
AVR1518: XMEGA-A1 Xplained Training - XMEGA Clock System



Prerequisites

- Required knowledge
 - Basic knowledge of microcontrollers and the C programming language
 - Completed AVR1512 XMEGA-A1 Xplained Training – XMEGA Basics
- Software prerequisites
 - Atmel® AVR Studio® 5
- Hardware prerequisites
 - XMEGA-A1 Xplained
 - JTAGICE3 (or JTAGICE mkII or AVR ONE!)
- Estimated completion time:
 - 2 hours

1 Introduction

Atmel AVR® XMEGA® has an advanced clock system, supporting a large number of clock sources. It incorporates both integrated oscillators, and external crystal oscillators and resonators. A high frequency Phase Locked Loop (PLL) and clock prescalers can be used to generate a wide range of clock frequencies. A calibration feature (DFLL) is available, and can be used for automatic run-time calibration of the internal oscillators. A Crystal Oscillator Failure Monitor can be enabled to issue a Non-Maskable Interrupt and switch to internal oscillator if the external oscillator fails. This training will cover the basics, but you will find more details of the XMEGA clock system in Application Note AVR1003.

8-bit Atmel Microcontrollers

Application Note

Rev. 8404A-AVR-07/11



2 Introduction to the XMEGA clock system

The Atmel® AVR® XMEGA® Clock System provides a large portfolio of clock sources, both internal and external. In addition, an internal PLL can be used to multiply selected clock sources with a factor ranging from 1x to 31x.

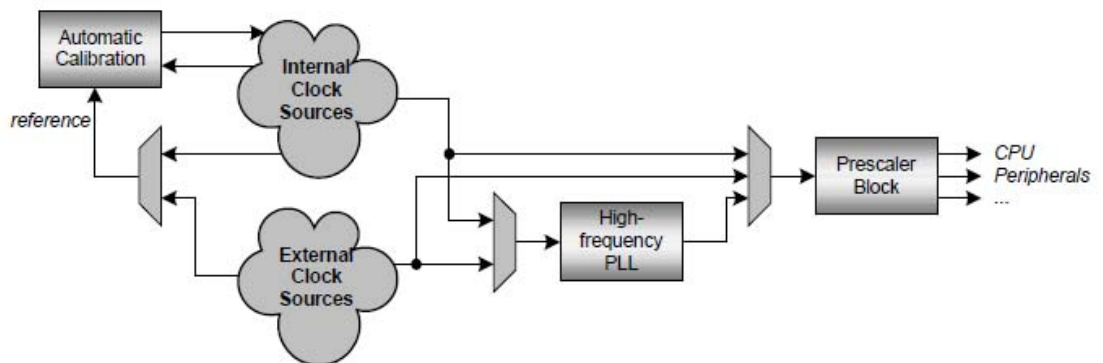
The internal 2MHz and 32MHz oscillators available have hardware support for automatic calibration against a 32kHz clock source with the help of the built-in Digital Frequency Locked Loop (DFLL).

The Atmel XMEGA also has a dedicated Real Time Counter (RTC) which is typically used to keep track of time in more human friendly units like milliseconds, seconds, etc.

In the following chapters we will take a more detailed look into the different parts of the clock system.

2.1 Clock sources

In order to ease implementations, the default clock setting for the XMEGA is to start up running from an internal 2 MHz factory-calibrated source. In this way, if the default settings are sufficient, no external components or software configuration is required to start executing code.

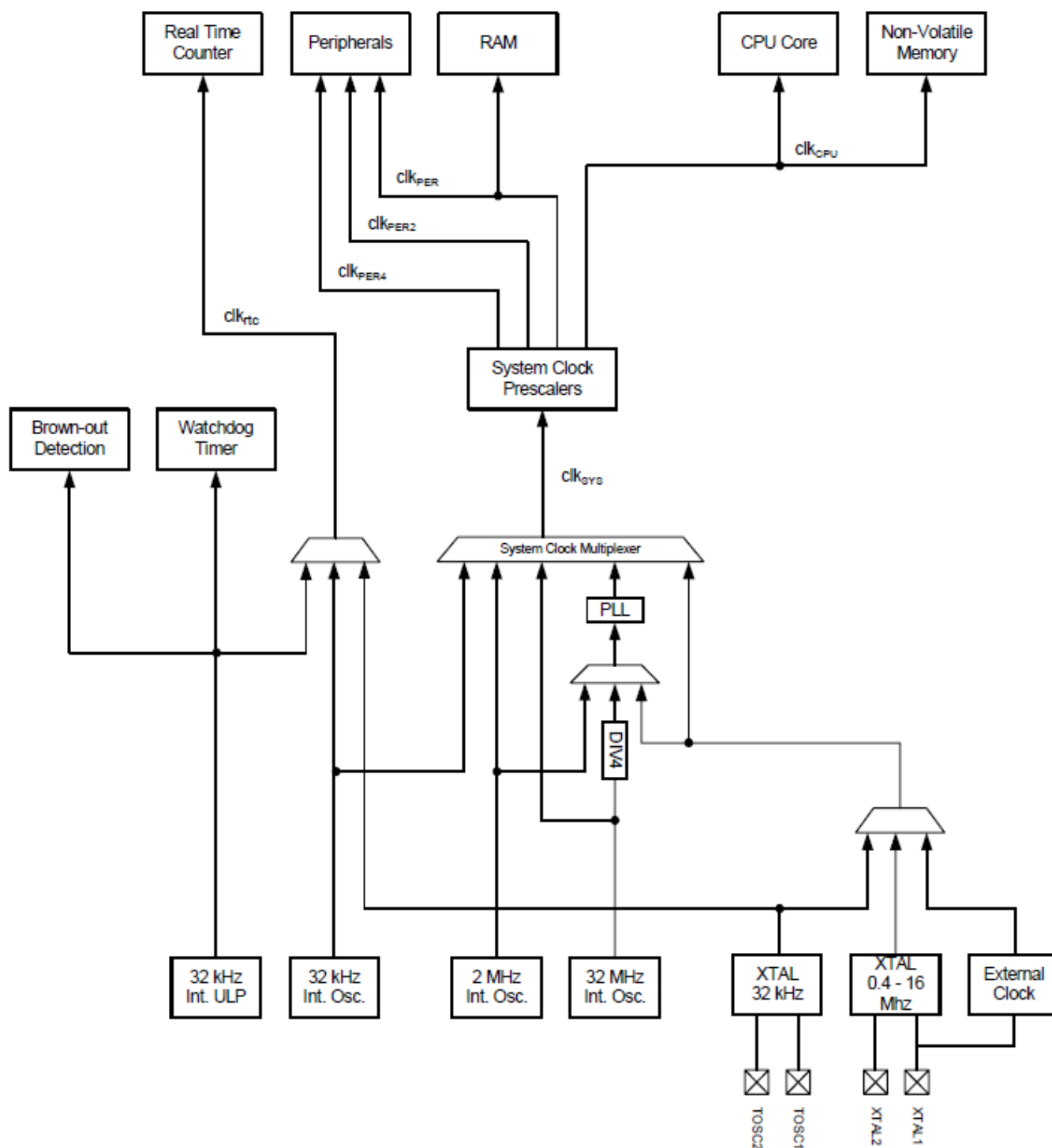


There are five internal clock sources (including the internal PLL), ranging from an ultra low-power (ULP) 32kHz RC oscillator to a 32MHz factory-calibrated ring oscillator with auto-calibration features. All sources except the ULP 32kHz RC oscillator can be used for the main system clock.

Any number of the internal sources can be enabled at any given time, even if none are used for the main system clock. Also, some clock sources might even be used for multiple purposes, such as the 32kHz RC oscillator that can be used as a main system clock and as a clock source for the Real Time Counter module at the same time.

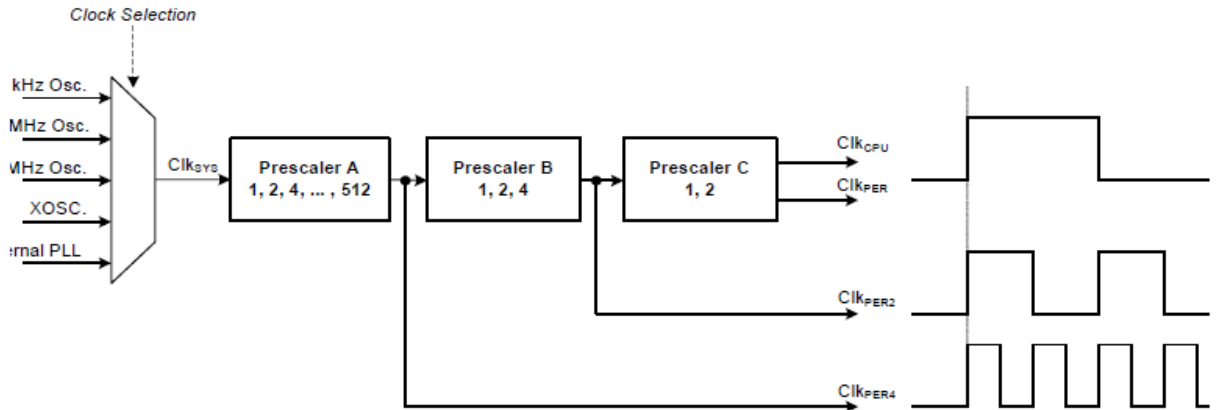
Some of the internal clock sources can be used as a reference to the internal PLL in order to generate even higher frequencies.

The following illustration shows the XMEGA clock system in more detail:



2.2 System clock prescaler

The system clock prescalers as shown in the illustration above can be illustrated in more details as follows:



The A, B and C prescalers gives us even more flexibility in the input clock selection. By using any of the available clock sources together with the prescalers, we are able to run the peripherals and CPU at various frequencies just by reconfiguring the A, B and C prescalers.

2.3 Runtime calibration

The Atmel XMEGA has two built-in Digital Frequency Locked Loops (DFLLs) which can be used to improve the accuracy of the 2MHz and 32MHz internal oscillators. The DFLL compares the oscillator frequency with a more accurate reference clock to do automatic run-time calibration of the oscillator. The choices for the reference clock sources are:

- 32kHz Calibrated Internal Oscillator
- 32kHz Crystal Oscillator connected to the TOSC pins

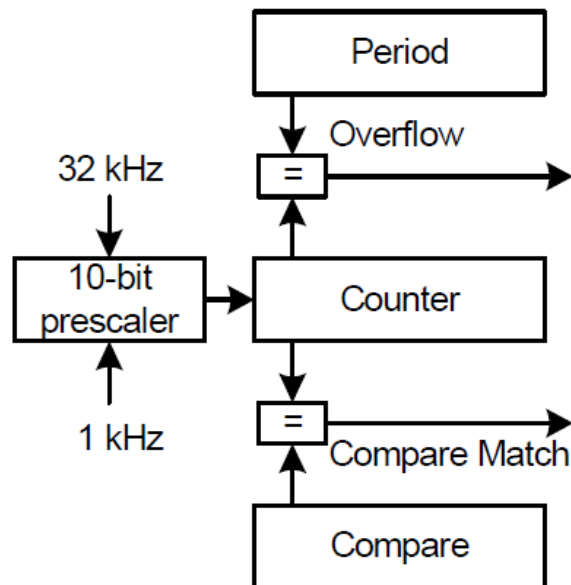
When the DFLL is enabled it will count each oscillator clock cycle, and for each reference clock edge, the counter value is compared to the fixed ideal relationship between the two clocks. If the internal oscillator runs too fast or too slow, the DFLL will decrement or increment the corresponding DFLL Calibration Register value by one to adjust the oscillator frequency slightly.

2.4 Real-time counter

The XMEGA has a dedicated counter to keep track of time in a more human friendly way than regular timers, the Real Time Counter or RTC.

The reference clock for this timer is typically an external high accuracy 32 kHz watch crystal. This timer/counter is clocked asynchronously and is active in several of the low power sleep modes. This allows the microcontroller to keep track of time even when in sleep.

If less accuracy is needed, it is possible to use the internal 32 kHz RC oscillator to clock the RTC instead of an external crystal.



The input clock for the RTC can be the 32kHz clock directly or a 1kHz prescaled version of the 32kHz clock. In addition the RTC has a separate prescaler block in order to choose accuracy over long time-out periods. With a maximum resolution of 30.5µs, time-out periods range up to 2000 seconds. With a resolution of 1 second, the maximum time-out period is over 18 hours (65536 seconds).

The RTC includes both a period and compare register that can be used to generate interrupts and/or events.

2.5 External clock source failure monitor

The Atmel XMEGA has functionality for handling external clock source failures. We will not show this functionality in this training. For more information please refer to the device datasheet and manual.



3 Overview

Here is a short overview of the tasks in this training:

Task 1: Changing system clock

In this task we will show how to change between the different oscillators as the main clock.

Task 2: Using the internal PLL

In this task we will use the internal PLL and show how to change the settings for the PLL. We also look at the internal prescalers that are available.

Task 3: Runtime calibration

In this task we will show you how to enable and disable the automatic runtime calibration of the internal oscillators.

Task 4: Real time counter

In this task we will show you how to configure the RTC and how to keep track of time in milliseconds, seconds, minutes, hours and days.

Good luck!

4 Task 1: Clock switching

Atmel XMEGA has a lot of clock sources. In this task we will show how to dynamically switch between the internal oscillators while the MCU is running.

The goals for this task are that you know how to:

- Use the available clock system drivers
- Enable the available clock sources
- Dynamically switch between the different clock sources



Todo

TASK:

1. Locate the XMEGA-ClockSystem folder, open the xmega_clock_system.avrsrc and set Task1 active by selecting it as StartUp Project.
2. Look through the code and ensure you understand how things are set up
3. Build the project; ensure there are no errors
4. Start the debugging session
5. Use the switches to change the oscillator. Do you observe a difference in the rate of blinks on the LEDs?
6. Reset the debugging and restart the code, and notice the rate in which the LEDs are blinking



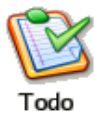
Which oscillator is the default after a reset?

5 Task 2: Using the internal PLL

Atmel XMEGA also has a built-in Phase Locked Loop (PLL) that can be used to generate runtime frequency dynamically. In this task we show you how to configure the XMEGA PLL and how to reconfigure it.

The goal for this task is that you know how to:

- Change the system prescalers
- Enable the PLL and use it as the main clock source
- Modify the PLL parameters



TASK:

1. Locate the XMEGA-ClockSystem folder, open the `xmega_clock_system.avrsln` solution and set Task2 active by selecting it as StartUp Project.
2. Look through the code and ensure you understand how things are set up
3. Build the project; ensure there are no errors
4. Start the debugging session
5. Use the switches to change the oscillator. Do you observe a difference in the rate of blinks on the LEDs?



Why are we able to switch to the 2MHz internal oscillator as main clock without enabling it and waiting for it to be stable as we did in Task 1?

What is the procedure for reconfiguring the running speed of the internal PLL?

Can the system clock prescalers be changed at any time?

6. If you have time: Add code to allow you to generate 62MHz to the `clkper4` and 15.5MHz to the CPU

6 Task 3: Runtime calibration

The Atmel XMEGA has a built-in Digital Frequency Locked Loop (DFLL) which can be used to calibrate the internal 32MHz and/or the 2 MHz RC oscillator. The DFLL can use either an external 32kHz crystal connected to the TOSCx pins, or the internal 32KHz oscillator, as input. When the DFLL is enabled the frequency of the 2 MHz/32 MHz oscillator is constantly being tuned to be in sync with the 32kHz source. The best accuracy of the runtime calibration is achieved by using an external 32kHz crystal.

The goal for this task is that you know how to:

- Enable and disable the runtime calibration for the 2MHz and 32MHz internal oscillator



Todo

TASK:

1. Locate the XMEGA-ClockSystem folder, open the `xmega_clock_system.avrsln` solution and set Task3 active by selecting it as StartUp Project.
2. Look through the code and ensure you understand how things are set up
3. Build the project; ensure there are no errors
4. Start the debugging session
5. Use the switches to enable and disable the oscillator



Do you observe any difference in the rate of blinks on the LEDs?
Why not?



Is it possible to use an external 32kHz crystal as reference for the 2MHz and the internal 32kHz oscillator as a reference for the 32MHz?

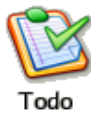
7 Task 4: Real time counter

The Atmel XMEGA A1 includes a 16-bit Real Time Counter (RTC). The RTC can be clocked from an accurate 32.768kHz Crystal Oscillator, the 32.768kHz Calibrated Internal Oscillator, or from the 32kHz Ultra Low Power Internal Oscillator.

In this task we will show you how to use the real time counter to keep track of time similar to a real time clock.

The goal for this task is that you know how to:

- Enable the Real Time Counter (RTC)
- Use the RTC to keep track of time



TASK:

1. Locate the XMEGA-ClockSystem folder, open the `xmega_clock_system.avrsln` solution and set Task4 active by selecting it as StartUp Project.
2. Look through the code and ensure you understand how things are set up
3. Build the project; ensure there are no errors
4. Start the debugging session
5. Holding the switches you can show the current value of the `RTC_ticks`, `RTC_seconds`, `RTC_minutes` and `RTC_hours`



What is the purpose of setting the `RTC.PER` and what value should be set in order to correspond to one second?

8 Summary

In this training we have looked at different aspects of the Atmel XMEGA Clock System. We have shown how to dynamically change the clock, how to use the internal PLL, how to runtime calibrate the internal oscillators and how to use the real time counter. We have also learned how to use the drivers for the clock system from application note AVR1003 to more efficiently start using the XMEGA clock system functionality.

9 Resources

- Atmel XMEGA Manual and Datasheets
 - <http://www.atmel.com/xmega>
- Atmel AVR Studio 5
 - <http://www.atmel.com/avrstudio>
- Atmel IAR Embedded Workbench® compiler
 - <http://www.iar.com/>

10 Atmel Technical Support Center

Atmel has several support channels available:

- | | | |
|---------------|---|----------------------------|
| • Web portal: | http://support.atmel.no/ | All Atmel microcontrollers |
| • Email: | avr@atmel.com | All Atmel AVR products |
| • Email: | avr32@atmel.com | All 32-bit AVR products |

Please register on the web portal to gain access to the following services:

- Access to a rich FAQ database
- Easy submission of technical support requests
- History of all your past support requests
- Register to receive Atmel microcontrollers' newsletters
- Get information about available trainings and training material



Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chou-ku, Tokyo 104-0033
JAPAN
Tel: (+81) 3523-3551
Fax: (+81) 3523-7581

© 2011 Atmel Corporation. All rights reserved.

Atmel®, Atmel logo and combinations thereof, XMEGA®, AVR Studio®, AVR®, AVR® logo and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. **EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.