synch pattern followed by an identifier. Among these three fields, only the break pattern cannot be transmitted as a normal 8-bit UART data. The break must contain a dominant value of 13 bits or more to ensure proper synchronization of slave nodes.

In the LIN communication, a slave task is required to be synchronized at the beginning of the protected identifier field of frame. For this purpose, every frame starts with a sequence consisting of a break field followed by a synch byte field. This sequence is unique and provides enough information for any slave task to detect the beginning of a new frame and be synchronized at the start of the identifier field.

Upon entering LIN communication, a connection is established and the transfer speed (baud rate) of the serial communication partner (host) is automatically synchronized in the following steps:

STEP 1: Initialize interface for reception and timer for baud rate measurement

STEP 2: Wait for an incoming LIN frame from host

STEP 3: Synchronize the baud rate to the host

STEP 4: Enter for Master Request Frame or for Slave Response Frame

*Note: Re-synchronization and setup of baud rate are always done for **every** Master Request Header or Slave Response Header LIN frame.*

### 3.14 High-Speed Synchronous Serial Interface

The High-Speed Synchronous Serial Interface (SSC) supports full-duplex and half-duplex synchronous communication. The serial clock signal can be generated by the SSC internally (master mode), using its own 16-bit baud-rate generator, or can be received from an external master (slave mode). Data width, shift direction, clock polarity and phase are programmable. This allows communication with SPI-compatible devices or devices using other synchronous serial interfaces.

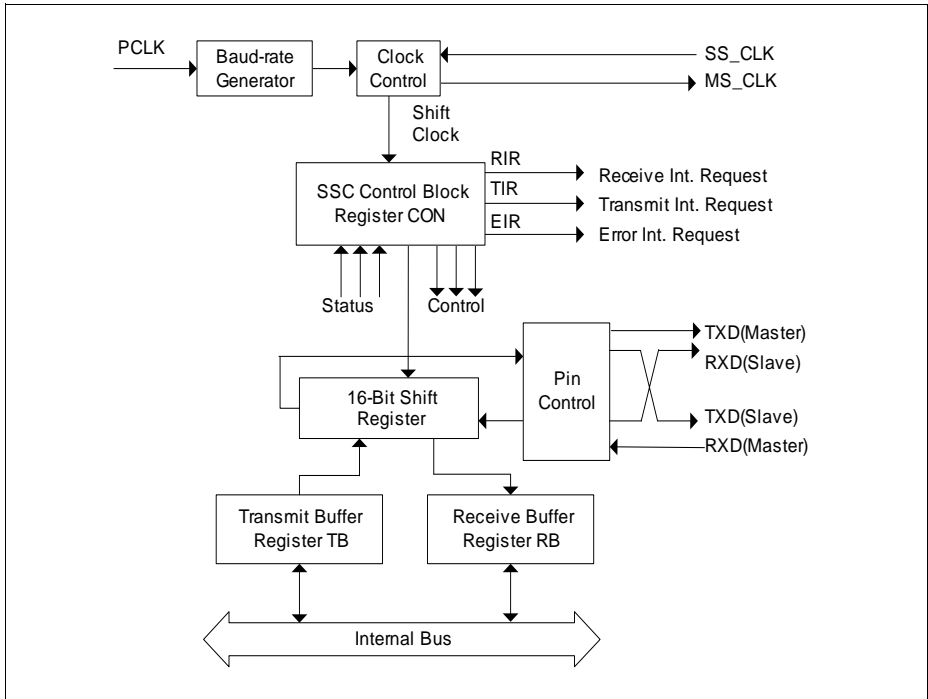
#### Features:

- Master and slave mode operation
  - Full-duplex or half-duplex operation
- Transmit and receive buffered
- Flexible data format
  - Programmable number of data bits: 2 to 8 bits
  - Programmable shift direction: LSB or MSB shift first
  - Programmable clock polarity: idle low or high state for the shift clock
  - Programmable clock/data phase: data shift with leading or trailing edge of the shift clock
- Variable baud rate
- Compatible with Serial Peripheral Interface (SPI)
- Interrupt generation
  - On a transmitter empty condition
  - On a receiver full condition
  - On an error condition (receive, phase, baud rate, transmit error)

**Functional Description**

Data is transmitted or received on lines TXD and RXD, which are normally connected to the pins MTSR (Master Transmit/Slave Receive) and MRST (Master Receive/Slave Transmit). The clock signal is output via line MS\_CLK (Master Serial Shift Clock) or input via line SS\_CLK (Slave Serial Shift Clock). Both lines are normally connected to the pin SCLK. Transmission and reception of data are double-buffered.

**Figure 33** shows the block diagram of the SSC.



**Figure 33 SSC Block Diagram**

### 3.15 Timer 0 and Timer 1

Timer 0 and Timer 1 can function as both timers or counters. When functioning as a timer, Timer 0 and Timer 1 are incremented every machine cycle, i.e. every 2 input clocks (or 2 PCLKs). When functioning as a counter, Timer 0 and Timer 1 are incremented in response to a 1-to-0 transition (falling edge) at their respective external input pins, T0 or T1.

Timer 0 and 1 are fully compatible and can be configured in four different operating modes for use in a variety of applications, see [Table 31](#). In modes 0, 1 and 2, the two timers operate independently, but in mode 3, their functions are specialized.

**Table 31 Timer 0 and Timer 1 Modes**

Mode	Operation
<b>0</b>	<b>13-bit timer</b> The timer is essentially an 8-bit counter with a divide-by-32 prescaler. This mode is included solely for compatibility with Intel 8048 devices.
<b>1</b>	<b>16-bit timer</b> The timer registers, TLx and THx, are concatenated to form a 16-bit counter.
<b>2</b>	<b>8-bit timer with auto-reload</b> The timer register TLx is reloaded with a user-defined 8-bit value in THx upon overflow.
<b>3</b>	<b>Timer 0 operates as two 8-bit timers</b> The timer registers, TL0 and TH0, operate as two separate 8-bit counters. Timer 1 is halted and retains its count even if enabled.

### 3.16 Timer 2 and Timer 21

Timer 2 and Timer 21 are 16-bit general purpose timers (THL2) that are fully compatible and have two modes of operation, a 16-bit auto-reload mode and a 16-bit one channel capture mode. As a timer, the timers count with an input clock of PCLK/12 (if prescaler is disabled). As a counter, they count 1-to-0 transitions on pin T2. In the counter mode, the maximum resolution for the count is PCLK/24 (if prescaler is disabled).

**Table 32 Timer 2 Modes**

Mode	Description
<b>Auto-reload</b>	<b>Up/Down Count Disabled</b> <ul style="list-style-type: none"> <li>• Count up only</li> <li>• Start counting from 16-bit reload value, overflow at FFFF<sub>H</sub></li> <li>• Reload event configurable for trigger by overflow condition only, or by negative/positive edge at input pin T2EX as well</li> <li>• Programmable reload value in register RC2</li> <li>• Interrupt is generated with reload event</li> </ul>
	<b>Up/Down Count Enabled</b> <ul style="list-style-type: none"> <li>• Count up or down, direction determined by level at input pin T2EX</li> <li>• No interrupt is generated</li> <li>• Count up <ul style="list-style-type: none"> <li>– Start counting from 16-bit reload value, overflow at FFFF<sub>H</sub></li> <li>– Reload event triggered by overflow condition</li> <li>– Programmable reload value in register RC2</li> </ul> </li> <li>• Count down <ul style="list-style-type: none"> <li>– Start counting from FFFF<sub>H</sub>, underflow at value defined in register RC2</li> <li>– Reload event triggered by underflow condition</li> <li>– Reload value fixed at FFFF<sub>H</sub></li> </ul> </li> </ul>
<b>Channel capture</b>	<ul style="list-style-type: none"> <li>• Count up only</li> <li>• Start counting from 0000<sub>H</sub>, overflow at FFFF<sub>H</sub></li> <li>• Reload event triggered by overflow condition</li> <li>• Reload value fixed at 0000<sub>H</sub></li> <li>• Capture event triggered by falling/rising edge at pin T2EX</li> <li>• Captured timer value stored in register RC2</li> <li>• Interrupt is generated with reload or capture event</li> </ul>



### 3.17 Capture/Compare Unit 6

The Capture/Compare Unit 6 (CCU6) provides two independent timers (T12, T13), which can be used for Pulse Width Modulation (PWM) generation, especially for AC-motor control. The CCU6 also supports special control modes for block commutation and multi-phase machines.

The timer T12 can function in capture and/or compare mode for its three channels. The timer T13 can work in compare mode only.

The multi-channel control unit generates output patterns, which can be modulated by T12 and/or T13. The modulation sources can be selected and combined for the signal modulation.

#### Timer T12 Features:

- Three capture/compare channels, each channel can be used either as a capture or as a compare channel
- Supports generation of a three-phase PWM (six outputs, individual signals for highside and lowside switches)
- 16-bit resolution, maximum count frequency = peripheral clock frequency
- Dead-time control for each channel to avoid short-circuits in the power stage
- Concurrent update of the required T12/13 registers
- Generation of center-aligned and edge-aligned PWM
- Supports single-shot mode
- Supports many interrupt request sources
- Hysteresis-like control mode

#### Timer T13 Features:

- One independent compare channel with one output
- 16-bit resolution, maximum count frequency = peripheral clock frequency
- Can be synchronized to T12
- Interrupt generation at period-match and compare-match
- Supports single-shot mode

#### Additional Features:

- Implements block commutation for Brushless DC-drives
- Position detection via Hall-sensor pattern
- Automatic rotational speed measurement for block commutation
- Integrated error handling
- Fast emergency stop without CPU load via external signal ( $\overline{\text{CTRAP}}$ )
- Control modes for multi-channel AC-drives
- Output levels can be selected and adapted to the power stage

Functional Description

The block diagram of the CCU6 module is shown in **Figure 34**.

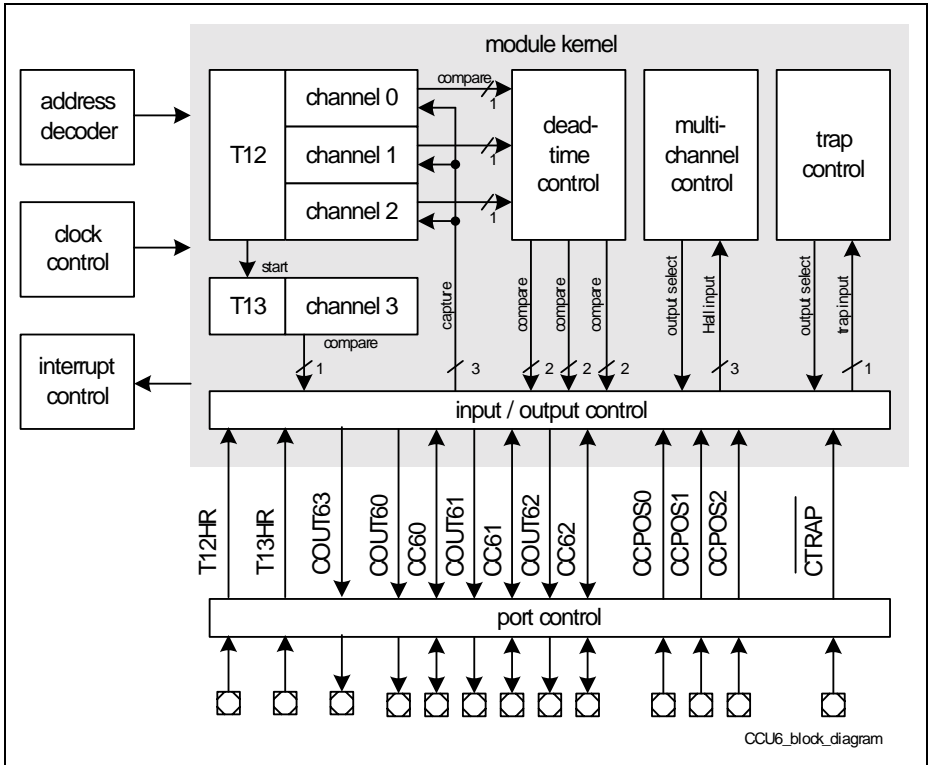


Figure 34 CCU6 Block Diagram

### 3.18 Analog-to-Digital Converter

The XC886/888 includes a high-performance 10-bit Analog-to-Digital Converter (ADC) with eight multiplexed analog input channels. The ADC uses a successive approximation technique to convert the analog voltage levels from up to eight different sources. The analog input channels of the ADC are available at Port 2.

#### Features:

- Successive approximation
- 8-bit or 10-bit resolution  
(TUE of  $\pm 1$  LSB and  $\pm 2$  LSB, respectively)
- Eight analog channels
- Four independent result registers
- Result data protection for slow CPU access  
(wait-for-read mode)
- Single conversion mode
- Autoscan functionality
- Limit checking for conversion results
- Data reduction filter  
(accumulation of up to 2 conversion results)
- Two independent conversion request sources with programmable priority
- Selectable conversion request trigger
- Flexible interrupt generation with configurable service nodes
- Programmable sample time
- Programmable clock divider
- Cancel/restart feature for running conversions
- Integrated sample and hold circuitry
- Compensation of offset errors
- Low power modes

### 3.18.1 ADC Clocking Scheme

A common module clock  $f_{ADC}$  generates the various clock signals used by the analog and digital parts of the ADC module:

- $f_{ADCA}$  is input clock for the analog part.
- $f_{ADCI}$  is internal clock for the analog part (defines the time base for conversion length and the sample time). This clock is generated internally in the analog part, based on the input clock  $f_{ADCA}$  to generate a correct duty cycle for the analog components.
- $f_{ADCD}$  is input clock for the digital part.

The internal clock for the analog part  $f_{ADCI}$  is limited to a maximum frequency of 10 MHz. Therefore, the ADC clock prescaler must be programmed to a value that ensures  $f_{ADCI}$  does not exceed 10 MHz. The prescaler ratio is selected by bit field CTC in register GLOBCTR. A prescaling ratio of 32 can be selected when the maximum performance of the ADC is not required.

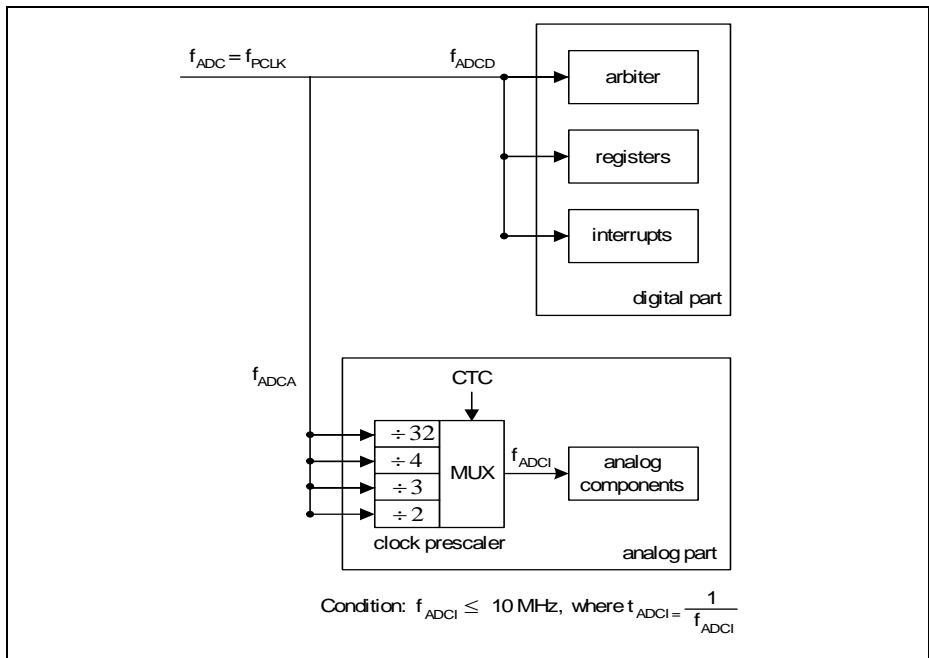


Figure 35 ADC Clocking Scheme

Functional Description

For module clock  $f_{ADC} = 24$  MHz, the analog clock  $f_{ADCI}$  frequency can be selected as shown in [Table 33](#).

**Table 33**  $f_{ADCI}$  Frequency Selection

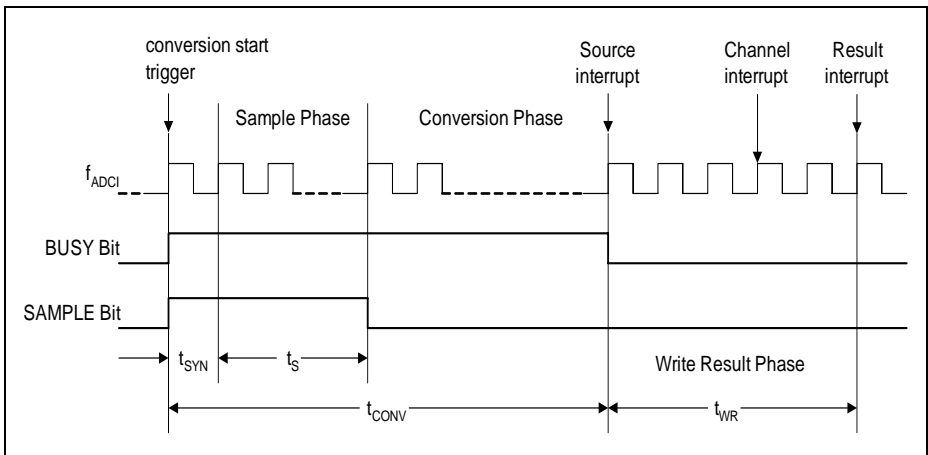
Module Clock $f_{ADC}$	CTC	Prescaling Ratio	Analog Clock $f_{ADCI}$
24 MHz	00 <sub>B</sub>	÷ 2	12 MHz (N.A)
	01 <sub>B</sub>	÷ 3	8 MHz
	10 <sub>B</sub>	÷ 4	6 MHz
	11 <sub>B</sub> (default)	÷ 32	750 kHz

As  $f_{ADCI}$  cannot exceed 10 MHz, bit field CTC should not be set to 00<sub>B</sub> when  $f_{ADC}$  is 24 MHz. During slow-down mode where  $f_{ADC}$  may be reduced to 12 MHz, 6 MHz etc., CTC can be set to 00<sub>B</sub> as long as the divided analog clock  $f_{ADCI}$  does not exceed 10 MHz. However, it is important to note that the conversion error could increase due to loss of charges on the capacitors, if  $f_{ADCI}$  becomes too low during slow-down mode.

### 3.18.2 ADC Conversion Sequence

The analog-to-digital conversion procedure consists of the following phases:

- Synchronization phase ( $t_{SYN}$ )
- Sample phase ( $t_S$ )
- Conversion phase
- Write result phase ( $t_{WR}$ )



**Figure 36** ADC Conversion Timing

### 3.19 On-Chip Debug Support

The On-Chip Debug Support (OCDS) provides the basic functionality required for the software development and debugging of XC800-based systems.

The OCDS design is based on these principles:

- use the built-in debug functionality of the XC800 Core
- add a minimum of hardware overhead
- provide support for most of the operations by a Monitor Program
- use standard interfaces to communicate with the Host (a Debugger)

#### Features:

- Set breakpoints on instruction address and on address range within the Program Memory
- Set breakpoints on internal RAM address range
- Support unlimited software breakpoints in Flash/RAM code region
- Process external breaks via JTAG and upon activating a dedicated pin
- Step through the program code

The OCDS functional blocks are shown in [Figure 37](#). The Monitor Mode Control (MMC) block at the center of OCDS system brings together control signals and supports the overall functionality. The MMC communicates with the XC800 Core, primarily via the Debug Interface, and also receives reset and clock signals.

After processing memory address and control signals from the core, the MMC provides proper access to the dedicated extra-memories: a Monitor ROM (holding the code) and a Monitor RAM (for work-data and Monitor-stack).

The OCDS system is accessed through the JTAG<sup>1)</sup>, which is an interface dedicated exclusively for testing and debugging activities and is not normally used in an application. The dedicated MBC pin is used for external configuration and debugging control.

*Note: All the debug functionality described here can normally be used only after XC886/888 has been started in OCDS mode.*

<sup>1)</sup> The pins of the JTAG port can be assigned to either the primary port (Port 0) or either of the secondary ports (Ports 1 and 2/Port 5).  
User must set the JTAG pins (TCK and TDI) as input during connection with the OCDS system.

Functional Description

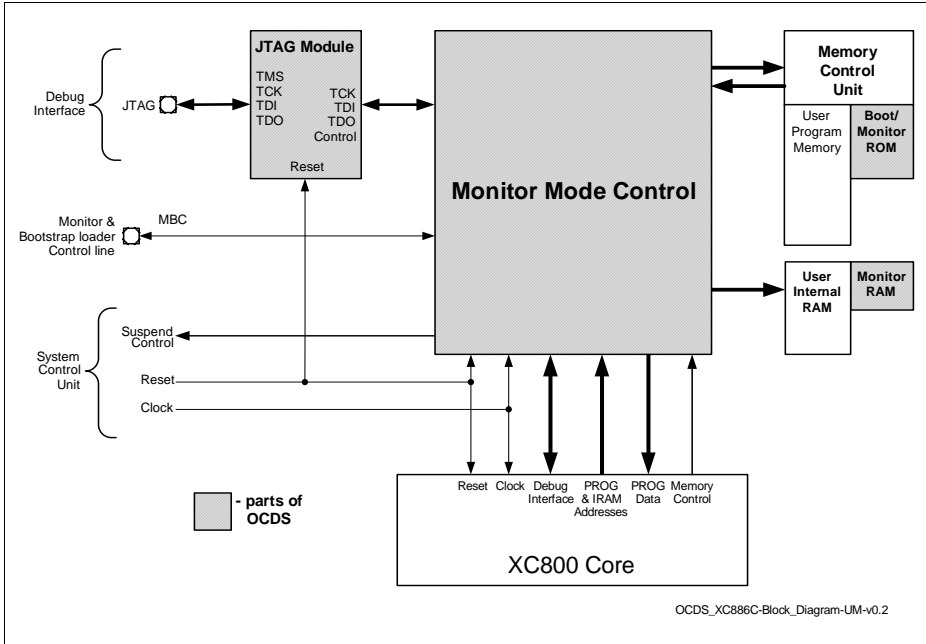


Figure 37 OCDS Block Diagram

### 3.19.1 JTAG ID Register

This is a read-only register located inside the JTAG module, and is used to recognize the device(s) connected to the JTAG interface. Its content is shifted out when INSTRUCTION register contains the IDCODE command (opcode 04<sub>H</sub>), and the same is also true immediately after reset.

The JTAG ID register contents for the XC886/888 Flash devices are given in [Table 34](#).

Table 34 JTAG ID Summary

Device Type	Device Name	JTAG ID
Flash	XC886/888*-8FF	1012 0083 <sub>H</sub>
	XC886/888*-6FF	1012 5083 <sub>H</sub>

Note: The asterisk (\*) above denotes all possible device configurations.

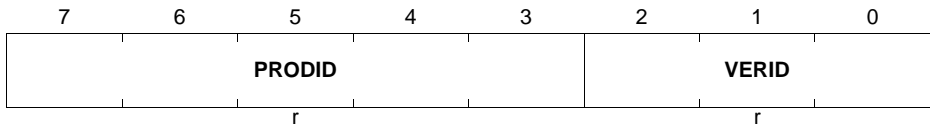
### 3.20 Identification Register

The XC886/888 identity register is located at Page 1 of address B3<sub>H</sub>.

#### ID

#### Identity Register

Reset Value: 0000 1001<sub>B</sub>



Field	Bits	Type	Description
VERID	[2:0]	r	Version ID 001 <sub>B</sub>
PRODID	[7:3]	r	Product ID 00001 <sub>B</sub>



## 4 Electrical Parameters

### 4.1 General Parameters

#### 4.1.1 Parameter Interpretation

The parameters listed in this section represent partly the characteristics of the XC886/888 and partly its requirements on the system. To aid interpreting the parameters easily when evaluating them for a design, they are indicated by the abbreviations in the "Symbol" column:

- **CC**

These parameters indicate **C**ontroller **C**haracteristics, which are distinctive features of the XC886/888 and must be regarded for a system design.

- **SR**

These parameters indicate **S**ystem **R**equirements, which must be provided by the microcontroller system in which the XC886/888 designed in.

### 4.1.2 Absolute Maximum Rating

Maximum ratings are the extreme limits to which the XC886/888 can be subjected to without permanent damage.

**Table 35 Absolute Maximum Rating Parameters**

Parameter	Symbol	Limit Values		Unit	Notes
		min.	max.		
Ambient temperature	$T_A$	-40	125	°C	under bias
Storage temperature	$T_{ST}$	-65	150	°C	
Junction temperature	$T_J$	-40	150	°C	under bias
Voltage on power supply pin with respect to $V_{SS}$	$V_{DDP}$	-0.5	6	V	
Input current on any pin during overload condition	$I_{IN}$	-10	10	mA	
Absolute sum of all input currents during overload condition	$\Sigma I_{IN} $	–	tbd	mA	

*Note: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. During absolute maximum rating overload conditions ( $V_{IN} > V_{DDP}$  or  $V_{IN} < V_{SS}$ ) the voltage on  $V_{DDP}$  pin with respect to ground ( $V_{SS}$ ) must not exceed the values defined by the absolute maximum ratings.*

### 4.1.3 Operating Conditions

The following operating conditions must not be exceeded in order to ensure correct operation of the XC886/888. All parameters mentioned in the following table refer to these operating conditions, unless otherwise noted.

**Table 36 Operating Condition Parameters**

Parameter	Symbol	Limit Values		Unit	Notes/ Conditions
		min.	max.		
Digital power supply voltage	$V_{DDP}$	4.5	5.5	V	5V range
		3.0	3.6	V	3.3V range
Digital ground voltage	$V_{SS}$	0		V	
Digital core supply voltage	$V_{DDC}$	2.3	2.7	V	
System Clock Frequency <sup>1)</sup>	$f_{SYS}$	88.8	103.2	MHz	
Ambient temperature	$T_A$	-40	85	°C	SAF-XC886/ 888...
		-40	125	°C	SAK-XC886/ 888...

<sup>1)</sup>  $f_{SYS}$  is the PLL output clock. During normal operating mode, CPU clock is  $f_{SYS} / 4$ . Please refer to [Figure 27](#) for detailed description.

## 4.2 DC Parameters

### 4.2.1 Input/Output Characteristics

**Table 37 Input/Output Characteristics (Operating Conditions apply)**

Parameter	Symbol	Limit Values		Unit	Test Conditions
		min.	max.		
<b><math>V_{DDP} = 5V</math> Range</b>					
Output low voltage	$V_{OL}$ CC	–	1.0	V	$I_{OL} = 15$ mA
		–	0.4	V	$I_{OL} = 5$ mA
Output high voltage	$V_{OH}$ CC	$V_{DDP} - 1.0$	–	V	$I_{OH} = -15$ mA
		$V_{DDP} - 0.4$	–	V	$I_{OH} = -5$ mA
Input low voltage on port pins (all except P0.0 & P0.1)	$V_{ILP}$ SR	–	$0.3 \times V_{DDP}$	V	CMOS Mode
Input low voltage on P0.0 & P0.1	$V_{ILP0}$ SR	-0.2	$0.3 \times V_{DDP}$	V	CMOS Mode
Input high voltage on port pins (all except P0.0 & P0.1)	$V_{IHP}$ SR	$0.7 \times V_{DDP}$	–	V	CMOS Mode
Input high voltage on P0.0 & P0.1	$V_{IHP0}$ SR	$0.7 \times V_{DDP}$	$V_{DDP}$	V	CMOS Mode
Input Hysteresis <sup>1)</sup>	$HYS$ CC	$0.08 \times V_{DDP}$	–	V	CMOS Mode
Pull-up current	$I_{PU}$ SR	–	-10	$\mu$ A	$V_{IH,min}$
		-150	–	$\mu$ A	$V_{IL,max}$
Pull-down current	$I_{PD}$ SR	–	10	$\mu$ A	$V_{IL,max}$
		150	–	$\mu$ A	$V_{IH,min}$
Input leakage current <sup>2)</sup>	$I_{OZ1}$ CC	-1	1	$\mu$ A	$0 < V_{IN} < V_{DDP}$ , $T_A \leq 125^\circ\text{C}$
Overload current on any pin	$I_{OV}$ SR	-5	5	mA	
Absolute sum of overload currents	$\sum  I_{OV} $ SR	–	tbd	mA	<sup>3)</sup>

**Electrical Parameters**
**Table 37 Input/Output Characteristics (Operating Conditions apply)**

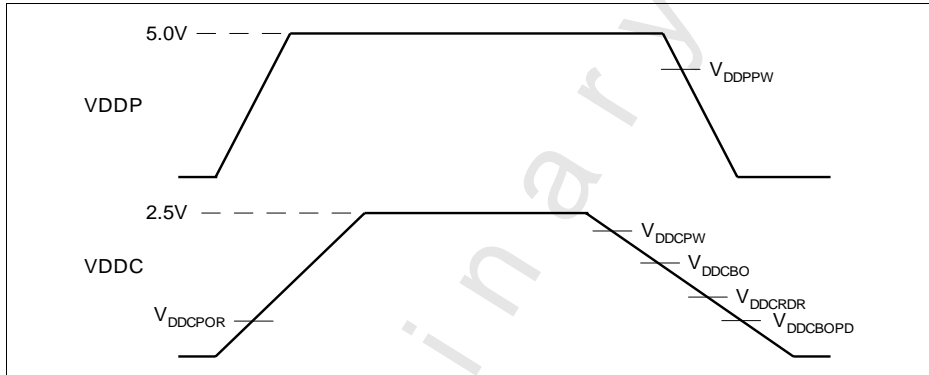
Parameter	Symbol	Limit Values		Unit	Test Conditions
		min.	max.		
<b><math>V_{DDP} = 3.3V</math> Range</b>					
Output low voltage	$V_{OL}$ CC	-	1.0	V	$I_{OL} = 8\text{ mA}$
		-	0.4	V	$I_{OL} = 2.5\text{ mA}$
Output high voltage	$V_{OH}$ CC	$V_{DDP} - 1.0$	-	V	$I_{OH} = -8\text{ mA}$
		$V_{DDP} - 0.4$	-	V	$I_{OH} = -2.5\text{ mA}$
Input low voltage on port pins (all except P0.0 & P0.1)	$V_{ILP}$ SR	-	$0.3 \times V_{DDP}$	V	CMOS Mode
Input low voltage on P0.0 & P0.1	$V_{ILP0}$ SR	-0.2	$0.3 \times V_{DDP}$	V	CMOS Mode
Input high voltage on port pins (all except P0.0 & P0.1)	$V_{IHP}$ SR	$0.7 \times V_{DDP}$	-	V	CMOS Mode
Input high voltage on P0.0 & P0.1	$V_{IHP0}$ SR	$0.7 \times V_{DDP}$	$V_{DDP}$	V	CMOS Mode
Input Hysteresis <sup>1)</sup>	$HYS$ CC	$0.03 \times V_{DDP}$	-	V	CMOS Mode
Pull-up current	$I_{PU}$ SR	-	-5	$\mu\text{A}$	$V_{IH,min}$
		-50	-	$\mu\text{A}$	$V_{IL,max}$
Pull-down current	$I_{PD}$ SR	-	5	$\mu\text{A}$	$V_{IL,max}$
		50	-	$\mu\text{A}$	$V_{IH,min}$
Input leakage current <sup>2)</sup>	$I_{OZ1}$ CC	-1	1	$\mu\text{A}$	$0 < V_{IN} < V_{DDP}$ , $T_A \leq 125^\circ\text{C}$
Overload current on any pin	$I_{OV}$ SR	-5	5	mA	
Absolute sum of overload currents	$\Sigma  I_{OV} $ SR	-	tbd	mA	<sup>3)</sup>

<sup>1)</sup> Not subjected to production test, verified by design/characterization. Hysteresis is implemented to avoid meta stable states and switching due to internal ground bounce. It cannot be guaranteed that it suppresses switching due to external system noise.

<sup>2)</sup> An additional error current ( $I_{Nj}$ ) will flow if an overload current flows through an adjacent pin. TMS pin and RESET pin have internal pull devices and are not included in the input leakage current characteristic.

<sup>3)</sup> Not subjected to production test, verified by design/characterization.

### 4.2.2 Supply Threshold Characteristics



**Figure 38 Supply Threshold Parameters**

**Table 38 Supply Threshold Parameters (Operating Conditions apply)**

Parameters	Symbol		Limit Values			Unit
			min.	typ.	max.	
V <sub>DDC</sub> prewarning voltage <sup>1)</sup>	V <sub>DDCPW</sub>	CC	2.2	2.3	2.4	V
V <sub>DDC</sub> brownout voltage in active mode <sup>1)</sup>	V <sub>DDCBO</sub>	CC	2.0	2.1	2.2	V
RAM data retention voltage	V <sub>DDCRDR</sub>	CC	0.9	1.0	1.1	V
V <sub>DDC</sub> brownout voltage in power-down mode <sup>2)</sup>	V <sub>DDCBOPD</sub>	CC	1.3	1.5	1.7	V
V <sub>DDP</sub> prewarning voltage <sup>3)</sup>	V <sub>DDPPW</sub>	CC	3.4	4.0	4.6	V
Power-on reset voltage <sup>2)4)</sup>	V <sub>DDCPOR</sub>	CC	1.3	1.5	1.7	V

<sup>1)</sup> Detection is disabled in power-down mode.

<sup>2)</sup> Detection is enabled in both active and power-down mode.

<sup>3)</sup> Detection is enabled for external power supply of 5.0V.  
Detection must be disabled for external power supply of 3.3V.

<sup>4)</sup> The reset of EVR is extended by 300 μs typically after the VDDC reaches the power-on reset voltage.

### 4.2.3 ADC Characteristics

The values in the table below are given for an analog power supply between 4.5 V to 5.5 V. The ADC can be used with an analog power supply down to 3 V. But in this case, the analog parameters may show a reduced performance. All ground pins ( $V_{SS}$ ) must be externally connected to one single star point in the system. The voltage difference between the ground pins must not exceed 200mV.

**Table 39 ADC Characteristics (Operating Conditions apply;  $V_{DDP} = 5V$  Range)**

Parameter	Symbol	Limit Values			Unit	Test Conditions/ Remarks
		min.	typ .	max.		
Analog reference voltage	$V_{AREF}$ SR	$V_{AGND} + 1$	$V_{DDP}$	$V_{DDP} + 0.05$	V	
Analog reference ground	$V_{AGND}$ SR	$V_{SS} - 0.05$	$V_{SS}$	$V_{AREF} - 1$	V	
Analog input voltage range	$V_{AIN}$ SR	$V_{AGND}$	–	$V_{AREF}$	V	
ADC clocks	$f_{ADC}$	–	24	25.8	MHz	module clock
	$f_{ADCI}$	–	–	10	MHz	internal analog clock See <a href="#">Figure 35</a>
Sample time	$t_S$ CC	$(2 + INPCR0.STC) \times t_{ADCI}$			$\mu s$	
Conversion time	$t_C$ CC	See <a href="#">Section 4.2.3.1</a>			$\mu s$	
Total unadjusted error	TUE <sup>1)</sup> CC	–	–	$\pm 1$	LSB	8-bit conversion. <sup>2)</sup>
		–	–	$\pm 2$	LSB	10-bit conversion.
Switched capacitance at the reference voltage input	$C_{AREFSW}$ CC	–	10	20	pF	<sup>2)3)</sup>
Switched capacitance at the analog voltage inputs	$C_{AINSW}$ CC	–	5	7	pF	<sup>2)4)</sup>
Input resistance of the reference input	$R_{AREF}$ CC	–	1	2	k $\Omega$	<sup>2)</sup>
Input resistance of the selected analog channel	$R_{AIN}$ CC	–	1	1.5	k $\Omega$	<sup>2)</sup>

Electrical Parameters

- 1) TUE is tested at  $V_{AREF} = 5.0\text{ V}$ ,  $V_{AGND} = 0\text{ V}$ ,  $V_{DDP} = 5.0\text{ V}$ .
- 2) Not subject to production test, verified by design/characterization
- 3) This represents an equivalent switched capacitance. This capacitance is not switched to the reference voltage at once. Instead of this, smaller capacitances are successively switched to the reference voltage.
- 4) The sampling capacity of the conversion C-Network is pre-charged to  $V_{AREF}/2$  before connecting the input to the C-Network. Because of the parasitic elements, the voltage measured at ANx is lower than  $V_{AREF}/2$ .

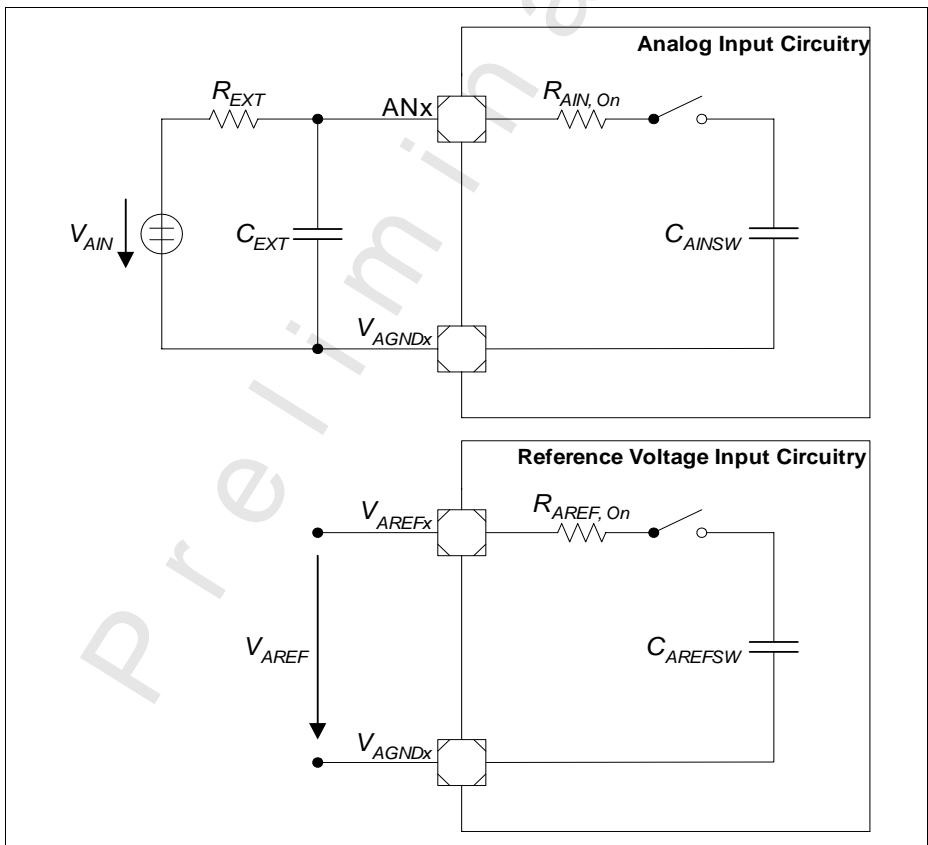


Figure 39 ADC Input Circuits



#### 4.2.3.1 ADC Conversion Timing

Conversion time,  $t_C = t_{ADC} \times (1 + r \times (3 + n + STC))$ , where

$r = CTC + 2$  for  $CTC = 00_B, 01_B$  or  $10_B$ ,

$r = 32$  for  $CTC = 11_B$ ,

$CTC =$  Conversion Time Control (GLOBCTR.CTC),

$STC =$  Sample Time Control (INPCR0.STC),

$n = 8$  or  $10$  (for 8-bit and 10-bit conversion respectively),

$t_{ADC} = 1 / f_{ADC}$

Preliminary

#### 4.2.4 Power Supply Current

**Table 40 Power Supply Current Parameters (Operating Conditions apply;  
 $V_{DDP} = 5V$  range )**

Parameter	Symbol	Limit Values		Unit	Test Condition
		typ. <sup>1)</sup>	max. <sup>2)</sup>		
<b><math>V_{DDP} = 5V</math> Range</b>					
Active Mode	$I_{DDP}$	29	tbd	mA	<sup>3)</sup>
Idle Mode	$I_{DDP}$	21.1	tbd	mA	<sup>4)</sup>
Active Mode with slow-down enabled	$I_{DDP}$	tbd	tbd	mA	<sup>5)</sup>
Idle Mode with slow-down enabled	$I_{DDP}$	tbd	tbd	mA	<sup>6)</sup>
Power-Down Mode	$I_{PDP}$	10	tbd	$\mu$ A	<sup>7)</sup>

<sup>1)</sup> The typical  $I_{DDP}$  values are based on preliminary measurements and are to be used as reference only. These values are periodically measured at  $T_A = +25^\circ\text{C}$  and  $V_{DDP} = 5.0\text{V}$ .

<sup>2)</sup> The maximum  $I_{DDP}$  values are measured under worst case conditions ( $T_A = +125^\circ\text{C}$  and  $V_{DDP} = 5.5\text{V}$ ).

<sup>3)</sup>  $I_{DDP}$  (active mode) is measured with: CPU clock and input clock to all peripherals running at 24 MHz (set by on-chip oscillator of 9.6 MHz and NDIV in PLL\_CON to 1001<sub>B</sub>),  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>4)</sup>  $I_{DDP}$  (idle mode) is measured with: CPU clock disabled, watchdog timer disabled, input clock to all peripherals enabled and running at 24 MHz,  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>5)</sup>  $I_{DDP}$  (active mode with slow-down mode) is measured with: CPU clock and input clock to all peripherals running at 8 MHz by setting CLKREL in CMCON to 0110<sub>B</sub>,  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>6)</sup>  $I_{DDP}$  (idle mode with slow-down mode) is measured with: CPU clock disabled, watchdog timer disabled, input clock to all peripherals enabled and running at 8 MHz by setting CLKREL in CMCON to 0110<sub>B</sub>,  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>7)</sup>  $I_{PDP}$  (power-down mode) is measured with:  $\overline{\text{RESET}} = V_{DDP}$ ,  $V_{AGND} = V_{SS}$ ,  $\text{RXD/INT0} = V_{DDP}$ ; rest of the ports are programmed to be input with either internal pull devices enabled or driven externally to ensure no floating inputs.

**Electrical Parameters**
**Table 41 Power Supply Current Parameters (Operating Conditions apply;  
 $V_{DDP} = 3.3V$  range)**

Parameter	Symbol	Limit Values		Unit	Test Condition
		typ. <sup>1)</sup>	max. <sup>2)</sup>		
<b><math>V_{DDP} = 3.3V</math> Range</b>					
Active Mode	$I_{DDP}$	tbd	tbd	mA	<sup>3)</sup>
Idle Mode	$I_{DDP}$	tbd	tbd	mA	<sup>4)</sup>
Active Mode with slow-down enabled	$I_{DDP}$	tbd	tbd	mA	<sup>5)</sup>
Idle Mode with slow-down enabled	$I_{DDP}$	tbd	tbd	mA	<sup>6)</sup>
Power-Down Mode	$I_{PDP}$	tbd	tbd	$\mu A$	<sup>7)</sup>

<sup>1)</sup> The typical  $I_{DDP}$  values are periodically measured at  $T_A = +25\text{ }^\circ\text{C}$  and  $V_{DDP} = 3.3\text{ V}$ .

<sup>2)</sup> The maximum  $I_{DDP}$  values are measured under worst case conditions ( $T_A = +125\text{ }^\circ\text{C}$  and  $V_{DDP} = 3.6\text{ V}$ ).

<sup>3)</sup>  $I_{DDP}$  (active mode) is measured with: CPU clock and input clock to all peripherals running at 24 MHz (set by on-chip oscillator of 9.6 MHz and NDIV in PLL\_CON to 1001<sub>B</sub>),  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>4)</sup>  $I_{DDP}$  (idle mode) is measured with: CPU clock disabled, watchdog timer disabled, input clock to all peripherals enabled and running at 24 MHz,  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>5)</sup>  $I_{DDP}$  (active mode with slow-down mode) is measured with: CPU clock and input clock to all peripherals running at 8 MHz by setting CLKREL in CMCON to 0110<sub>B</sub>,  $\overline{\text{RESET}} = V_{DDP}$ .

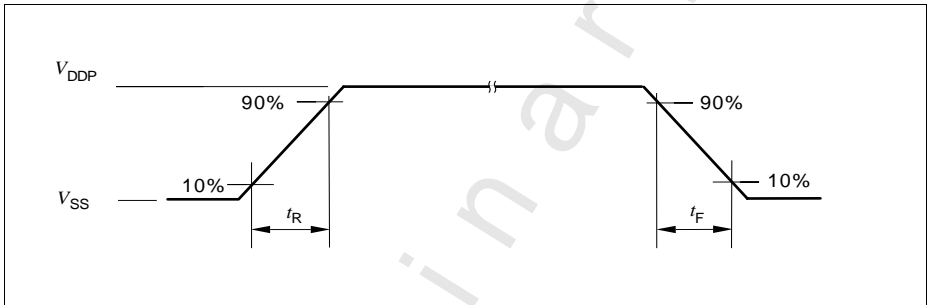
<sup>6)</sup>  $I_{DDP}$  (idle mode with slow-down mode) is measured with: CPU clock disabled, watchdog timer disabled, input clock to all peripherals enabled and running at 8 MHz by setting CLKREL in CMCON to 0110<sub>B</sub>,  $\overline{\text{RESET}} = V_{DDP}$ .

<sup>7)</sup>  $I_{PDP}$  (power-down mode) is measured with:  $\overline{\text{RESET}} = V_{DDP}$ ,  $V_{AGND} = V_{SS}$ ,  $\text{RXD}/\text{INT0} = V_{DDP}$ ; rest of the ports are programmed to be input with either internal pull devices enabled or driven externally to ensure no floating inputs

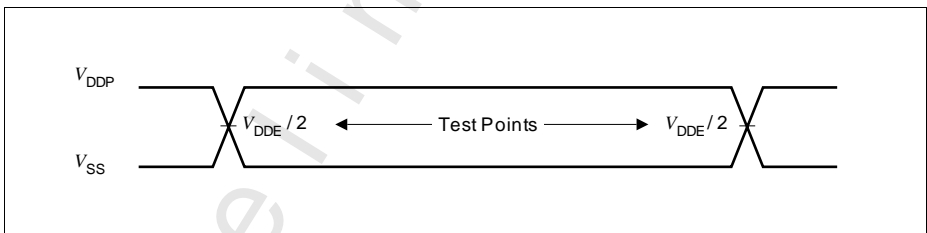
### 4.3 AC Parameters

#### 4.3.1 Testing Waveforms

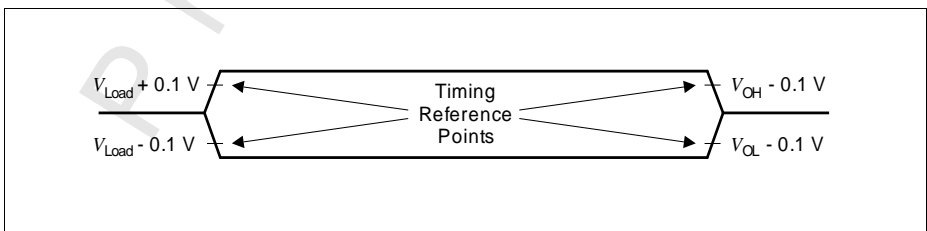
The testing waveforms for rise/fall time, output delay and output high impedance are shown in **Figure 40**, **Figure 41** and **Figure 42**.



**Figure 40** Rise/Fall Time Parameters



**Figure 41** Testing Waveform, Output Delay



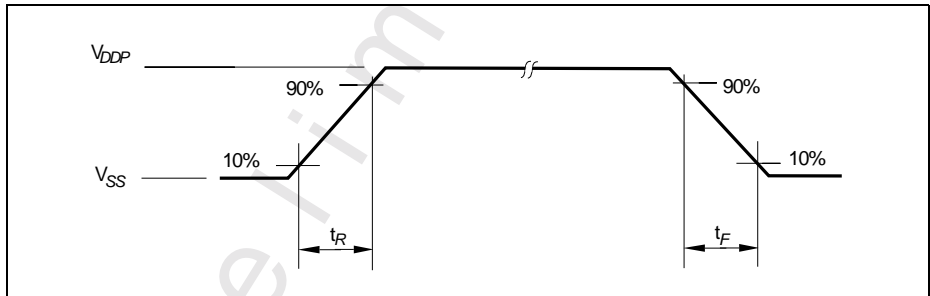
**Figure 42** Testing Waveform, Output High Impedance

### 4.3.2 Output Rise/Fall Times

**Table 42 Output Rise/Fall Times Parameters (Operating Conditions apply)**

Parameter	Symbol	Limit Values		Unit	Test Conditions
		min.	max.		
<b><math>V_{DDP} = 5V</math> Range</b>					
Rise/fall times <sup>1) 2)</sup>	$t_R, t_F$	–	10	ns	20 pF. <sup>3)</sup>
<b><math>V_{DDP} = 3.3V</math> Range</b>					
Rise/fall times <sup>1) 2)</sup>	$t_R, t_F$	–	10	ns	20 pF. <sup>4)</sup>

- 1) Rise/Fall time measurements are taken with 10% - 90% of the pad supply.
- 2) Not all parameters are 100% tested, but are verified by design/characterization and test correlation.
- 3) Additional rise/fall time valid for  $C_L = 20pF - 100pF @ 0.125 ns/pF$ .
- 4) Additional rise/fall time valid for  $C_L = 20pF - 100pF @ 0.225 ns/pF$ .



**Figure 43 Rise/Fall Times Parameters**

### 4.3.3 Power-on Reset and PLL Timing

**Table 43 Power-On Reset and PLL Timing (Operating Conditions apply)**

Parameter	Symbol	Limit Values			Unit	Test Conditions
		min.	typ.	max.		
Pad operating voltage	$V_{PAD}$ CC	2.3	–	–	V	
On-Chip Oscillator start-up time	$t_{OSCST}$ CC	–	–	500	ns	
Flash initialization time	$t_{FINIT}$ CC	–	160	–	$\mu$ s	
$\overline{RESET}$ hold time <sup>1)</sup>	$t_{RST}$ SR	–	500	–	$\mu$ s	$V_{DDP}$ rise time (10% – 90%) $\leq 500\mu$ s
PLL lock-in in time	$t_{LOCK}$ CC	–	–	200	$\mu$ s	
PLL accumulated jitter	$D_P$	–	–	tbd	ns	<sup>2)</sup>

<sup>1)</sup>  $\overline{RESET}$  signal has to be active (low) until  $V_{DDC}$  has reached 90% of its maximum value (typ. 2.5V).

<sup>2)</sup> PLL lock at 96 MHz using a 4 MHz external oscillator. The PLL Divider settings are K = 2, N = 48 and P = 1.

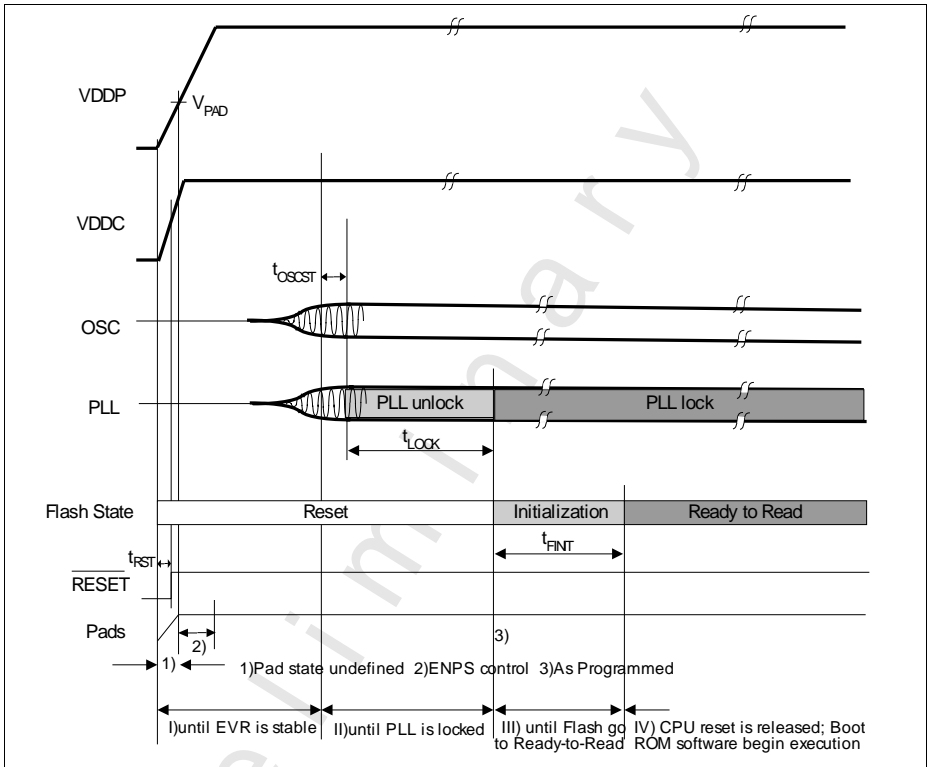


Figure 4-1 Power-on Reset Timing

### 4.3.4 On-Chip Oscillator Characteristics

**Table 44 On-chip Oscillator Characteristics (Operating Conditions apply)**

Parameter	Symbol	Limit Values			Unit	Test Conditions
		min.	typ.	max.		
Nominal frequency	$f_{NOM}$ CC	–	9.6	–	MHz	under nominal conditions <sup>1)</sup> after IFX-backend trimming
Chip-to-chip frequency deviation	$\Delta f_{CC}$ CC	-2.5	–	2.5	%	with respect to $f_{NOM}$
Long term frequency deviation	$\Delta f_{LT}$ CC	-5.0	–	5.0	%	with respect to $f_{NOM}$ , over lifetime and temperature, for one given device after trimming
Short term frequency deviation	$\Delta f_{ST}$ CC	-1.0	–	1.0	%	with respect to $f_{NOM}$ , within one LIN message (<10 ms .... 100 ms)

<sup>1)</sup> Nominal condition:  $V_{DDC} = 2.5$  V,  $T_A = +25^\circ\text{C}$ .



### 4.3.5 JTAG Timing

Table 45 TCK Clock Timing (Operating Conditions apply;  $C_L = 50$  pF)

Parameter	Symbol	Limits		Unit
		min	max	
TCK clock period	$t_{TCK}$ SR	50	–	ns
TCK high time	$t_1$ SR	tbd	–	ns
TCK low time	$t_2$ SR	tbd	–	ns
TCK clock rise time	$t_3$ SR	–	tbd	ns
TCK clock fall time	$t_4$ SR	–	tbd	ns

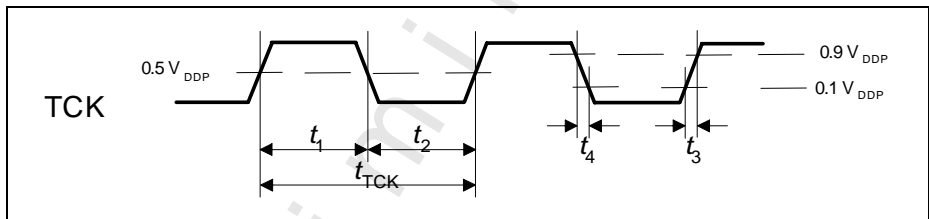
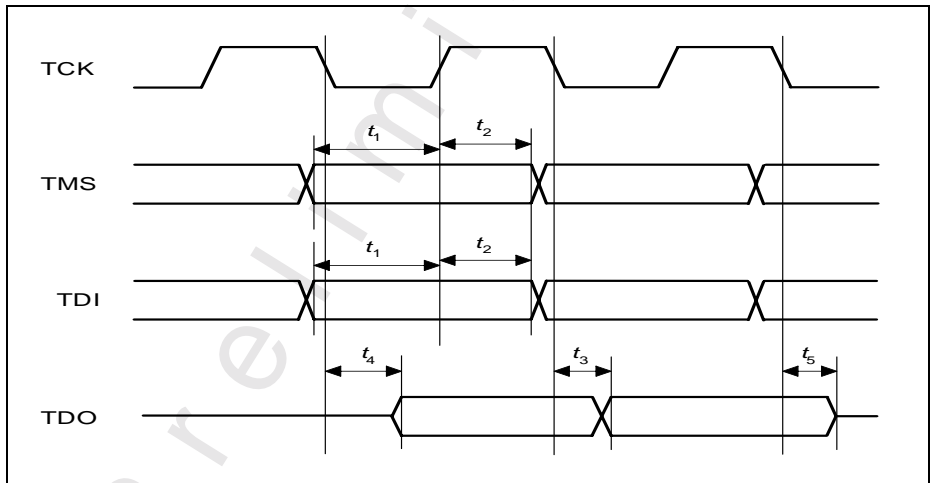


Figure 44 TCK Clock Timing

**Table 46 JTAG Timing (Operating Conditions apply;  $C_L = 50$  pF)**

Parameter	Symbol	Limits		Unit
		min	max	
TMS setup to TCK ↗	$t_1$ SR	tbd	–	ns
TMS hold to TCK ↘	$t_2$ SR	tbd	–	ns
TDI setup to TCK ↗	$t_1$ SR	tbd	–	ns
TDI hold to TCK ↘	$t_2$ SR	tbd	–	ns
TDO valid output from TCK ↘	$t_3$ CC	–	tbd	ns
TDO high impedance to valid output from TCK ↗	$t_4$ CC	–	tbd	ns
TDO valid output to high impedance from TCK ↗	$t_5$ CC	–	tbd	ns



**Figure 45 JTAG Timing**

### 4.3.6 SSC Master Mode Timing

Table 47 SSC Master Mode Timing (Operating Conditions apply;  $C_L = 50\text{ pF}$ )

Parameter	Symbol		Limit Values		Unit
			min.	max.	
SCLK clock period	$t_0$	CC	$2 \cdot T_{SSC}^{1)}$	–	ns
MSTR delay from SCLK	$t_1$	CC	0	tbd	ns
MRST setup to SCLK	$t_2$	SR	tbd	–	ns
MRST hold from SCLK	$t_3$	SR	tbd	–	ns

<sup>1)</sup>  $T_{SSCmin} = T_{CPU} = 1/f_{CPU}$ . When  $f_{CPU} = 24\text{MHz}$ ,  $t_0 = 83.3\text{ns}$ .  $T_{CPU}$  is the CPU clock period.

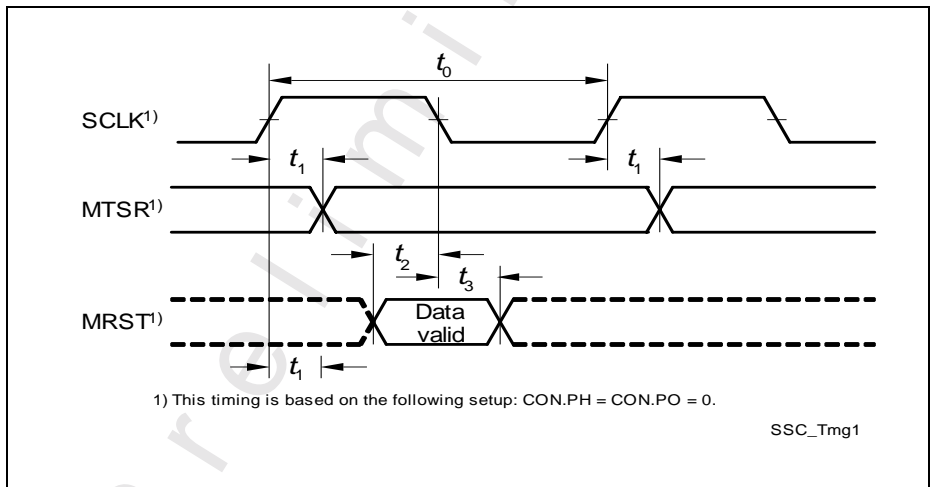


Figure 46 SSC Master Mode Timing

## 5 Package and Quality Declaration

### 5.1 Package Outline

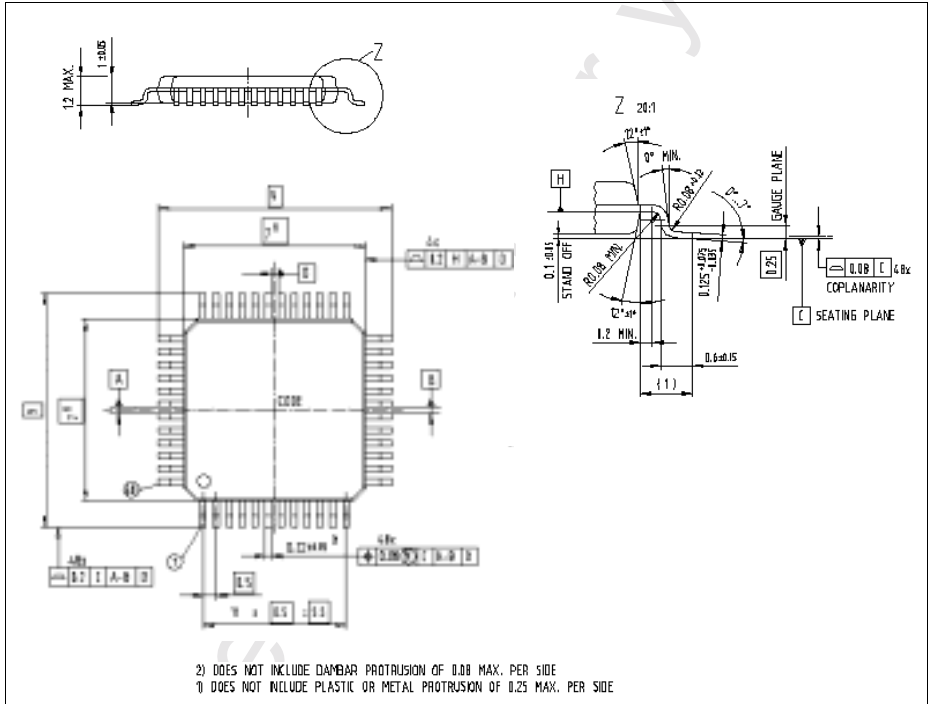


Figure 47 PG-TQFP-48-4 Package Outline

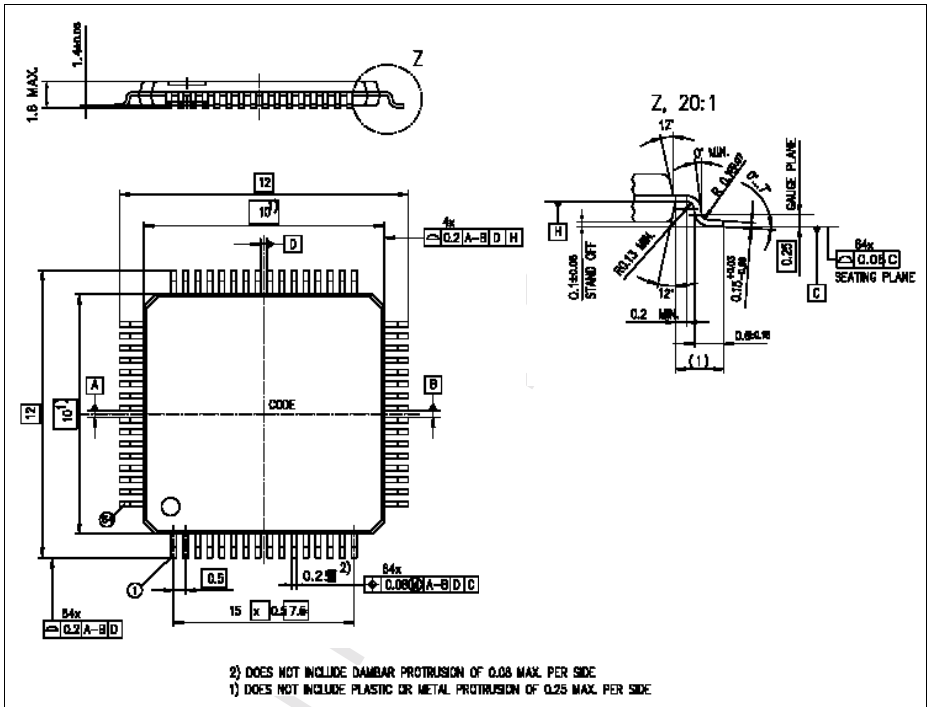


Figure 48 PG-TQFP-64-8 Package Outline

## 5.2 Quality Declaration

**Table 48** shows the characteristics of the quality parameters in the XC886/888.

**Table 48 Quality Parameters**

Parameter	Symbol	Limit Values		Unit	Notes
		Min.	Max.		
ESD susceptibility according to Human Body Model (HBM)	$V_{HBM}$	–	2000	V	Conforming to EIA/JESD22-A114-B
ESD susceptibility according to Charged Device Model (CDM) pins	$V_{CDM}$	–	500	V	Conforming to JESD22-C101-C

<http://www.infineon.com>