AVR1907: Xplain Hardware User's Guide

8-bit AVR® Microcontrollers

Application Note

Preliminary

Features

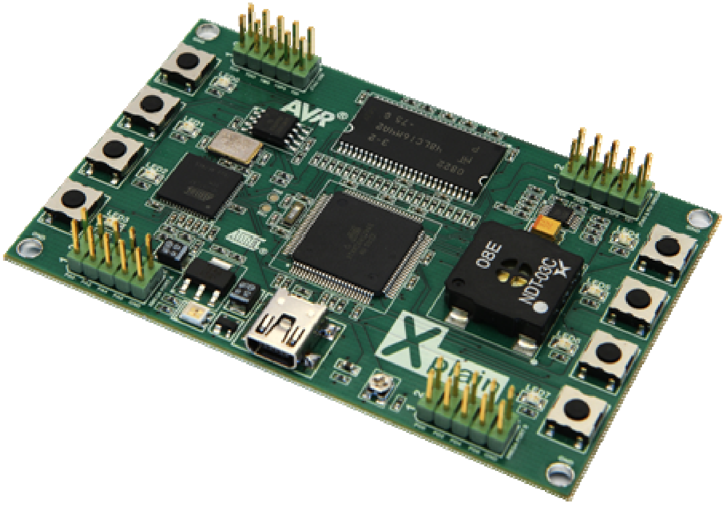
- ATxmega128A1 microcontroller
- External Memory
  - 8MB SDRAM
  - 8MB Serial Data Flash
- AT90USB1287
  - Communication gateway
  - Programmer for XMEGA
- Analog Input (to ADC)
  - Temperature sensor
  - Potentiometer
- Analog Output (from DAC)
  - Mono speaker via audio amplifier
- Digital IO
  - UART communication through USB gateway
  - 8 micro switch buttons
  - 8 LEDs
  - 8 spare analog pins
  - 8 spare digital pins

1 Introduction

The Xplain evaluation kit is a hardware platform to evaluate the ATxmega128A1.

The kit offers a larger range of features that enables the XMEGA™ user to get started using the XMEGA's peripherals right away and to get an understanding of how to integrate the XMEGA in their own design.

Figure 1-1. Xplain evaluation kit.



Rev. 8203C-AVR-12/09

2 Related Items

FLIP (Flexible In-System Programmer)
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3886

AVR Studio 4 (Atmel's free IDE)
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725

JTAGICE mkII (On-chip programming and debugging tool)
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3353

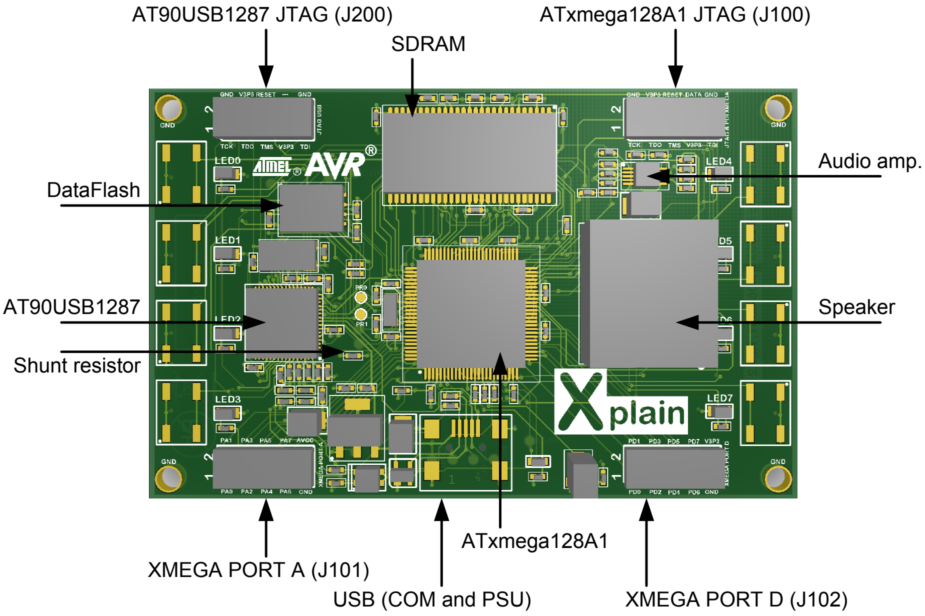
AVR ONE! (On-chip programming and debugging tool)
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=4279

3 General information

This document targets the Xplain evaluation kit revision 3 and parts of the document may therefore be inconsistent with earlier revisions of the product. For earlier revision please refer to the schematics, which is this only documentation available for these revision.

The Xplain kit is intended to demonstrate the ATxmega128A1, and the hardware that relates to the AT90USB1287 is therefore not covered in this document.

Figure 3-1. Overview of the Xplain kit.



3.1 Preprogrammed firmware

The Xplain kit comes with both the ATxmega128A1 and the AT90USB1287 preprogrammed.

The preprogrammed firmware in the XMEGA plays different sounds when the micro switch buttons are pushed.

The preprogrammed AT90USB1287 firmware offers features such as a bootloader for self-programming and a UART-to-USB gateway.

3.2 Power supply

The kit is powered from USB which leaves two options to power it: Either connects the kit to a PC through an USB cable, or to a 5V USB power supply (AC/DC adapter).

3.3 Measuring the XMEGA power consumption

As part of an evaluation of the ATxmega128A1 it can be of interest to measure its power consumption. The R105 resistor (0 Ohm) is the only connection point between the XMEGA power plane (the V_XM net) and common 3.3V supply of the board (the V3P3 net). By replacing the R105 shunt resistor with a higher value resistor it is possible to determine the current consumption of the ATxmega128A1.

3.4 Programming the XMEGA through the USB gateway

Programming of the ATxmega128A1 through the USB is not supported in the preliminary release.

3.5 Communication through UART-to-USB gateway

The XMEGA's UARTC0 is connected to a software UART on the AT90USB1287. The AT90USB1287 UART is communicating at 9600 baud, using one start bit, eight data bits, one stop bit and no parity.

When the AT90USB1287 device is enumerated (connected to a PC) the data transmitted from the XMEGA is passed on to a (virtual) COM port. This means that it is possible to use a terminal program to receive the transmitted data on a PC. Similarly data transmitted from the PC COM port is passed on to the XMEGA UART through the gateway.

If the USB device is not enumerated, e.g. powering the kit from an USB supply adapter, the AT90USB1287 device's UART will operate in loop back mode (echo back everything it receives).

4 Connectors

The Xplain kit has four 10-pin 100mill headers. Two are used for programming the ATxmega128A1 and AT90USB1287, and two are to access spare analog and digital pins on the XMEGA (expansion headers).

4.1 Programming headers

The XMEGA can be programmed and debugged by connecting an external programming/debugging tool to the "JTAG & PDI XMEGA" pin header (J100). The pin header is having a standard JTAG programmer pinout (refer to online help in AVR Studio®), and tools like the JTAGICE mkII or the AVR ONE! can thus be connected directly to the header. If it is desired to use PDI programming/debugging an adapter must be used.

Table 4-1. XMEGA programming and debugging interface – JTAG and PDI.

J100 pin	JTAG[1]	PDI[2]
J100-1	TCK	-
J100-2	GND	GND[3]
J100-3	TDO	-
J100-4	VCC	VCC[3]
J100-5	TMS	-
J100-6	nSRST	CLK
J100-7	VCC	VCC[3]
J100-8	-	DATA
J100-9	TDI	-
J100-10	GND	GND[3]

Notes:
1. Standard pin-out for JTAGICE mkII and other Atmel programming tools.
2. Requires adapter to connect a JTAGICE mkII (refer to AVR Studio help).
3. It is only required to connect on VCC/GND pin.

The AT90USB1287 can be programmed through its bootloader. The bootloader is evoked by shorting the (PORTF4) pin to GND before applying power to the board. This can be done by shorting pin 1 and 2 on the “JTAG USB” pin header (J200) with a jumper. Programming is performed through the “FLIP” plug-in in AVR Studio (can be started as a stand-alone application also).

FLIP (FLexible In-system Programmer) is a free Atmel’s proprietary software that runs Windows 9x/Me/NT/2000/XP and Linux x86. FLIP supports in-system programming of Flash devices through RS232, USB or CAN.

Alternatively, the AT90USB1287 can be also programmed by connecting a programming tool, such as a JTAGICE mkII, to the “JTAG USB” pin header (J200).

Note that it is not recommended to program the AT90USB1287 using a programming tool as this will erase the bootloader.

4.2 IO expansion headers

The XMEGA’s analog PORTA is available on the “XMEGA PORT A” pin header (J101). This allows the user to connect external signals to the ADC, DAC and Analog Comparators on PORTA.

The XMEGA’s digital PORTD is available on the “XMEGA PORT D” pin header (J102). This port features general purpose IO and various communication modules (USART, SPI and TWI).

Note that the communication modules on PORTD can be interconnected to test out various functions and features: The USART can loop back communication with a jumper, or communicate between the two UARTs on the PORT. The native SPI and the USART in SPI master mode can be connected, and the TWI module can be enabled in both master and slave mode at the same time to get loop back behavior (pull-up resistors required).

5 Attached Memories

The Xplain kit demonstrates how to use the External Bus Interface (EBI) module to interface a 4-bit SDRAM. An 8MB SDRAM (16M x 4) is attached in 3-port EBI mode (PORTH, PORTK and PORTJ).

An 8MB serial Dataflash® is connected to the UARTC1 which can operate in SPI master mode to interface the dataflash.

6 Miscellaneous IO

6.1 Micro switch buttons

Eight micro switch buttons are connected to the XMEGA's PORTF. Internal pull-ups should be enabled to detect when the buttons are pushed as they short the respective line to GND.

6.2 LEDs

Eight LEDs are connected to the XMEGA's PORTE. The LEDs are active low – and thus lights up when the respective lines are drawn low by the XMEGA.

6.3 Analog IO

An NTC and an adjustable resistor divider (potentiometer) are connected to PORTB on PINB0 and PINB1 respectively. These analog references can be used as input to the ADC.

An audio amplifier (and mono speaker) is connected to PORTB on PINB2. This pin is connected to the XMEGA's DAC and thus offers a way to generate sound.

7 Available code examples and drivers

A Getting-Started training for the Xplain kit can be downloaded from the Atmel web site. This training is a general introduction to XMEGA peripherals.

Further information and drivers for XMEGA can be downloaded as application notes, also distributed from Atmel's web site.

8 Quick start guide

8.1 Powering the Xplain board

Power the Xplain kit by connecting it to a PC host or a USB power adapter. The Xplain USB connector is the mini-B type used for most portable products today.

If connecting the Xplain kit to a PC, it is required to install the Xplain USB driver the first time it is connected. Please find the install file in the /software/USB driver/ folder of the zip file distributed with this application note (http://www.atmel.com/avr/). Select the driver corresponding to the Xplain/AT90USB1287 firmware revision used. It is possible to replace the AT90USB1287 firmware of a revision 1 kit with later firmware, in that case it is required to uninstall the USB driver and install the new driver.

8.2 Using the preprogrammed firmware

The preprogrammed firmware in the ATxmega128A1 runs a demo application where sounds/tunes are played when the buttons are pushed.

The preprogrammed firmware of the AT90USB1287 is composed of two distinct sections: the bootloader and the application.

The bootloader provides a simple way for the user to update the application section of the AT90USB1287 with the only need of FLIP (a free PC application) and a USB cable.

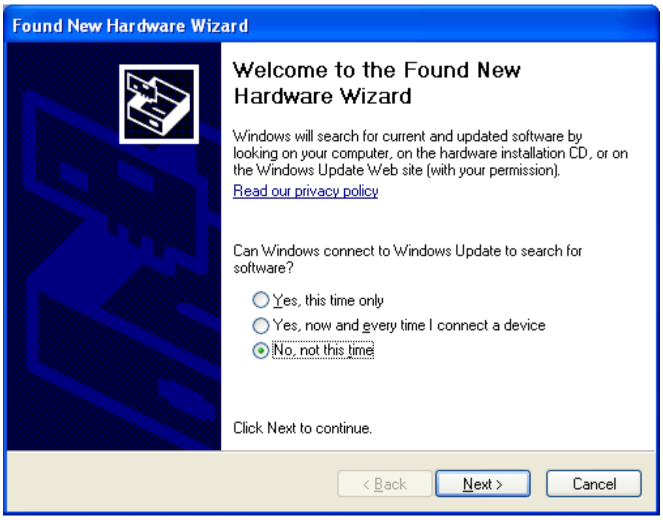
The preprogrammed application is a USB to serial port bridge. When the USB is connected to a PC host and correctly enumerated, the AT90USB1287 will operate as a communication gateway, and pass on data between the UART and the USB (COM port). When powered from a USB power adapter (and therefore not enumerated) the AT90USB1287 will echo back data received on the UART.

8.3 Reprogramming the AT90USB1287 application firmware

This guide requires that both AVR Studio and FLIP are installed. It is a walkthrough which describes how to download the USB application firmware. To use FLIP as a stand alone application, please refer to the application note: “AVR282: USB Firmware Upgrade for AT90USB”**.**

1. Disconnect the USB power source.
2. Unplug the JTAGICE mkII and place a jumper between pins 1 and 2 on the “JTAG USB” pin header (J200).
3. Connect the USB cable to a computer.
4. The first time this is done, the bootloader driver will have to be installed. A “Found new hardware wizard” will pop up. Select “No, not this time” and next as shown bellow.

Figure 8-1. Found new hardware



Found New Hardware Wizard

Welcome to the Found New Hardware Wizard

Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).

Read our privacy policy

Can Windows connect to Windows Update to search for software?

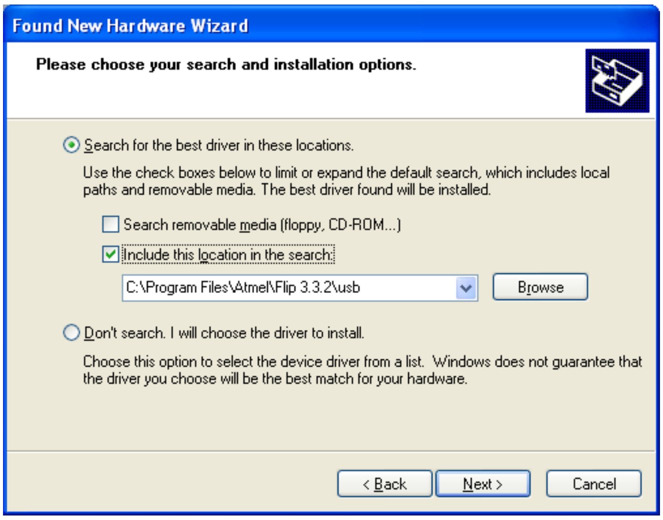
○ Yes, this time only
○ Yes, now and every time I connect a device
● No, not this time

Click Next to continue.

5. Select “Install from a list or specific location (Advanced). Click next.

6. Select “Include this location in the search:”. Select the FLIP installation folder and the usb folder as shown bellow. Click next.

Figure 8-2. Driver path selection



Found New Hardware Wizard

Please choose your search and installation options.

● Search for the best driver in these locations.

Use the check boxes below to limit or expand the default search, which includes local paths and removable media. The best driver found will be installed.

□ Search removable media (floppy, CD-ROM...)

☑ Include this location in the search:

C:\Program Files\Atmel\Flip 3.3.2\usb

○ Don't search. I will choose the driver to install.

Choose this option to select the device driver from a list. Windows does not guarantee that the driver you choose will be the best match for your hardware.

7. Click finish.

8. Open AVR Studio and select Tools -> Flip X.X.X.

9. Select Device -> Select... -> AT90USB1287 and click “OK”

10. Select Settings -> Communication -> USB and click “Open”.

11. Under “Operations Flow” check out “Erase, “Program” and “Verify”.

12. Choose the hex file to be programmed: select File -> Load hex file. Choose the USB application firmware hex file and click “OK”.

13. Press “Run”.

The device is now programmed with the serial gateway application firmware. The same procedure can be used to download your own application code. Remove the jumper at the “JTAG USB” pin header (J200) and cycle power to start the application code.

8.4 Reprogramming the ATxmega128A1 firmware

This guide requires that AVR Studio 4 is installed. It is a walkthrough which describes how to download the original firmware using a JTAGICE mkII with AVR Studio 4.16.

1. Make sure the Xplain board is powered (please refer to the “8.1 Powering the Xplain board” section).

2. Connect the JTAGICE mkII to the “JTAG & PDI XMEGA” pin header (J100).

3. Open the programming dialog in AVR Studio selecting Tools -> Program AVR -> Connect. Select “JTAGICE mkII” and Auto and click “Connect”.

4. Select the “Main” tab and choose the “ATxmega128A1” from the dropdown list. Choose “JTAG mode” as programming mode.

5. Select the “Program” tab and choose the XMEGA hex file provided with the application note as Flash input hex file (/firmware/revision_XXX/ATxmega128A1/ATxmega128A1.hex). Click Program.

(Programming of the ATxmega128A1 through the USB bootloader is expected to be available late Q2 2009).

8.5 Reprogramming the AT90USB1287 firmware

This guide requires that AVR Studio is installed. It is a walkthrough which describes how to download the bootloader, the application as well as setting the correct fuse settings. The firmware is downloaded with the JTAGICE mkII and AVR Studio 4.

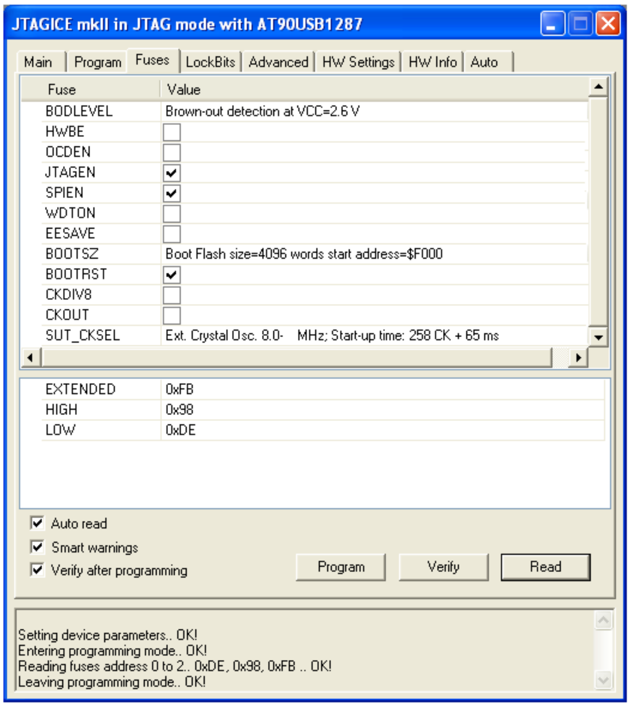
Downloading the firmware:

1. Make sure the Xplain board is powered (please refer to the “8.1 Powering the Xplain board” section).
2. Connect the JTAGICE mkII to the “JTAG USB” pin header (J200).
3. Open the programming dialog in AVR Studio selecting Tools -> Program AVR -> Connect. Select “JTAGICE mkII” and Auto and click “Connect”.
4. Select the “Main” tab and choose the “AT90USB1287” from the dropdown list. Choose “JTAG mode” as programming mode.
5. Select the “Program” tab and choose the USB bootloader hex file provided with the application note as Flash input hex file (/firmware/revision_XXX/AT90USB1287/ AT90USB1287.hex). Click Program.

Selecting the correct fuse settings:

1. Select the “Fuses” tab in the programming dialog. The fuses should be set as shown bellow:

Figure 8-3. USB fuse settings



Fuse	Value
BODLEVEL	Brown-out detection at VCC=2.6 V
HWBE	□
OCDEN	□
JTAGEN	☑
SPIEN	☑
WDTON	□
EESAVE	□
BOOTSZ	Boot Flash size=4096 words start address=$F000
BOOTRST	☑
CKDIV8	□
CKOUT	□
SUT_CKSEL	Ext. Crystal Osc. 8.0- MHz; Start-up time: 258 CK + 65 ms

EXTENDED	0xFB
HIGH	0x98
LOW	0xDE

☑ Auto read
☑ Smart warnings
☑ Verify after programming

```
Setting device parameters.. OK!
Entering programming mode.. OK!
Reading fuses address 0 to 2.. 0xDE, 0x98, 0xFB .. OK!
Leaving programming mode.. OK!
```

9 Revision history

Different revisions of the Xplain kit is released, these revisions are not fully identical it can therefore be relevant to identify the board used.

9.1 Revision 1

Revision 1 of the Xplain kit can be identified by **not** having a bar-code sticker on the back side.

Table 9-1. Xplain revision 1 – known issues.

Issue	Cause	Solution
Speaker is noisy during programming and debugging of the ATxmega128A1 when using JTAG.	Problem due to cross-talk contaminating the audio amplifier input.	Use the PDI interface or remove resistor R305.
AT90USB1287 cannot connect to FLIP programming software to reprogram the AT90USB1287 application section of the flash.	No bootloader programmed in the AT90USB1287	Upgrade to the AT90USB1287 with the latest firmware using an external programming tool such as the JTAG ICE mkII.

9.2 Revision 2

Revision 2 of the Xplain kit can be identified by a bar-code sticker on the back side with the following product ID: A09-0560/2.

Table 9-2. Difference between Xplain revision 1 and revision 2.

Difference	Comment
Dataflash changed to 45DB642D-CNU	The 45DB642D-CNU is replacing the 45DB041D-SU.

Difference	Comment
AT90USB1287 bootloader	Bootloader is entered by shorting pin 1 and 2 on the J200.
R305 removed	To eliminate noise on speaker during JTAG programming and debugging of the ATxmega128A1.
Two jumpers added – default placed on the J101 and J102.	For convenience of the end-user.

9.3 Revision 3

Revision 3 of the Xplain kit can be identified by a bar-code sticker on the back side with the following product ID: A09-0560/3.

Table 9-3. Difference between Xplain revision 2 and revision 3.

Difference	Comment
Series resistors on SDRAM lines removed	Not required – and therefore removed.
Shunt resistor (R105) added	This is a zero ohm resistor that enables measurement of the ATxmega128A1 current consumption.
Audio amplifier is default off (R300 is pulling down the SD line of the U300).	Reduce power consumption and reduce faint noise from speaker.

9.4 Revision 4

Revision 4 of the Xplain kit can be identified by a bar-code sticker on the back side with the following product ID: A09-0560/4.

Table 9-4. Difference between Xplain revision 3 and revision 4.

Difference	Comment
Replaced resistor R400 with a 0 ohm resistor.	Reduce the noise on the power line but disable the Xplain board current measurement.

Headquarters

__Atmel Corporation__
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

International

__Atmel Asia__
Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
Tel: (852) 2245-6100
Fax: (852) 2722-1369

__Atmel Europe__
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
Tel: (33) 1-30-60-70-00
Fax: (33) 1-30-60-71-11

__Atmel Japan__
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Product Contact

__Web Site__
http://www.atmel.com/

__Technical Support__
avr@atmel.com

__Sales Contact__
www.atmel.com/contacts

__Literature Request__
www.atmel.com/literature

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 ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, 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 OF 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's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.