Safety Management for Space Environment Applications Using ATmegaS128

Introduction

This document highlights the key parameters of the ATmegaS128 that should be handled with care by any hardware and/or software developer in order to develop space-safe applications.

This document focuses on features that may be sensitive to a radiation environment and that must be considered at the application level. In addition, some tips to improve the safety of the global application are provided.
# Table of Contents

Introduction

1. General Considerations
   1.1. Code Behavior and Code Limits
   1.2. Interrupts
   1.3. Special Function Registers (SFR)
   1.4. Bootloader
   1.5. FMEA at System Level

2. System Control Mitigation
   2.1. Clocks
   2.2. Watchdog
   2.3. Fuse Bits

3. Memory Management
   3.1. Flash Memory Management
   3.2. EEPROM Memory Management
   3.3. SRAM Memory Management

4. Communication Interfaces
   4.1. I/O Conflict Management
   4.2. USART
   4.3. SPI
   4.4. TWI

5. Analog Functions
   5.1. ADC
   5.2. Comparator

6. Revision History
   6.1. Rev. A - 01/2018

The Microchip Web Site

Customer Change Notification Service

Customer Support

Microchip Devices Code Protection Feature

Legal Notice

Trademarks

Quality Management System Certified by DNV
1. General Considerations

1.1 Code Behavior and Code Limits

1.1.1 Default State of the Flash Memory

In case of unexpected loss of PC or SP, the application can fetch anywhere in the Flash memory.

By default, in Flash memory, unprogrammed bytes are set to 0xFF. 0xFFFF (16-bit instruction code) corresponds to a valid opcode in the ATmegaS128 instruction set:

“sbrs r31,7" (Skip if Bit 7 in Register R31 is Set)

In case of loss of the PC, if the PC is going above the end of your program, it will continue to fetch and execute all the Flash memory up to rollover to 0x0000.

Tip: Fill unused bytes of the Flash memory with an opcode allowing infinite loop and let the watchdog expire. The opcode 0xFCFF “rjump @PC” is the infinite loop opcode.

Remember: The PC is always aligned by words (2 bytes). If all unused code memory is filled with 0xFCFF (“rjump @PC”), the processor will never fetch 0xFFFC.

1.1.2 Unknown Opcodes

If an SEU event occurs, data may be corrupted while fetching the operations to be executed. In case of a fetch of unexpected opcode – opcode that is not allowed in the ATmegaS128 instruction set – the core executes a NOP.

1.2 Interrupts

In case of an unexpected interrupt (e.g., an interrupt from an unused peripheral), the user software may jump in and out of the interrupt subroutine without doing anything inside this routine. The user software may also enter an endless loop while waiting for the watchdog to reset the part.

Tip: Initialize all the interrupt vectors, even if they are not used, to reach a predictable state in case of error.

1.3 Special Function Registers (SFR)

It is recommended that the user periodically refresh the SFRs with their relevant values to correct any SEU effects on SFRs.
1.4 **Bootloader**

The ATmegaS128 bootloader is intended for reprogramming of the application throughout the life of the device. If the application must be reprogrammed during the mission (in-flight write-to-flash) through the bootloader, the tips listed below should be considered.

- To avoid any corruption on the bootloader area (critical software for reprogramming), the boot code section must be locked. See the section “Avoiding Flash Memory Content Loss”.
- The following startup sequence should be used to secure the correct user application execution:
  1. Reset the boot section and check if bootloader activation is requested.
  2. If the bootloader is requested (bootloader start requested), check the bootloader section content (CRC or checksum).
     - If the result is correct (bootloader content not modified), resume the bootloader.
     - If the bootloader check is incorrect, run the current application without downloading a new application through the bootloader.
  3. If the bootloader is not requested (application start requested), check the application section content (CRC or checksum).
     - If the result is correct (application content not modified), then run the application by jumping at address 0x0000.
     - If the application check is incorrect, run the bootloader to download a correct application.

**Remember:** The user must pay careful attention to Watchdog behavior, in particular if the WDTON bit is programmed during the startup sequence. See the section “Watchdog” for recommendations regarding Watchdog behavior.

1.5 **FMEA at System Level**

At system level, the end user must consider the criticality of the different signals/events managed by the ATmegaS128 to elaborate an adequate mitigation between the consecutive refreshes of the application.

**Attention:** Key points to be answered are:

- What happens if a signal is missing for a short period of time?
- What happens if a signal is wrong for a short period of time?
2. System Control Mitigation

2.1 Clocks

2.1.1 Internal Oscillator
The OSCCAL value is copied at reset from the signature row into the OSCCAL register. In case of use of the OSCCAL register in the application, it is recommended to make copies of this parameter in the RAM memory in order to control its integrity throughout the lifetime of the application. The SRAM is sensitive to SEU events, and so multiple locations should be used as copies of the OSCCAL parameter to allow efficient checking. In any case, a reset reconfigures the OSCCAL register with the default factory value.

2.1.2 Source Clock Selection
The clock source for the device can be selected between internal RC sources and external sources. If an application requires precise timing/clocking, it is recommended to use the external source. In this case, the clock is configured so that the external source is a reference; however, there is a risk to switch from this clock to an internal source in case of SEU. In order to detect the event, it is recommended to have the OSCCAL register configured to 0. This allows detection of the event if it occurs.

2.2 Watchdog
The ATmegaS128 embeds watchdog features that the user must consider carefully in his software design to avoid unexpected timeout of the application.

When WDTON (Watchdog Enable On) is programmed, the watchdog is running directly after reset with a default configuration of 16K clock cycles before it expires.

Tip: To ensure correct behavior of the application without spurious time-out, first clear the watchdog and then set it to the value consistent with the application requirement in the application startup file. This should be done before any other applicative task.

Remember: Watchdog frequency is dependent on temperature and voltage, thus the watchdog must be configured with sufficient margin to fit the application requirements.

2.3 Fuse Bits
The ATmegaS128 embeds a full set of fuse bits for configuration of the device parameters. As for the Flash memory array itself, the fuse bits are immune to SEU. The effect of the fuses on the application occurs only once at power-on when they are latched in volatile memory cells. The volatile cells in which the parameters have been latched are not SEU immune and, as such, can be affected by SEU events.
If an error is induced in this volatile cell by a heavy ion, the configuration of the device will remain in its ‘faulty’ state as long as no power-off/power-on sequence is applied.

- The loss of fuse bits is not recovered by the internal watchdog which simply generates a reset of the chip.
- As the fuse bits are only latched at power-on, the only way to recover from the faulty state is to apply a power-off/power-on sequence to the device.

**Tip:** Some fuse bit functionalities are critical for the application, such as the clock selection, the BOD level or the boot reset. To avoid deadlock of the device, implement an external mechanism to cycle power-off then power-on of the device when the application no longer answers. Occurrences of such events are very rare. For more details, refer to the ATmegaS128 radiation report.

**CAUTION** Although not related to radiation considerations, an incorrect configuration of the fuse bits may lead to a deadlock of the device, without any possibility to recover on-board. Particular attention must be given to the alignment between hardware and software prior to any configuration of the fuse bits.
3. Memory Management

3.1 Flash Memory Management

3.1.1 Avoiding Flash Memory Content Loss

The ATmegaS128 has lock bits to lock the content of the Flash memory. It is mandatory to use the lock bits to prevent any loss of content of the Flash memory.

Tip: If no reprogramming functionality is needed, lock the part using BLB0x and BLB1x lock bits (without using LBx bits).

Tip: If the bootloader functionality is used, lock the bootloader section using BLB1x fuse bits (without using LBx bits).

See the section “Bootloader” for more details on risks related to programming in-flight.

3.1.2 In-Flight Flash Reprogramming Considerations

The global application of an ATmegaS128 relies on a bootloader that must fit in the Bootloader section of the Flash memory and, on a user application, in the application area of the memory.

CAUTION: It is highly recommended to lock the Flash memory content as described in the section “Avoiding Flash Memory Content Loss” and to avoid in-flight Flash memory reprogramming.
When the memory is fully locked, both the bootloader and the application are protected against any write operation, thus preventing unexpected data loss. Data retention is also guaranteed. Since the application section is locked, no application reprogramming is possible.

Whereas it is not recommended to use Flash reprogramming during flight, it is possible to use the capability to reprogram the device. If the end user wants to use in-flight Flash programming, either of the two setups described below may be used.

3.1.2.1 Application reprogramming from the Boot section

The reference configuration hypotheses for this use case are the following:

- Bootloader section is locked
- Application section is not locked

See the section "Avoiding Flash Memory Content Loss".

When the Boot section is locked to avoid any corruption in the critical area of the software, the ATmegaS128 still allows reprogramming of the application from the boot section.

The bootloader is always protected against unexpected loss of data. Data retention of the bootloader section is guaranteed.

The application section is not protected against unexpected data loss.

Data retention of the application section after in-flight programming must be re-assessed with respect to the post-write TID characterization results presented in the ATmegaS128 radiation report.

3.1.2.2 Application reprogramming from the ISP interface

The reference configuration hypotheses for this use case are the following:

- Bootloader section is locked
- Application section is locked

See the section "Avoiding Flash Memory Content Loss".
When both the boot section and the application section are locked, it is not possible to reprogram the application section through the bootloader. The only way to reprogram the application is to perform reprogramming through the external ISP.

The bootloader and application are always protected against unexpected data loss.

Retention by the two sections after in-flight programming must be re-assessed with respect to the post-write TID characterization results presented in the ATmegaS128 radiation report.

### 3.2 EEPROM Memory Management

#### 3.2.1 Avoiding EEPROM Memory Content Loss

The ATmegaS128 has lock bits to lock the content of the EEPROM memory.

It is mandatory to use the lock bits to prevent any loss of content from the EEPROM memory.

Tip: If no reprogramming functionality is needed, lock the part using the LBx bits.

### 3.3 SRAM Memory Management

#### 3.3.1 Avoiding Erroneous Data in SRAM

As described in the radiation report, the SRAM memory of the ATmegaS128 is not SEU immune. Single bit events (SBU) and multiple bit events (MBU) can be observed on the memory.

Since the SRAM is sensitive to SEU events, there is a risk of operating on a corrupted data in the application. To avoid this, multiple memory locations should be used as copy of the data/parameters to allow efficient checking.
Tip: As an example, if a data is saved in three locations in the SRAM memory, the verification of the correctness of the data can be easily done by applying a voting algorithm at software level before use of the data.

**CAUTION** Depending on the ‘check’ strategy used in minimizing the ‘erroneous read’ risk, the performance of the application can be decreased from a wide range. Safety/performance paradigms should be consistent with application constraints.
4. Communication Interfaces

4.1 I/O Conflict Management
The registers used for configuration of the I/Os can be affected by SEU by modifying the PIN/PORT direction registers and I/O values. An optimized I/O configuration process is recommended for each I/O access.

4.1.1 Reading PIN/PORT Registers

Tip: Systematically configure the PIN/PORT as input before reading any port.

Tip: Execute multiple PIN/PORT reads to get the value.

4.1.2 Writing PIN/PORT Registers

Tip: Systematically configure the PIN/PORT as output before writing to any port.

4.1.3 I/O Conflict Management
In case of SEU affecting the PORT direction, conflict on the I/O lines may appear (risk of multiple drivers on the same line). To avoid conflicts, it is recommended to

- Add a line resistor on all input pins to avoid conflict in case of SEU changing I/O direction to output.
- Refresh port direction on a fast time basis to avoid long term switch to the faulty direction.

4.2 USART
As SEU can affect USART communication, it is recommended to implement data integrity check mechanisms at application level. In case of error, the transmitter must be warned and determines whether to resend (or not) the data/frame to the receiver.

At hardware level, on the USART, byte control can be activated – use of the parity bits inside the USART configuration.

At application level, frame control with CRC, checksum, security check, etc can be instantiated.
4.3 SPI
As SEU can affect SPI communication, it is recommended to implement data integrity check mechanisms at application level. In case of error, the transmitter must be warned and determines whether to resend the data/frame to the receiver.

At application level, frame control with CRC, checksum, security check, etc. can be instantiated.

**Tip:** If the SPI is used for memory access, execute several read cycles of the required memory cell to ensure correct data read.

4.4 TWI
As SEU can affect TWI communication, it is recommended to implement data integrity check mechanisms at application level. In case of error, the transmitter must be warned and determines whether to resend (or not) the data/frame to the receiver.

At application level, frame control with CRC, checksum, security check, etc. can be instantiated.

**Tip:** If the TWI is used for memory access, execute several read cycles of the required memory cell to ensure correct data read.
5. Analog Functions

5.1 ADC
As SEU can affect ADC conversion by corrupting a conversion result, it is recommended to execute multiple conversions before taking the converted value into account.

The effect of SEU in an ADC conversion chain can be considered as a noise over the acquisition chain. The standard techniques used in signal processing for improving SNR and optimizing measurement resolution can be of interest in order to minimize the impact of the SEU on the quality of the acquisition.

Tip: Standard oversampling and/or averaging techniques can be used for the mitigation of SEU events over the ADC acquisition chain.

5.2 Comparator
As SEU can affect comparator behavior by generating unexpected events, it is recommended to apply a digital filtering on the comparison event flags before considering the event as relevant.

The effect of SEU on the comparator is to indicate a spurious wrong comparison result on the ACO output. In such a case, all the comparison information flags would be updated while no modification of the status is expected.

On reception of a comparison event (interrupt configured in the ACIS field of the Analog Comparator Control and Status (ACSR) register), it is recommended that the application sample the ACO field of the ACSR multiple times. This checks whether the event was due to a spurious faulty comparison result, or if it relies on a relevant comparison result.

Tip: The ACSR.ACO field is read-only. In case of a spurious comparison, reading the field twice or three times determines if the event was transient or if it is relevant.
6. Revision History

6.1 Rev. A - 01/2018

- General
  - Template update: Moved from Atmel to Microchip template.
  - The application note is assigned a new document number (DS00002635) and revision letter is reset to A.
    - Document number DS00002635 revision A corresponds to what would have been document 41086 revision B.
    - ISBN number assigned.
  - Editorial changes throughout.
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.


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.


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.
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, KEELQ® 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.
## 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

### 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

### Europe

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Tel:</th>
<th>Fax:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>Wels</td>
<td>43-7242-2244-39</td>
<td>43-7242-2244-393</td>
</tr>
<tr>
<td>Denmark</td>
<td>- Copenhagen</td>
<td>45-4450-2828</td>
<td>45-4485-2829</td>
</tr>
<tr>
<td>Finland</td>
<td>- Espoo</td>
<td>358-9-4520-820</td>
<td>358-9-4520-820</td>
</tr>
<tr>
<td>Germany</td>
<td>- Garching</td>
<td>49-8931-9700</td>
<td>49-8931-9700</td>
</tr>
<tr>
<td>Germany</td>
<td>- Haan</td>
<td>49-2129-3766400</td>
<td>49-2129-3766400</td>
</tr>
<tr>
<td>Germany</td>
<td>- Heilbronn</td>
<td>49-7131-67-3636</td>
<td>49-7131-67-3636</td>
</tr>
<tr>
<td>Germany</td>
<td>- Karlsruhe</td>
<td>49-721-625370</td>
<td>49-721-625370</td>
</tr>
<tr>
<td>Germany</td>
<td>- Munich</td>
<td>49-89-627-144-0</td>
<td>49-89-627-144-44</td>
</tr>
<tr>
<td>Germany</td>
<td>- Rosenheim</td>
<td>49-8031-354-560</td>
<td>49-8031-354-560</td>
</tr>
<tr>
<td>Israel</td>
<td>- Ra’anana</td>
<td>972-9-744-7705</td>
<td>972-9-744-7705</td>
</tr>
<tr>
<td>Italy</td>
<td>- Milan</td>
<td>39-0331-742611</td>
<td>39-0331-466781</td>
</tr>
<tr>
<td>Italy</td>
<td>- Padova</td>
<td>39-049-7625286</td>
<td>39-049-7625286</td>
</tr>
<tr>
<td>Netherlands</td>
<td>- Drunen</td>
<td>31-416-609399</td>
<td>31-416-609399</td>
</tr>
<tr>
<td>Norway</td>
<td>- Trondheim</td>
<td>47-7289-7561</td>
<td>47-7289-7561</td>
</tr>
<tr>
<td>Poland</td>
<td>- Warsaw</td>
<td>49-22-3232573</td>
<td>49-22-3232573</td>
</tr>
<tr>
<td>Romania</td>
<td>- Bucharest</td>
<td>40-21-407-87-50</td>
<td>40-21-407-87-50</td>
</tr>
<tr>
<td>Spain</td>
<td>- Madrid</td>
<td>34-91-708-08-90</td>
<td>34-91-708-08-91</td>
</tr>
<tr>
<td>Sweden</td>
<td>- Gothenberg</td>
<td>46-31-704-60-40</td>
<td>46-31-704-60-40</td>
</tr>
<tr>
<td>Sweden</td>
<td>- Stockholm</td>
<td>46-8-5090-4654</td>
<td>46-8-5090-4654</td>
</tr>
<tr>
<td>UK</td>
<td>- Wokingham</td>
<td>44-118-921-5800</td>
<td>44-118-921-5820</td>
</tr>
</tbody>
</table>