AN2466

Using Atmel-ICE for AVR® Programming In Mass Production

Introduction

For mass production, it is recommended to use dedicated production programmers that are designed to operate in production environments and which have software intended for volume production. The application note "AT06015: Production Programming of Microchip AVR® and SAM microcontrollers" lists 3rd party production programming tools for both Microchip AVR and SAM microcontrollers.

The Atmel-ICE is a powerful development tool for debugging and programming AVR and ARM® Cortex®-M-based SAM microcontrollers. Though the Atmel-ICE is not intended for production programming, some users choose to use it for small-scale production programming because it is very affordable, and because it is already well known from the development phase of the product. To speed up the development of the user's production programming tool, this application note gives a detailed introduction to the Atmel-ICE setup, driver installation, user guide of the Atmel-Studio command line utility (atprogram.exe), and a PC programming tool project written in C#. The Visual C# example project can be downloaded along with this application note and run with the ATmega328PB Xplained Mini kit.

Features

- Atmel-ICE introduction
- "atprogram" utility user guide
- Example mass production programming tool development

Table of Contents

1. Atmel-ICE Introduction

Atmel-ICE is a powerful developing and programming tool for the ARM Cortex-M-based SAM and AVR microcontrollers. It supports:

- Programming (TPI) of all tinyAVR® 8-bit microcontrollers with support for this interface
- Programming and on-chip debugging of all AVR XMEGA® family devices on both JTAG and PDI 2-wire interfaces
- Programming and on-chip debugging of all AVR UC3 microcontrollers on both JTAG and aWire interfaces
- Programming (JTAG and SPI) and debugging of all AVR 8-bit microcontrollers on both JTAG and debugWIRE interfaces
- Programming and debugging of all SAM ARM Cortex-M-based microcontrollers on both SWD and JTAG interfaces

1.1 Atmel-ICE Features

- Fully compatible with Atmel Studio
- Supports programming and debugging of all AVR UC3 32-bit microcontrollers
- Supports programming and debugging of all AVR XMEGA 8-bit devices
- Supports programming and debugging of all megaAVR® and tinyAVR 8-bit devices with OCD
- Supports programming and debugging of all SAM ARM Cortex-M-based microcontrollers
- Target operating voltage range of 1.62 to 5.5V
- Draws less than 3 mA from target VTref when using debugWIRE interface and less than 1 mA for all other interfaces
- Supports JTAG clock frequencies from 32 kHz to 7.5 MHz
- Supports PDI clock frequencies from 32 kHz to 7.5 MHz
- Supports debugWIRE baud rates from 4 kbit/s to 0.5 Mbit/s
- Supports aWire baud rates from 7.5 kbit/s to 7 Mbit/s
- Supports SPI clock frequencies from 8 kHz to 5 MHz
- Supports SWD clock frequencies from 32 kHz to 10 MHz
- USB 2.0 high-speed host interface
- ITM serial trace capture up to 3 MB/s
- Supports DGI SPI and USART interfaces when not debugging or programming
- Supports 10-pin 50-mil JTAG connector with both AVR and Cortex pin-outs. The standard probe cable supports AVR 6-pin ISP/PDI/TPI 100-mil headers as well as 10-pin 50-mil. An adapter is available to support 6-pin 50-mil, 10-pin 100-mil, and 20-pin 100-mil headers. Several kit options are available with different cabling and adapters.

1.2 Atmel-ICE System Requirements

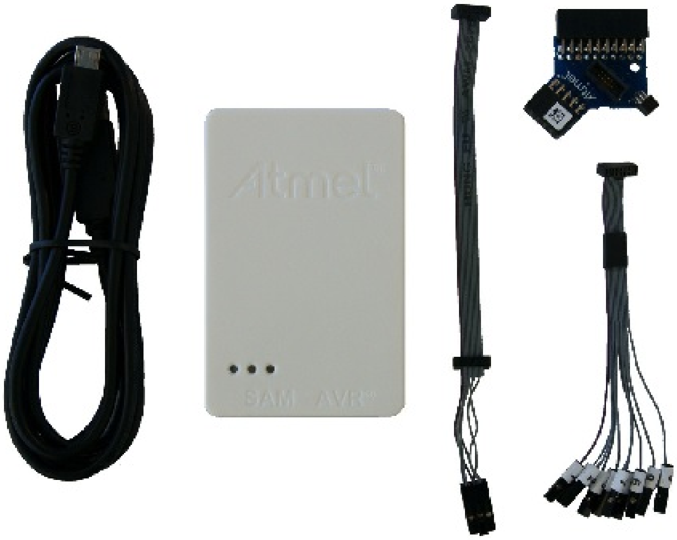
The Atmel-ICE unit requires that the front-end debugging environment Atmel Studio version 6.2 or later is installed on your computer and needs to connect to the host computer using the USB cable provided, or a certified Micro-USB cable.

1.3 Atmel-ICE Kit Contents

The Atmel-ICE full kit contains these items (also shown in the figure below):

- One Atmel-ICE unit
- One USB cable (1.8m, high-speed, Micro-B)
- One adapter board containing 50-mil AVR, 100-mil AVR/SAM, and 100-mil 20-pin SAM adapters
- One IDC flat cable with 10-pin 50-mil connector and 6-pin 100-mil connector
- One 50-mil 10-pin mini squid cable with 10 x 100-mil sockets

Figure 1-1. Atmel-ICE Full Kit Contents

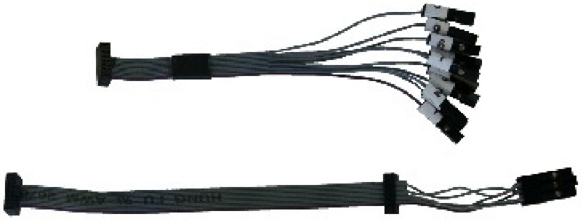


1.4 Assemble Atmel-ICE

There are two kinds of cables, which can be used to connect the Atmel-ICE kit to AVR or SAM chip programming interface pins, as shown in the figure below, provided in the full kit:

- One 50-mil 10-pin IDC flat cable with 6-pin ISP and 10-pin connectors
- One 50-mil 10-pin mini squid cable with 10 x 100-mil sockets

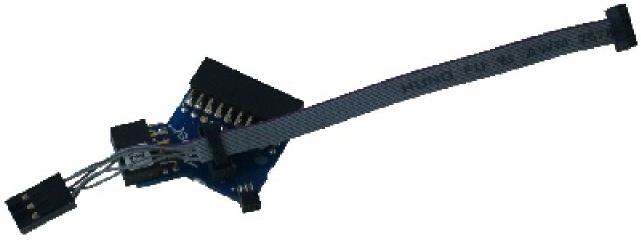
Figure 1-2. Atmel-ICE Cables



For most purposes, the 50-mil 10-pin IDC flat cable can be used, connecting either natively to its 10-pin or 6-pin connectors, or connecting via the adapter board, as shown in the figure below. Three adapters are provided on one small PCBA. The following adapters are included:

- One 100-mil 10-pin JTAG/SWD adapter
- One 100-mil 20-pin SAM JTAG/SWD adapter
- One 50-mil 6-pin SPI/debugWIRE/PDI/aWire adapter

Figure 1-3. Atmel-ICE Adapter



To assemble your Atmel-ICE into its default configuration, connect the 10-pin 50-mil IDC cable to the unit, as shown in the figures below. Be sure to orient the cable so that the red wire (pin 1) on the cable aligns with the triangular indicator on the blue belt of the enclosure. The cable should connect upwards from the unit. Be sure to connect to the port corresponding to the pinout of your target, AVR or SAM.

Figure 1-4. Atmel-ICE AVR Probe Connection

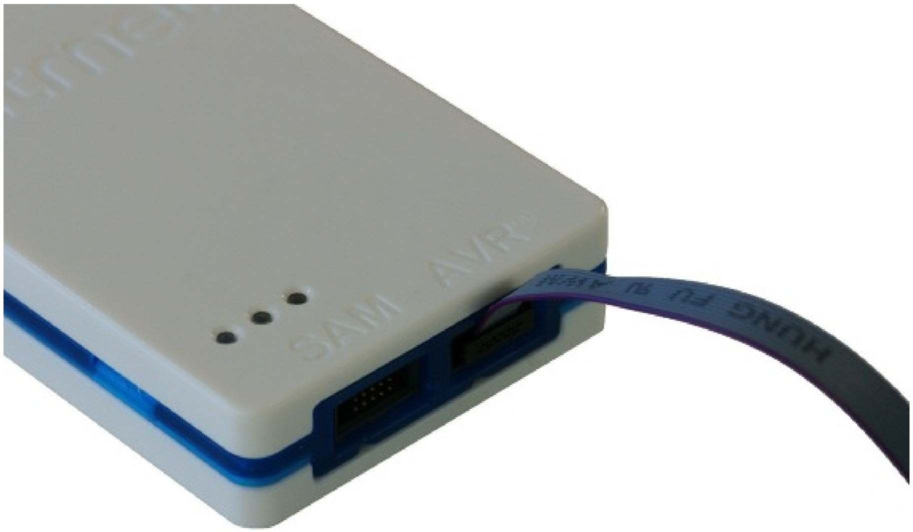
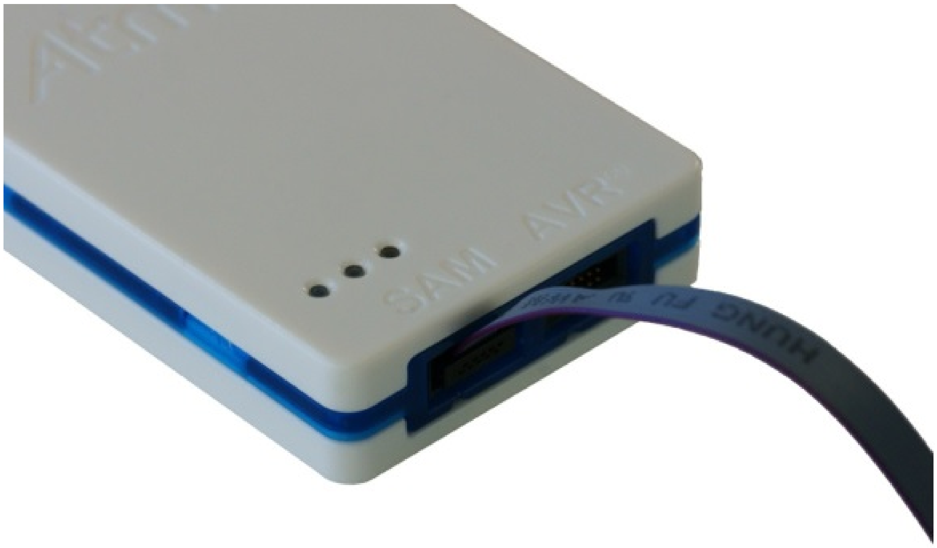


Figure 1-5. Atmel-ICE SAM Probe Connection



1.5 Connecting to the Host Computer and USB Driver Installation

The Atmel-ICE communicates primarily using a standard HID interface and does not require a special driver on the host computer. To use the advanced Data Gateway functionality of the Atmel-ICE, be sure to install the USB driver on the host computer. This is automatically done when installing the front-end software, which is provided free by Microchip. See http://www.microchip.com/ for further information and to download the latest front-end software.

The Atmel-ICE must be connected to an available USB port on the host computer by using the USB cable provided, or a suitable and certified Micro-USB cable. The Atmel-ICE contains a USB 2.0 compliant controller and can operate in both full-speed and high-speed modes. For best results, connect the Atmel-ICE directly to a USB 2.0 compliant high-speed hub on the host computer using the cable provided.

The Atmel-ICE is powered by the USB bus voltage. It requires less than 100 mA to operate, and can, therefore, be powered by a USB hub. The power LED will illuminate when the unit is plugged in. When not connected in an active programming or debugging session, the unit will enter low-power consumption mode to preserve your computer's battery. The Atmel-ICE cannot be powered down. It should be unplugged when not in use.

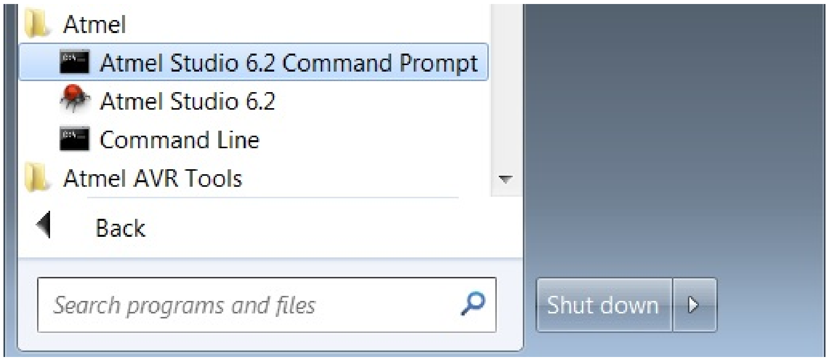
1.6 Software Integration in Atmel Studio

Atmel Studio is an Integrated Development Environment (IDE) for programming and debugging AVR and SAM applications with JTAG, aWire, SPI, PDI, TPI, and SWD interface selected in Windows® environments.

For AVR or SAM devices programmed with Atmel-ICE, use the "Device programming" option integrated in the Atmel Studio IDE or the command line utility called atprogram provided by Atmel Studio. During the Atmel Studio installation a shortcut called Atmel Studio 7.0 (version number depending on the specific Atmel Studio version installed) Command Prompt was created in the Atmel folder on the Start menu. By double clicking this shortcut a command prompt will be opened and programming commands can be entered. The command line utility is installed in the Atmel Studio installation path in the folder Atmel/Atmel Studio 7.0/atbackend.

2. Atprogram Utility User Guide

During the Atmel Studio installation, a shortcut called Atmel Studio xx (xx is the Atmel Studio version number) Command Prompt is created in the Atmel folder on the Start menu, as shown in the figure below. The users can run the atprogram.exe directly by clicking the shortcut or calling the atprogram from other applications to perform the device programming function, but unfortunately, we haven't a standalone installer for the "atprogram.exe" and the users must, therefore, install Atmel Studio first before using it.



With atprogram.exe, it can be used to:

- Program a .bin, .hex, or .elf file to the device
- Verify that programming was correct
- Read, write, and erase the device memories
- Program fuses, lock bits, security bits, user page, and user signature
- Program a production file to a device
- List out all connected tools
- Set interface and interface clock speed

In the later sections, the command usage of the atprogram utility is introduced, including commands of chip erasing, image file programming, and fuse lockbit set. To know more about atprogram utility, type the command: atprogram --help.

2.1 Atprogram Application Usage

For atprogram application usage, refer to the following description:

Usage: atprogram [options] <command> [arguments] [<command> [arguments] ...]

Options:

- -t --tool <arg>: Tool name: Atmel-ICE, AVR Dragon, AVR ISP mkII, AVR ONE, JTAG ICE3, JTAG ICE mkII, QT600, STK500, STK600, SAM-ICE™, EDBG, MEDBG, Power Debugger, Mega DFU or FLIP.
- -s --serialnumber <arg>: The programmer/debugger serial number. Must be specified when more than one debugger is connected.
- -i --interface <arg>: Physical interface: aWire, DebugWIRE, HVPP, HVSP, ISP, JTAG, PDI, UPDI, TPI, or SWD.
- -d --device <arg>: Device name. E.g. ATxmega128A1 or AT32UC3A0256.
- -l --logfile <arg>: Log file for atpbackend output.

- -f --force: Force command even if the firmware is not up to date.
- -cl --clock <arg>: The frequency to use for communication with a device (Hz, kHz, MHz, default Hz). E.g. -cl 10 MHz.
- -mb --max-baudrate <arg>: The maximum baud rate for communication on aWire (bps).
- -xr --external reset: Apply external reset when starting a session.
- -tv --target-voltage <arg>: Set the STK600, STK500, or Power debugger target voltage (float value).
- -a0 --aref0 <arg>: Set the STK600 AREF0 or STK500 AREF generator voltage (float value).
- -a1 --aref1 <arg>: Set the STK600 Aref1 generator voltage (float value).
- -cg --clock-generator <arg>: Set the STK600 or STK500 clock generator frequency (Hz, kHz, MHz, default Hz); --timeout <arg>: Set the timeout value in seconds for commands. The default is 180 seconds. Set to 0 for no timeout.
- -? --help: Display help information.
- -V --version: Display version information.

Commands:

- calibrate: Performs the oscillator calibration procedure.
- chiperase: Full erase of chip.
- dwdisable: Disable DebugWIRE interface.
- erase: Erase the specified memory.
- help: Display help for a specific command.
- info: Display information about a device.
- program: Program device with data from <file>.
- read: Read the contents of the memory on the device.
- reset: Reset all domains and jump to the reset vector.
- secure: Set the security bit on UC3 and ARM devices.
- selftest: Performs the self-test procedure on Atmel-ICE.
- verify: Verify content of memory based on a file.
- write: Write to the memory with values entered on the command line.

Arguments or more command specific help:

- Use atprogram help <command> to get more command usage information.

Example:

- atprogram -t atmelice -i ISP -d ATmega328PB program -f file.elf
  Program file.elf image to ATmega328PB flash with Atmel-ICE tool, ISP interface.
- atprogram -t atmelice -i ISP -d ATmega328PB chiperase
  Full erase ATmega328PB chip with Atmel-ICE tool, ISP interface.

2.2 Chiperase Command Usage

Usage: atprogram [options] chiperase.

Information: Full erase of the chip.

Note: Different behavior on different device architectures, see the data sheet for your architecture.

Options: Execute the atprogram without arguments to list available options.

Examples:

- atprogram -t samice -i jtag -d atsam3s4c chiperase
  Perform chiperase on ATSAM3S4C.
- atprogram -t atmelice-i ISP -d atmega328pb chiperase
  Perform chiperase on ATmega328PB.

2.3 Program Command Usage

Usage: atprogram [options] program <arguments>

Options: Execute the atprogram without arguments to list the available options.

Arguments:

-fl --flash: Program flash address space. tinyAVR, megaAVR, and AVR XMEGA only.

-ee --eeprom: Program EEPROM address space.

-us --usersignature: Program user signature.

-up --userpage: Program user page.

-fs --fuses: Program fuses.

-lb --lockbits: Program lockbits.

-f --file (file): File to be programmed. Intel® .hex, .elf, or binary.

-o --offset (offset): Input file contents will be written to this offset. Default offset is 0. Only valid for binary file format.

--format (format): Specify the file format. Supported input formats are 'elf', 'hex', and 'bin'.

-c --chiperase: Perform a chip erase before programming.

-e --erase: Erase only affected pages before programming. AVR UC3 and AVR XMEGA only. SAM devices will always do this.

--verify: Verify memory after programming.

Examples:

- atprogram -t atmelice -i jtag -d atmega2560 program -c -fl -f source.elf
  Perform chip erase and program only the segments of source.elf that maps to the flash.
- atprogram -t atmelice -i ISP -d atmega328pb program -f d:\GCCBoard1.hex --verify
  Program firmware image file d:\GCCBoard1.hex to the ATmega328PB chip and verify after programming.

2.4 Memory Writing Command Usage

Usage: atprogram [options] write <arguments>.

Information: Write to the memory with values entered on the command line. The values provided will be written to all selected address spaces. At least one address space must be provided.

Options: Execute the atprogram without arguments to list the available options.

Arguments: -fl --flash: Write to flash. tinyAVR, megaAVR, and AVR XMEGA only.

-ee --eeprom: Write to EEPROM. tinyAVR, megaAVR, and AVR XMEGA only.

-us --usersignature: Write to user signature.

-fs --fuses: Write to fuses.

-lb --lockbits: Write to lockbits.

--values (value): Hex encoded values to write, e.g.: 0102040A0F.

-o --offset (offset): Values are written from this offset.

-v --verify: Verify memory contents after write.

Examples:

- atprogram -t atmelice -i ISP -d atmega328pb write -lb --values c0
  Write value 0xc0 to ATmega328PB chip fuse lockbits.

3. Program Software Example

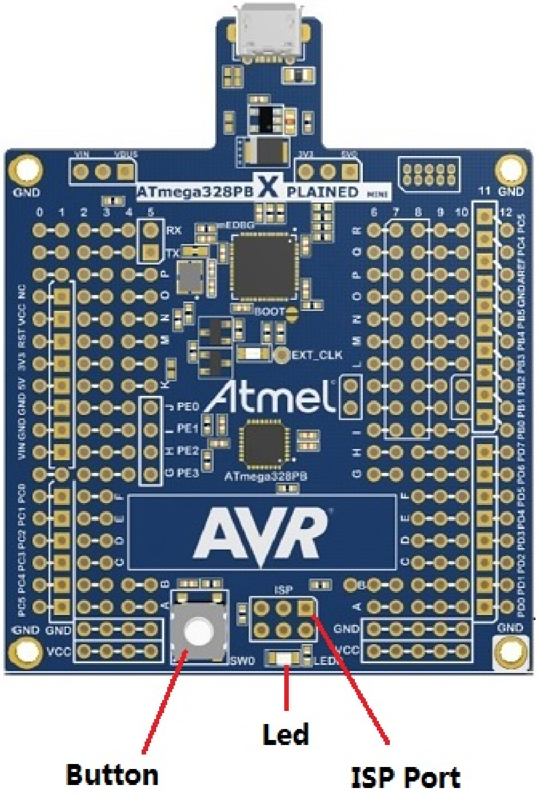
Together with this application note there is a .zip-file with a C# example project that shows how to develop programming software based on the Atmel Studio command line utility "atprogram". The example project source code can be downloaded along with this application note and run with the ATmega328PB Xplained Mini kit.

3.1 Prerequisites

The programming tool example discussed in this document requires:

- Atmel Studio 7.0 or later
- The example AVR_massproduction_programming_tool_with_atmelice.zip file downloaded from the Microchip website
- .Net framework 4, which can be downloaded from the Microsoft® website
- Microsoft Visual C# 2010 Express or later
- 1 set of Atmel-ICE full kit
- 1 set of ATmega328PB Xplained Mini kit, as shown in the figure below

Figure 3-1. ATmega328PB Xplained Mini Kit



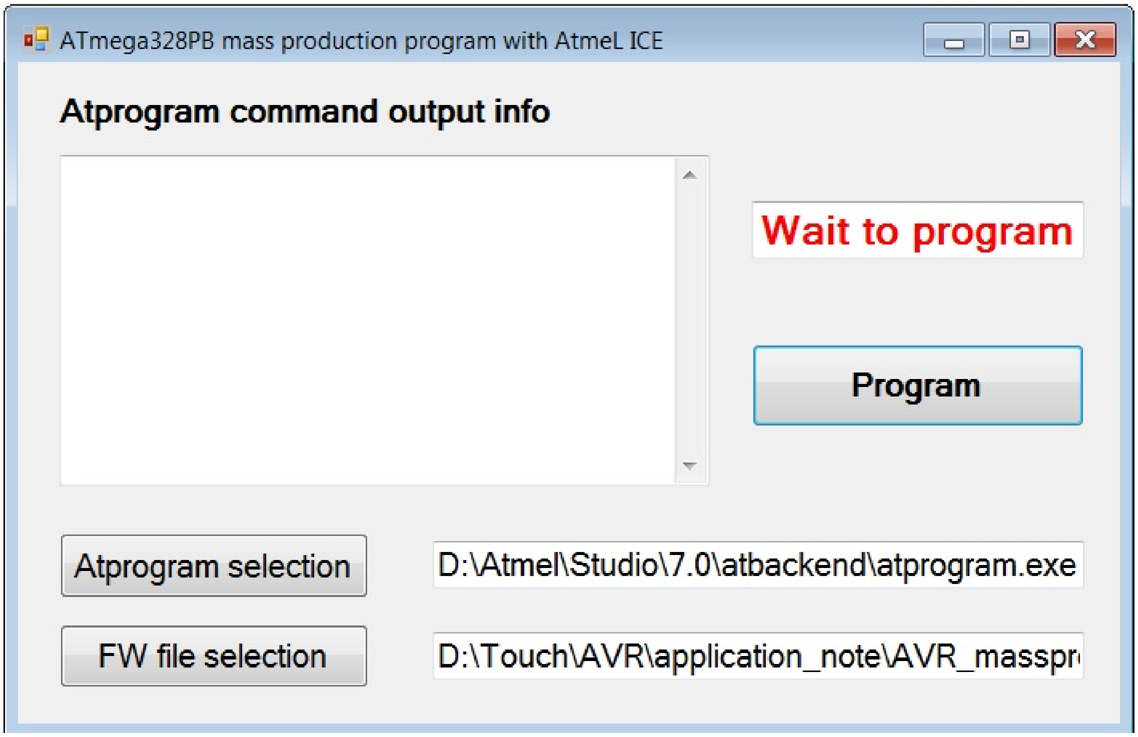
3.2 Features

- One button click to start the chiperase. The firmware image will then be programmed and verified, and the fuse lock bit will automatically be written.
- Output programming result information - pass or fail
- Real-time display programming output information
- Image and atprogram.exe file path selection

3.3 User Guide

This application note provides an example ATmega328PB production programming software tool as shown in the figure below. When the users click the "Program" button in the programming tool GUI window, it will perform the device programming and output *success* or *fail* information after the programming operation has finished. Once the programming is successful, LED0 will switch ON when the users press the SW0 button of the ATmega328PB Xplained Mini Kit and OFF when the SW0 button is released.

Figure 3-2. ATmega328PB Mass Production Programming Tool with Atmel-ICE



ATmega328PB mass production program with AtmeL ICE

Wait to program

D:\Atmel\Studio\7.0\atbackend\atprogram.exe

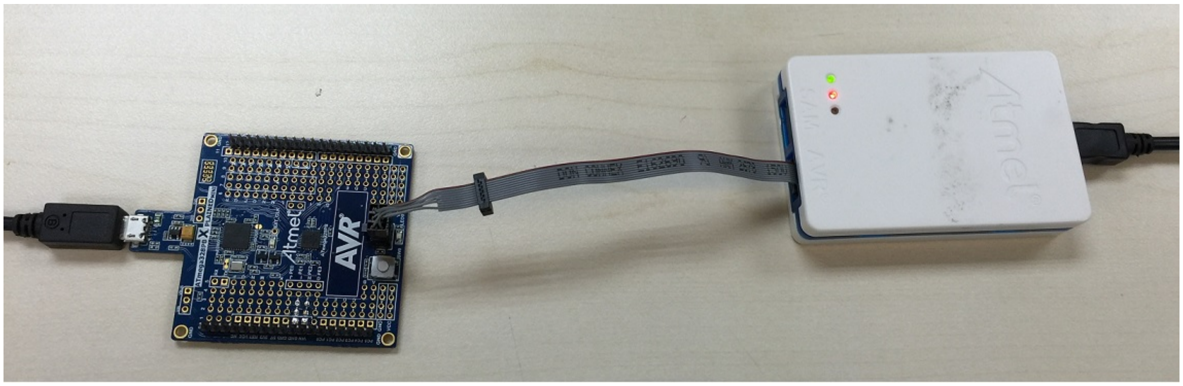
D:\Touch\AVR\application_note\AVR_masspr

The detailed steps on how to use this tool are listed below:

- Step 1: Download the AVR_massproduction_programming_tool_with_atmelice.zip file from the Microchip website and unzip it. After unzipping, it will have two folders; one is the Mass_production_programming_tool and the other is atprogram_example_application_project. In the mass_production_programming_tool directory, you will find the tool application and example firmware image file Button_led.hex, which can run on the ATmega328PB Xplained Mini Kit. The atprogram_example_application_project directory contains the source project developed with Visual C# 2010 Express.

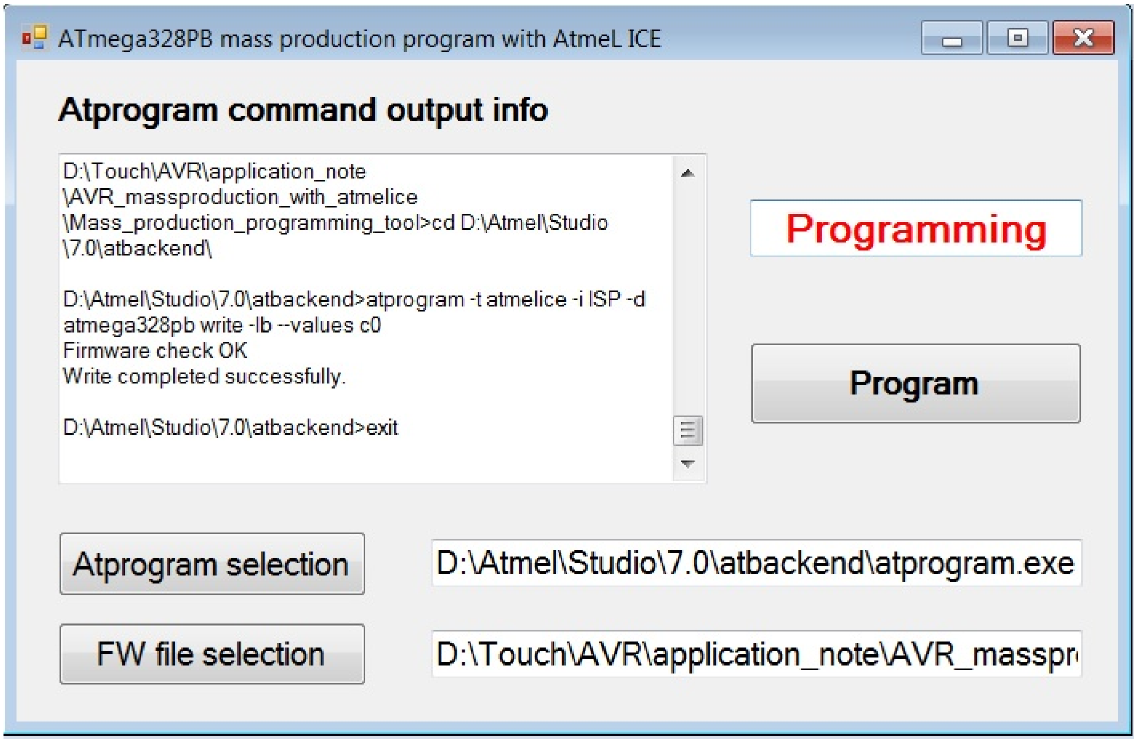
- Step 2: Connect the Atmel-ICE kit to the ISP programming port of the ATmega328PB Xplained Mini Kit with a 6-pin 100-mil IDC flat cable as shown in the figure below.

Figure 3-3. ATmega328PB Xplained Mini Kit's ISP Connection with Atmel-ICE



- Step 3: Plug the Atmel-ICE kit's USB cable into the PC and launch the mass_production_programming \Atprogram_application_example.exe application from the directory where you unzip the AVR_massproduction_programming_tool_with_Atmelice.zip file.
- Step 4: Click the "Atprogram selection" button in the programming GUI window and choose the Studio command line utility "atprogram.exe" file path. The atprogram.exe is located in the Atmel Studio installation directory \atbackend.
- Step 5: Click the "FW file selection" button in the programming GUI window and choose the "Button_Led.hex" file from the directory where you unzip the AVR_massprodcution_programming_tool_with_Atmelice.zip file.
- Step 6: Click the "Program" button. The "programming" string in program_status textbox will show up, as shown in the figure below.

Figure 3-4. ATmega328PB Chip In-programming Status Display



ATmega328PB mass production program with AtmeL ICE
Atprogram command output info

```
D:\Touch\AVR\application_note
\AVR_massproduction_with_atmelice
\Mass_production_programming_tool>cd D:\Atmel\Studio
\7.0\atbackend\

D:\Atmel\Studio\7.0\atbackend>atprogram -t atmelice -i ISP -d
atmega328pb write -lb --values c0
Firmware check OK
Write completed successfully.

D:\Atmel\Studio\7.0\atbackend>exit
```

Programming

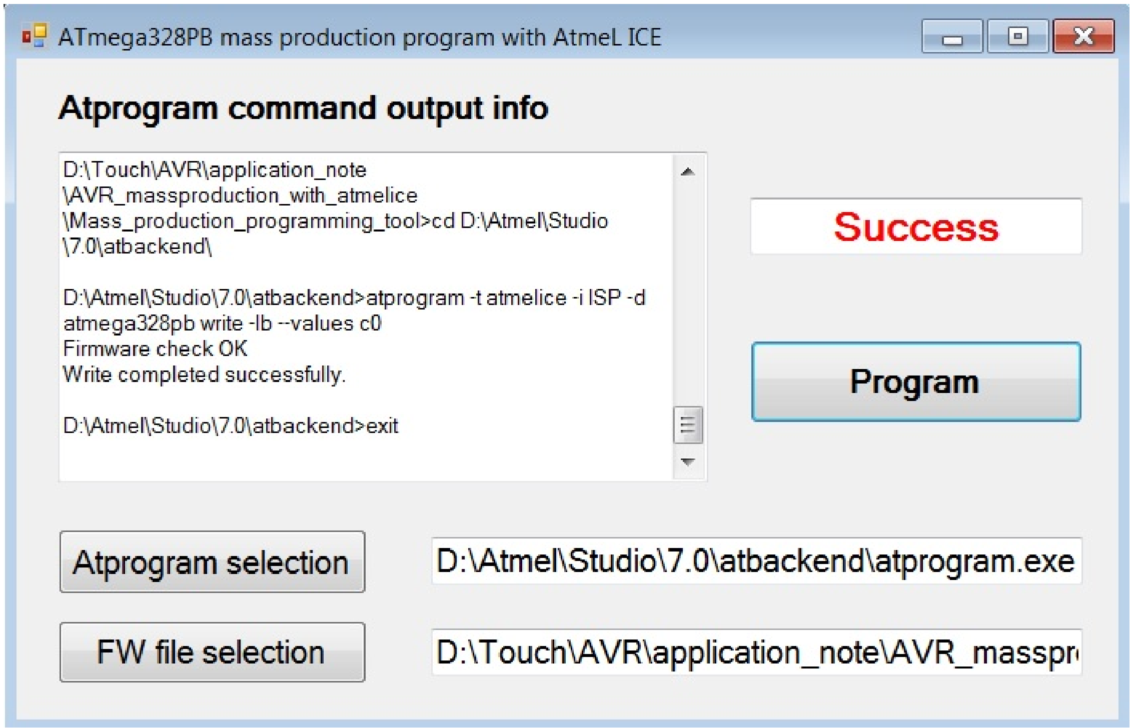
Program
Atprogram selection

D:\Atmel\Studio\7.0\atbackend\atprogram.exe

D:\Touch\AVR\application_note\AVR_masspr

- Step 7: Once programming is finished, it will show either the "success" string, as shown in figure ATmega328PB chip programming success info display, or the "fail" string, as shown in figure ATmega328PB chip programming failure info display, in the program_status textbox. If device programming fails, the users must first check the connection between Atmel-ICE and ATmega328PB Xplained Mini Kit's ISP programming connector; secondly, the users must check the ATmega328PB chip in the ISP programming mode because the kit programming mode may be changed from default ISP mode to debugWire mode. For ATmega328PB chip programming mode selection, refer to the AN42469: ATmega328PB Xplained Mini kit user guide.

Figure 3-5. ATmega328PB Chip Programming Success Info Display



ATmega328PB mass production program with AtmeL ICE
Atprogram command output info
Program
Atprogram selection
FW file selection

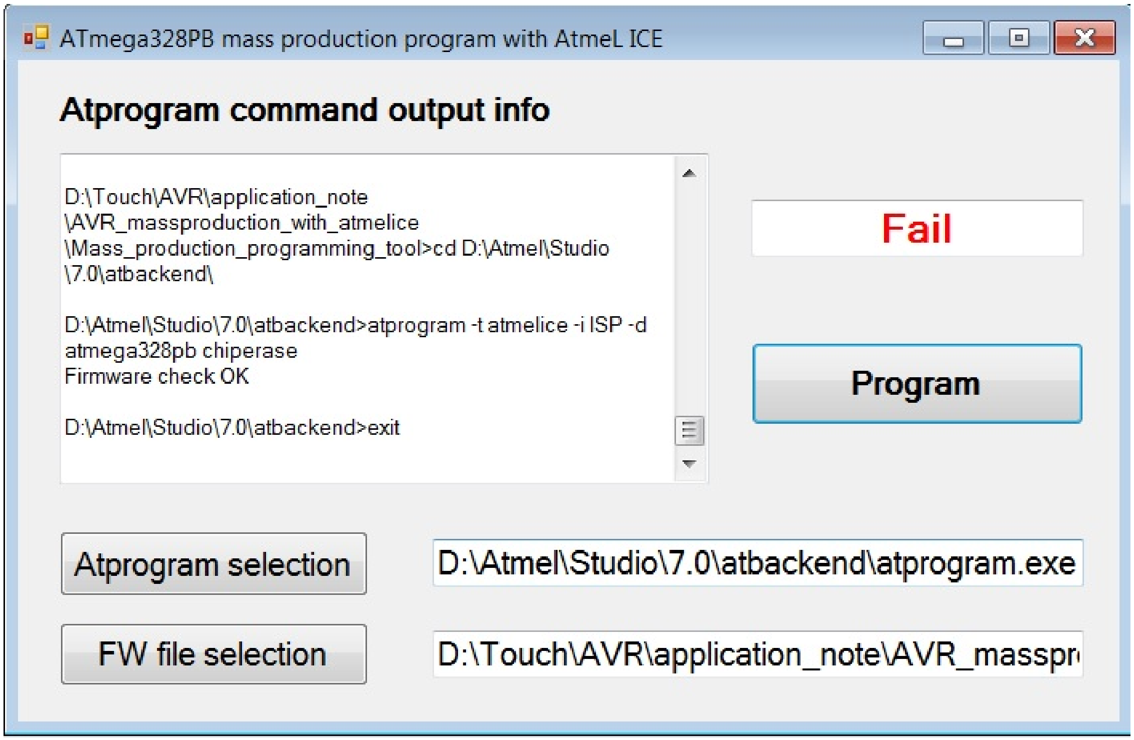
```
D:\Touch\AVR\application_note
\AVR_massproduction_with_atmelice
\Mass_production_programming_tool>cd D:\Atmel\Studio
\7.0\atbackend\

D:\Atmel\Studio\7.0\atbackend>atprogram -t atmelice -i ISP -d
atmega328pb write -lb --values c0
Firmware check OK
Write completed successfully.

D:\Atmel\Studio\7.0\atbackend>exit
```

Success

D:\Atmel\Studio\7.0\atbackend\atprogram.exe

D:\Touch\AVR\application_note\AVR_masspr

Figure 3-6. ATmega328PB Chip Programming Failure Info Display


ATmega328PB mass production program with AtmeL ICE
Atprogram command output info
Fail
Program
Atprogram selection
FW file selection

```
D:\Touch\AVR\application_note
\AVR_massproduction_with_atmelice
\Mass_production_programming_tool>cd D:\Atmel\Studio
\7.0\atbackend\

D:\Atmel\Studio\7.0\atbackend>atprogram -t atmelice -i ISP -d
atmega328pb chiperase
Firmware check OK

D:\Atmel\Studio\7.0\atbackend>exit
```

D:\Atmel\Studio\7.0\atbackend\atprogram.exe

D:\Touch\AVR\application_note\AVR_masspr

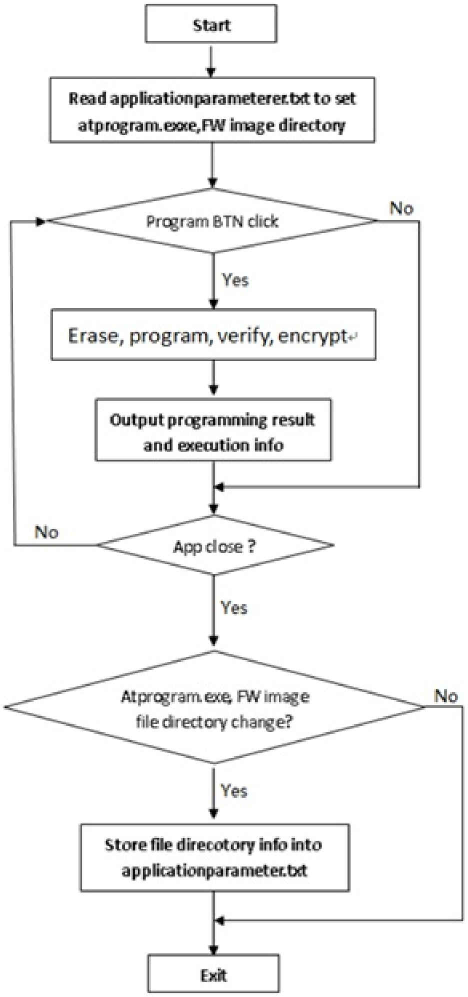
- Step 8: Verify if the example firmware image Button_led.hex is programmed successfully into the ATmega328PB Xplained Mini kit. Once successfully programmed, LED0 will be ON when the SW0 mechanical button is pressed and OFF when the SW0 button is released.

3.4 Source Overview

To help the users to better understand the program software development, this chapter introduces example software flow, atprogram commands calling code in example project file, and how to modify them for the users' own programming tool creation.

3.4.1 Example Software Flow

Figure 3-7. Example Programming Tool Software Flow


Start
Read applicationparameterer.txt to set atprogram.exxe,FW image directory
Program BTN click
No
Yes
Erase, program, verify, encrypt↵
Output programming result and execution info
No
App close ?
Yes
Atprogram.exe, FW image file directory change?
No
Yes
Store file direcotory info into applicationparameter.txt
Exit

3.4.2 Atprogram Commands Calling Code Review

When clicking the "program" button of the example software GUI, the example software project will execute the program_Click function in Form1.cs file. As shown below, in part of the program_Click function source code there are three atprogram commands: chiperase, program, and write.

```
private void program_Click(object sender, EventArgs e)
{
 StreamReader reader;
 .................

 // Full erase of the chip
 reader = RunProgram_atprogram("cmd.exe", "atprogram -t atmelice -i ISP -d atmega328pb
chiperase", textBox1.Text);
 .................

 // Program firmware image file assigned by text of textBox2 to flash and verify after
programming
 reader = RunProgram_atprogram("cmd.exe", "atprogram -t atmelice -i ISP -d atmega328pb
program -f "+textBox2.Text+" --verify",textBox1.Text);
 .................
 // Writing fuse lock bits value-0xc0(The fuse lock bits value need refer to specific device
datasheet, 0xc0 just for ATmega328PB device )
 reader = RunProgram_atprogram("cmd.exe", "atprogram -t atmelice -i ISP -d atmega328pb write -
lb --values c0 ", textBox1.Text);
 .................

 }
```

As the example software tool is developed for ATmega328PB device programming with ISP interface, the user must replace the device name (ATmega328PB) and programming interface name (ISP) with those on the user's platform. After modification and project rebuilding, a customized mass production programming tool with Atmel-ICE will be generated.

4. References

- ATmega328PB datasheet (http://www.microchip.com/wwwproducts/en/atmega328pb)
- ATmega328PB Xplained Mini kit (http://www.microchip.com/developmenttools/productdetails.aspx?partno=atmega328pb-xmini)

5. Revision History

Doc. Rev.	Date	Comments
B	12/2017	Introduction updated to recommend professional production programmers than Atmel-ICE.
A	07/2017	Initial document release.

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.

- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip’s code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BeaconThings, BitCloud, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, KeeLoq logo, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, RightTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, CryptoAuthentication, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, QMatrix, RightTouch logo, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

ISBN: 978-1-5224-2476-5

Quality Management System Certified by DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/
support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820