

## 24. Boot Loader Support – Read-While-Write Self-Programming, ATmega88P, ATmega168P and ATmega328P

### 24.1 Features

- Read-While-Write Self-Programming
- Flexible Boot Memory Size
- High Security (Separate Boot Lock Bits for a Flexible Protection)
- Separate Fuse to Select Reset Vector
- Optimized Page<sup>(1)</sup> Size
- Code Efficient Algorithm
- Efficient Read-Modify-Write Support

Note: 1. A page is a section in the Flash consisting of several bytes (see [Table 25-11 on page 299](#)) used during programming. The page organization does not affect normal operation.

### 24.2 Overview

In ATmega88P, ATmega168P and ATmega328P, the Boot Loader Support provides a real Read-While-Write Self-Programming mechanism for downloading and uploading program code by the MCU itself. This feature allows flexible application software updates controlled by the MCU using a Flash-resident Boot Loader program. The Boot Loader program can use any available data interface and associated protocol to read code and write (program) that code into the Flash memory, or read the code from the program memory. The program code within the Boot Loader section has the capability to write into the entire Flash, including the Boot Loader memory. The Boot Loader can thus even modify itself, and it can also erase itself from the code if the feature is not needed anymore. The size of the Boot Loader memory is configurable with fuses and the Boot Loader has two separate sets of Boot Lock bits which can be set independently. This gives the user a unique flexibility to select different levels of protection.

### 24.3 Application and Boot Loader Flash Sections

The Flash memory is organized in two main sections, the Application section and the Boot Loader section (see [Figure 24-2](#)). The size of the different sections is configured by the BOOTSZ Fuses as shown in [Table 24-7 on page 289](#) and [Figure 24-2](#). These two sections can have different level of protection since they have different sets of Lock bits.

#### 24.3.1 Application Section

The Application section is the section of the Flash that is used for storing the application code. The protection level for the Application section can be selected by the application Boot Lock bits (Boot Lock bits 0), see [Table 24-2 on page 281](#). The Application section can never store any Boot Loader code since the SPM instruction is disabled when executed from the Application section.

#### 24.3.2 BLS – Boot Loader Section

While the Application section is used for storing the application code, the The Boot Loader software must be located in the BLS since the SPM instruction can initiate a programming when executing from the BLS only. The SPM instruction can access the entire Flash, including the BLS itself. The protection level for the Boot Loader section can be selected by the Boot Loader Lock bits (Boot Lock bits 1), see [Table 24-3 on page 281](#).