

## Configure Boot Memory as Extension of Main Memory (AP Mode)

---

### Introduction

This application note introduces the method and sample routines of configuring the boot memory of MCU with AP mode as the extension of main memory.

Applicable products:

Part number	AT32 (with AP mode)
-------------	---------------------

## Contents

<b>1</b>	<b>Overview .....</b>	<b>5</b>
<b>2</b>	<b>Sample routines .....</b>	<b>6</b>
2.1	Use Artery ICP Programmer to set boot memory as main memory extension .....	6
2.2	run_in_boot_memory: execute application in boot memory .....	6
<b>3</b>	<b>Revision history .....</b>	<b>10</b>

## List of Tables

Table 1. Document revision history..... 10

## List of Figures

Figure 1. ICP interface .....	6
Figure 2. Set boot memory address range in IROM2 .....	7
Figure 3. Configure .c file.....	7
Figure 4. Configure c. file compile address .....	8
Figure 5. Scatter file.....	8
Figure 6. Add FLASH algorithm file .....	9

# 1 Overview

Boot memory is used by default to store the original boot code in BOOT mode. For MCUs with AP mode, the boot memory can also be configured as the extension of main memory (AP mode) to store user-defined code.

***Note: The boot memory AP mode can only be set once irreversibly, and the original boot memory BOOT mode function cannot be restored after setting.***

This application note takes AT32F415 series as an example to explain the use of extension of main memory.

- Section 2.1 introduces how to use Artery ICP Programmer to enable AP mode of boot memory before configuring it as the extension of main memory.
- Section 2.2 introduces the “run\_in\_boot\_memory” project.
- The routine is saved in BSP standard library in utilities\at32xx\_boot\_memory\_ap\_demo\run\_in\_boot\_memory.

## 2 Sample routines

### 2.1 Use Artery ICP Programmer to set boot memory as main memory extension

Use ICP Programmer and complete the following procedures.

- Connect J-Link or AT-Link emulator to AT32F415 and power on;
- Open ICP programmer, and select J-Link or AT-Link;
- Click “Target” -- “Boot memory AP mode”, as shown below.

Figure 1. ICP interface



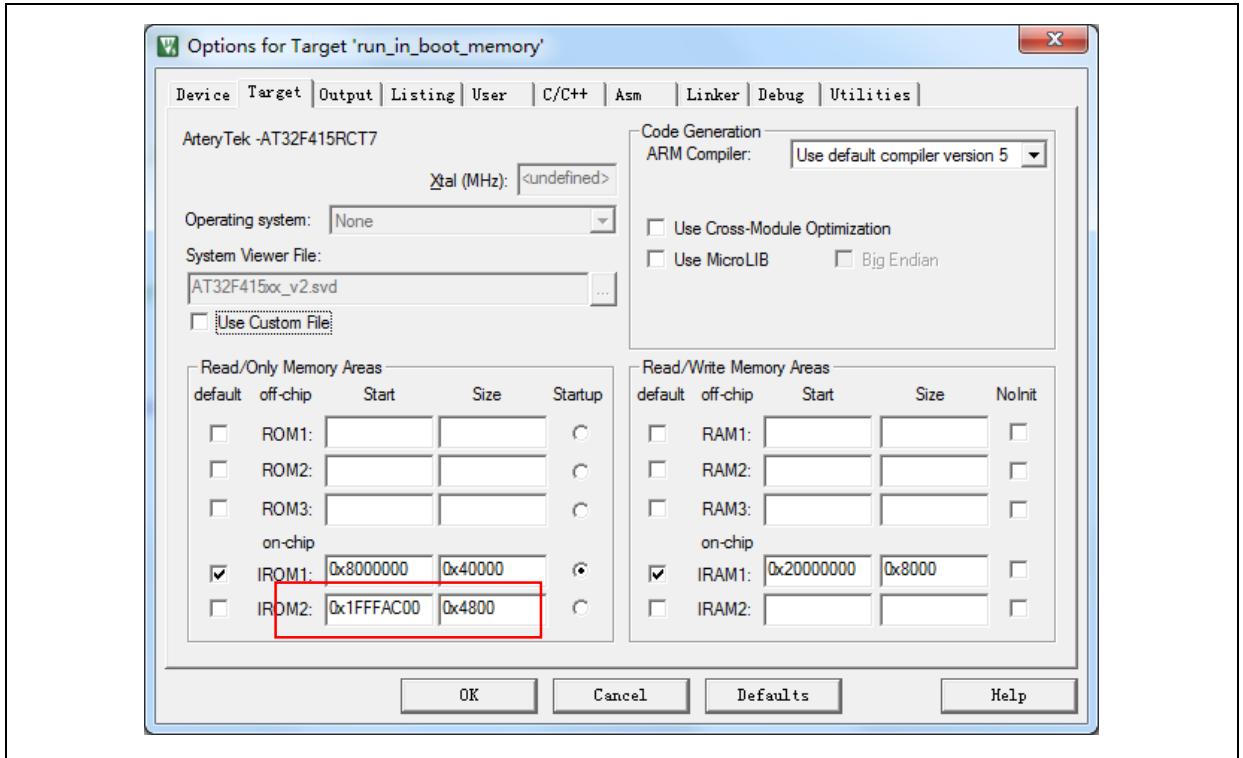
- In order to prevent misoperation, users need to manually enter the enable key 0xA35F6D24, and then a success or failure prompt will appear in the “Flash info”.
- For details about ICP Programmer, refer to ICP Programmer user manual.

### 2.2 run\_in\_boot\_memory: execute application in boot memory

In this routine, hardware boots from the main memory. The code of main memory outputs information from the serial port USART1, and then the code in boot memory will initialize LED light and enable LED2 flashes. Users need configure Keil as follows before executing application in boot memory.

- Set the start address and range of boot memory.

**Figure 2. Set boot memory address range in IROM2**



- This routine writes the application code to be loaded into boot memory into a separate “.c” file. Relevant code enabling LED light flashes is saved in “run\_in\_boot\_memory.c”. Right click on the “.c” file to open “Options”, and compile the address to IROM2.

**Figure 3. Configure .c file**

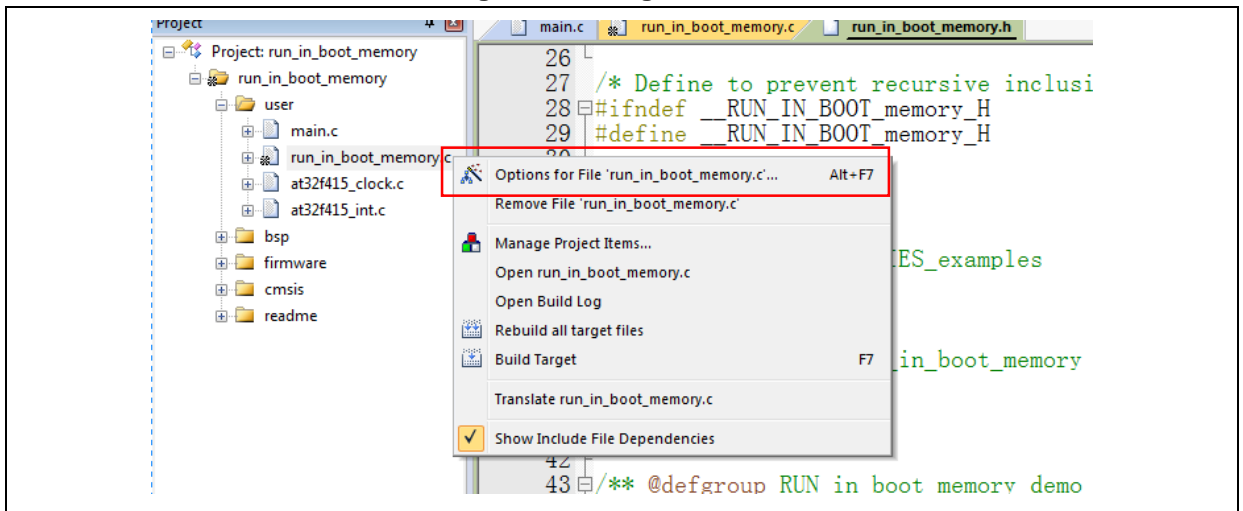
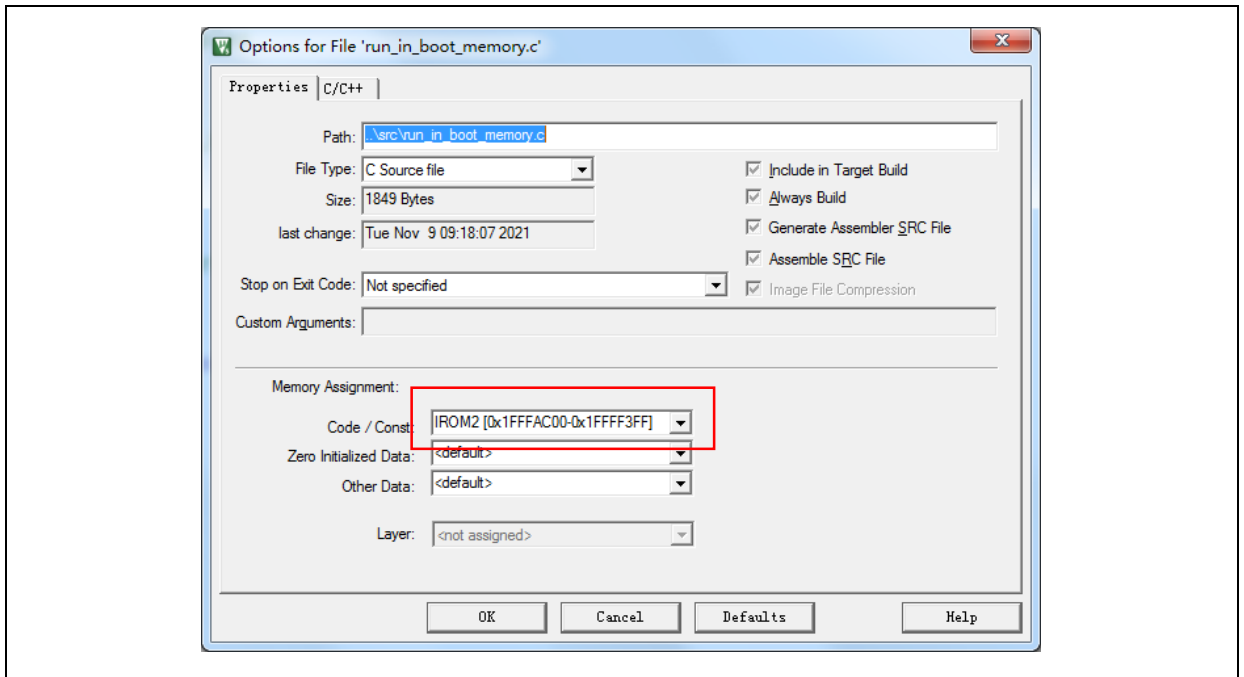
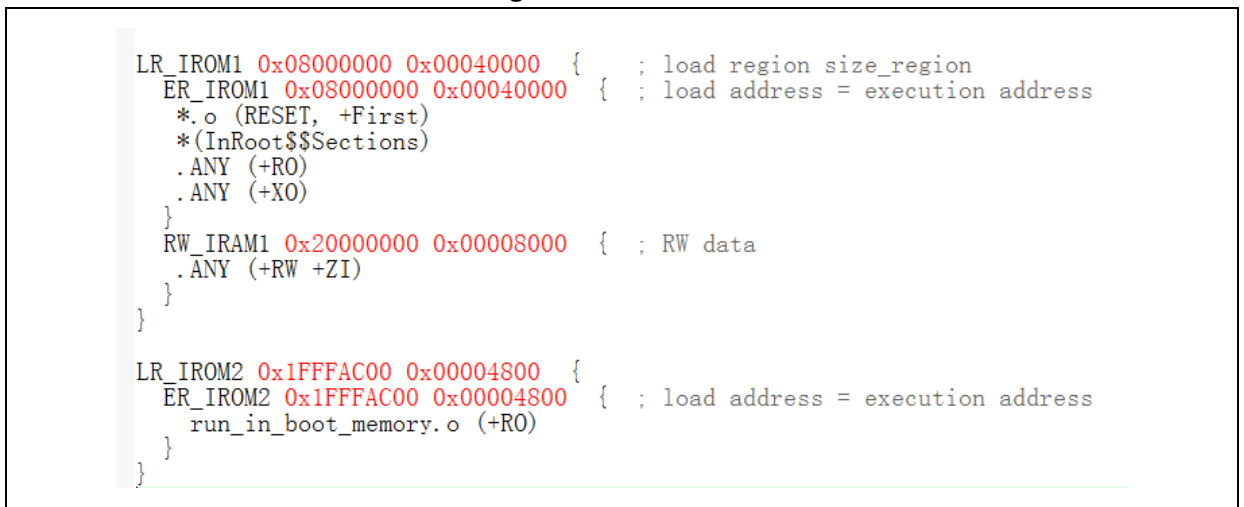


Figure 4. Configure c. file compile address



- After compiling, check the scatter file, and the object file “run\_in\_boot\_memory.o” is in boot memory.

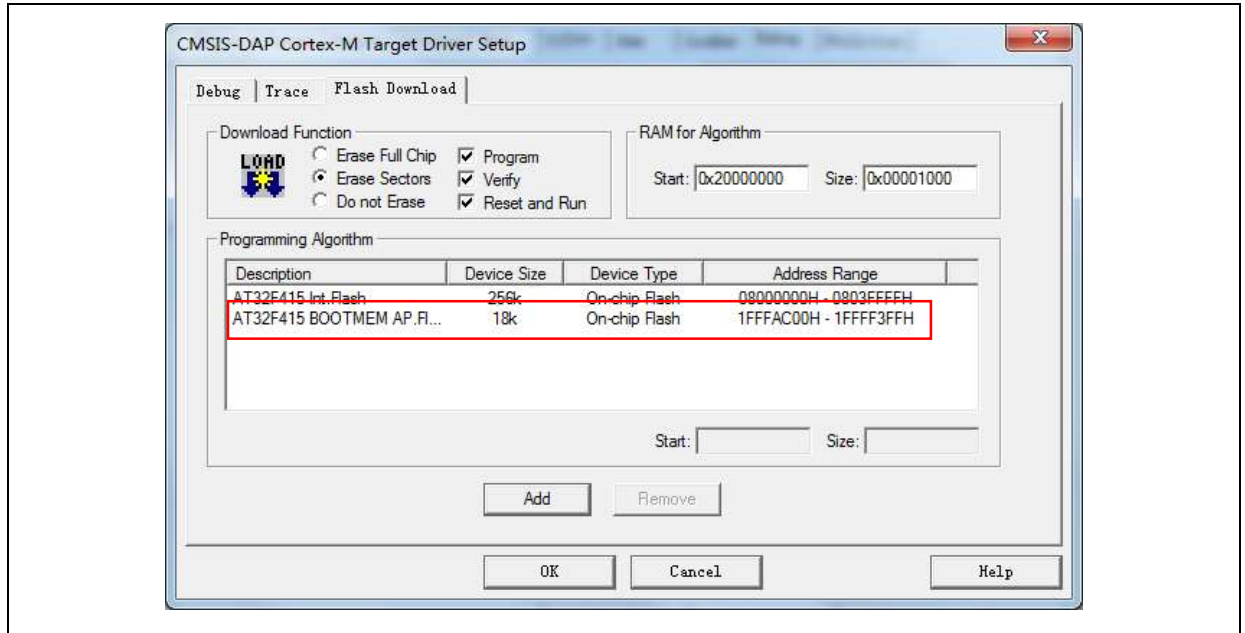
Figure 5. Scatter file



- Add Flash algorithm to boot memory.



Figure 6. Add FLASH algorithm file



- Download and run, and the LED of the corresponding execution flashes.

### 3 Revision history

Table 1. Document revision history

Date	Version	Revision note
2022.01.19	2.0.0	Initial release

## IMPORTANT NOTICE – PLEASE READ CAREFULLY

Purchasers are solely responsible for the selection and use of ARTERY's products and services; ARTERY assumes no liability for purchasers' selection or use of the products and the relevant services.

No license, express or implied, to any intellectual property right is granted by ARTERY herein regardless of the existence of any previous representation in any forms. If any part of this document involves third party's products or services, it does NOT imply that ARTERY authorizes the use of the third party's products or services, or permits any of the intellectual property, or guarantees any uses of the third party's products or services or intellectual property in any way.

Except as provided in ARTERY's terms and conditions of sale for such products, ARTERY disclaims any express or implied warranty, relating to use and/or sale of the products, including but not restricted to liability or warranties relating to merchantability, fitness for a particular purpose (based on the corresponding legal situation in any unjudicial districts), or infringement of any patent, copyright, or other intellectual property right.

ARTERY's products are not designed for the following purposes, and thus not intended for the following uses: (A) Applications that have specific requirements on safety, for example: life-support applications, active implant devices, or systems that have specific requirements on product function safety; (B) Aviation applications; (C) Auto-motive application or environment; (D) Aerospace applications or environment, and/or (E) weapons. Since ARTERY products are not intended for the above-mentioned purposes, if purchasers apply ARTERY products to these purposes, purchasers are solely responsible for any consequences or risks caused, even if any written notice is sent to ARTERY by purchasers; in addition, purchasers are solely responsible for the compliance with all statutory and regulatory requirements regarding these uses.

Any inconsistency of the sold ARTERY products with the statement and/or technical features specification described in this document will immediately cause the invalidity of any warranty granted by ARTERY products or services stated in this document by ARTERY, and ARTERY disclaims any responsibility in any form.