XC82x/XC83x

AP08101

Current Consumption in Power Saving Modes
For Low Power Applications

Application Note

V1.0, 2010-08

Microcontrollers

Edition 2010-08

Published by
Infineon Technologies AG
81726 Munich, Germany

XC82x/XC83x
Revision History: V1.0 2010-08

Previous Version(s):

Page	Subjects (major changes since last revision)
–	

We Listen to Your Comments
Is there any information in this document that you feel is wrong, unclear or missing? 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

Table of Contents

1 Overview . . . 5

2 Stopping the CPU Clock (Idle Mode) . . . 6

2.1 Entering and Exiting Idle Mode . . . 6

3 Power Down of the Entire System (Power Down Mode) . . . 8

3.1 Powering Down XC82x . . . 8

3.2 Powering Down XC83x . . . 8

3.3 Power Down Current Consumption and Wake-up Timing . . . 9

4 Stopping/Reducing Peripheral Clock Speeds(Peripheral Management) . . . 10

5 Power Saving Checklist . . . 13

6 Summary . . . 14

1 Overview

The XC82x and XC83x products from Infineon are designed to help achieve low power consumption in applications, as both of these devices allow the user to configure the operating mode via software.

In this application note, XC82x and XC83x users are provided with information and guidelines on how to apply power saving features via a combination of techniques, including:

- Stopping the CPU clock (Idle Mode)
- Powering down the entire system with fast restart capability (Power-Down Mode)
- Stopping the clocks of individual system components (Peripheral Management)
- Reducing the clock speed of some peripheral components (Peripheral Management)

In this document, users can find measurements of the current consumed by the XC82x and XC83x devices when operating in different power modes. There is also a breakdown of the current consumed by the peripherals at different clock speeds. With these measurements, users will have a better understanding of how the different power saving features affect the power consumption of the device and therefore gain a greater understanding of how to apply them more effectively. Note however, that these results are meant as a reference only as the values may vary from device to device. The power consumption can also be affected by various factors such as the type of instructions used in the code, program flow, peripheral settings, measuring equipment, power supply, temperature and the frequency of the oscillator for example.

The measurements recorded in this document are the averages of the readings taken from three different XC822 devices and two different XC836 devices.

Unless otherwise stated, all the readings were carried out under the following conditions:

- Power supply of 3.3V, 5V
- CPU clock of 8MHz, 24MHz
- Ambient temperature of approximately 25 ^{0}C
- Port 2 (analog inputs) were disabled
- All the other port pins were configured as output and set to high
- Peripherals were not programmed to perform any operations

2 Stopping the CPU Clock (Idle Mode)

The microcontroller can reduce power consumption by stopping the CPU clock. This can be achieved by putting the device in idle mode. In this mode, the oscillator continues to run, but the CPU is stopped with its clock disabled. Peripherals whose input clocks are not disabled are still functional.

Note: If the WatchDog Timer (WDT) is still active when the device goes into idle mode, it will generate an internal reset when an overflow occurs. It is therefore necessary to disable the WDT before entering idle mode.

The CPU status is preserved in its entirety; the stack pointer, program counter, program status word, accumulator, and all other registers maintain their data during idle mode. The port pins hold the logical state they had at the time the idle mode was activated, unless they are modified by the peripherals. For example, UART, SPI data transfer in progress.

2.1 Entering and Exiting Idle Mode

Idle mode can be entered by setting the bit PCON.IDLE.

```
PCON |= 0x01;
```

Table 1, **Table 2**, **Table 3** and **Table 4** show the savings in current consumption between active and idle mode of the XC82x and XC83x devices.

Table 1 Active and Idle mode current measurements for XC822 at V_{DDP} = 3.3V

Device	CPU Clock (MHz)	Peripherals Enabled?1)	Current (mA)		
			Active2)	Idle	Savings3)
XC822-1FRI4)	8.00	Yes	13.28	12.49	0.79
XC822-1FRI	8.00	No	5.86	5.07	0.79
XC822-1FRI	24.00	Yes	17.97	16.53	1.44
XC822-1FRI	24.00	No	9.80	8.37	1.43

1) Peripherals are disabled by default. They can be enabled via PMCON1.
2) Program waits in a continuous loop. i.e. while(1);
3) Savings = Active - Idle.
4) 1F indicates 4-Kbyte flash size.

Table 2 Active and Idle mode current measurements for XC836 at V_{DDP} = 3.3V

Device	CPU Clock (MHz)	Peripherals Enabled?1)	Current (mA)		
			Active2)	Idle	Savings3)
XC836-2FRI4)	8.00	Yes	16.78	15.75	1.03
XC836-2FRI	8.00	No	6.81	5.79	1.02
XC836-2FRI	24.00	Yes	22.06	20.29	1.77
XC836-2FRI	24.00	No	11.30	9.53	1.77

1) Peripherals are disabled by default. They can be enabled via PMCON1.
2) Program waits in a continuous loop. i.e. while(1);
3) Savings = Active - Idle.
4) 2F indicates 8-Kbyte flash size.

Table 3 Active and Idle mode current measurements for XC22 at V_{DDP} = 5V

Device	CPU Clock (MHz)	Peripherals Enabled?[1]	Current (mA)		
			Active[2]	Idle	Savings[3]
XC822-1FRI[4]	8.00	Yes	13.38	12.62	0.76
XC822-1FRI	8.00	No	5.94	5.18	0.76
XC822-1FRI	24.00	Yes	18.10	16.70	1.40
XC822-1FRI	24.00	No	9.88	8.50	1.38

1) Peripherals are disabled by default. They can be enabled via PMCON1.
2) Program waits in a continuous loop. i.e. while(1);
3) Savings = Acive - Idle.
4) 1F indicates 4-Kbyte flash size.

Table 4 Active and Idle mode current measurements for XC36 at V_{DDP} = 5V

Device	CPU Clock (MHz)	Peripherals Enabled?[1]	Current (mA)		
			Active[2]	Idle	Savings[3]
XC836-2FRI[4]	8.00	Yes	17.06	15.93	1.13
XC836-2FRI	8.00	No	7.08	5.92	1.16
XC836-2FRI	24.00	Yes	22.36	20.49	1.87
XC836-2FRI	24.00	No	11.59	9.68	1.91

1) Peripherals are disabled by default. They can be enabled via PMCON1.
2) Program waits in a continuous loop. i.e. while(1);
3) Savings = Active - Idle.
4) 2F indicates 8-Kbyte flash size.

Consider a program that is constantly waiting to service a Timer interrupt. The program can:

a) wait in an endless loop for the interrupt event to occur (active mode), or
b) wait for the interrupt event to occur while the CPU is disabled (idle mode) and then enable the CPU to service the interrupt routine when an interrupt occurs.

Method (b) will consume less power as indicated in **Table 1**, **Table 2**, **Table 3** and **Table 4**.

The device in idle mode can exit to active mode in the event of any of these two conditions:

1. Hardware reset. The device resets and code execution starts from address 0_H.
2. An interrupt has occured from an enabled interrupt source. The device will service the interrupt routine and resumes its operation from the next instruction after the instruction that has previously set the PCON.IDLE bit to 1.

Entering and exiting idle mode each takes approximately 2 CCLK cycles. The system returns to either 8 MHz or 24 MHz active mode upon exiting idle mode, depending on operating mode prior to entering the idle state.

3 Power Down of the Entire System (Power Down Mode)

Low power is achieved in power down mode by switching off the main Embedded Voltage Regulator (EVR). Only the Low Power Embedded Voltage Regulator (LPEVR) is still operating (for more information regarding these 2 voltage regulators, please refer to the User Manual). In addition, the 48 MHz oscillator and the Flash memory are put in power down. Therefore, most of microcontroller functions are stopped, while the contents of the Flash, on-chip RAM, XRAM, and the SFRs are maintained.

In power down mode, all port pins are disabled except for the External Interrupt 0 (EXINT0) pin, if it is enabled as a wake-up source. Because of this the port pins are tri-stated. If the application requires any port pins to maintain an output level of “1” or “0”, the user should enable the pull-up/down device for the respective pins before entering power down mode. The internal pull-up/down device on the port pins will remain active during power down mode, therefore the pins can be made to stay high or low via this method. The port pins will be enabled per the last configuration and status, on wake-up. Note also that the user has to manually restore the status of the internal pull-up/down devices respectively, on wake-up.

For a more detailed description of the power down modes and their implementation details, please refer to the note (**AP08098**: Low Power Modes with Periodic/Real-Time Clock Wake-up in XC82x/XC83x).

3.1 Powering Down XC82x

The XC82x has two power down modes (Mode 1 and 2). All of the general description on the power-down mode mentioned above, applies to both modes.

Essentially, the differences between the two power modes are the modules that are still active, listed in **Table 5**, and the available wake-up sources. All other modules are powered-down in both modes.

Table 5 Status of Modules in Power Down Mode for XC82x

Modules	Mode 1	Mode 2
Low Power Embedded Voltage Regulator (LPEVR)	Active	Active
Real Time Clock (RTC)	Inactive	Active (Mode 1)
75 KHz Oscillator	Inactive	Active

3.2 Powering Down XC83x

The XC83x has four power down modes (Mode 1, 2, 3 and 4). The differences between the four power modes are the modules that are still active during these modes, as listed in **Table 6**, and the available wake-up sources. All other modules that are powered-down in both modes.

Table 6 Status of Modules in Power Down Mode for XC83x

Modules	Mode 1	Mode 2	Mode 3	Mode 4
Low Power Embedded Voltage Regulator (LPEVR)	Active	Active	Active	Active
Real Time Clock (RTC)	Inactive	Active (Mode 0)	Active (Mode 0)	Active (Mode 1)
75 KHz Oscillator	Inactive	Active	Inactive	Active
32.768 kHz Oscillator Pad	Inactive	Active	Active	Inactive
32.768 kHz Oscillator Watchdog	Inactive	Active	Inactive	Inactive

3.3 Power Down Current Consumption and Wake-up Timing

This section provides the measured values of the devices' current consumption in the power down modes at V_{ddp} = 3.3V (**Table 7**) and V_{ddp} = 5V (**Table 8**).

Table 7 Current consumed in power down mode at V_{ddp} = 3.3V

Device	Current (µA)			
	Mode 1[1]	Mode 2	Mode 3[2]	Mode 4[3]
XC822-1FRI	2.23	4.09[4]	-	-
XC836-2FRI	2.23	4.20[5]	2.82	4.05

1) Power Down Mode 1 with EXINT0 as wake-up source and pull-up/down device disabled for all other port pins during power down.
2) Power Down Mode 3 with RTC running in Mode 0 (Time-Keeping Mode with 32.768 kHz Crystal Clock), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
3) Power Down Mode 4 with RTC running in Mode 1 (Periodic Wake-up Mode), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
4) Power Down Mode 2 with RTC running in Mode 1 (Periodic Wake-up Mode), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
5) Power Down Mode 2 with RTC running in Mode 0 (Time-Keeping Mode with 32.768 kHz Crystal Clock), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.

Table 8 Current consumed in power down mode at V_{ddp} = 5V

Device	Current (µA)			
	Mode 1[1]	Mode 2	Mode 3[2]	Mode 4[3]
XC822-1FRI	3.34	5.24[4]	-	-
XC836-2FRI	3.24	5.22[5]	3.83	5.05

1) Power Down Mode 1 with EXINT0 as wake-up source and pull-up/down device disabled for all other port pins during power down.
2) Power Down Mode 3 with RTC running in Mode 0 (Time-Keeping Mode with 32.768 kHz Crystal Clock), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
3) Power Down Mode 4 with RTC running in Mode 1 (Periodic Wake-up Mode), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
4) Power Down Mode 2 with RTC running in Mode 1 (Periodic Wake-up Mode), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.
5) Power Down Mode 2 with RTC running in Mode 0 (Time-Keeping Mode with 32.768 kHz Crystal Clock), 5-minute wake-up time configured, and pull-up/down device disabled for all port pins during power down.

Table 9 Wake-up timing for power down modes

Wake-up reset sequence	XC822-1FRI		XC836-2FRI	
	PMCON0.WKSEL=0	PMCON0.WKSEL=1	PMCON0.WKSEL=0	PMCON0.WKSEL=1
EVR, 48MHz OSC and Flash ready	166µs	166µs	180µs	180µs
BootROM startup	0µs	377µs	0µs	388µs
Total wake-up time	166µs	543µs	180µs	568µs

4 Stopping/Reducing Peripheral Clock Speeds(Peripheral Management)

Depending on the requirement of the application, each of the XC82x/XC83x peripherals can be individually enabled by programming the assigned register bits in PMCON1, which would gate on clock inputs to the individual peripherals. This optimizes the overall power consumption of the device. The analog part of the ADC module may also be disabled by resetting the GLOBCTR.ANON bit when no conversion is needed, allowing a further reduction in power consumption.

Table 10, **Table 11**, **Table 12** and **Table 13** provides the relationship between the amount of current that is consumed when each/all of the peripheral module(s) are enabled and the frequency of the CPU clock (CCLK).

The SSC, T2 and IIC are clocked by peripheral clock (PCLK) which is equal to CCLK in normal running mode.

The ADC, CCU6, MDU, LTS (for both XC82x and XC83x) and Cordic (for XC83x only) are clocked by the fast peripheral clock (FPCLK = 48 MHz) which is driven directly by a 48 MHz oscillator and is unaffected by the clock mode selected in active mode (8 or 24 MHz). This explains the differences in the current consumption values for the peripherals clocked by PCLK between the 2 CCLK frequencies. Because of this, the user may wish to consider running the program at a slower clock mode if the performance speed of the SSC, T2 or IIC is not a critical factor, to achieve some savings on the power consumption.

Software example to enable all peripherals:

```
SCU_PAGE = 0x01;            //Open SCU page 1 to access PMCON1
PMCON1 |= 0x00;             //Enable ADC, SSC, CCU, T2, MDU, Cordic, LTS and IIC
```

Table 10 Current consumed by XC822 peripherals at different CCLK frequencies with V_{DDP} = 3.3V

Peripheral	Current (mA)	
CPU Clock Frequency (MHz)	**8**	**24**
ADC[1]	2.25[2]	
SSC	0.22	0.58
CCU	3.38[2]	
T2	0.14	0.30
MDU	0.90[2]	
LEDTSCU	0.78[2]	
IIC	0.18	0.42
All	7.48	8.20

1) The internal analog clock of the ADC, f_{ADCI}, can also be reduced to gain some savings on the power consumption. However, do take note that there will be some implications on the performance of the peripheral, which in this case would be the conversion and sample time of the ADC.

2) The ADC, CCU, MDU and LTS are clocked at FPCLK = 48MHz, regardless of the CCLK frequency.

Table 11 Current consumed by XC822 peripherals at different CCLK frequencies with V_{DDP} = 5V

Peripheral	Current (mA)	
CPU Clock Frequency (MHz)	**8**	**24**
ADC[1]	2.23[2]	
SSC	0.21	0.53
CCU	3.36[2]	
T2	0.13	0.26
MDU	0.87[2]	
LEDTSCU	0.75[2]	
IIC	0.16	0.38
All	7.46	8.23

1) The internal analog clock of the ADC, f_{ADCI}, can also be reduced to gain some savings on the power consumption. However, do take note that there will be some implications on the performance of the peripheral, which in this case would be the conversion and sample time of the ADC.

2) The ADC, CCU, MDU and LTS are clocked at FPCLK = 48MHz, regardless of the CCLK frequency.

Table 12 Current consumed by XC836 peripherals at different CCLK frequencies with V_{DDP} = 3.3V

Peripheral	Current (mA)	
CPU Clock Frequency (MHz)	**8**	**24**
ADC[1]	2.89[2]	
SSC	0.31	0.67
CCU	3.45[2]	
T2	0.20	0.36
MDU	0.94[2]	
Cordic	2.06[2]	
LEDTSCU	0.93[2]	
IIC	0.25	0.48
All	10.12	10.90

1) The internal analog clock of the ADC, f_{ADCI}, can also be reduced to gain some savings on the power consumption. However, do take note that there will be some implications on the performance of the peripheral, which in this case would be the conversion and sample time of the ADC.

2) The ADC, CCU, MDU, Cordic and LTS are clocked at FPCLK = 48MHz, regardless of the CCLK frequency.

Table 13 Current consumed by XC836 peripherals at different CCLK frequencies with V_{DDP} = 5V

Peripheral	Current (mA)	
CPU Clock Frequency (MHz)	**8**	**24**
ADC[1]	2.92[2]	
SSC	0.33	0.69
CCU	3.47[2]	
T2	0.21	0.36
MDU	0.95[2]	
Cordic	2.07[2]	
LEDTSCU	0.94[2]	
IIC	0.25	0.48
All	10.16	10.96

1) The internal analog clock of the ADC, f_{ADCI}, can also be reduced to gain some savings on the power consumption. However, do take note that there will be some implications on the performance of the peripheral, which in this case would be the conversion and sample time of the ADC.

2) The ADC, CCU, MDU, Cordic and LTS are clocked at FPCLK = 48MHz, regardless of the CCLK frequency.

5 Power Saving Checklist

The list below is provided to serve as an aid for the user when implementing any of the power saving features.

- Port 2 is a general purpose input-only port. If unused, pins should be disabled by resetting register P2_EN.
- Input port pins should not be left floating. Port pins should be pulled to a defined level (as input or output) instead of being left floating. Output pins that are pulled-up should be set to high or have the pull-up disabled.
- The analog part of the ADC module can be disabled by resetting the bit GLOBCTR.ANON. This feature causes the generation of f_{ADCI} to be stopped and allows a reduction in power consumption when no conversion is needed.
- Use idle mode instead of polling loops (see **Chapter 2.1**).
- Only enable peripherals that are to be used in the application to optimize power consumption.
- When using the power down modes, the wake-up time is recommended to be at least 500µsec, as this is the time required to shut down the modules that are inactive in power down mode. Choosing a wake-up time shorter than the recommended minimum time will not result in any power saving.

6 Summary

- Idle mode and power down mode are the two main power saving modes available in the XC82x and XC83x devices.
- Idle mode reduces the device's power consumption by shutting off the CPU. This is implemented by stopping the clock to the CPU.
- Power down mode reduces the device's power consumption by stopping the clock to the CPU and the peripherals. Only the contents of the FLASH, on-chip RAM, XRAM and the SFRs are maintained.
- XC82x offers two power down modes.
- XC83x offers four power down modes.
- Peripherals can be enabled according to the requirement of the application to optimize the device's power consumption, by programming the assigned bits in PMCON1.

w w w . i n f i n e o n . c o m

Published by Infineon Technologies AG