Atmel

APPLICATION NOTE

Atmel AVR1638: XMEGA RTC Calibration

8-bit Atmel Microcontrollers

Features

- Example software project which performs RTC calibration with help of an external precise clock source
- Software has been implemented as Atmel® Studio ASF example project
- Support for Atmel XMEGA® E devices

Introduction

XMEGA E devices support correction on the RTC clock by taking ppm error value from CALIB register. The CALIB register has to be written by user software after external calibration or temperature correction.

This application note provide example software project which performs RTC calibration with help of an external accurate clock source. The software will calculate the RTC calibration value for connected RTC clock source for current environmental condition. This value will be stored in the device EEPROM location.

Later, user application code, which uses RTC peripheral and intents to perform automatic RTC clock correction, has to read this calibration value from stored EEPROM location and load in CALIB register.

42042A–AVR–11/2012

Table of Contents

1. RTC clock correction

Atmel XMEGA E devices support a feature by which RTC can do internal correction on the RTC crystal clock by taking the ppm error value from the CALIB register. The CALIB register has to be written by user software after external calibration or temperature correction. Correction is done within an interval of approximately 1 million RTC clock cycles. The correction operation is performed as a single cycle operation – adding or removing one cycle, depending on the nature of error. These single cycle operations will be performed repeatedly the error number of times (ERROR [6:0] - CALIB Register) spread throughout the 1 million cycle correction interval. The final correction of the clock will be reflected in the RTC count value available through the CNTL and CNTH registers.

This feature allows user to choose relatively cheaper and not very highly accurate RTC crystal/clock for RTC application. To achieve higher timing accuracy from such RTC, prior to download of user application, the error from such crystal clock has to be calibrated by separate software which will find the calibration value. Later, in user application code, which uses RTC, need to load this calibration value in CALIB register as part of RTC init routine.

In short, XMEGA E devices support only RTC clock correction, but not calibration. Hence, finding the calibration value is the responsibility of the user.

2. RTC calibration

As explained in Chapter 1, finding calibration value is responsibly of user. This application note will help exactly to meet this task with ready to use source code.

The software has been implemented as an example project in Atmel Studio ASF. The ASF project is “RTC Calibration Demo application”.

2.1 Overview of calibration software

- Calibration set-up uses very precise external clock connected at PC4 (more and more precise clock means lesser error in RTC calibration value calculation) as system clock source. With a minimum clock of 2MHz, the RTC can be calibrated to an accuracy of ±0.5ppm
- RTC can use either external clock from TOSC1 or crystal oscillator on TOSC
- By comparing number of RTC clocks with at least 1 million system clock (assumed no error in system clock), RTC calibration value will be calculated
- The calibration value, which can be directly used to load in to RTC-CALIB register, will be stored in configurable EEPROM address
- Calibration status will also be loaded in another configurable EEPROM address
- XMEGA E support maximum correction of 127ppm
- If RTC clock error is more than 127ppm, correction is not possible and calibration status will be failure. Failure is indicated with status value 0x02
- If RTC clock error is less than or equal to 127ppm, correction is possible and calibration status will be success. Success is indicated with status value 0x01

2.2 How calibration works – Calibration Algorithm

This section explains device configuration and algorithm used to find calibration value.

2.2.1 Resources and configuration used in software

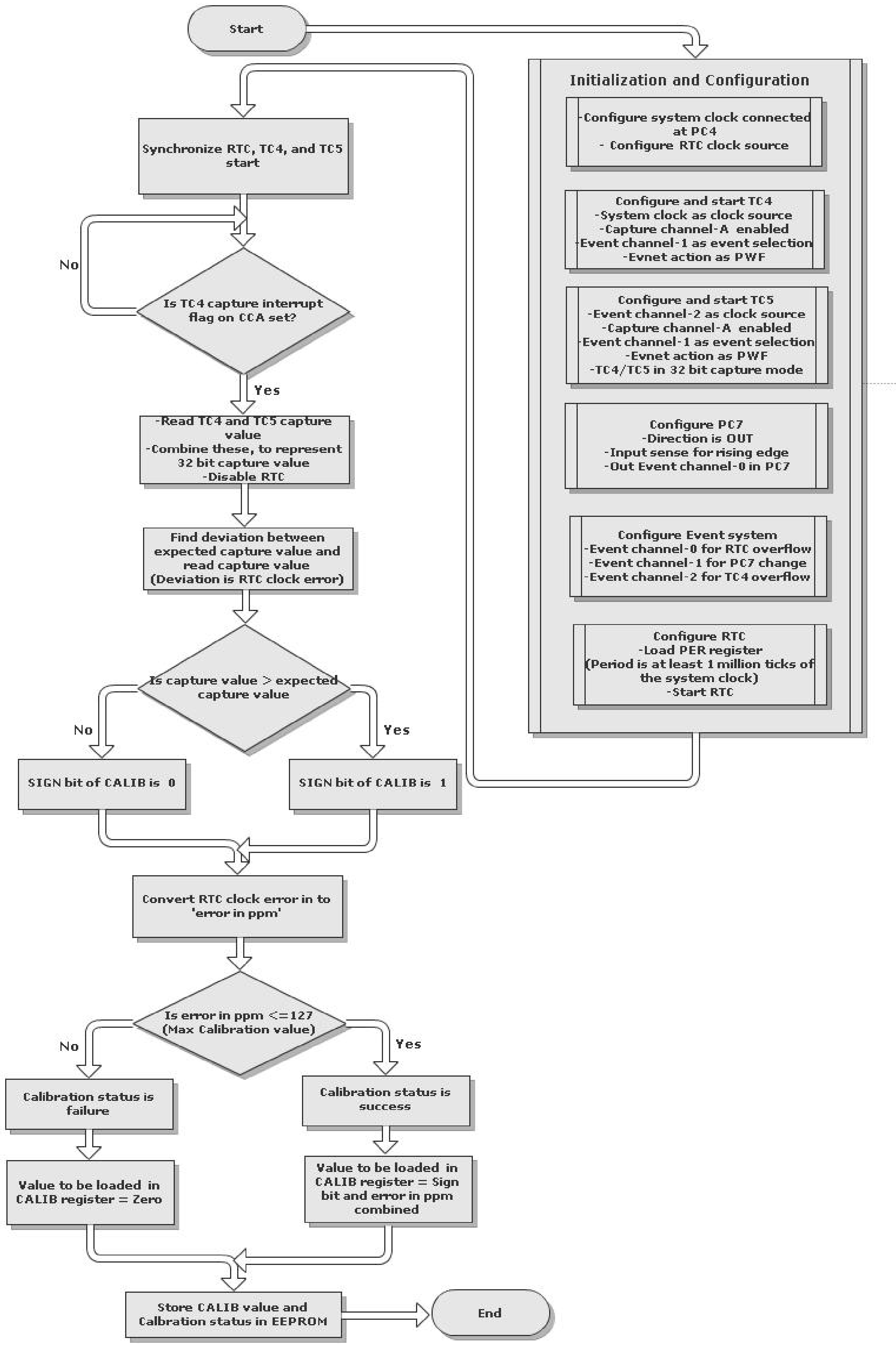
- Precise external clock connected at PC4 used as system clock
- RTC with clock source as either external clock from TOSC1 or crystal oscillator on TOSC
  - Load RTC Period Register with a value so that the duration of the period is equal to at least 1 million periods of the system clock. As an example, with system clock of 2MHz, RTC clock of 32.768kHz and RTC PER register with 0x8000 will generate 2 million system clock periods at RTC overflow. This configuration will lead to ±0.5ppm calibration accuracy
- Timer counter 4 and 5 together works in 32-bit input capture mode by cascading both
- Timer counter 4 used with below configuration
  - In pulse width and frequency capture mode, with event selection as event channel-1
  - Clock source is system clock
  - Capture channel A is enabled
- Timer counter 5 used with below configuration
  - In pulse width and frequency capture mode, with event selection as event channel-1
  - Clock source is event channel-2
  - Event delay is enabled to operate in 32-bit input capture mode
  - Capture channel A is enabled
- Port C, Pin7 (PC7) configuration
  - PC7 direction is output
  - Configure PC7 for input sense in rising edge
  - Configure to out event channel-0 on PC7
- Event system configuration
  - Event Channel-0 is configured for RTC overflow
  - Event Channel-1 is configured Port C, Pin7 (PC7) change
  - Event Channel-2 is configured TC4 overflow

2.2.2 Algorithm

- After configurations made as given above, start both RTC and TC4/5
- Synchronize between RTC start TC4/5 start before getting 32-bit capture value to ensure correct capture value. For this, wait for the TC4 capture flag is set and clear it
- Wait for new capture flag of TC4 and read out both TC4 and TC5 capture value
- Compare this 32-bit capture value with expected value. Any deviation indicates the error in RTC clock
- Convert the error into ppm. That is error per million
- If capture value is more than expected value, it means RTC clock is slower. Hence need to make the RTC clock faster for correction. That is, sign bit of CALIB is 1
- If capture value is less than expected value, it means RTC clock is faster. Hence need to make the RTC clock slower for correction. That is, sign bit of CALIB is 0
- Note that when the required correction is speeding up the clock by skipping cycles (that is, if sign bit of CALIB is 1), it is required to run the RTC prescaler at a minimum setting of DIV2(RTC clock/2)

A flow chart of the RTC calibration algrothm is shown in Figure 2-1.

Figure 2-1. Flow chart of the RTC calibration algorithm.



2.3 RTC calibration configuration in software

Refer the header file ‘conf_rtc_calibration.h’, under src/config folder from RTC calibration demo ASF project.

- From ‘conf_rtc_calibration.h’, make below configurations as per user requirement/setup
  - RTCCALIB_SYSTEM_CLOCK_IN_MHZ – System clock value in MHz
  - RTCCALIB_EEPROM_ADDR_TO_STORE – EEPROM address where calculated calibration value to be stored
  - RTCCALIB_STAT_EEPROM_ADDR – EEPROM address where calibration status to be stored
  - RTCCALIB_RTC_CLK_SRC – RTC clock source

Refer the header file ‘conf_clock.h’, under src/config folder from RTC calibration demo ASF project.

- From ‘conf_clock.h’, make below configuration as per user requirement/setup
  - BOARD_XOSC_HZ – External system clock frequency value

2.4 Quick start guide

Here it is assumed that, working user board based on an Atmel ATxmega E device will be used to run this application note software. If default settings are made, then it is not necessary to change any configuration values.

1. Connect external clock of value 2MHz (default system clock value) at port pin PC4.
2. Connect RTC clock 32.768 kHz at TOSC1 pin (Default RTC clock source). This is the clock to be calibrated.
3. Open Atmel Studio ASF project ‘RTC calibration demo project’ and select proper XMEGA E device. Default device selected is ATxmega32E5.
4. To enable proper debugging, from Atmel studio, Project -> Properties -> AVR®/GNU C Compiler -> Optimization, select “use rjmp/rcall(limited range) on >8K devices”.
5. Build and download the .hex file to user board.
6. After running this .hex file in user board, calibration value will be stored in EEPROM address 0 (0 is default address configuration) and calibration status will be stored in EEPROM address 1 (1 is default address configuration).
7. Here calibration routine is finished and now user application can be downloaded.
8. Ensure that EEPROM content is preserved before downloading user application code.
9. User application code should first read calibration status from stored EEPROM location.
10. If calibration status is success (if status is 0x01), read the calibration value from stored EEPROM location and load in to CALIB register to enable RTC correction.
11. If calibration status is failure (if status is 0x02), it means RTC clock error is more than 127ppm and hence correction is not possible.

3. Revision History

Doc. Rev.	Date	Comments
42042A	11/2012	Initial document release

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 G.K.
16F Shin-Osaki Kangyo Building
1-6-4 Osaki
Shinagawa-ku, Tokyo 141-0032
JAPAN
Tel: (+81)(3) 6417-0300
Fax: (+81)(3) 6417-0370