AN209716

Implementing the BootROM Interface with Traveo™ Family

Author: Hiroto Nishihata

Associated Part Family: Traveo™ Family S6J3110/3120/3200/3310/3320/3330/3340/3350/3360/3370/3400 Series

Related Documents: For a complete list, see Related Documents.

This application note describes how to implement the BootROM interface of the Cypress Traveo™ family S6J3110/S6J3120/S6J3200/S6J3300/S6J3350/S6J3360/S6J3370/S6J3400 series.

1 Introduction

This application note is intended for users of the Cypress Traveo family S6J3110/ S6J3120/ S6J3200/ S6J3300/ S6J3350/ S6J3360/ S6J3370/ S6J3400 series MCUs. It describes how to implement the BootROM interface.

2 BootROM Interface

The Traveo family S6J3110/S6J3120/S6J3200/S6J3300/S6J3350/S6J3360/S6J3370/S6J3400 series MCUs have two BootROM interfaces:

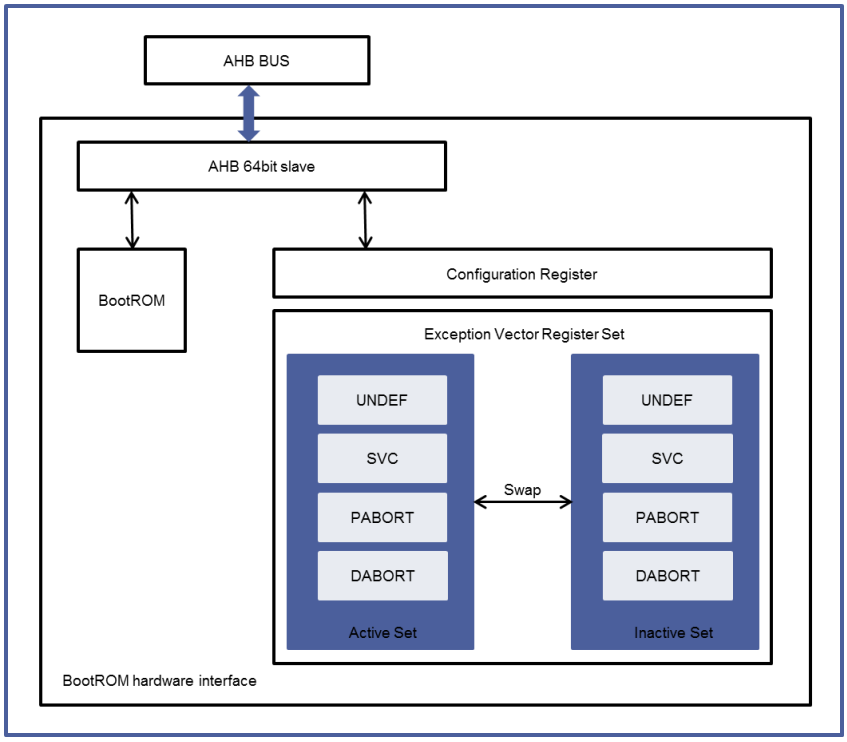
- BootROM hardware interface
- BootROM software interface

2.1 BootROM Hardware Interface

The BootROM hardware interface is used to configure the setting registers for the user-defined exception handler. It contains both the active and inactive sets of the exception vector registers. Figure 1 shows the BootROM hardware interface block diagram.

The address to which the active set is mapped is at an active position that the BootROM exception entry refers to when an exception occurs. The address to which the inactive set is mapped is at an inactive position, in which the values can be changed. Writing a ‘1’ to the swap bit of the setting register (EXCFG_CNFG:SWAP) swaps the contents of the active and inactive sets. This action changes the settings of the active set.

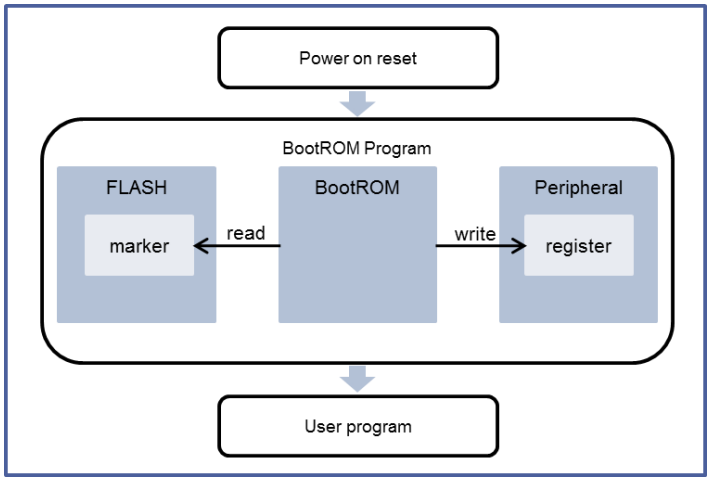
Figure 1. BootROM Hardware Interface Block Diagram



2.2 BootROM Software Interface

The BootROM software is automatically executed after a reset and before the user program starts executing. Figure 2 shows the interfaces of the BootROM software.

Figure 2. BootROM Software Interface Image



The BootROM software mainly does the following:

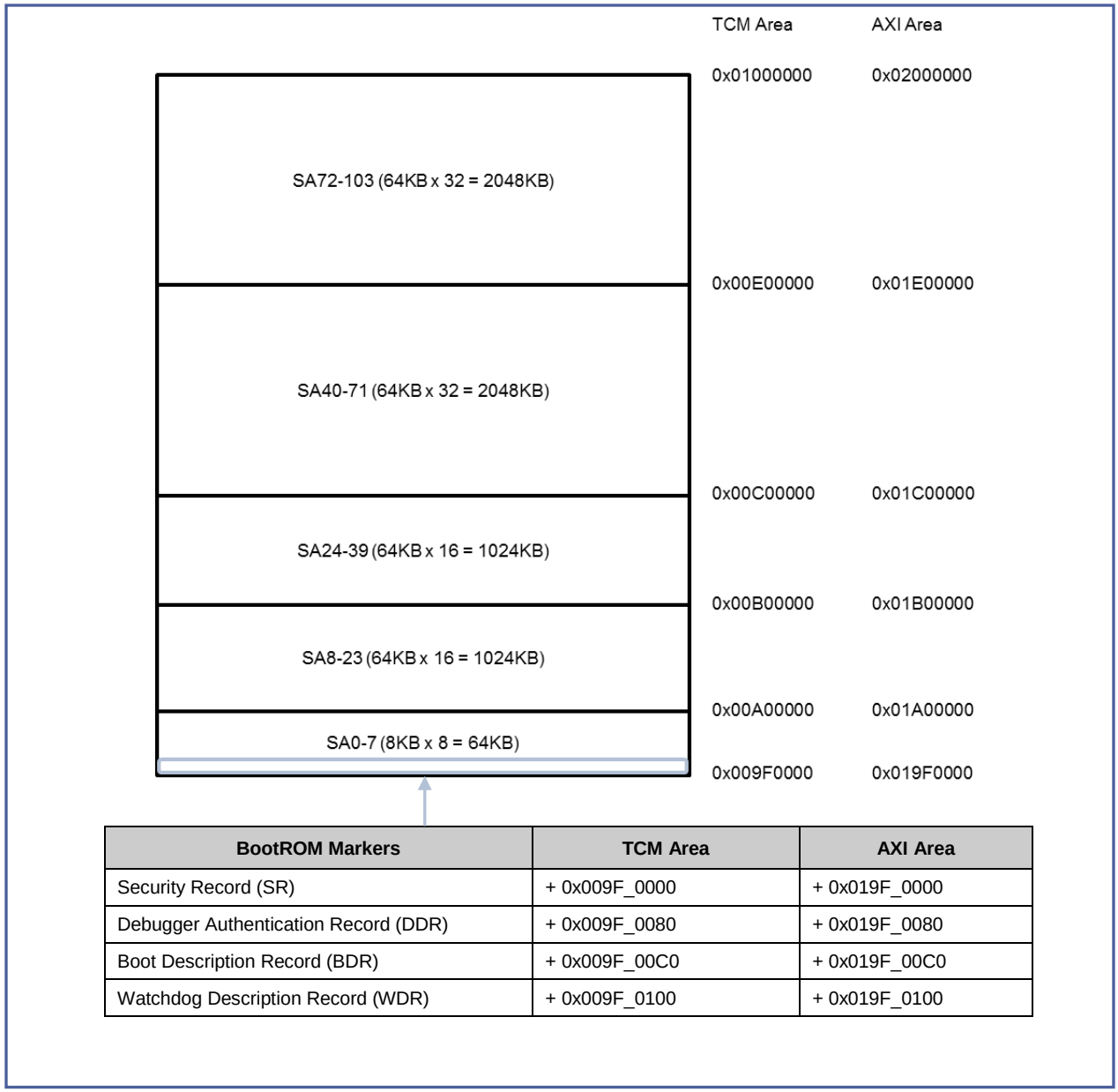
- Determines the mode performs mode judgment
- Sets the security
- Sets the watchdog timer.
- Reads the BootROM marker and determines the hardware behavior.

2.3 BootROM Markers

BootROM markers are registers in the TCFLASH. Figure 3 and Figure 4 show the BootROM marker structure.

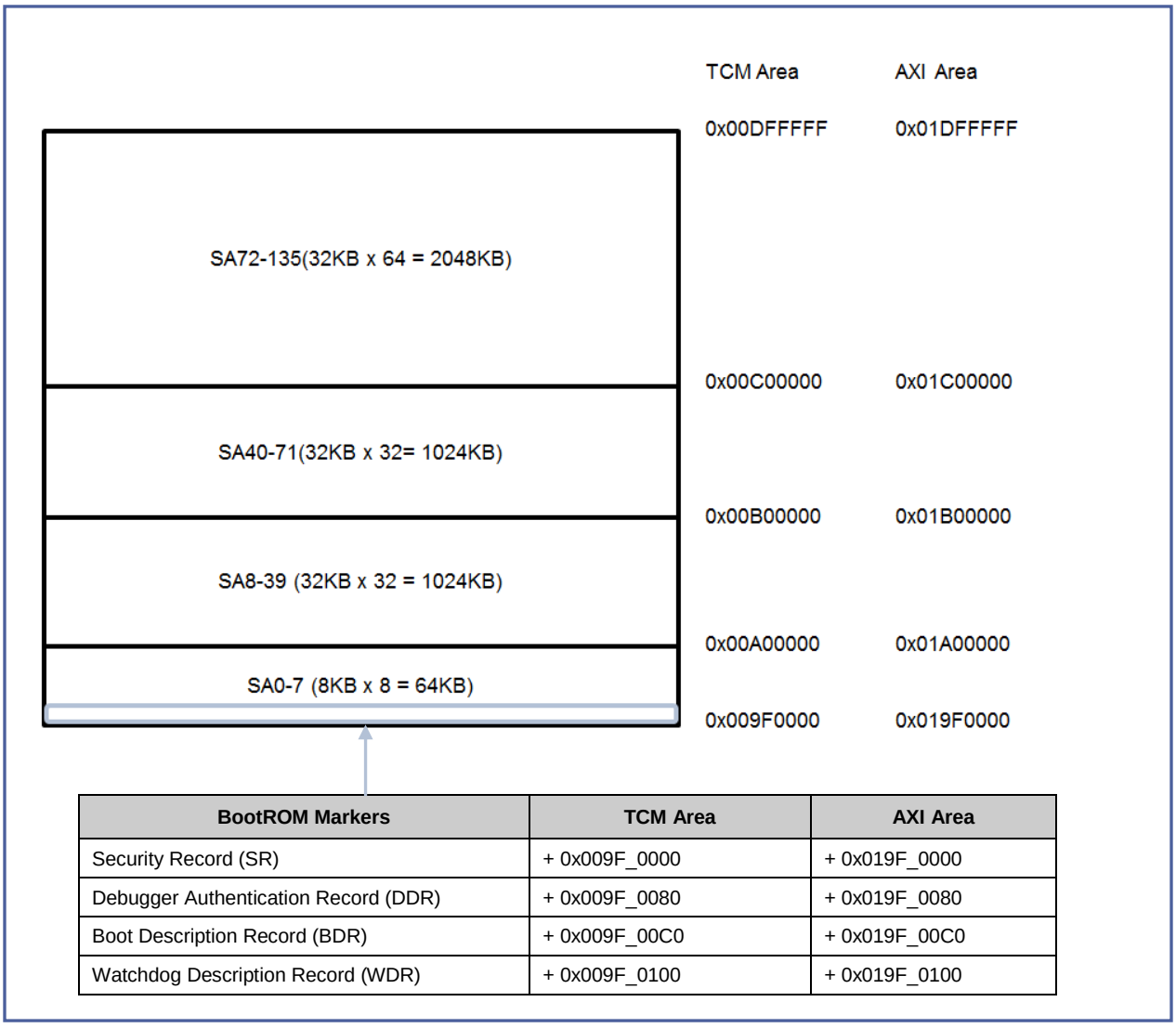
A BootROM marker is different from a usual register, and the register value does not control the function of the hardware directly. It is read by BootROM software, and the function of the hardware is controlled via BootROM software processing.

Figure 3. BootROM Marker Structure of S6J3110/S6J3120/S6J3200 Series



BootROM Markers	TCM Area	AXI Area
Security Record (SR)	+ 0x009F_0000	+ 0x019F_0000
Debugger Authentication Record (DDR)	+ 0x009F_0080	+ 0x019F_0080
Boot Description Record (BDR)	+ 0x009F_00C0	+ 0x019F_00C0
Watchdog Description Record (WDR)	+ 0x009F_0100	+ 0x019F_0100

Figure 4. BootROM Marker Structure of S6J3300/S6J3350/S6J3360/S6J3370/S6J3400 Series



BootROM Markers	TCM Area	AXI Area
Security Record (SR)	+ 0x009F_0000	+ 0x019F_0000
Debugger Authentication Record (DDR)	+ 0x009F_0080	+ 0x019F_0080
Boot Description Record (BDR)	+ 0x009F_00C0	+ 0x019F_00C0
Watchdog Description Record (WDR)	+ 0x009F_0100	+ 0x019F_0100

2.3.1 Types of BootROM Markers

There are four types of BootROM markers:

- Security Record (SR): This marker is used for configuring the security settings (such as security enable, chip erase enable, and work/code flash sector write permissions).
- Debugger Authentication Record (DDR): This marker is used for configuring the debugger connection settings (such as debugger connection enable and debugger security key settings).
- Boot Description Record (BDR): This marker is used for configuring the startup settings (such as SHE secure boot mode and alternative boot vector enable settings).
- Watchdog Description Record (WDR): This marker is used for configuring the hardware watchdog timer settings (such as hardware watchdog interrupt, trigger, and lower/upper limit settings).

3 BootROM Interface Settings

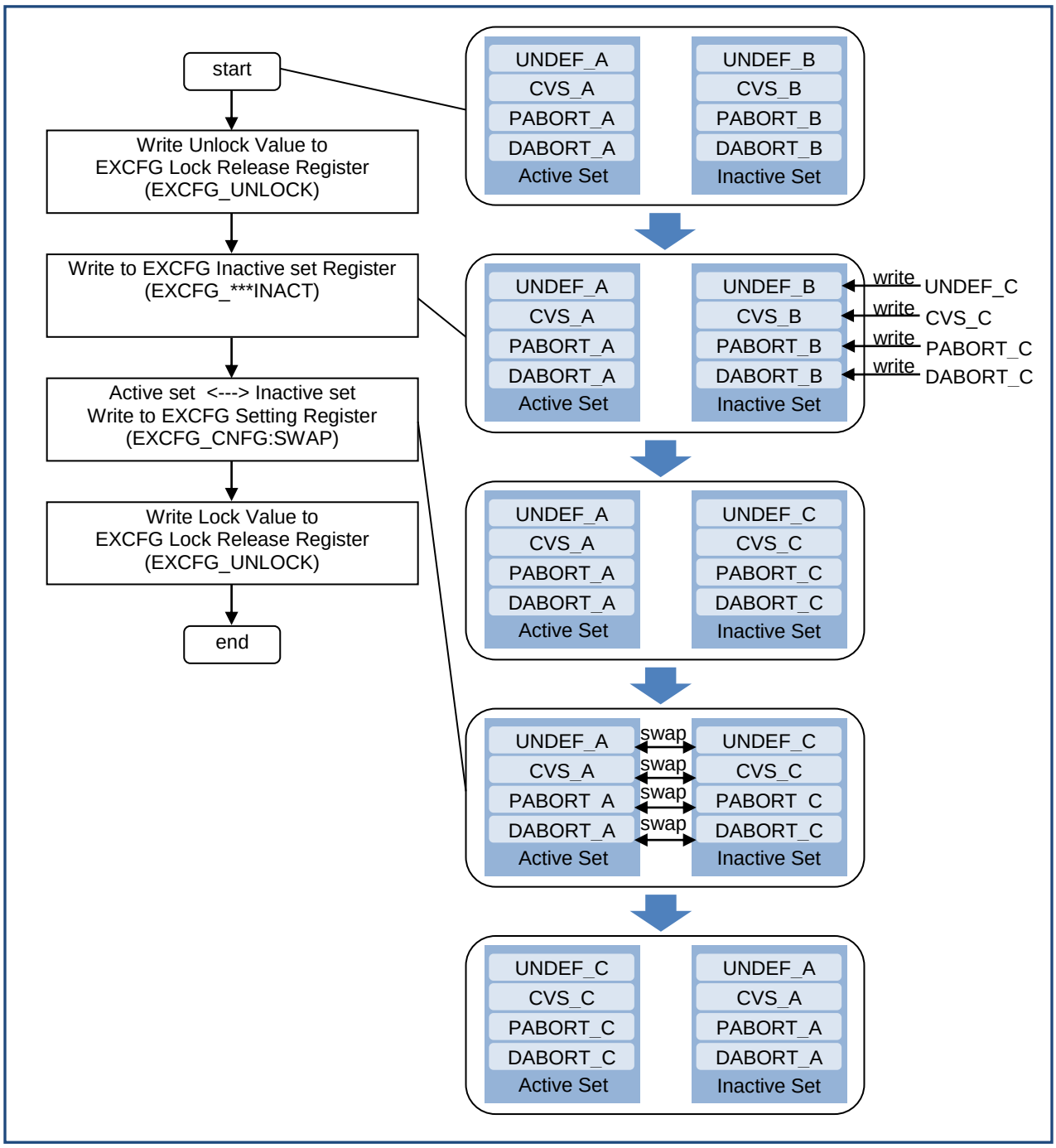
Specific settings are required to use the BootROM interface of the S6J3110/ S6J3120/ S6J3200/ S6J3300/ S6J3350/ S6J3360/ S6J3370/ S6J3400 series MCUs. This section describes how to set up the BootROM hardware interface and BootROM software interface.

3.1 BootROM Hardware Interface Settings

With the BootROM hardware interface, you can configure the user-defined exception handler. It contains both the active and inactive sets of the exception vector registers. These registers can be used to set the starting address of the exception handler for the undefined instruction exception, the supervisor call exception, the prefetch abort exception, and the data abort exception. Writing a '1' to the set swap bit of the setting register (EXCFG_CNFG:SWAP) swaps the contents of the active and inactive sets. This action changes the settings of the active set.

Figure 5 shows an example of the BootROM hardware interface flow chart.

Figure 5. Example of BootROM Hardware Interface Flow Chart



1. Write an unlock value (0xACC5B007) to the EXCFG Lock Release Register to unlock the BootROM hardware interface.
2. Write the setting to the inactive set of the exception vector register.
3. Write the setting to the swap bit of the EXCFG Setting Register so the inactive set is swapped for the active set.
4. Write the lock value (0xB007ECF6) to the EXCFG Lock Release Register to lock the BootROM hardware interface.

To use the BootROM hardware interface, it is necessary to set the registers listed in Table 1.

Table 1. BootROM Hardware Interface Registers

Register Abbreviation	Bit Name	Description
EXCFG_UNLOCK	UNLOCK	Controls the write lock of the register
EXCFG_CNFG	SWAP	Swaps the contents of the exception vector registers
EXCFG_UNDEFINACT	UNDEFVEC	Starting address of the exception handler for the undefined instruction exception (inactive set)
EXCFG_SVCINACT	SVCVEC	Starting address of the exception handler for the supervisor call exception (inactive set)
EXCFG_PABORTINACT	PABORTVEC	Starting address of the exception handler for the prefetch abort exception (inactive set)
EXCFG_DABORTINACT	DABORTVEC	Starting address of the exception handler for the data abort exception (inactive set)
EXCFG_UNDEFACT	UNDEFVEC	Starting address of the exception handler for the undefined instruction exception (active set)
EXCFG_SVCACT	SVCVEC	Starting address of the exception handler for the supervisor call exception (active set)
EXCFG_PABORTACT	PABORTVEC	Starting address of the exception handler for the prefetch abort exception (active set)
EXCFG_DABORTACT	DABORTVEC	Starting address of the exception handler for the data abort exception (active set)

3.2 BootROM Software Interface Settings

Using the BootROM software interface, it is possible to configure the security settings and the hardware watchdog timer settings with BootROM markers. The BootROM marker is a register that is located at the head of the SA0 area in a small sector area of the TCFLASH. As described in Table 2, write the setting value of each marker to the appropriate address. Table 3 shows example settings for the Security Record (SR).

Table 2. BootROM Marker Addresses

BootROM Marker	TCM Area	AXI Area
Security Record (SR)	+ 0x009F_0000	+ 0x019F_0000
Debugger Authentication Record (DDR)	+ 0x009F_0080	+ 0x019F_0080
Boot Description Record (BDR)	+ 0x009F_00C0	+ 0x019F_00C0
Watchdog Description Record (WDR)	+ 0x009F_0100	+ 0x019F_0100

Table 3. Example Settings for Security Record (SR) for Flash Protection

Address	Setting Value for Flash Protection
0x019F_0000 (MK_SER)	0x00000001
0x019F_0004	0xFFFFFFFF
0x019F_0008 (MK_SSR)	0xFFFFFFFF
0x019F_000C	0xFFFFFFFF

The next section describes the setting value of each marker.

3.2.1 Security Record (SR) Settings

This marker can be used to configure security settings, as described in Table 4.

Table 4. Security Record Settings

Register Abbreviation	Bit Name	Setting
MK_SER (common parameter)	SER	This marker is used to enable the security feature. When the setting value is 0x00000001, the security is enabled. When the setting value is 0xFFFFFFFF, the security is disabled. Any other value enables the security, but should not be used.
MK_SSR (common parameter)	SSR	This marker is used to set the security scope of the device. When the setting value is 0xFFFFFFFF, the security scope is set to “Flash protection.” When the setting value is “Other than above,” the security scope is set to “Device protection.”
MK_SOER (common parameter)	SOER	This marker is used to set security register overwrite. When the setting value is 0xFFFFFFFF, the security settings in the registers can be overwritten. Any other value disables the write access to the security settings registers.
MK_SWPOER (common parameter)	SWPOER	This marker is used to set sector write permission overwrite. When the setting value is 0xFFFFFFFF, the sector write permissions can be overwritten. Any other value disables the write access to the sector write permission registers.
MK_WSWPR (S6J3110/S6J3120/S6J3200)	WSWP	This marker is used to set the sector permission for WorkFlash sector SA0-SA13. When the setting value is ‘0’, WorkFlash SA0-SA13 cannot be programmed or erased. When the setting value is ‘1’, WorkFlash SA0-SA13 program or erase is permitted.
MK_C0SWPR (S6J3110/S6J3120/S6J3200)	C0SWP	This marker is used to set the sector permission for TCFlash small sector SA0-SA7. When the setting value is ‘0’, TCFlash SA0-SA07 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA0-SA07 program or erase is permitted.
MK_C1SWPR (S6J3110/S6J3120/S6J3200)	C1SWP	This marker is used to set the sector permission for TCFlash large sector SA8-SA39. When the setting value is ‘0’, TCFlash SA8-SA39 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA8-SA39 program or erase is permitted.
MK_C2SWPR (S6J3110/S6J3120/S6J3200)	C2SWP	This marker is used to set the sector permission for TCFlash large sector SA40-SA71. When the setting value is ‘0’, TCFlash SA40-SA71 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA40-SA71 program or erase is permitted.
MK_CSWP0 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	CSWP0	This marker is used to set the sector permission for TCFlash small sector SA0-SA7. When the setting value is ‘0’, TCFlash SA0-SA7 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA0-SA7 program or erase is permitted.
MK_CSWP1 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	CSWP1	This marker is used to set the sector permission for TCFlash large sector SA8-SA39. When the setting value is ‘0’, TCFlash SA8-SA39 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA8-SA39 program or erase is permitted.
MK_CSWP2 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	CSWP2	This marker is used to set the sector permission for TCFlash large sector SA40-SA71. When the setting value is ‘0’, TCFlash SA40-SA71 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA40-SA71 program or erase is permitted.
MK_CSWP3 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	CSWP3	This marker is used to set the sector permission for TCFlash large sector SA72-SA103. When the setting value is ‘0’, TCFlash SA72-SA103 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA72-SA103 program or erase is permitted.
MK_CSWP4 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	CSWP4	This marker is used to set the sector permission for TCFlash large sector SA104-SA135. When the setting value is ‘0’, TCFlash SA104-SA135 cannot be programmed or erased. When the setting value is ‘1’, TCFlash SA104-SA135 program or erase is permitted.
MK_WSWP0 (S6J3300/S6J3350/S6J3360/ S6J3370/S6J3400)	WSWP0	This marker is used to set the sector permission for WorkFlash sector SA0-SA27. When the setting value is ‘0’, WorkFlash SA0-SA27 cannot be programmed or erased. When the setting value is ‘1’, WorkFlash SA0-SA27 program or erase is permitted.

3.2.2 Debugger Authentication Record (DDR) Settings

This marker can be used to set the permissions for the debugger connection and its authentication key, as described in Table 5.

Table 5. Debugger Authentication Record Settings

Register Abbreviation	Bit Name	Setting
DDR_DSM	DSEM	This marker is used to enable debugger connection. Setting the value of the debugger connection permission to 0x59F71234 enables the debugger connection.
DDR_DSKM[0-3]	DSKM	This marker is used to set a 128-bit authentication key for debugger connection when a debugger connection is enabled by DDR_DSM.

If security is disabled with the SR MK_SER register, a 128-bit authentication key is not required when a debugger is connected.

3.2.3 Boot Description Record (BDR) Settings

This marker is used to decide the behavior of the SHE secure boot. These markers are used to set the SHE secure boot mode, the size of the area for the program to be tested with the SHE, and the start address of the area for the program to be tested with SHE, as described in Table 6.

Table 6. Boot Description Record Settings

Register Abbreviation	Bit Name	Setting
BDR_SBMM	SBMM	This marker is used to select between the “Measuring during application” mode and “Measuring before application” mode. The setting value of “Measuring during application” mode is 0x00000000. The setting value of “Measuring before application” mode is other than the above.
BDR_SBSM	SBSM	This marker is used to set the size of the area for the program to be tested with the SHE in bytes.
BDR_DWEM	DWEM	This marker is used to specify whether to enable wait for debugger connection after clearing a hard reset. When the setting value is 0x292D3A7B, “Wait for debugger connection after clearing a hard reset” is not enabled. When the setting value is “Other than above,” “Wait for debugger connection after clearing a hard reset” is enabled.
BDR_ABVM	ABVM	This marker is used to set the start address of a user program.
BDR_ABVEM	ABVEM	This marker is used to enable setting the start address of a user program. When the setting value is 0x292D3A7B, the start address of a user program is the value set in the Alternative Boot Vector Marker (BDR_ABVM). When the setting value is “Other than above,” the start address of a user program is 0x00A00000 (fixed address).

3.2.4 Watchdog Description Record (WDR) Settings

This marker can be used to configure the hardware watchdog timer, as described in Table 7.

If WDR_CEM is enabled, the hardware watchdog timer will be started based on the settings defined with the WDR. If WDR_CEM is disabled, the hardware watchdog timer will be started based on the default settings .

Table 7. Watchdog Description Record Settings

Register Abbreviation	Bit Name	Setting
WDR_INTM	RSTENM	This marker is used to set the value that controls the output when a watchdog error occurs. When the setting value is ‘0’, an NMI (Non Maskable Interrupt) is generated. When the setting value is ‘1’, a reset is generated.
	IRQENM	This marker is used to specify whether to enable a prior warning interrupt. When the setting value is ‘0’, it is not enabled. When the setting value is 1, it is enabled.

Register Abbreviation	Bit Name	Setting
WDR_TRG0CFGM	WDGTRG0CFGM	This marker is used to set a counter clear value.
WDR_TRG1CFGM	WDGTRG1CFGM	This marker is used to set a counter clear value.
WDR_RUNLLM	WDGRUNLLM	This marker is used to set a window lower limit value for RUN mode. The lower limit of the range that can clear the watchdog counter is set in this marker. When the setting value is '0', the window function is disabled.
WDR_RUNULM	WDGRUNULM	This marker is used to set a window upper limit value for RUN mode. The upper limit of the range that can clear the watchdog counter is set in this marker.
WDR_PSSLLM	WDGPSSLLM	This marker is used to set a window lower limit value for PSS mode. The lower limit of the range that can clear the watchdog counter is set in this marker. When the setting value is 0, the window function is disabled.
WDR_PSSULM	WDGPSSULM	This marker is used to set a window upper limit value for PSS mode. The upper limit of the range that can clear the watchdog counter is set in this marker.
WDR_RSTDLYM	WDGRSTDLYM	This marker is used to set the number of cycles for the delay time that is to be inserted before generating a watchdog reset request or watchdog interrupt request.
WDR_CFGM	OBSSELM	This marker is used to select any bit in the watchdog counter (32 bits) as the output of the watchdog counter monitor bit.
	CLKSELM	This marker is used to set a source clock of the watchdog counter. When the setting value is '0', it is high-speed CR clock. When the setting value is '1', it is low-speed CR clock.
WDR_CEM	CEM	This marker is used to enable various markers of the hardware watchdog. When the setting value is 0x292D3A7B, the hardware watchdog timer will be started based on the settings defined with the WDR. When the setting value is "Other than above," the hardware watchdog timer will be started based on the default settings .

4 Summary

The BootROM interface can help you change the initialization values set at MCU startup. This application note guides you in implementing the BootROM interface with the S6J3110/ S6J3120/ S6J3200/ S6J3300/ S6J3350/ S6J3360/ S6J3370/ S6J3400 series easily.

5 Related Documents

- S6J311E/D/C/B Series Datasheet (Doc. No.002-05681)
- S6J311A/9/8 Series Datasheet (Doc. No.002-04632)
- S6J3110 Series Hardware Manual (Doc.No.002-10667)
- S6J3120 Series Datasheet (Doc.No.002-04863)
- S6J3120 Series Hardware Manual (Doc.No.002-04855)
- S6J3200 Series Datasheet (Doc.No.002-05682)
- S6J3200 Series Hardware Manual (Doc.No.002-04852)
- S6J32E/F/G Series Datasheet (Doc.No.002-10689)
- S6J32E/F/G Series Hardware Manual (Doc.No.002-12500)
- Traveo Family Hardware Manual Platform Part for S6J3200 Series (Doc.No.002-04854)
- S6J3310/20/30/40 Series Datasheet (Doc.No.002-10635)
- S6J3350 Series Datasheet (Doc.No.002-10634)
- S6J3310/20/30/40/50 Series Hardware Manual (Doc.No.002-10185)
- Traveo Family HardwareManual Platform Part for S6J3310/3320/3330/3340/3350 Series (Doc.No.002-07884)
- S6J3360/70 Series Datasheet (Doc.No.002-03359)
- S6J3360/70 Series Hardware Manual (Doc.No.002-18302)
- Traveo Family HardwareManual Platform Part for S6J3360/3370 Series (Doc.No.002-07884)
- S6J3400 Series Datasheet (Doc.No.001-97829)
- S6J3400 Series Hardware Manual (Doc.No.002-09919)
- Traveo Family Hardware Manual Platform Part for S6J3400 Series (Doc.No.002-07884)

Document History

Document Title: AN209716 – Implementing the BootROM Interface with Traveo™ Family

Document Number: 002-09716

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5062215	HNIS	05/13/2016	New application note
*A	5293748	HNIS	06/15/2016	Added target products S6J3200/S6J3300/S6J3350/S6J3400 series.
*B	5668243	HNIS	03/30/2017	Added S6J3360/S6J3370 series to target products. Updated the Related Documents section

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709