
Hardware and Software Getting Started

ATmegaS128-STK600

Introduction

This document guides you step by step to setup the hardware and software, to program and run your first code with the ATmegaS128 device.

Requirements

To use this “Getting Started” document you need to gather the following materials:

- One STK600 kit (ATSTK600)
- One routing board (ATSTK600-RC09)
- One ATmegaS128 board with the device soldered on the board. (ATmegaS128-ZC-EK).
- Atmel Studio available from : <http://www.atmel.com/Microsite/atmel-studio>
- “ATmegaS128_led_chaser.zip” software File

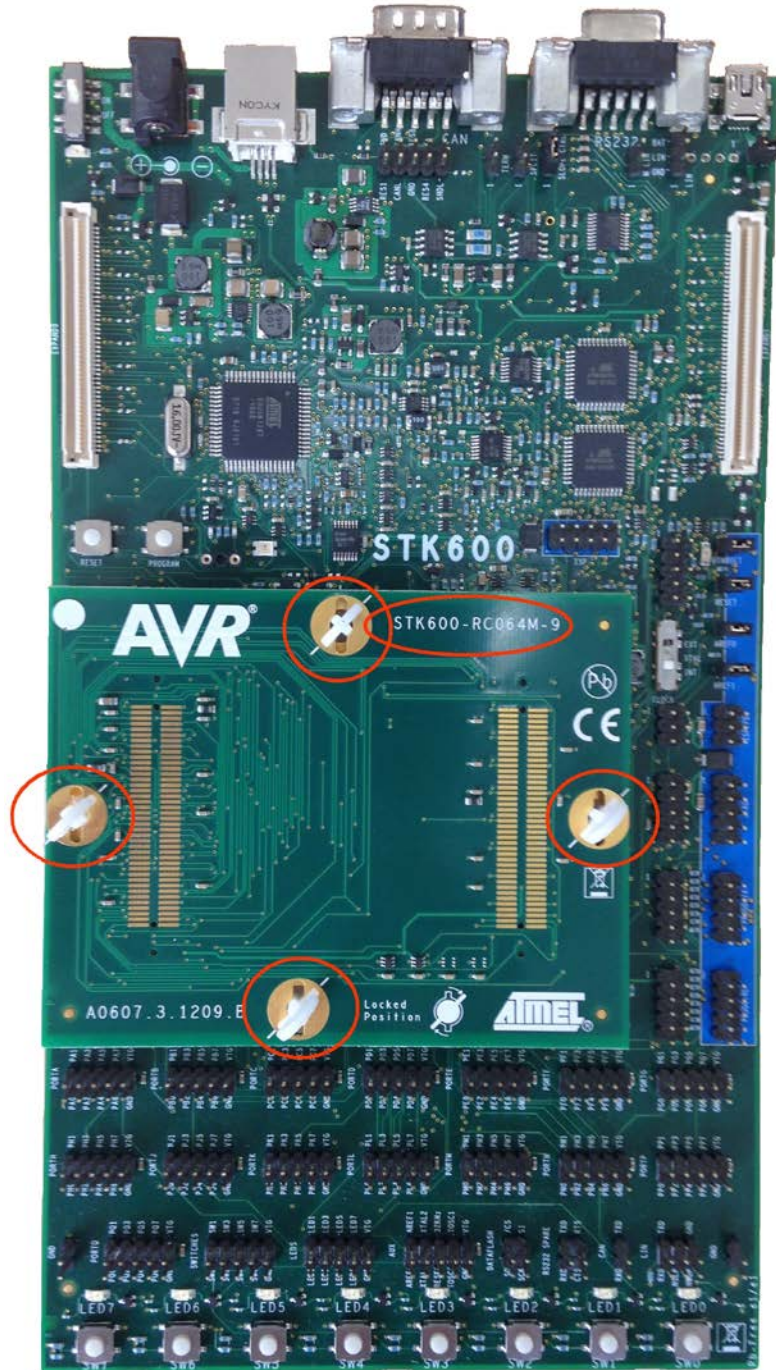
Objective

The aim is prepare the hardware, program and run a short demo based on the “led chaser” software.

1 Hardware Configuration

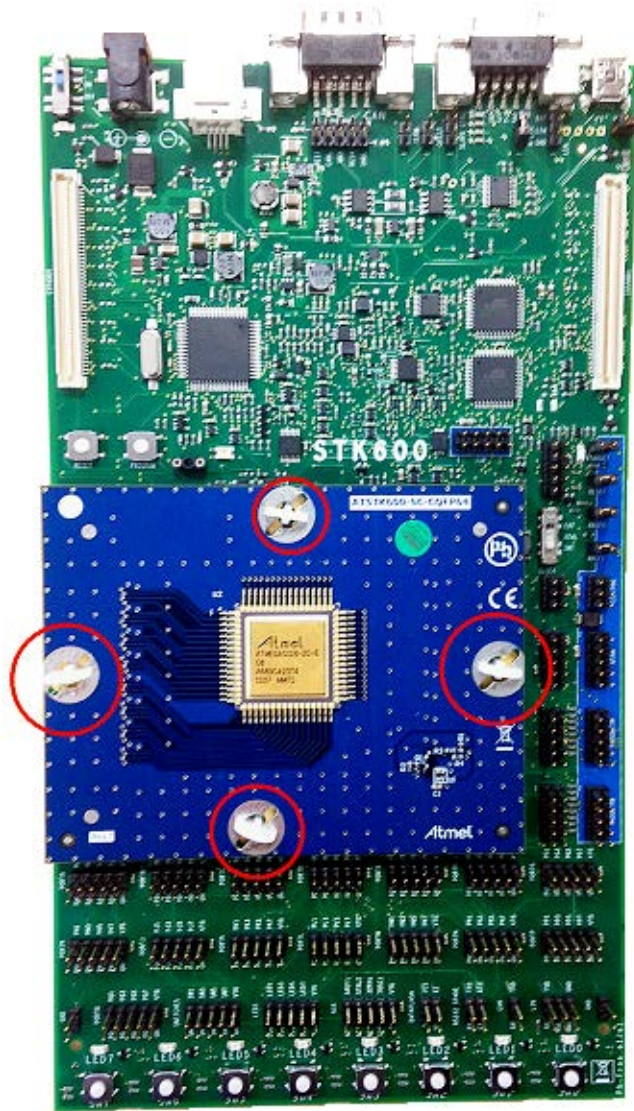
1.1 Step 1.

Unpack the STK600 kit and install the RC064M-9 routing board as shown on the figure below.



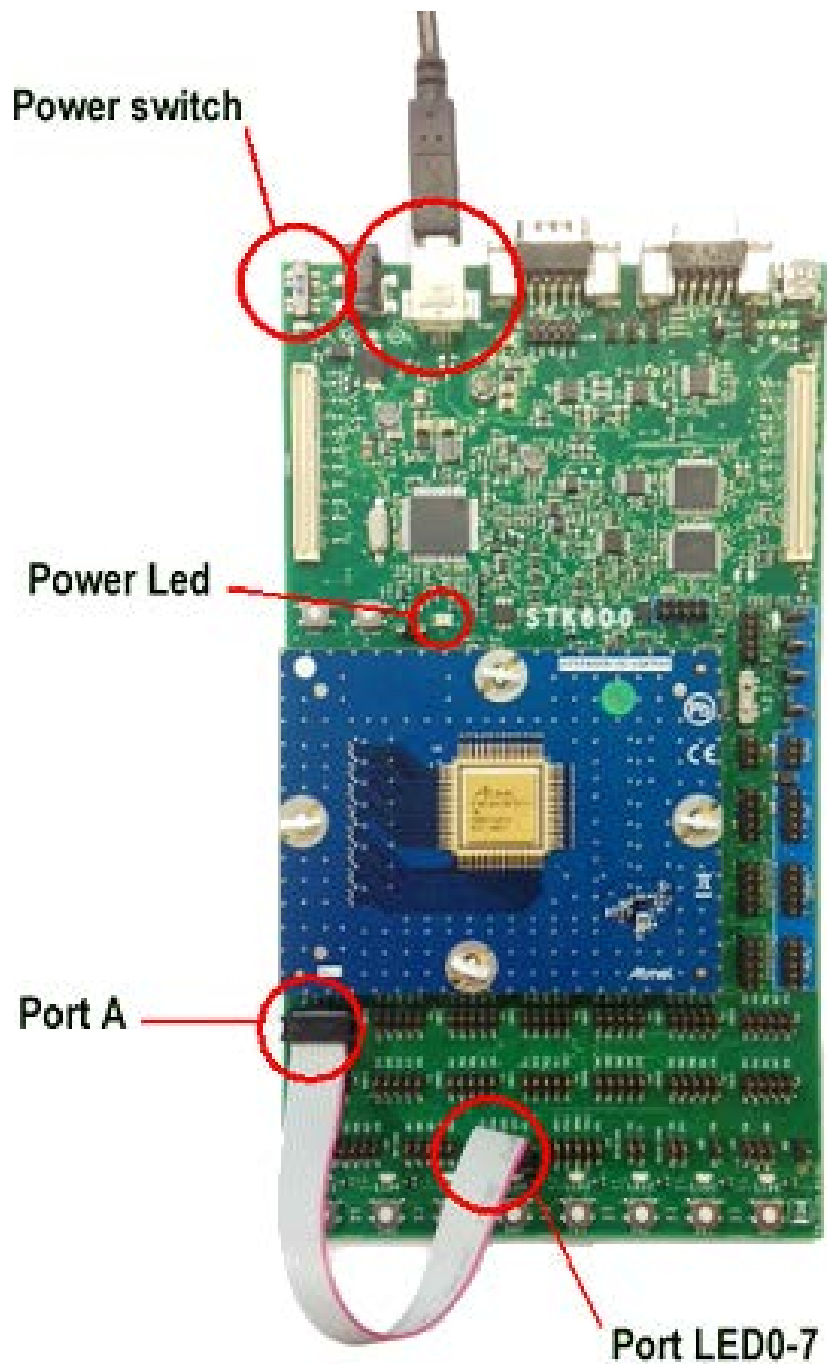
1.2 Step 2.

Install the ATmegaS128-ZC-EK board as shown on the figure below.



1.3 Step 3.

Connect an USB cable between your workstation and the STK600 board and connect an 8-wires flat cable between the ports A(0-7) and LED(0-7) as shown in the figure below. Before connecting the USB cable, set the power switch to “Off” position. Those two cables are supplied in the STK600 kit.



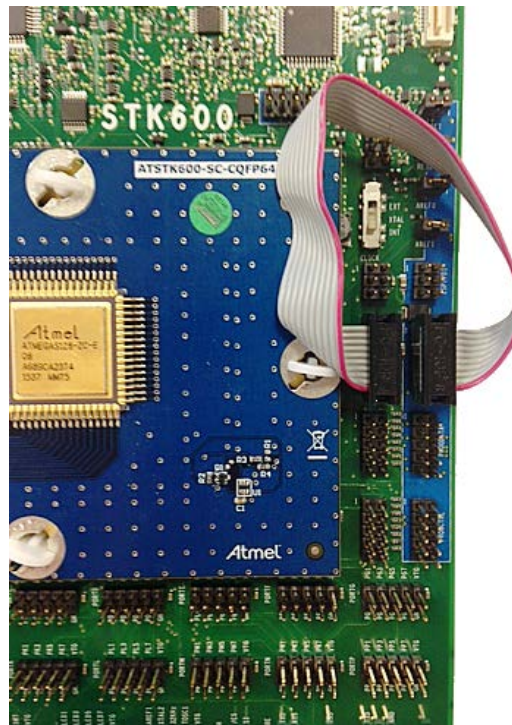
1.4 Step 4.

Verify that the jumpers and the switch are configured as shown on the figure below. The jumpers VTARGET, RESET, AREF0, AREF1 must be set and the clock switch must be set to “INT” position.



1.5 Step 5.

Connect a 10-wires flat cable on the JTAG connector as shown on the figure below. The cable is supplied in the STK600 kit.



Move the power switch to “ON” position. The hardware configuration is now complete. The next step will consist to run the application software.

2 Software Configuration

This section guides you in few steps to prepare your workstation to program and run the “led chaser” demo.

Prerequisite 1.: download Atmel Studio from the URL <http://www.atmel.com/Microsite/atmel-studio> and install it on your workstation.

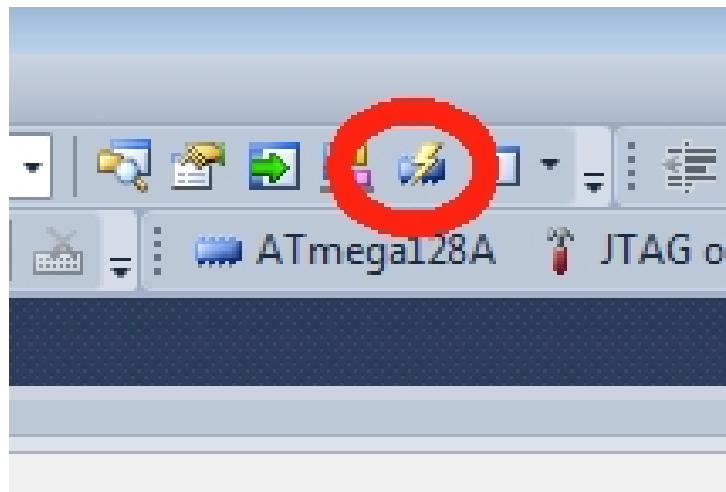
Prerequisite 2.: unzip the “led chaser” demo file.

2.1 Step 6.

Double click on the file “Led_Chaser.atsln”. It makes run Atmel Studio (if a window requesting an update appears, you can close it).

2.2 Step 7.

Click on the programming Icon as shown on the figure below.

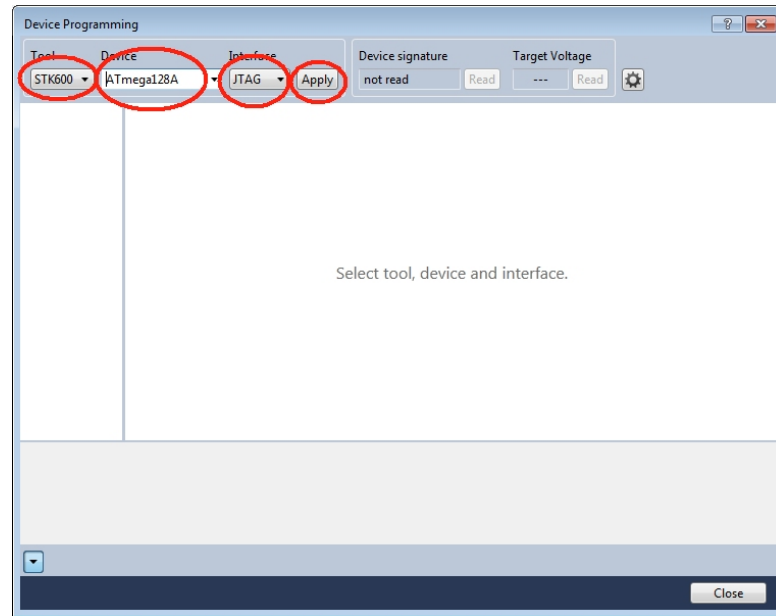


2.3 Step 8.

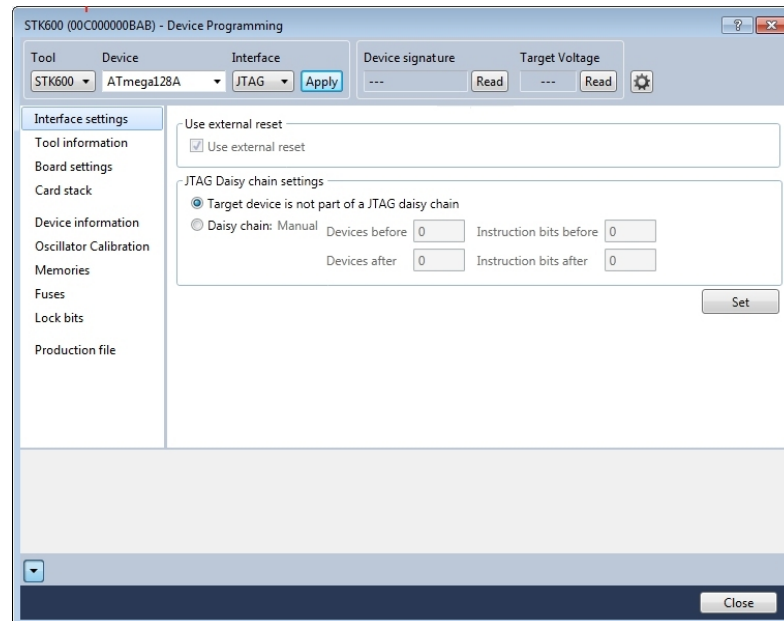
A window should pop as shown on the figure below.

Select STK600 as tools, ATmega128A as device and JTAG as Interface then click on the “Apply” button.

Note: For the development tools, the ATmegaS128, is “seen” as an ATmega128A.

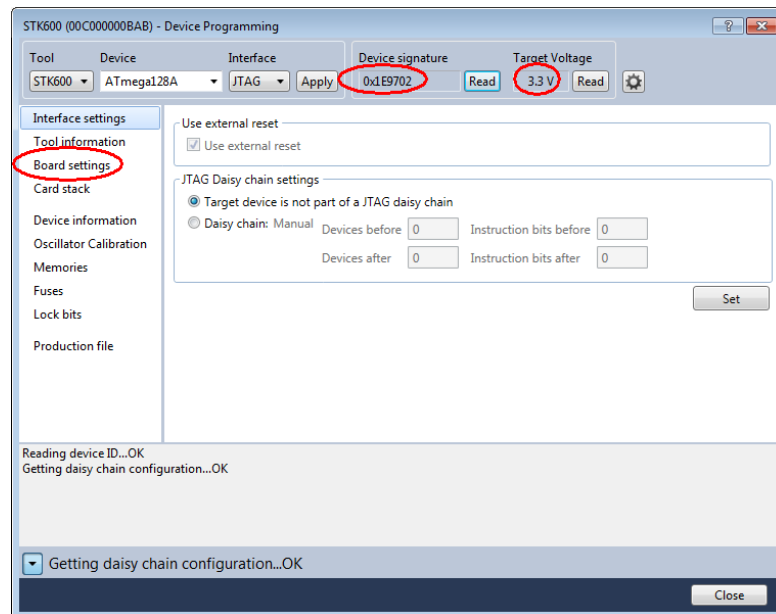


In return, the system should display the information shown on the figure below, meaning that the hardware configuration has been recognized by the system.



2.4 Step 9.

Click on the “Read” button to get the signature bytes of the ATmegaS128.

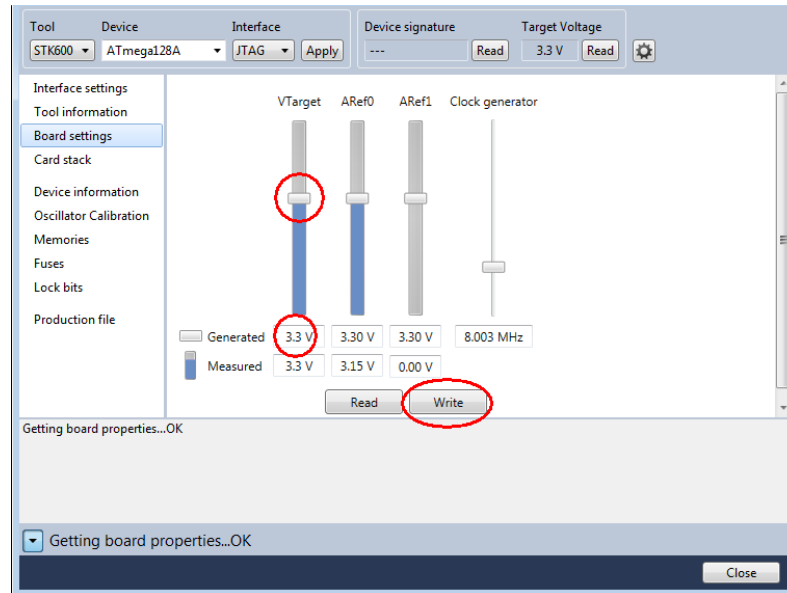


The voltage supply of the ATmegaS128 device can be adjusted from your workstation. If the voltage level is close to zero, it is not possible to read the signature bytes. Therefore the voltage supply must be tuned.

You should read the value 0x1E9702 meaning that the system is able to communicate with the ATmegaS128 device. If an error message comes up, check the device voltage by clicking on the “Read” button. If it is not possible, click on the “board settings” option from the left menu.

Note: The ATmegaS128 is a 3.3V device while the ATmega128A is a 2.7-5.5V device. Therefore, the ATmega128A lets you the choice between 3.3V or 5V. For the ATmegaS128 please set voltage to 3.3V.

Adjust the voltage level by means of the cursor as shown on the figure below then click on the “Write” button. Verify the target voltage and the availability of the signature bytes (0x1E9702).

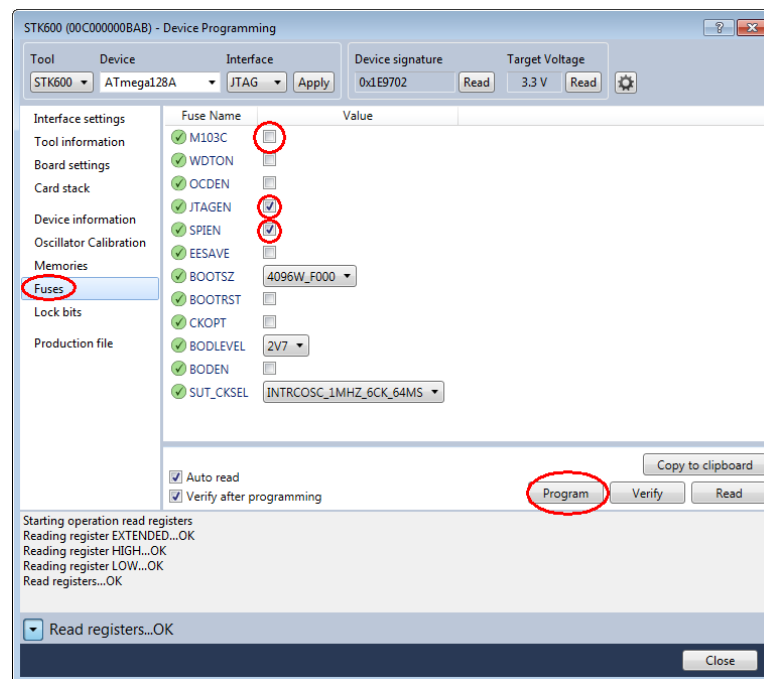


2.5 Step 10. Configuring the fuse bytes of the ATmegaS128.

Click on the “Fuses” option from the left menu.

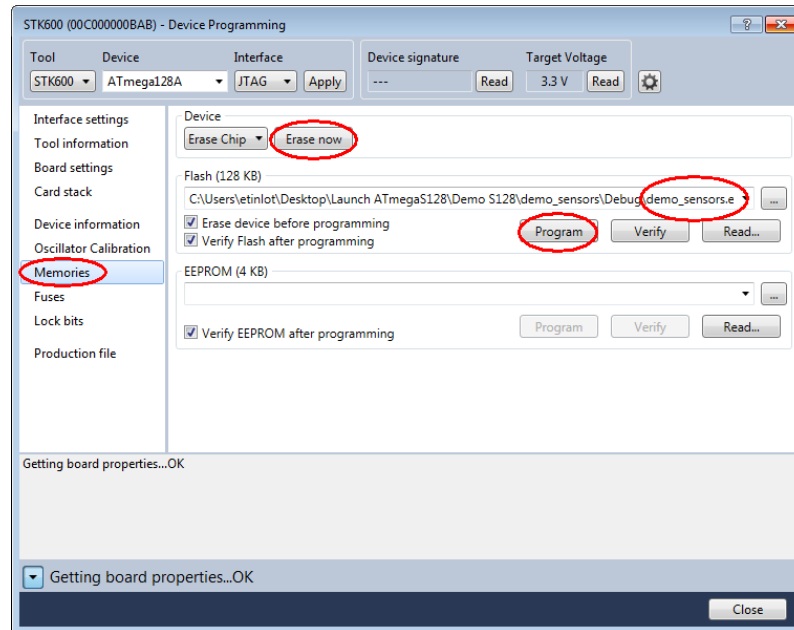
Those fuse bytes enable to configure several device hardware parameters.

Verify that the M103C fuse is unchecked, that the JTAGEN and SPIEN are checked, then click on the “Program” button.



2.6 Step 11. Programming the code.

Click on the “Memories” option from the left menu then click on “Erase now” button to erase the device. Find the file “Led_Chaser.elf” on your workstation from the path \led_chaser\GccBoardProject2\Debug, then click on the “Program” button.



The demo should start after few seconds.

Congratulations. You have now completely set up your hardware and software environment and programmed your first AVR based demo. Enjoy by changing the source code and reprogramming the chip.

3 Revision History

Doc Rev.	Date	Comments
A	11/2015	Initial release.
B	05/2016	Page 1 : corrected an error on the ordering number of the STK600-RC064M-9 routing card



Atmel® | Enabling Unlimited Possibilities®



Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA T: (+1)(408) 441.0311 F: (+1)(408) 436.4200 | www.atmel.com

© 2014 Atmel Corporation. / Rev.:Atmel-41075B-Aero-Hardware and Software Getting Started -05/2016.

Atmel®, Atmel logo and combinations thereof, Enabling Unlimited Possibilities®, and others are registered trademarks or trademarks of Atmel Corporation in U.S. and other countries. ARM®, ARM Connected® logo, and others are the registered trademarks or trademarks of ARM Ltd. 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 products 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.

SAFETY-CRITICAL, MILITARY, AND AUTOMOTIVE APPLICATIONS DISCLAIMER: Atmel products are not designed for and will not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death ("Safety-Critical Applications") without an Atmel officer's specific written consent. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Atmel products are not designed nor intended for use in military or aerospace applications or environments unless specifically designated by Atmel as military-grade. Atmel products are not designed nor intended for use in automotive applications unless specifically designated by Atmel as automotive-grade.