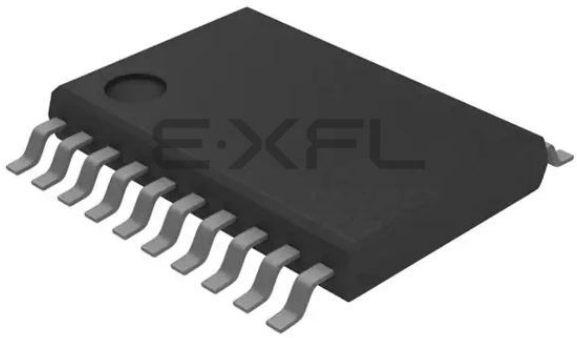

Welcome to E-XFL.COM

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	86MHz
Connectivity	LINbus, SSI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	9
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	PG-TSSOP-20
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/saf-xc864l-1fri-3v-aa

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
  - 8 Kbytes of Boot ROM
  - 256 bytes of RAM
  - 512 bytes of XRAM
  - 4 Kbytes of Flash for code (and data) (includes memory protection strategy)
- I/O port supply at 3.3 V/5.0 V and core logic supply at 2.5 V (generated by embedded voltage regulator)

(further features are on next page)

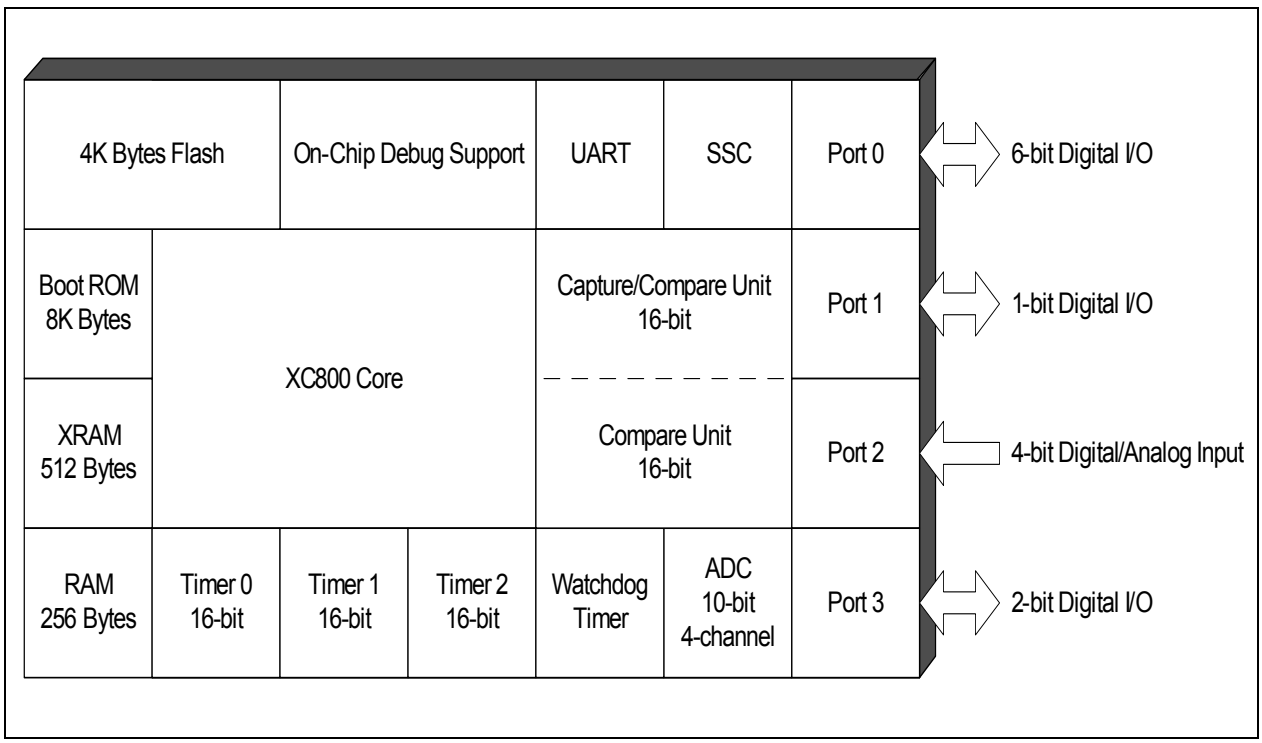

Figure 1 XC864 Functional Units

2.2 Logic Symbol

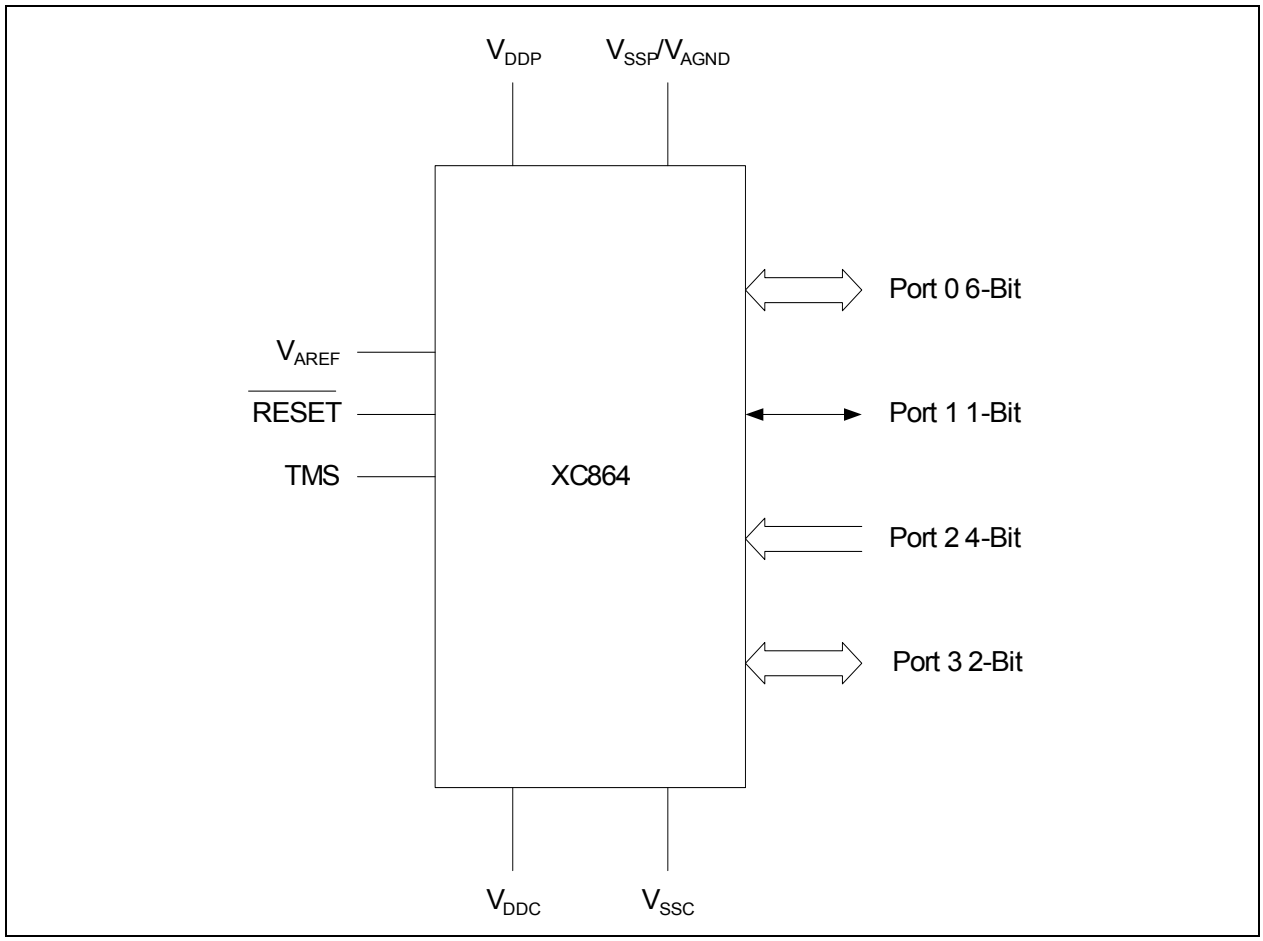

Figure 3 XC864 Logic Symbol

3 Functional Description

3.1 Processor Architecture

The XC864 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 XC864 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 XC864 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 5 shows the CPU functional blocks.

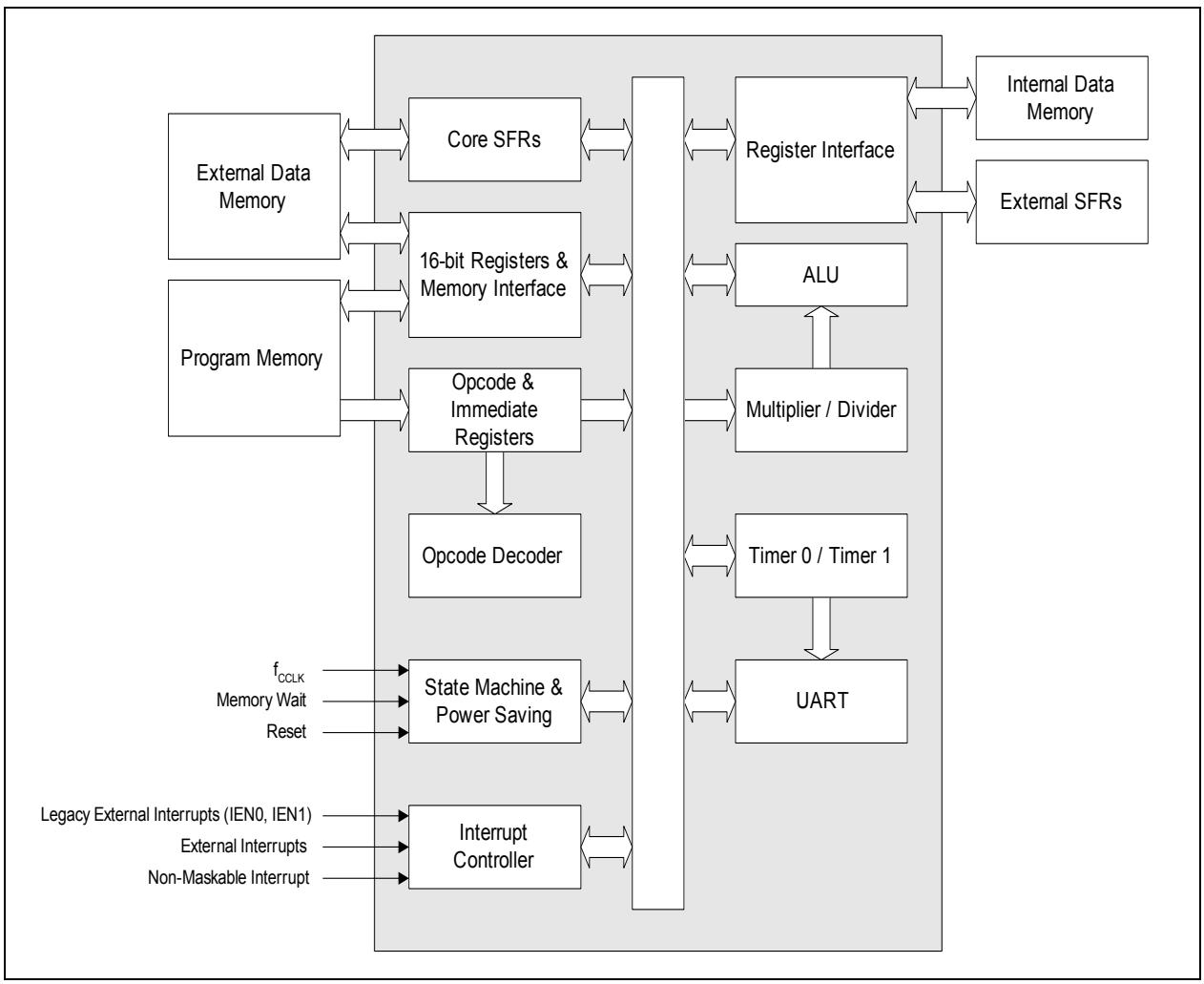

Figure 5 CPU Block Diagram

3.2.2 Special Function Register

The Special Function Registers (SFRs) occupy direct internal data memory space in the range 80_H to FF_H. 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_H to FF_H, 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 $8F_H$. 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 7**.

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

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

Field	Bits	Type	Description
RMAP	0	rw	**Special Function Register Map Control** 0 The access to the standard SFR area is enabled. 1 The access to the mapped SFR area is enabled.
1	2	rw	**Reserved** Returns the last value if read; should be written with 1.
0	1,[7:3]	r	**Reserved** Returns 0 if read; should be written with 0.

Table 6 WDT Register Overview

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 1										
BB_H	**WDTCON** **Reset: 00_H** Watchdog Timer Control Register	Bit Field	0		WINB EN	WDT PR	0	WDT EN	WDT RS	WDT IN
		Type	r		rw	rh	r	rw	rwh	rw
BC_H	**WDTREL** **Reset: 00_H** Watchdog Timer Reload Register	Bit Field	WDTREL							
		Type	rw							
BD_H	**WDTWINB** **Reset: 00_H** Watchdog Window-Boundary Count Register	Bit Field	WDTWINB							
		Type	rw							
BE_H	**WDTL** **Reset: 00_H** Watchdog Timer Register Low	Bit Field	WDT[7:0]							
		Type	rh							
BF_H	**WDTH** **Reset: 00_H** Watchdog Timer Register High	Bit Field	WDT[15:8]							
		Type	rh							

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

Table 7 Port Register Overview

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
RMAP = 0										
$B2_H$	**PORT_PAGE** **Reset: 00_H** Page Register for PORT	Bit Field	OP		STNR		0	PAGE		
		Type	w		w		r	rwh		
RMAP = 0, Page 0										
80_H	**P0_DATA** **Reset: 00_H** P0 Data Register	Bit Field	0		P5	P4	P3	P2	P1	P0
		Type	r		rwh	rwh	rwh	rwh	rwh	rwh
86_H	**P0_DIR** **Reset: 00_H** P0 Direction Register	Bit Field	0		P5	P4	P3	P2	P1	P0
		Type	r		rw	rw	rw	rw	rw	rw
90_H	**P1_DATA** **Reset: 00_H** P1 Data Register	Bit Field	0						P1	P0
		Type	rwh						rwh	rwh
91_H	**P1_DIR** **Reset: 00_H** P1 Direction Register	Bit Field	0						P1	P0
		Type	rw						rw	rw
$A0_H$	**P2_DATA** **Reset: 00_H** P2 Data Register	Bit Field	P7	0				P2	P1	P0
		Type	rwh	rwh				rwh	rwh	rwh
$A1_H$	**P2_DIR** **Reset: 00_H** P2 Direction Register	Bit Field	P7	0				P2	P1	P0
		Type	rw	rw				rw	rw	rw
$B0_H$	**P3_DATA** **Reset: 00_H** P3 Data Register	Bit Field	0						P1	P0
		Type	rwh						rwh	rwh
$B1_H$	**P3_DIR** **Reset: 00_H** P3 Direction Register	Bit Field	0						P1	P0
		Type	rw						rw	rw
RMAP = 0, Page 1										
80_H	**P0_PUDSEL** **Reset: FF_H** P0 Pull-Up/Pull-Down Select Register	Bit Field	1		P5	P4	P3	P2	P1	P0
		Type	r		rw	rw	rw	rw	rw	rw
86_H	**P0_PUDEN** **Reset: $C4_H$** P0 Pull-Up/Pull-Down Enable Register	Bit Field	1		P5	P4	P3	P2	P1	P0
		Type	r		rw	rw	rw	rw	rw	rw
90_H	**P1_PUDSEL** **Reset: FF_H** P1 Pull-Up/Pull-Down Select Register	Bit Field	1						P1	P0
		Type	rw						rw	rw
91_H	**P1_PUDEN** **Reset: FF_H** P1 Pull-Up/Pull-Down Enable Register	Bit Field	1						P1	P0
		Type	rw						rw	rw

Table 12 OCDS Register Summary (cont'd)

Addr	Register Name	Bit	7	6	5	4	3	2	1	0
$F4_H$	**MMICR** **Reset: 00_H** Monitor Mode Interrupt Control Register	Bit Field	DVECT	DRETR	COMRST	MSTSEL	MMUIE_P	MMUIE	RRIE_P	RRIE
		Type	rwh	rwh	rwh	rh	w	rw	w	rw
$F5_H$	**MMDR** **Reset: 00_H** Monitor Mode Data Transfer Register *Receive*	Bit Field	MMRR							
		Type	rh							
$F6_H$	**HWBPSR** **Reset: 00_H** Hardware Breakpoints Select Register	Bit Field	0			BPSEL_P	BPSEL			
		Type	r			w	rw			
$F7_H$	**HWBPDR** **Reset: 00_H** Hardware Breakpoints Data Register	Bit Field	HWBPxx							
		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
- 32-byte minimum program width[1)]
- 1-sector minimum erase width
- 1-byte read access
- Operating supply voltage: 2.5 V ± 7.5 %
- Read access time: 3 × t_{CCLK} = 112.5 ns[2)]
- Program time: 209440 / f_{SYS} = 2.6 ms[3)]
- Erase time: 8175360 / f_{SYS} = 102 ms[3)]

Table 13 shows the Flash data retention and endurance targets.

Table 13 Flash Data Retention and Endurance (Operating Conditions apply)

Retention	Endurance[1)]	Size
20 years	1,000 cycles	up to 4 Kbytes
5 years	10,000 cycles	1 Kbyte
2 years	70,000 cycles	512 bytes
2 years	100,000 cycles	128 bytes

1) One cycle refers to the programming of all wordlines in a sector and erasing of sector. The Flash endurance data specified in **Table 13** is valid only if the following conditions are fulfilled:
- the maximum number of erase cycles per Flash sector must not exceed 100,000 cycles.
- the maximum number of erase cycles per Flash bank must not exceed 300,000 cycles.
- the maximum number of program cycles per Flash bank must not exceed 2,500,000 cycles.

1) 32-byte wordline can be programmed twice, i.e., two gate disturbs allowed.
2) Values shown here are typical values. f_{sys} = 80 MHz ± 7.5% (f_{CCLK} = 26.7 MHz ± 7.5 %) is the maximum frequency range for Flash read access.
3) Values shown here are typical values. f_{sys} = 80 MHz ± 7.5% is the only frequency range for Flash programming and erasing. f_{sysmin} is used for obtaining the worst case timing.

3.3.2 Flash Programming Without Erase

The same WL can be programmed twice before erasing is required as the Flash cells are able to withstand two gate disturbs. This means if the number of data bytes that needs to be written is smaller than the 32-byte minimum programming width, the user can opt to program this number of data bytes (x; where x can be any integer from 1 to 31) first and program the remaining bytes (32 - x) later. Hence, it is possible to program the same WL, for example, with 16 bytes of data in two times (see **Figure 11**).

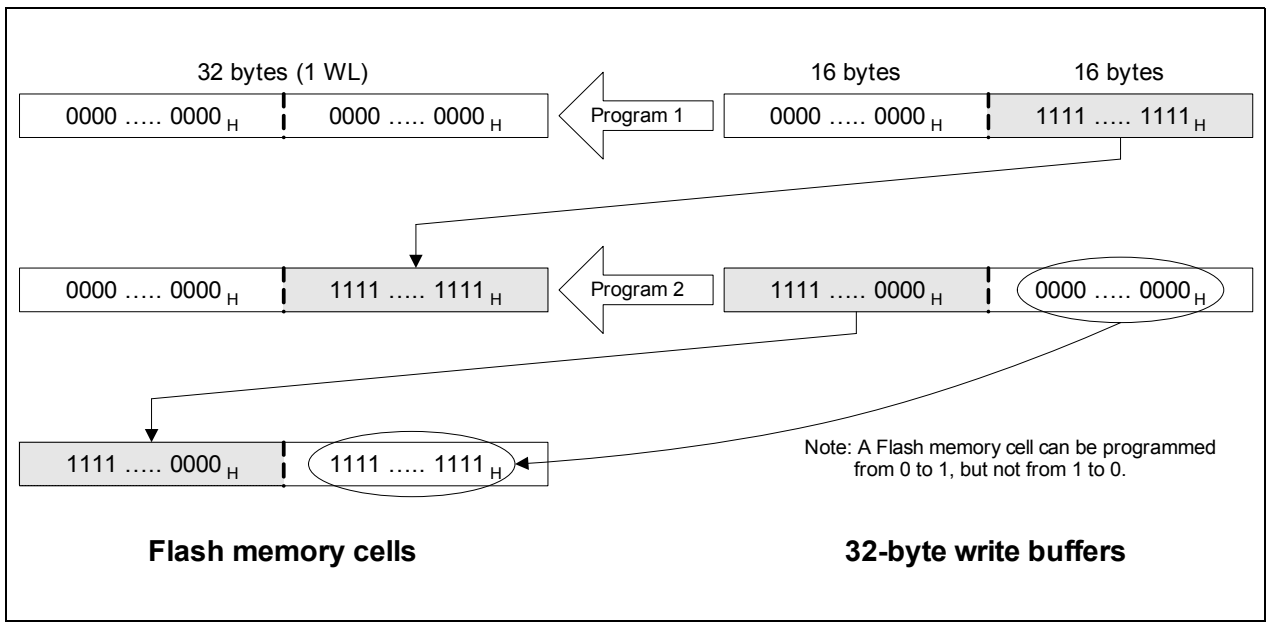

Figure 11 Flash Programming Without Erase

Note: When programming a 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”.

Table 15 Flash Program Subroutine Type 2 (cont'd)

Resource used/ destroyed	ACC, B, SCU_PAGE
	R0 – R7 of Register Bank 3 (IRAM address 18_H – $1F_H$) (8 bytes) IRAM address 36_H – $3D_H$ (8 bytes)

1) The last 5 LSB of the DPL is 0 for an aligned WL address, for e.g. 00_H, 20_H, 40_H, 60_H, 80_H, $A0_H$, $C0_H$ and $E0_H$..

2) DPTR is only incremented by 20_H when PSW.CY is 0.

3.3.3.2 Flash Erasing

Each call of the Flash erase subroutine allows the user to select one sector or a combination of several sectors for erase. Before calling the Flash erase subroutine, the user must ensure that required inputs (**Table 16** and **Table 17**) are provided. Also, protected Flash banks should not be targeted for erase.

Flash Erase Subroutine Type 1

If valid inputs have been set up, calling the subroutine begins flash erasing. The subroutine exits and returns to the user code, while the target Flash bank is still in erase mode, and is not accessible by user code.

Table 16 Flash Erase Subroutine Type 1

Subroutine	$DFF9_H$: FLASH_ERASE
Input[1)]	R3 of Register Bank 3 (IRAM address $1B_H$): Select sector(s) to be erased. LSB represents sector 0, MSB represents sector 7.
	R4 of Register Bank 3 (IRAM address $1C_H$): Select sector(s) to be erased. LSB represents sector 8, bit 1 represents sector 9.
	Flash NMI (NMICON.NMIFLASH) is enabled (1) or disabled (0)
	MISC_CON.DFLASHEN[2)] bit = 1
Output	PSW.CY: 0 = Flash erasing is in progress 1 = Flash erasing is not started Flag FNMIFLASH will be set when Flash erasing has successfully completed.
Stack size required	10
Resource used/ destroyed	ACC, B, SCU_PAGE
	R0 – R7 of Register Bank 3 (IRAM address 18_H – $1F_H$) (8 bytes)
	IRAM address 36_H – $3D_H$ (8 bytes)

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 19**.

Table 19 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 Normal Divider Overflow, LIN	6
ADC Interrupt	7
SSC Interrupt	8
External Interrupt 2	9
External Interrupt 3	10
CCU6 Interrupt Node Pointer 0	11
CCU6 Interrupt Node Pointer 1	12
CCU6 Interrupt Node Pointer 2	13
CCU6 Interrupt Node Pointer 3	14

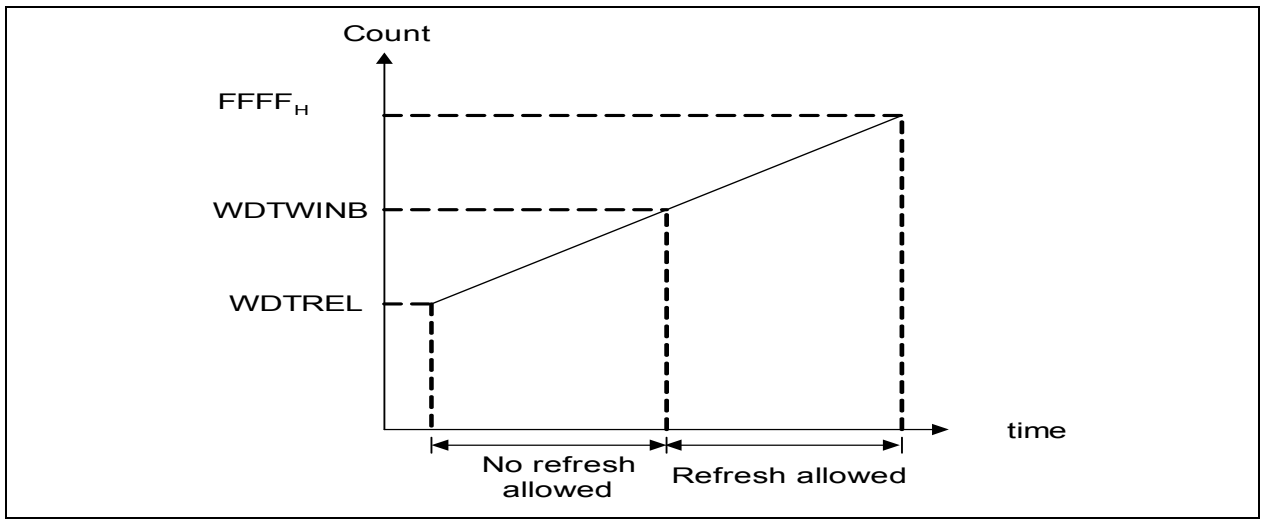

Figure 26 WDT Timing Diagram

Table 24 lists the possible watchdog time range that can be achieved for different module clock frequencies . Some numbers are rounded to 3 significant digits.

Table 24 Watchdog Time Ranges

Reload value in WDTREL	Prescaler for f_{PCLK}	
	2 (WDTIN = 0)	**128 (WDTIN = 1)**
	26.7 MHz	**26.7 MHz**
FF_H	19.2 µs	1.23 ms
$7F_H$	2.48 ms	159 ms
00_H	4.92 ms	315 ms

- 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:

[5]

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

[6]

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

The maximum baud rate that can be generated is limited to $f_{PCLK}/32$. Hence, for a module clock of 26.7 MHz, the maximum achievable baud rate is 0.83 MBaud.

Standard LIN protocal 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 26 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 26.7 MHz is used.

Table 26 Typical Baud rates for UART with Fractional Divider disabled

Baud rate	Prescaling Factor (2^{BRPRE})	Reload Value (BR_VALUE + 1)	Deviation Error
19.2 kBaud	1 (BRPRE=000_B)	87 (57_H)	-0.22 %
9600 Baud	1 (BRPRE=000_B)	174 (AE_H)	-0.22 %
4800 Baud	2 (BRPRE=001_B)	174 (AE_H)	-0.22 %
2400 Baud	4 (BRPRE=010_B)	174 (AE_H)	-0.22 %

The fractional divider allows baud rates of higher accuracy (lower deviation error) to be generated. **Table 27** lists the resulting deviation errors from generating a baud rate of

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.*

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 29 shows the block diagram of the SSC.

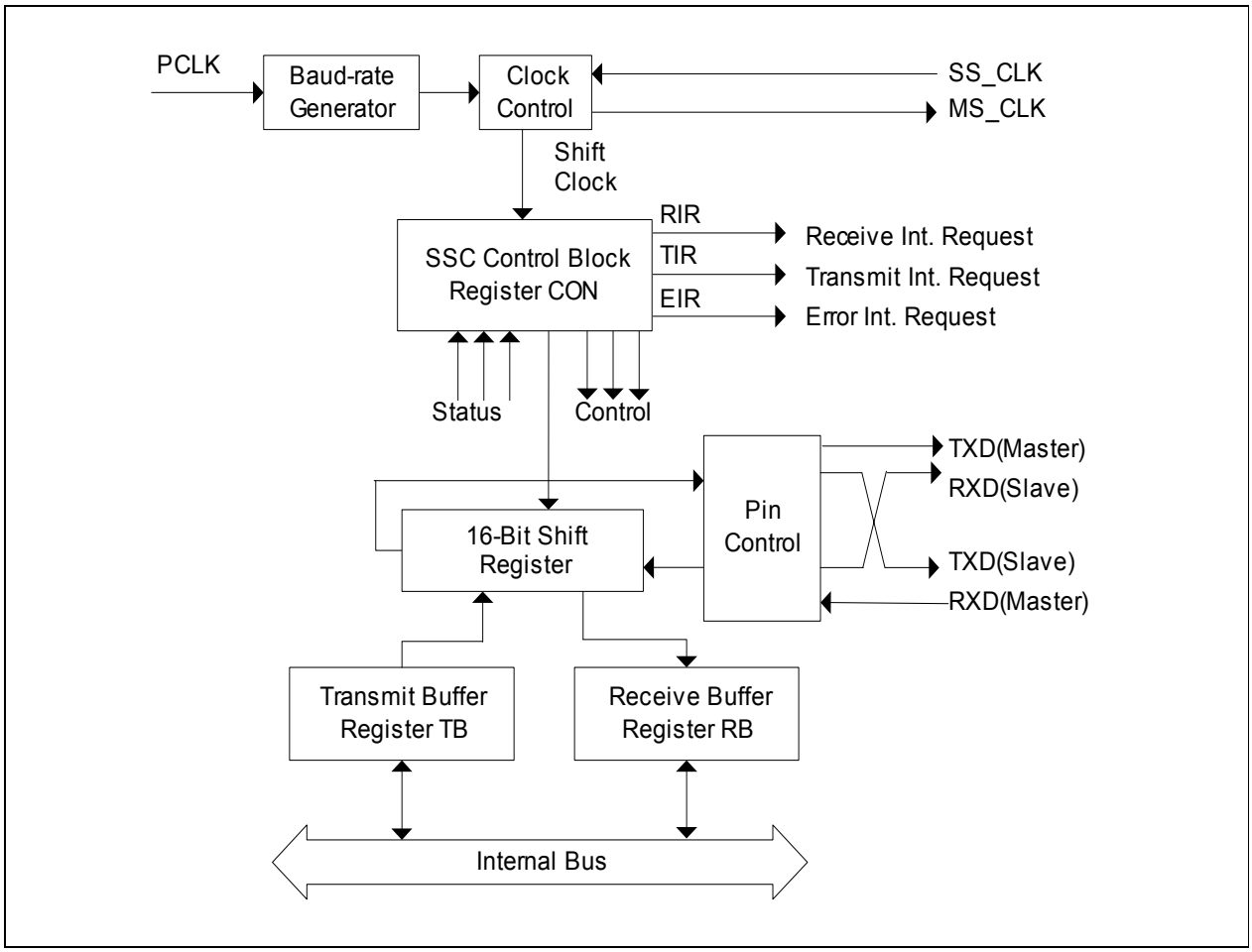

Figure 29 SSC Block Diagram

3.15 Timer 0 and Timer 1

Timers 0 and 1 are count-up timers which are incremented every machine cycle, or in terms of the input clock, every 2 PCLK cycles. Timer 0 can also be incremented in response to a 1-to-0 transition (falling edge) at the external input pin, T0.

Both timers are fully compatible and can be configured in four different operating modes for use in a variety of applications, see **Table 28**. In modes 0, 1 and 2, the two timers operate independently, but in mode 3, their functions are specialized.

Table 28 Timer 0 and Timer 1 Modes

Mode	Operation
0	**13-bit timer** 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.
1	**16-bit timer** The timer registers, TLx and THx, are concatenated to form a 16-bit counter.
2	**8-bit timer with auto-reload** The timer register TLx is reloaded with a user-defined 8-bit value in THx upon overflow.
3	**Timer 0 operates as two 8-bit timers** The timer registers, TL0 and TH0, operate as two separate 8-bit counters. Timer 1 is halted and retains its count even if enabled.

3.17 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

4 Electrical Parameters

This chapter provides the characteristics of the electrical parameters which are implementation-specific for the XC864.

4.1 General Parameters

The general parameters are described here to aid the users in interpreting the parameters mainly in **Chapter 4.2** and **Chapter 4.3**.

4.1.1 Parameter Interpretation

The parameters listed in this section represent partly the characteristics of the XC864 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 XC864 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 XC864 is designed in.

4.2.4 Power Supply Current

Table 36 Power Supply Current Parameters (Operating Conditions apply; V_{DDP} = 5V range)

Parameter	Symbol	Limit Values		Unit	Test Condition
		typ.[1]	max.[2]		
V_{DDP} = 5V Range					
Active Mode	I_{DDP}	22.6	24.5	mA	[3]
Idle Mode	I_{DDP}	12.5	14	mA	[4]
Active Mode with slow-down enabled	I_{DDP}	5.6	7.5	mA	[5]
Idle Mode with slow-down enabled	I_{DDP}	5.1	7.2	mA	[6]

1) The typical I_{DDP} values are periodically measured at T_A = + 25 °C and V_{DDP} = 5.0 V.

2) The maximum I_{DDP} values are measured under worst case conditions (T_A = + 125 °C and V_{DDP} = 5.5 V).

3) I_{DDP} (active mode) is measured with: CPU clock and input clock to all peripherals running at 26.7 MHz(set by on-chip oscillator of 10 MHz and NDIV in PLL_CON to 0010_B), $\overline{RESET}$ = V_{DDP}, no load on ports.

4) I_{DDP} (idle mode) is measured with: CPU clock disabled, watchdog timer disabled, input clock to all peripherals enabled and running at 26.7 MHz, $\overline{RESET}$ = V_{DDP}, no load on ports.

5) I_{DDP} (active mode with slow-down mode) is measured with: CPU clock and input clock to all peripherals running at 833 KHz by setting CLKREL in CMCON to 0101_B, $\overline{RESET}$ = V_{DDP}, no load on ports.

6) 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 833 KHz by setting CLKREL in CMCON to 0101_B, $\overline{RESET}$ = V_{DDP}, no load on ports.

Table 37 Power Down Current (Operating Conditions apply; V_{DDP} = 5V range)

Parameter	Symbol	Limit Values		Unit	Test Condition
		typ.[1]	max.[2]		
V_{DDP} = 5V Range					
Power-Down Mode[3]	I_{PDP}	1	10	µA	T_A = + 25 °C.[4]
		-	35	µA	T_A = + 85 °C.[4][5]

1) The typical I_{PDP} values are measured at V_{DDP} = 5.0 V.

2) The maximum I_{PDP} values are measured at V_{DDP} = 5.5 V.

3) I_{PDP} (power-down mode) has a maximum value of 200 µA at T_A = + 125 °C.

4) I_{PDP} (power-down mode) is measured with: RESET = V_{DDP}, V_{AGND}= V_{SS}, 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.

5) Not subject to production test, verified by design/characterization.

4.3.5 JTAG Timing

Table 43 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	20	–	ns
TCK low time	t_2	SR	20	–	ns
TCK clock rise time	t_3	SR	–	4	ns
TCK clock fall time	t_4	SR	–	4	ns

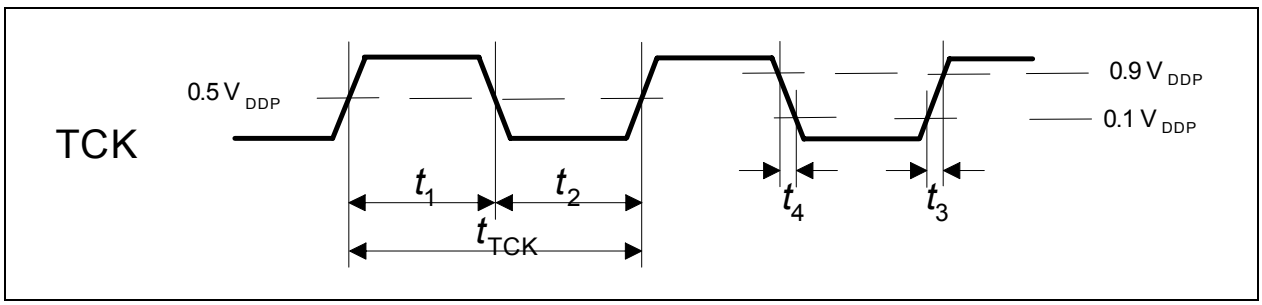

Figure 41 TCK Clock Timing

5.2 Package Outline

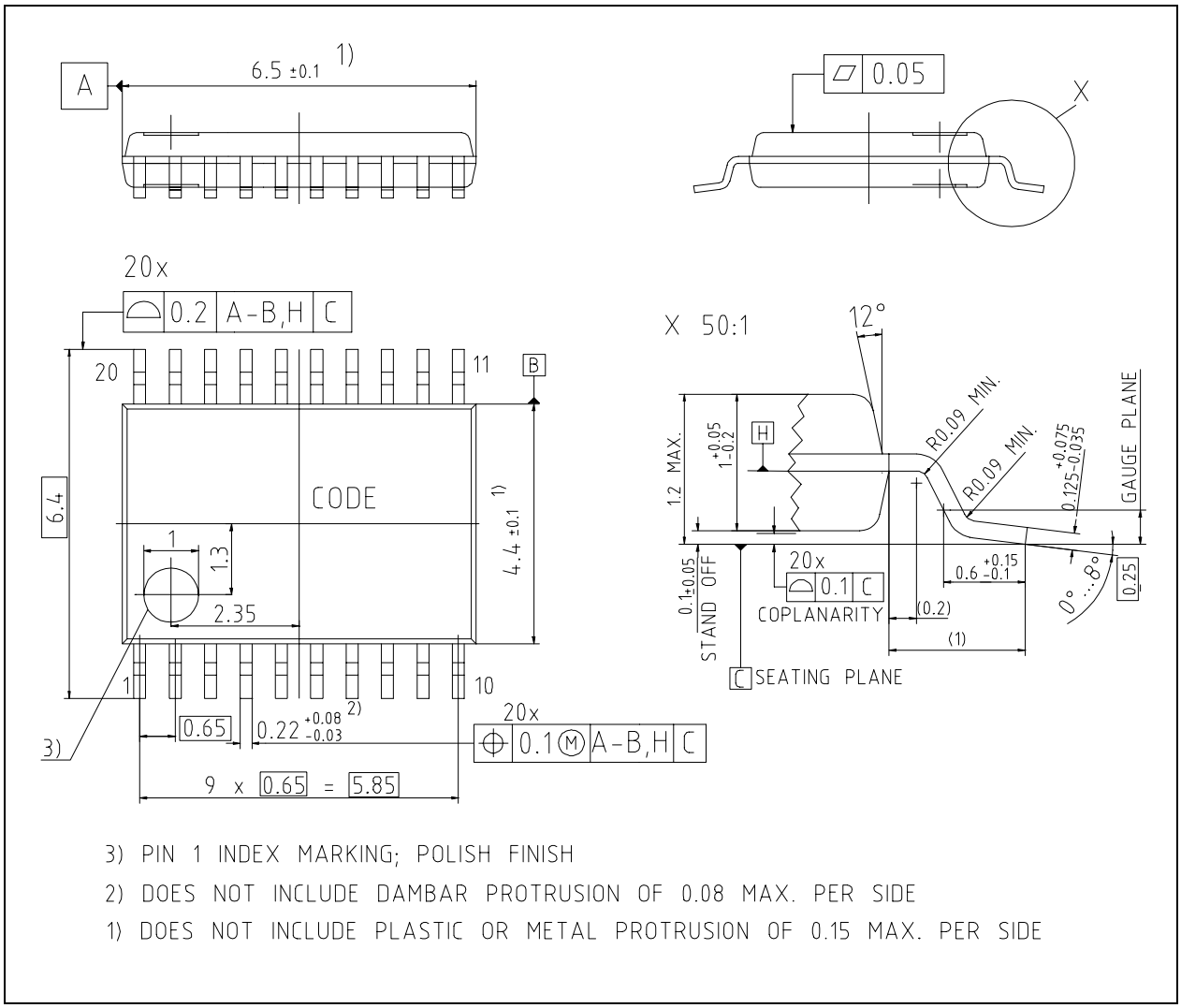

Figure 43 PG-TSSOP-20 Package Outline