Application Note, V 0.1, July 2003

# AP08009

# C868P

## Data Scrambling Feature in C868P Microcontroller

Microcontrollers

Infineon technologies

Never stop thinking.

**Edition 2003-07-01**

**Published by**
**Infineon Technologies AG**
**81726 München, Germany**



**LEGAL DISCLAIMER**

THE INFORMATION GIVEN IN THIS APPLICATION NOTE IS GIVEN AS A HINT FOR THE IMPLEMENTATION OF THE INFINEON TECHNOLOGIES COMPONENT ONLY AND SHALL NOT BE REGARDED AS ANY DESCRIPTION OR WARRANTY OF A CERTAIN FUNCTIONALITY, CONDITION OR QUALITY OF THE INFINEON TECHNOLOGIES COMPONENT. THE RECIPIENT OF THIS APPLICATION NOTE MUST VERIFY ANY FUNCTION DESCRIBED HEREIN IN THE REAL APPLICATION. INFINEON TECHNOLOGIES HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND LIABILITIES OF ANY KIND (INCLUDING WITHOUT LIMITATION WARRANTIES OF NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF ANY THIRD PARTY) WITH RESPECT TO ANY AND ALL INFORMATION GIVEN IN THIS APPLICATION NOTE.

**Information**

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office (**www.infineon.com**).

**Warnings**

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

**C868P**

| **Revision History:** | **2003-07** | V 0.1 |
|---|---|---|
| Previous Version: | - | |

| Page | Subjects (major changes since last revision) |
|---|---|
| | |
| | |
| | |
| | |
| | |

Controller Area Network (CAN): License of Robert Bosch GmbH

**We Listen to Your Comments**

Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document. Please send your proposal (including a reference to this document) to:

**mcdocu.comments@infineon.com**

**Table of Content** **Page**



**List of Figures** **Page**

# 1 Introduction

The C868P microcontroller offers the feature of Data Scrambling which provides code protection for external EEPROM. The customer's original hex code can be scrambled using a dedicated key and PC tools provided by Infineon Technologies. The scrambled code is then burned into the EEPROM. The C868P microcontroller, interfacing with the external EEPROM, will then unscrambled the EEPROM code, load the descrambled code (customer's original hex code) to SRAM and finally run from SRAM.

This application note briefly describes the hardware interface of the EEPROM with the C868P microcontroller and the EEPROM functions implemented by the C868P bootstrap loader. At the later section, Data Scrambling features, tool, requirements, procedures, functions, checksum test and limitations are elaborated.

# 2 Hardware Interface

## 2.1 Type of EEPROM

The type of external EEPROM supported is SPI serial EEPROM. Serial EEPROM devices offer significant advantages over parallel devices in applications where lower data transfer rates are acceptable. They also require less board space and allow microcontroller I/O pins to be conserved. This is valuable to the low pin count C868P microcontroller. Furthermore, SPI specification is easy to be implemented in software. Additionally, please note that only EEPROM in 32-byte page mode is supported. There is no limitation on the size of the EEPROM. However, as the SRAM size is 8Kbyte, It is suggested to use EEPROM (≥8Kbyte) so that all SRAM content can be loaded to EEPROM.

## 2.2 Hardware Configuration

The SPI EEPROM is connected to the C868P microcontroller in three-wire configuration as shown in Fig.1. In this configuration, the SPI EEPROM serial data in (SI) and serial data out (SO) are both connected to the same C868P I/O pin, thereby saving a pin. This is possible because the C868P I/O pins can be dynamically reprogrammed as input or output.

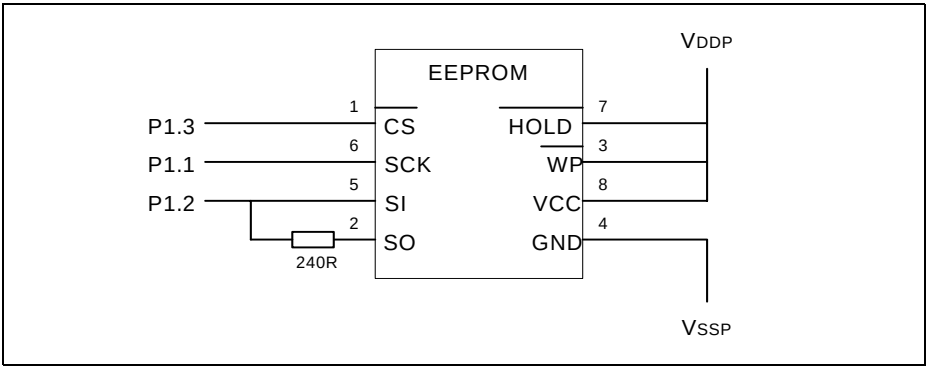


**Figure 1 Three-wire connection to SPI EEPROM**

# 3 Data Scrambling

## 3.1 Customer Specific Tool DataScrambler

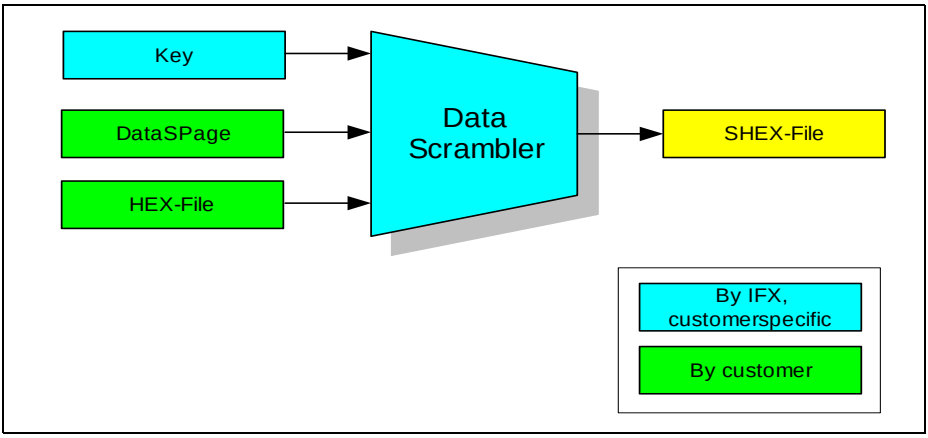


**Figure 2 Data Scrambling Flow**

**DataScrambler (GUI version)**

- DataScrambler (DataScramble.exe) is a dialog based application, which acts as a user friendly front end for Data Scrambling

  (Windows->Start->Programs->C868DataScrambler->DataScrambler)
- It can run on Win95/98/2000 and Win NT(4.0) (Intel processors)
- Three parameters: Key, DataSPage and HexFile.hex
- Generate scrambled SHexFile.hex (in C:\Temp)

**DataScrambler (Command prompt)**

- DataScrambler (DS.exe) is an executable file which can run on DOS-prompt.

  (Windows->Start->Programs->MS-DOS Prompt) on Win98;

  (Windows->Start->Programs->Accessories->Command Prompt) on Win2000
- Three parameters: Key, DataSPage and HexFile.hex
- Generate scrambled SHexFile.hex (in current directory)

The invocation of DataScrambler is done on DOS-prompt as in the following example:

```
DS  Key  DataSPage  HEX-File
DS  D5000084C1485350 A3 testhex.hex
```

### 3.1.1 Inputs

```
Key             [8-byte Key provided by Infineon Technologies]
DataSPage       [Starting page of data in EEPROM of SHEX-file]
HexFile.hex     [Input HEX-file for scrambling]
```

**Key**

The 8-byte Key is unique to each customer and project. It is provided to customer on request by Infineon Technologies.

**DataSPage**

DataSPage (one byte) ranges from 0x01 to 0xFF. It is the starting page number of data in EEPROM where the data portion of input HEX-file will not be scrambled. The customer should ensure that the data in the input HEX-file starts from the address:

**Starting address of data (in input HEX-file) = $32_{10}$*DataSPage -$10_{10}$**

*For example:*

*When DataSPage is 0xF8, the start address should be $32_{10}$*$248_{10}$ -$10_{10}$=$7926_{10}$*

*Therefore the data in the input HEX-file should start from address 0x1EF6.*

**HexFile.hex**

The purpose of Data Scrambling is to offer code protection for customer's code. Thus for enhanced security measure, customer is encouraged to fill the GAPs in the input HEX-file with random code if the GAPs are large.

### 3.1.2 Output

The scrambled SHEX-file (SHexFile.hex) is generated based on the Key and DataSPage values entered by customer. This SHEX-file contains:

- Scrambler header
- Scrambled code
- Unscrambled data
- End of file record

*Note: DataScrambler is capable of generating 8Kbyte of SHEX-file. In addition to the 10-byte scrambler header, the input HEX-file should not be more than (8K-10)byte, otherwise error message will be generated when executing the DataScrambler.*

## 3.2 Requirements

- DataScrambler Setup <ap0800901_DataScrambling_C868P.exe>
- The 8-byte Key
- SPI EEPROM
- Customer's Hex code

## 3.3 Procedure

Step 1: Unzip the ap0800901_DataScrambling_C868P.exe file and double click on the *setup.exe* to install the DataScrambler.

– If path is not amended, the DataScramble.exe and DS.exe should be installed in C:\Program Files\C868 DataScrambler\DataScrambler

Step 2: To execute Data Scrambler GUI version[1)], double click on *DataScramble.exe* found in the mentioned directory.

– A dialog based application will popped up.

Step 3: Enter the 8-byte Key and the DataSPage.

Step 4: Choose the input HEX-file for data scrambling and the resultant SHEX-file.

Step 5: Click on *Scramble* button.

– SHEX-file (SHexFile.hex) will be generated at the chosen directory.

Step 6: Load/program this SHEX-file into the external EEPROM.

Step 7: Connect this EEPROM to C868P microcontroller as described in **Section 2.2**.

– You are ready!

When power supply is switched on, the C868P will access the EEPROM directly. When 0D5H is detected, the C868P will undergo descrambling routine and load the descrambled code into SRAM. Finally it will jump to the beginning of SRAM and execute the instructions.

1) For Command Prompt version of DataScrambler, copy the DS.exe into other directory, C:\DataS for example. Open a MS-DOS console window and enter C:\DataS . To execute the DataScrambler, type DS followed by three command line parameters: 8-byte Key, DataSPage and the input HEX-file (with full path if it is not in the current directory). The output SHEX-file will then be generated in the current directory.

# 4 Functional Description

After boot-up from bootrom, the C868P microcontroller will access the EEPROM directly. It will try to read the first byte (Password) of the EEPROM. There are three different scenarios:

If the Password is **0A5H**, the C868P will directly load program from the EEPROM to SRAM and then execute from SRAM after loading is completed. The (re)programming of this EEPROM can be found in application note ap08006. The C868P will always load 8Kbytes of code.

If the Password is **0D5H**, the C868P will identify EEPROM as a scrambled code and unscrambled the EEPROM code. It will then load the descrambled code (customer's original hex code) to SRAM and finally run from SRAM. The C868P will always load 8Kbytes of code.

If the Password is **neither 0A5H nor 0D5H**, the C868P will enter a state where the content of the SRAM are cleared. This is for security purpose.

## 4.1 Checksum Test

The one-byte Checksum is obtained by XOR-ing all of the customer's original code (excluding the data). It is generated by DataScrambler and included in the output SHEX-file.

While descrambling and downloading the program from the EEPROM, the bootrom will XOR the code and calculate the checksum. If the checksum is correct (the calculated checksum tallies with the Checksum value obtained in the EEPROM), the C868P will continue to download the remaining data from EEPROM to the SRAM.

If the checksum is incorrect (the calculated checksum does not tally with the Checksum value obtained in the EEPROM), the C868P will attempt to re-access the EEPROM from the beginning. If the checksum remains incorrect after 10 attempts, the C868P will enter a state where the content of the SRAM are cleared.

## 4.2 Limitations

1. No UART bootup is allowed to prevent SRAM content being read out. i.e. the mini-debugger cannot be connected in the C868P microcontroller.

2. Scrambled code will have to be programmed/loaded into EEPROM using other means like Programmer or normal C868 microcontroller.

3. Only loading to SRAM is supported.

# 5 Glossary

| | |
|---|---|
| SRAM | On-chip ram mapped at address 0000H to 1FFFH |
| SPI | Serial Peripheral Interface |
| UART | Full-Duplex Serial Interface |
| DS | DataScrambler |
| GUI | Graphical User Interface |

# Infineon goes for Business Excellence

“Business excellence means intelligent approaches and clearly defined processes, which are both constantly under review and ultimately lead to good operating results.
Better operating results and business excellence mean less idleness and wastefulness for all of us, more professional success, more accurate information, a better overview and, thereby, less frustration and more satisfaction.”

Dr. Ulrich Schumacher

http://www.infineon.com

Published by Infineon Technologies AG