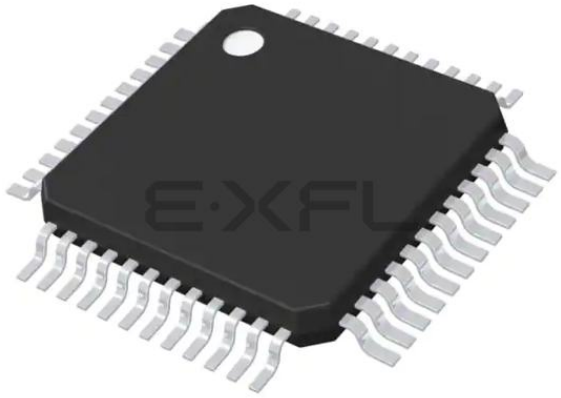
# Infineon Technologies - SAF-XC886-8FFA 5V AC Datasheet



**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | XC800 |
| Core Size | 8-Bit |
| Speed | 24MHz |
| Connectivity | SSI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 34 |
| Program Memory Size | 32KB (32K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1.75K x 8 |
| Voltage - Supply (Vcc/Vdd) | 4.5V ~ 5.5V |
| Data Converters | A/D 8x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 48-LQFP |
| Supplier Device Package | PG-TQFP-48 |
| Purchase URL | https://www.e-xfl.com/product-detail/infineon-technologies/saf-xc886-8ffa-5v-ac |

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
  - Up to 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 (T2 and T21)
- Multiplication/Division Unit for arithmetic operations (MDU)
- Software libraries to support floating point and MDU calculations
- CORDIC Coprocessor for computation of trigonometric, hyperbolic and linear functions
- MultiCAN with 2 nodes, 32 message objects
- 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)

## 2.2 Logic Symbol

The logic symbols of the XC886/888 are shown in **Figure 3**.

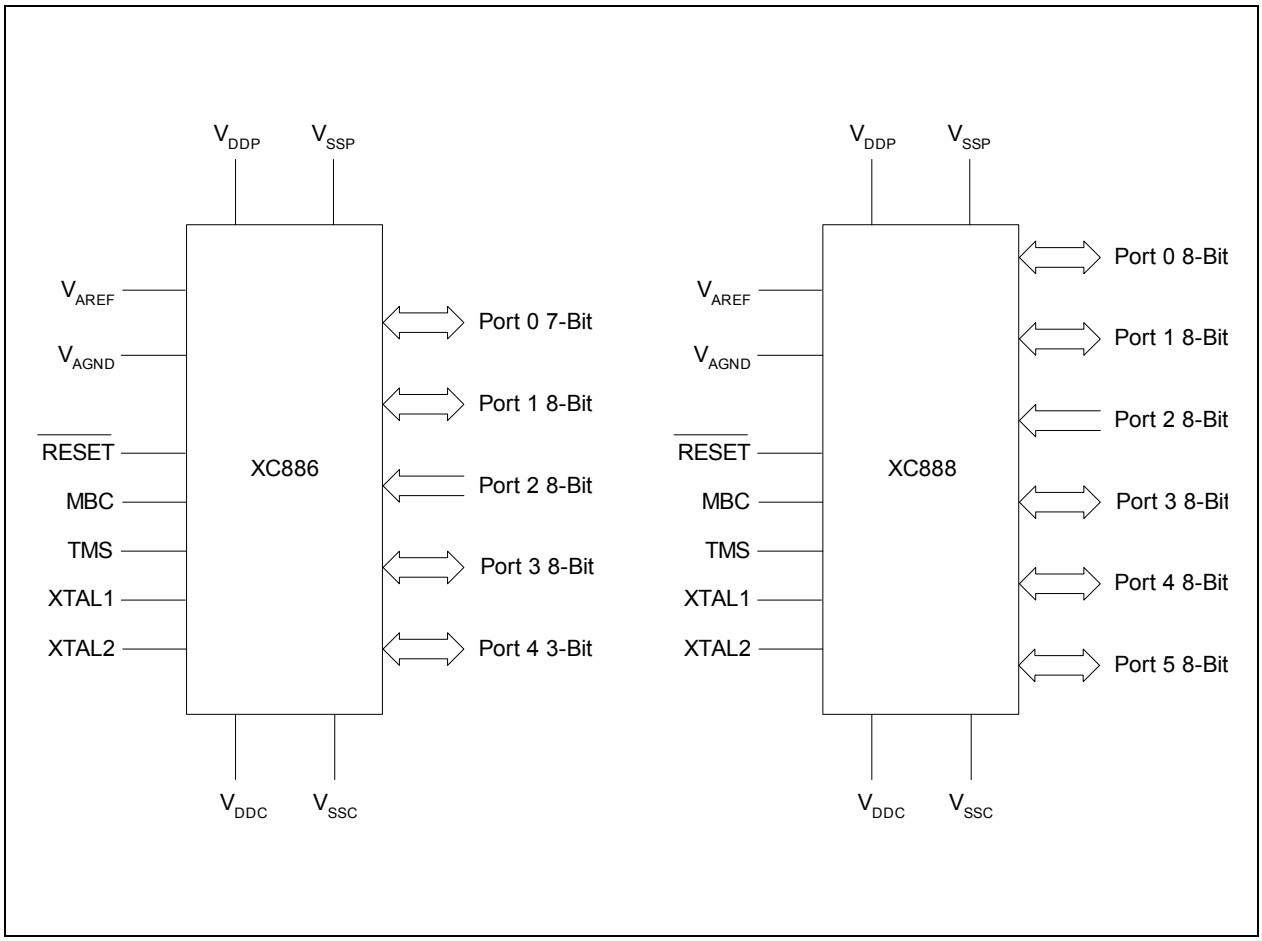



**Figure 3 XC886/888 Logic Symbol**

**SYSCON0**
**System Control Register 0** **Reset Value: $04_H$**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| | **0** | | **IMODE** | **0** | **1** | **0** | **RMAP** |
| | r | | rw | r | r | r | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RMAP** | 0 | rw | **Interrupt Node XINTR0 Enable**<br>0 The access to the standard SFR area is enabled<br>1 The access to the mapped SFR area is enabled |
| **1** | 2 | r | **Reserved**<br>Returns 1 if read; should be written with 1. |
| **0** | [7:5], 3,1 | r | **Reserved**<br>Returns 0 if read; should be written with 0. |

*Note: The RMAP bit should be cleared/set by ANL or ORL instructions.*

### 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**.

### 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 **Chapter 3.2.4.1** to **Chapter 3.2.4.14**.

*Note: The addresses of the bitaddressable SFRs appear in bold typeface.*

#### 3.2.4.1 CPU Registers

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 | | | | | | | | | | |
| $81_H$ | **SP** **Reset: $07_H$**<br>Stack Pointer Register | Bit Field | SP | | | | | | | |
| | | Type | rw | | | | | | | |
| $82_H$ | **DPL** **Reset: $00_H$**<br>Data Pointer Register Low | Bit Field | DPL7 | DPL6 | DPL5 | DPL4 | DPL3 | DPL2 | DPL1 | DPL0 |
| | | Type | rw | rw | rw | rw | rw | rw | rw | rw |
| $83_H$ | **DPH** **Reset: $00_H$**<br>Data Pointer Register High | Bit Field | DPH7 | DPH6 | DPH5 | DPH4 | DPH3 | DPH2 | DPH1 | DPH0 |
| | | Type | rw | rw | rw | rw | rw | rw | rw | rw |
| $87_H$ | **PCON** **Reset: $00_H$**<br>Power Control Register | Bit Field | SMOD | 0 | | | GF1 | GF0 | 0 | IDLE |
| | | Type | rw | r | | | rw | rw | r | rw |
| **$88_H$** | **TCON** **Reset: $00_H$**<br>Timer Control Register | Bit Field | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
| | | Type | rwh | rw | rwh | rw | rwh | rw | rwh | rw |
| $89_H$ | **TMOD** **Reset: $00_H$**<br>Timer Mode Register | Bit Field | GATE 1 | T1S | T1M | | GATE 0 | T0S | T0M | |
| | | Type | rw | rw | rw | | rw | rw | rw | |
| $8A_H$ | **TL0** **Reset: $00_H$**<br>Timer 0 Register Low | Bit Field | VAL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $8B_H$ | **TL1** **Reset: $00_H$**<br>Timer 1 Register Low | Bit Field | VAL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $8C_H$ | **TH0** **Reset: $00_H$**<br>Timer 0 Register High | Bit Field | VAL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $8D_H$ | **TH1** **Reset: $00_H$**<br>Timer 1 Register High | Bit Field | VAL | | | | | | | |
| | | Type | rwh | | | | | | | |
| **$98_H$** | **SCON** **Reset: $00_H$**<br>Serial Channel Control Register | Bit Field | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
| | | Type | rw | rw | rw | rw | rw | rwh | rwh | rwh |
| $99_H$ | **SBUF** **Reset: $00_H$**<br>Serial Data Buffer Register | Bit Field | VAL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $A2_H$ | **EO** **Reset: $00_H$**<br>Extended Operation Register | Bit Field | 0 | | | TRAP_EN | 0 | | | DPSEL0 |
| | | Type | r | | | rw | r | | | rw |

**Table 6 MDU Register Overview** (cont'd)

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $B3_H$ | **MR1** **Reset: $00_H$** MDU Result Register 1 | Bit Field | DATA | | | | | | | |
| | | Type | rh | | | | | | | |
| $B4_H$ | **MD2** **Reset: $00_H$** MDU Operand Register 2 | Bit Field | DATA | | | | | | | |
| | | Type | rw | | | | | | | |
| $B4_H$ | **MR2** **Reset: $00_H$** MDU Result Register 2 | Bit Field | DATA | | | | | | | |
| | | Type | rh | | | | | | | |
| $B5_H$ | **MD3** **Reset: $00_H$** MDU Operand Register 3 | Bit Field | DATA | | | | | | | |
| | | Type | rw | | | | | | | |
| $B5_H$ | **MR3** **Reset: $00_H$** MDU Result Register 3 | Bit Field | DATA | | | | | | | |
| | | Type | rh | | | | | | | |
| $B6_H$ | **MD4** **Reset: $00_H$** MDU Operand Register 4 | Bit Field | DATA | | | | | | | |
| | | Type | rw | | | | | | | |
| $B6_H$ | **MR4** **Reset: $00_H$** MDU Result Register 4 | Bit Field | DATA | | | | | | | |
| | | Type | rh | | | | | | | |
| $B7_H$ | **MD5** **Reset: $00_H$** MDU Operand Register 5 | Bit Field | DATA | | | | | | | |
| | | Type | rw | | | | | | | |
| $B7_H$ | **MR5** **Reset: $00_H$** MDU Result Register 5 | Bit Field | DATA | | | | | | | |
| | | Type | rh | | | | | | | |

### 3.2.4.3 CORDIC Registers

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 | | | | | | | | | | |
| $9A_H$ | **CD_CORDXL** **Reset: $00_H$** CORDIC X Data Low Byte | Bit Field | DATAL | | | | | | | |
| | | Type | rw | | | | | | | |
| $9B_H$ | **CD_CORDXH** **Reset: $00_H$** CORDIC X Data High Byte | Bit Field | DATAH | | | | | | | |
| | | Type | rw | | | | | | | |
| $9C_H$ | **CD_CORDYL** **Reset: $00_H$** CORDIC Y Data Low Byte | Bit Field | DATAL | | | | | | | |
| | | Type | rw | | | | | | | |
| $9D_H$ | **CD_CORDYH** **Reset: $00_H$** CORDIC Y Data High Byte | Bit Field | DATAH | | | | | | | |
| | | Type | rw | | | | | | | |
| $9E_H$ | **CD_CORDZL** **Reset: $00_H$** CORDIC Z Data Low Byte | Bit Field | DATAL | | | | | | | |
| | | Type | rw | | | | | | | |
| $9F_H$ | **CD_CORDZH** **Reset: $00_H$** CORDIC Z Data High Byte | Bit Field | DATAH | | | | | | | |
| | | Type | rw | | | | | | | |

**Table 8 SCU Register Overview** (cont'd)

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $BC_H$ | **NMISR** **Reset: $00_H$** NMI Status Register | 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 | rwh |
| $BD_H$ | **BCON** **Reset: $00_H$** Baud Rate Control Register | Bit Field | BGSEL | | 0 | BRDIS | BRPRE | | | R |
| | | Type | rw | | r | rw | rw | | | rw |
| $BE_H$ | **BG** **Reset: $00_H$** Baud Rate Timer/Reload Register | Bit Field | BR_VALUE | | | | | | | |
| | | Type | rwh | | | | | | | |
| $E9_H$ | **FDCON** **Reset: $00_H$** Fractional Divider Control Register | Bit Field | BGS | SYNE N | ERRS YN | EOFS YN | BRK | NDOV | FDM | FDEN |
| | | Type | rw | rw | rwh | rwh | rwh | rwh | rw | rw |
| $EA_H$ | **FDSTEP** **Reset: $00_H$** Fractional Divider Reload Register | Bit Field | STEP | | | | | | | |
| | | Type | rw | | | | | | | |
| $EB_H$ | **FDRES** **Reset: $00_H$** Fractional Divider Result Register | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| RMAP = 0, PAGE 1 | | | | | | | | | | |
| $B3_H$ | **ID** **Reset: $UU_H$** Identity Register | Bit Field | PRODID | | | | | VERID | | |
| | | Type | r | | | | | r | | |
| $B4_H$ | **PMCON0** **Reset: $00_H$** Power Mode Control Register 0 | Bit Field | 0 | WDT RST | WKRS | WK SEL | SD | PD | WS | |
| | | Type | r | rwh | rwh | rw | rw | rwh | rw | |
| $B5_H$ | **PMCON1** **Reset: $00_H$** Power Mode Control Register 1 | Bit Field | 0 | CDC_ DIS | CAN_ DIS | MDU_ DIS | T2_ DIS | CCU_ DIS | SSC_ DIS | ADC_ DIS |
| | | Type | r | rw | rw | rw | rw | rw | rw | rw |
| $B6_H$ | **OSC_CON** **Reset: $08_H$** OSC Control Register | Bit Field | 0 | | | OSC PD | XPD | OSC SS | ORD RES | OSCR |
| | | Type | r | | | rw | rw | rw | rwh | rh |
| $B7_H$ | **PLL_CON** **Reset: $90_H$** PLL Control Register | Bit Field | NDIV | | | | VCO BYP | OSC DISC | RESL D | LOCK |
| | | Type | rw | | | | rw | rw | rwh | rh |
| $BA_H$ | **CMCON** **Reset: $10_H$** Clock Control Register | Bit Field | VCO SEL | KDIV | 0 | FCCF G | CLKREL | | | |
| | | Type | rw | rw | r | rw | rw | | | |
| $BB_H$ | **PASSWD** **Reset: $07_H$** Password Register | Bit Field | PASS | | | | | PROT ECT_S | MODE | |
| | | Type | wh | | | | | rh | rw | |
| $BC_H$ | **FEAL** **Reset: $00_H$** Flash Error Address Register Low | Bit Field | ECCERRADDR | | | | | | | |
| | | Type | rh | | | | | | | |
| $BD_H$ | **FEAH** **Reset: $00_H$** Flash Error Address Register High | Bit Field | ECCERRADDR | | | | | | | |
| | | Type | rh | | | | | | | |

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

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $D3_H$ | **ADC_RESR3H Reset: $00_H$** Result Register 3 High | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| RMAP = 0, PAGE 3 | | | | | | | | | | |
| $CA_H$ | **ADC_RESRA0L Reset: $00_H$** Result Register 0, View A Low | Bit Field | RESULT | | | VF | DRC | CHNR | | |
| | | Type | rh | | | rh | rh | rh | | |
| $CB_H$ | **ADC_RESRA0H Reset: $00_H$** Result Register 0, View A High | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| $CC_H$ | **ADC_RESRA1L Reset: $00_H$** Result Register 1, View A Low | Bit Field | RESULT | | | VF | DRC | CHNR | | |
| | | Type | rh | | | rh | rh | rh | | |
| $CD_H$ | **ADC_RESRA1H Reset: $00_H$** Result Register 1, View A High | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| $CE_H$ | **ADC_RESRA2L Reset: $00_H$** Result Register 2, View A Low | Bit Field | RESULT | | | VF | DRC | CHNR | | |
| | | Type | rh | | | rh | rh | rh | | |
| $CF_H$ | **ADC_RESRA2H Reset: $00_H$** Result Register 2, View A High | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| $D2_H$ | **ADC_RESRA3L Reset: $00_H$** Result Register 3, View A Low | Bit Field | RESULT | | | VF | DRC | CHNR | | |
| | | Type | rh | | | rh | rh | rh | | |
| $D3_H$ | **ADC_RESRA3H Reset: $00_H$** Result Register 3, View A High | Bit Field | RESULT | | | | | | | |
| | | Type | rh | | | | | | | |
| RMAP = 0, PAGE 4 | | | | | | | | | | |
| $CA_H$ | **ADC_RCR0 Reset: $00_H$** Result Control Register 0 | Bit Field | VFCTR | WFR | 0 | IEN | 0 | | | DRCTR |
| | | Type | rw | rw | r | rw | r | | | rw |
| $CB_H$ | **ADC_RCR1 Reset: $00_H$** Result Control Register 1 | Bit Field | VFCTR | WFR | 0 | IEN | 0 | | | DRCTR |
| | | Type | rw | rw | r | rw | r | | | rw |
| $CC_H$ | **ADC_RCR2 Reset: $00_H$** Result Control Register 2 | Bit Field | VFCTR | WFR | 0 | IEN | 0 | | | DRCTR |
| | | Type | rw | rw | r | rw | r | | | rw |
| $CD_H$ | **ADC_RCR3 Reset: $00_H$** Result Control Register 3 | Bit Field | VFCTR | WFR | 0 | IEN | 0 | | | DRCTR |
| | | Type | rw | rw | r | rw | r | | | rw |
| $CE_H$ | **ADC_VFCR Reset: $00_H$** Valid Flag Clear Register | Bit Field | 0 | | | | VFC3 | VFC2 | VFC1 | VFC0 |
| | | Type | r | | | | w | w | w | w |
| RMAP = 0, PAGE 5 | | | | | | | | | | |
| $CA_H$ | **ADC_CHINFR Reset: $00_H$** Channel Interrupt Flag Register | Bit Field | CHINF7 | CHINF6 | CHINF5 | CHINF4 | CHINF3 | CHINF2 | CHINF1 | CHINF0 |
| | | Type | rh | rh | rh | rh | rh | rh | rh | rh |
| $CB_H$ | **ADC_CHINCR Reset: $00_H$** Channel Interrupt Clear Register | Bit Field | CHINC7 | CHINC6 | CHINC5 | CHINC4 | CHINC3 | CHINC2 | CHINC1 | CHINC0 |
| | | Type | w | w | w | w | w | w | w | w |

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

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $CC_H$ | **ADC_CHINSR Reset: $00_H$** Channel Interrupt Set Register | 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 |
| $CD_H$ | **ADC_CHINPR Reset: $00_H$** Channel Interrupt Node Pointer Register | 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 |
| $CE_H$ | **ADC_EVINFR Reset: $00_H$** Event Interrupt Flag Register | Bit Field | EVINF 7 | EVINF 6 | EVINF 5 | EVINF 4 | 0 | | EVINF 1 | EVINF 0 |
| | | Type | rh | rh | rh | rh | r | | rh | rh |
| $CF_H$ | **ADC_EVINCR Reset: $00_H$** Event Interrupt Clear Flag Register | Bit Field | EVINC 7 | EVINC 6 | EVINC 5 | EVINC 4 | 0 | | EVINC 1 | EVINC 0 |
| | | Type | w | w | w | w | r | | w | w |
| $D2_H$ | **ADC_EVINSR Reset: $00_H$** Event Interrupt Set Flag Register | Bit Field | EVINS 7 | EVINS 6 | EVINS 5 | EVINS 4 | 0 | | EVINS 1 | EVINS 0 |
| | | Type | w | w | w | w | r | | w | w |
| $D3_H$ | **ADC_EVINPR Reset: $00_H$** Event Interrupt Node Pointer Register | 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 | | | | | | | | | | |
| $CA_H$ | **ADC_CRCR1 Reset: $00_H$** Conversion Request Control Register 1 | Bit Field | CH7 | CH6 | CH5 | CH4 | 0 | | | |
| | | Type | rwh | rwh | rwh | rwh | r | | | |
| $CB_H$ | **ADC_CRPR1 Reset: $00_H$** Conversion Request Pending Register 1 | Bit Field | CHP7 | CHP6 | CHP5 | CHP4 | 0 | | | |
| | | Type | rwh | rwh | rwh | rwh | r | | | |
| $CC_H$ | **ADC_CRMR1 Reset: $00_H$** Conversion Request Mode Register 1 | Bit Field | Rsv | LDEV | CLRP ND | SCAN | ENSI | ENTR | 0 | ENGT |
| | | Type | r | w | w | rw | rw | rw | r | rw |
| $CD_H$ | **ADC_QMR0 Reset: $00_H$** Queue Mode Register 0 | Bit Field | CEV | TREV | FLUS H | CLRV | 0 | ENTR | 0 | ENGT |
| | | Type | w | w | w | w | r | rw | r | rw |
| $CE_H$ | **ADC_QSR0 Reset: $20_H$** Queue Status Register 0 | Bit Field | Rsv | 0 | EMPT Y | EV | 0 | | FILL | |
| | | Type | r | r | rh | rh | r | | rh | |
| $CF_H$ | **ADC_Q0R0 Reset: $00_H$** Queue 0 Register 0 | Bit Field | EXTR | ENSI | RF | V | 0 | REQCHNR | | |
| | | Type | rh | rh | rh | rh | r | rh | | |
| $D2_H$ | **ADC_QBUR0 Reset: $00_H$** Queue Backup Register 0 | Bit Field | EXTR | ENSI | RF | V | 0 | REQCHNR | | |
| | | Type | rh | rh | rh | rh | r | rh | | |
| $D2_H$ | **ADC_QINR0 Reset: $00_H$** Queue Input Register 0 | Bit Field | EXTR | ENSI | RF | 0 | | REQCHNR | | |
| | | Type | w | w | w | r | | w | | |

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

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $FA_H$ | **CCU6_CC60SRL Reset: $00_H$** Capture/Compare Shadow Register for Channel CC60 Low | Bit Field | CC60SL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $FB_H$ | **CCU6_CC60SRH Reset: $00_H$** Capture/Compare Shadow Register for Channel CC60 High | Bit Field | CC60SH | | | | | | | |
| | | Type | rwh | | | | | | | |
| $FC_H$ | **CCU6_CC61SRL Reset: $00_H$** Capture/Compare Shadow Register for Channel CC61 Low | Bit Field | CC61SL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $FD_H$ | **CCU6_CC61SRH Reset: $00_H$** Capture/Compare Shadow Register for Channel CC61 High | Bit Field | CC61SH | | | | | | | |
| | | Type | rwh | | | | | | | |
| $FE_H$ | **CCU6_CC62SRL Reset: $00_H$** Capture/Compare Shadow Register for Channel CC62 Low | Bit Field | CC62SL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $FF_H$ | **CCU6_CC62SRH Reset: $00_H$** Capture/Compare Shadow Register for Channel CC62 High | Bit Field | CC62SH | | | | | | | |
| | | Type | rwh | | | | | | | |
| RMAP = 0, PAGE 1 | | | | | | | | | | |
| $9A_H$ | **CCU6_CC63RL Reset: $00_H$** Capture/Compare Register for Channel CC63 Low | Bit Field | CC63VL | | | | | | | |
| | | Type | rh | | | | | | | |
| $9B_H$ | **CCU6_CC63RH Reset: $00_H$** Capture/Compare Register for Channel CC63 High | Bit Field | CC63VH | | | | | | | |
| | | Type | rh | | | | | | | |
| $9C_H$ | **CCU6_T12PRL Reset: $00_H$** Timer T12 Period Register Low | Bit Field | T12PVL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $9D_H$ | **CCU6_T12PRH Reset: $00_H$** Timer T12 Period Register High | Bit Field | T12PVH | | | | | | | |
| | | Type | rwh | | | | | | | |
| $9E_H$ | **CCU6_T13PRL Reset: $00_H$** Timer T13 Period Register Low | Bit Field | T13PVL | | | | | | | |
| | | Type | rwh | | | | | | | |
| $9F_H$ | **CCU6_T13PRH Reset: $00_H$** Timer T13 Period Register High | Bit Field | T13PVH | | | | | | | |
| | | Type | rwh | | | | | | | |
| $A4_H$ | **CCU6_T12DTCL Reset: $00_H$** Dead-Time Control Register for Timer T12 Low | Bit Field | DTM | | | | | | | |
| | | Type | rw | | | | | | | |
| $A5_H$ | **CCU6_T12DTCH Reset: $00_H$** Dead-Time Control Register for Timer T12 High | Bit Field | 0 | DTR2 | DTR1 | DTR0 | 0 | DTE2 | DTE1 | DTE0 |
| | | Type | r | rh | rh | rh | r | rw | rw | rw |
| $A6_H$ | **CCU6_TCTR0L Reset: $00_H$** Timer Control Register 0 Low | Bit Field | CTM | CDIR | STE12 | T12R | T12PRE | T12CLK | | |
| | | Type | rw | rh | rh | rh | rw | rw | | |
| $A7_H$ | **CCU6_TCTR0H Reset: $00_H$** Timer Control Register 0 High | Bit Field | 0 | | STE13 | T13R | T13PRE | T13CLK | | |
| | | Type | r | | rh | rh | rw | rw | | |
| $FA_H$ | **CCU6_CC60RL Reset: $00_H$** Capture/Compare Register for Channel CC60 Low | Bit Field | CC60VL | | | | | | | |
| | | Type | rh | | | | | | | |

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

| Addr | Register Name | Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| $FB_H$ | **CCU6_CC60RH** **Reset: $00_H$** Capture/Compare Register for Channel CC60 High | Bit Field | CC60VH | | | | | | | |
| | | Type | rh | | | | | | | |
| $FC_H$ | **CCU6_CC61RL** **Reset: $00_H$** Capture/Compare Register for Channel CC61 Low | Bit Field | CC61VL | | | | | | | |
| | | Type | rh | | | | | | | |
| $FD_H$ | **CCU6_CC61RH** **Reset: $00_H$** Capture/Compare Register for Channel CC61 High | Bit Field | CC61VH | | | | | | | |
| | | Type | rh | | | | | | | |
| $FE_H$ | **CCU6_CC62RL** **Reset: $00_H$** Capture/Compare Register for Channel CC62 Low | Bit Field | CC62VL | | | | | | | |
| | | Type | rh | | | | | | | |
| $FF_H$ | **CCU6_CC62RH** **Reset: $00_H$** Capture/Compare Register for Channel CC62 High | Bit Field | CC62VH | | | | | | | |
| | | Type | rh | | | | | | | |
| RMAP = 0, PAGE 2 | | | | | | | | | | |
| $9A_H$ | **CCU6_T12MSELL** **Reset: $00_H$** T12 Capture/Compare Mode Select Register Low | Bit Field | MSEL61 | | | | MSEL60 | | | |
| | | Type | rw | | | | rw | | | |
| $9B_H$ | **CCU6_T12MSELH** **Reset: $00_H$** T12 Capture/Compare Mode Select Register High | Bit Field | DBYP | HSYNC | | | MSEL62 | | | |
| | | Type | rw | rw | | | rw | | | |
| $9C_H$ | **CCU6_IENL** **Reset: $00_H$** Capture/Compare Interrupt Enable Register Low | Bit Field | ENT12PM | ENT12OM | ENCC62F | ENCC62R | ENCC61F | ENCC61R | ENCC60F | ENCC60R |
| | | Type | rw | rw | rw | rw | rw | rw | rw | rw |
| $9D_H$ | **CCU6_IENH** **Reset: $00_H$** Capture/Compare Interrupt Enable Register High | Bit Field | ENSTR | ENIDLE | ENWHE | ENCHE | 0 | ENTRPF | ENT13PM | ENT13CM |
| | | Type | rw | rw | rw | rw | r | rw | rw | rw |
| $9E_H$ | **CCU6_INPL** **Reset: $40_H$** Capture/Compare Interrupt Node Pointer Register Low | Bit Field | INPCHE | | INPCC62 | | INPCC61 | | INPCC60 | |
| | | Type | rw | | rw | | rw | | rw | |
| $9F_H$ | **CCU6_INPH** **Reset: $39_H$** Capture/Compare Interrupt Node Pointer Register High | Bit Field | 0 | | INPT13 | | INPT12 | | INPERR | |
| | | Type | r | | rw | | rw | | rw | |
| $A4_H$ | **CCU6_ISSL** **Reset: $00_H$** Capture/Compare Interrupt Status Set Register Low | Bit Field | ST12PM | ST12OM | SCC62F | SCC62R | SCC61F | SCC61R | SCC60F | SCC60R |
| | | Type | w | w | w | w | w | w | w | w |
| $A5_H$ | **CCU6_ISSH** **Reset: $00_H$** Capture/Compare Interrupt Status Set Register High | Bit Field | SSTR | SIDLE | SWHE | SCHE | SWHC | STRPF | ST13PM | ST13CM |
| | | Type | w | w | w | w | w | w | w | w |
| $A6_H$ | **CCU6_PSLR** **Reset: $00_H$** Passive State Level Register | Bit Field | PSL63 | 0 | PSL | | | | | |
| | | Type | rwh | r | rwh | | | | | |
| $A7_H$ | **CCU6_MCMCTR** **Reset: $00_H$** Multi-Channel Mode Control Register | Bit Field | 0 | | SWSYN | | 0 | SWSEL | | |
| | | Type | r | | rw | | r | rw | | |
| $FA_H$ | **CCU6_TCTR2L** **Reset: $00_H$** Timer Control Register 2 Low | Bit Field | 0 | T13TED | T13TEC | | | | T13SSC | T12SSC |
| | | Type | r | rw | rw | | | | rw | rw |

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

| Interrupt Source | Vector Address | Assignment for XC886/888 | Enable Bit | SFR |
|---|---|---|---|---|
| XINTR6 | $0033_H$ | MultiCAN Nodes 1 and 2 | EADC | IEN1 |
| | | ADC[1:0] | | |
| XINTR7 | $003B_H$ | SSC | ESSC | |
| XINTR8 | $0043_H$ | External Interrupt 2 | EX2 | |
| | | T21 | | |
| | | CORDIC | | |
| | | UART1 | | |
| | | UART1 Fractional Divider (Normal Divider Overflow) | | |
| | | MDU[1:0] | | |
| XINTR9 | $004B_H$ | External Interrupt 3 | EXM | |
| | | External Interrupt 4 | | |
| | | External Interrupt 5 | | |
| | | External Interrupt 6 | | |
| | | MultiCAN Node 3 | | |
| XINTR10 | $0053_H$ | CCU6 INP0 | ECCIP0 | |
| | | MultiCAN Node 4 | | |
| XINTR11 | $005B_H$ | CCU6 INP1 | ECCIP1 | |
| | | MultiCAN Node 5 | | |
| XINTR12 | $0063_H$ | CCU6 INP2 | ECCIP2 | |
| | | MultiCAN Node 6 | | |
| XINTR13 | $006B_H$ | CCU6 INP3 | ECCIP3 | |
| | | MultiCAN Node 7 | | |

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

**Table 25 VCO Range**

| $f_{VCOmin}$ | $f_{VCOmax}$ | $f_{VCOFREEmin}$ | $f_{VCOFREEmax}$ | Unit |
|---|---|---|---|---|
| 150 | 200 | 20 | 80 | MHz |
| 100 | 150 | 10 | 80 | MHz |

### 3.8.1 Recommended External Oscillator Circuits

The oscillator circuit, a Pierce oscillator, is designed to work with both, an external crystal oscillator or an external stable clock source. It basically consists of an inverting amplifier and a feedback element with XTAL1 as input, and XTAL2 as output.

When using a crystal, a proper external oscillator circuitry must be connected to both pins, XTAL1 and XTAL2. The crystal frequency can be within the range of 4 MHz to 12 MHz. Additionally, it is necessary to have two load capacitances $C_{X1}$ and $C_{X2}$, and depending on the crystal type, a series resistor $R_{X2}$, to limit the current. A test resistor $R_Q$ may be temporarily inserted to measure the oscillation allowance (negative resistance) of the oscillator circuitry. $R_Q$ values are typically specified by the crystal vendor. The $C_{X1}$ and $C_{X2}$ values shown in Figure 25 can be used as starting points for the negative resistance evaluation and for non-productive systems. The exact values and related operating range are dependent on the crystal frequency and have to be determined and optimized together with the crystal vendor using the negative resistance method. Oscillation measurement with the final target system is strongly recommended to verify the input amplitude at XTAL1 and to determine the actual oscillation allowance (margin negative resistance) for the oscillator-crystal system.

When using an external clock signal, the signal must be connected to XTAL1. XTAL2 is left open (unconnected).

The oscillator can also be used in combination with a ceramic resonator. The final circuitry must also be verified by the resonator vendor. Figure 25 shows the recommended external oscillator circuitries for both operating modes, external crystal mode and external input clock mode.

### 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 MultiCAN): FCLK = 24 or 48 MHz
- Peripheral clock: PCLK = 24 MHz
- Flash Interface clock: CCLK2 = 48 MHz and CCLK = 24 MHz

In addition, different clock frequencies can be output to pin CLKOUT (P0.0 or P0.7). The clock output frequency, which is derived from the clock output divider (bit COREL), can further be divided by 2 using toggle latch (bit TLEN is set to 1). The resulting output frequency has a 50% duty cycle. **Figure 26** shows the clock distribution of the XC886/888.

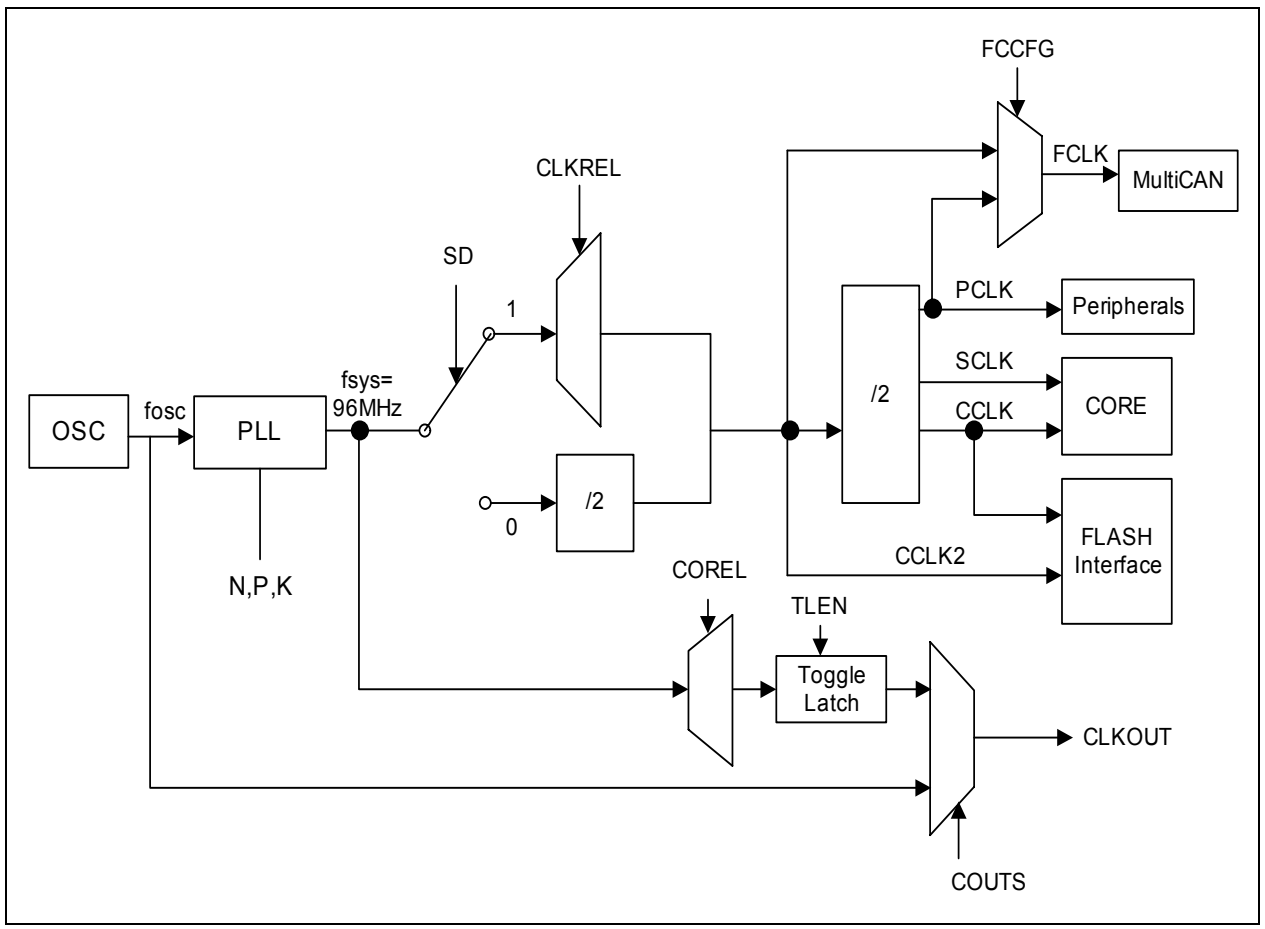



**Figure 26 Clock Generation from $f_{sys}$**

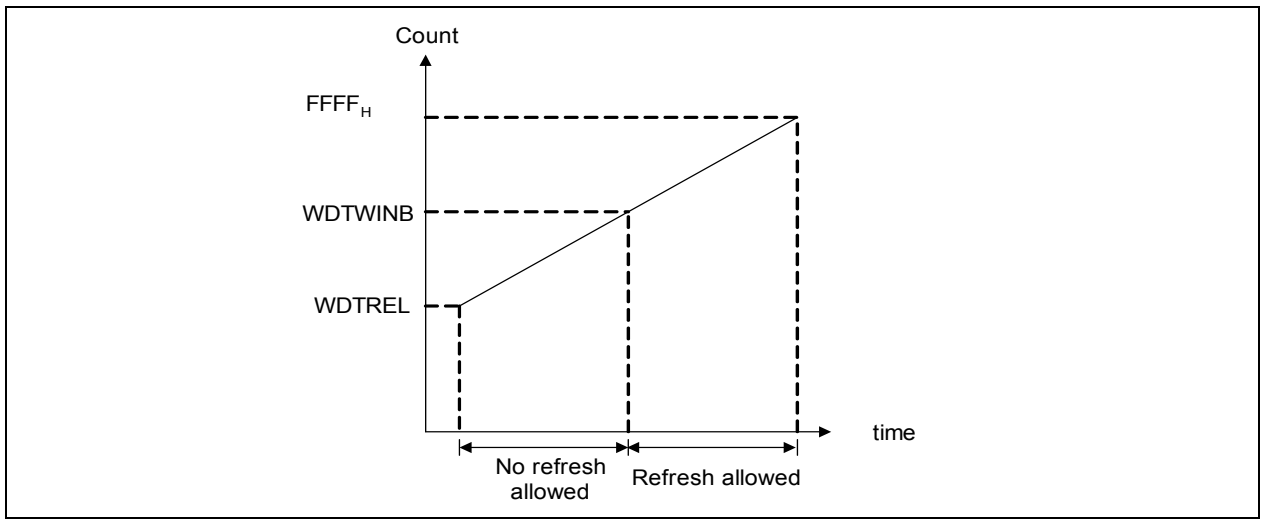



**Figure 29 WDT Timing Diagram**

Table 27 lists the possible watchdog time ranges that can be achieved using a certain module clock. 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** |
| $FF_H$ | 21.3 µs | 1.37 ms |
| $7F_H$ | 2.75 ms | 176 ms |
| $00_H$ | 5.46 ms | 350 ms |

needed for the handshaking between the master and slave tasks is provided by the master task through the header portion of the frame.

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.*

## 3.20 Controller Area Network (MultiCAN)

The MultiCAN module contains two Full-CAN nodes operating independently or exchanging data and remote frames via a gateway function. Transmission and reception of CAN frames is handled in accordance to CAN specification V2.0 B active. Each CAN node can receive and transmit standard frames with 11-bit identifiers as well as extended frames with 29-bit identifiers.

Both CAN nodes share a common set of message objects, where each message object may be individually allocated to one of the CAN nodes. Besides serving as a storage container for incoming and outgoing frames, message objects may be combined to build gateways between the CAN nodes or to setup a FIFO buffer.

The message objects are organized in double chained lists, where each CAN node has it's own list of message objects. A CAN node stores frames only into message objects that are allocated to the list of the CAN node. It only transmits messages from objects of this list. A powerful, command driven list controller performs all list operations.

The bit timings for the CAN nodes are derived from the peripheral clock ($f_{CAN}$) and are programmable up to a data rate of 1 MBaud. A pair of receive and transmit pins connects each CAN node to a bus transceiver.

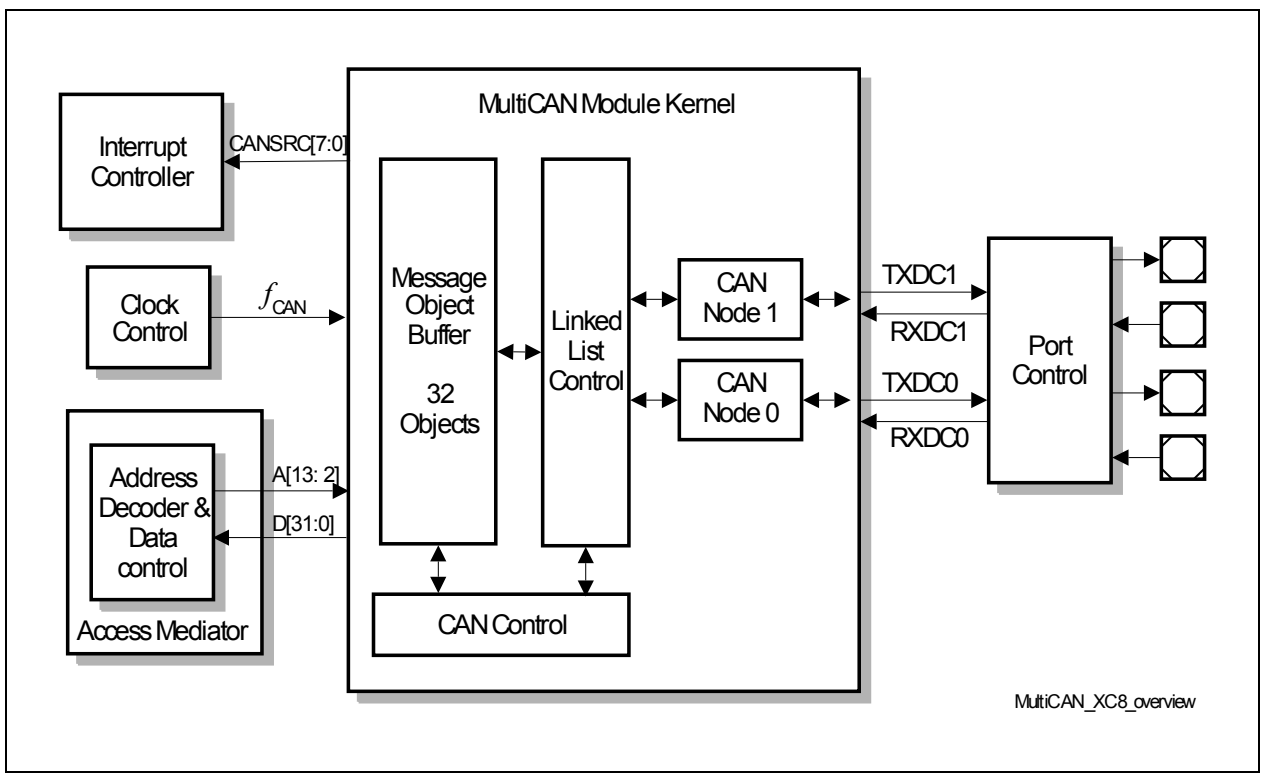



**Figure 34 Overview of the MultiCAN**

### Features

- Compliant to ISO 11898.

### 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 4-1 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 | 1) |
| Junction temperature | $T_J$ | -40 | 150 | °C | under bias1) |
| Voltage on power supply pin with respect to $V_{SS}$ | $V_{DDP}$ | -0.5 | 6 | V | 1) |
| Voltage on any pin with respect to $V_{SS}$ | $V_{IN}$ | -0.5 | $V_{DDP}$ + 0.5 or max. 6 | V | whichever is lower1) |
| Input current on any pin during overload condition | $I_{IN}$ | -10 | 10 | mA | 1) |
| Absolute sum of all input currents during overload condition | $\Sigma\|I_{IN}\|$ | – | 50 | mA | 1) |

1) Not subjected to production test, verified by design/characterization.

*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 37 Operating Condition Parameters**

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

1) $f_{SYS}$ is the PLL output clock. During normal operating mode, CPU clock is $f_{SYS}$ / 4. Please refer to **Figure 26** for detailed description.

## 4.3 AC Parameters

The electrical characteristics of the AC Parameters are detailed in this section.

### 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**.

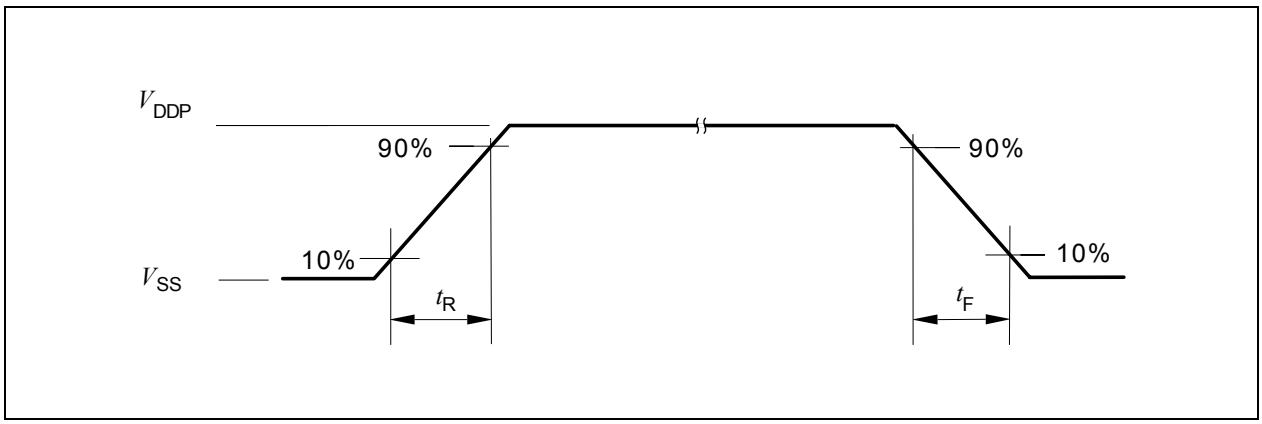



**Figure 40 Rise/Fall Time Parameters**

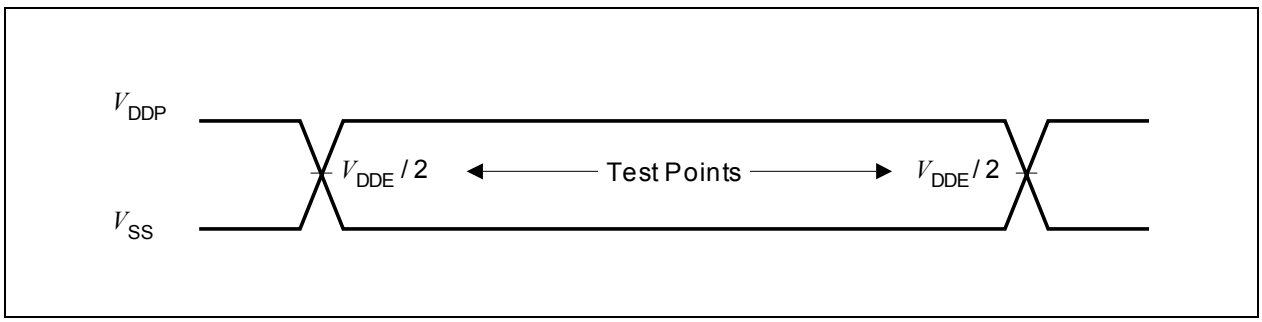



**Figure 41 Testing Waveform, Output Delay**

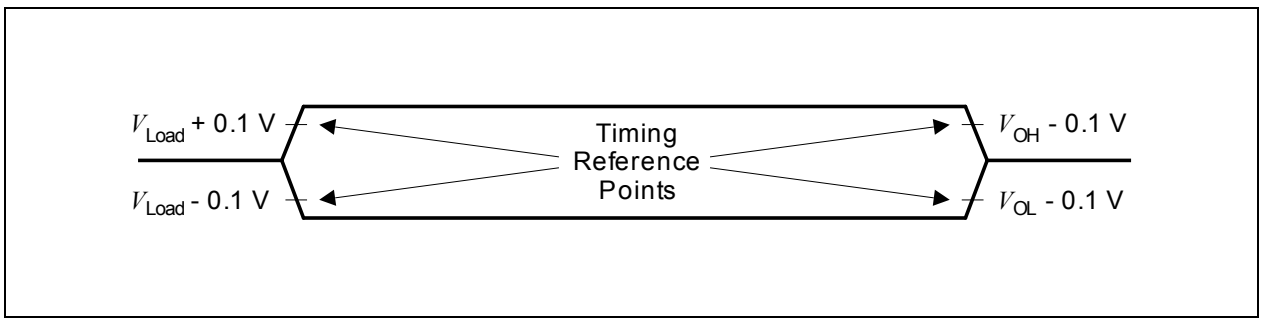



**Figure 42 Testing Waveform, Output High Impedance**

# 5 Package and Quality Declaration

**Chapter 5** provides the information of the XC886/888 package and reliability section.

## 5.1 Package Parameters

**Table 1** provides the thermal characteristics of the package used in XC886 and XC888.

**Table 1 Thermal Characteristics of the Packages**

| Parameter | Symbol | | Limit Values | | Unit | Notes |
|---|---|---|---|---|---|---|
| | | | Min. | Max. | | |
| **PG-TQFP-48 (XC886)** | | | | | | |
| Thermal resistance junction case | $R_{TJC}$ | CC | - | 13 | K/W | 1)2) |
| Thermal resistance junction lead | $R_{TJL}$ | CC | - | 32.5 | K/W | 1)2) |
| **PG-TQFP-64 (XC888)** | | | | | | |
| Thermal resistance junction case | $R_{TJC}$ | CC | - | 12.6 | K/W | 1)2) |
| Thermal resistance junction lead | $R_{TJL}$ | CC | - | 33.4 | K/W | 1)2) |

1) The thermal resistances between the case and the ambient ($R_{TCA}$) , the lead and the ambient ($R_{TLA}$) are to be combined with the thermal resistances between the junction and the case ($R_{TJC}$), the junction and the lead ($R_{TJL}$) given above, in order to calculate the total thermal resistance between the junction and the ambient ($R_{TJA}$). The thermal resistances between the case and the ambient ($R_{TCA}$), the lead and the ambient ($R_{TLA}$) depend on the external system (PCB, case) characteristics, and are under user responsibility.
The junction temperature can be calculated using the following equation: $T_J=T_A+R_{TJA} \times P_D$, where the $R_{TJA}$ is the total thermal resistance between the junction and the ambient. This total junction ambient resistance $R_{TJA}$ can be obtained from the upper four partial thermal resistances, by
a) simply adding only the two thermal resistances (junction lead and lead ambient), or
b) by taking all four resistances into account, depending on the precision needed.

2) Not all parameters are 100% tested, but are verified by design/characterization and test correlation.