

ABOV Cortex-Mx 사용자 Bootloader Guide

2019-03-08



■ 소개

- 본 문서는 ABOV 32비트 Cortex-Mx MCU의 사용자 bootloader를 사용하기 위한 문서입니다.
- 사용자 bootloader는 PC와 MCU 사이 uart 통신을 통하여 사용자 어플리케이션 코드를 MCU flash에 업데이트 할 수 있는 것입니다.
- 부트로더 프로젝트, 어플리케이션 프로젝트와 PC를 사용한 uart 다운로드로 구성되어 있으며, keil 환경에서 동작하도록 구성되어 있습니다.

- 자료는 다음과 같은 폴더로 구성되어 있습니다.
 - DeviceName_keil_bootcode 폴더
 - » Bootloader 관련 Project
 - DeviceName_keil_application 폴더
 - » Application 관련 Project
 - Downloader for UART_v1.xx 폴더
 - » UART를 이용하여 Application 코드를 updat하는 PC 용 s/w 프로그램

Flash 메모리 영역 구분

• Boot 영역

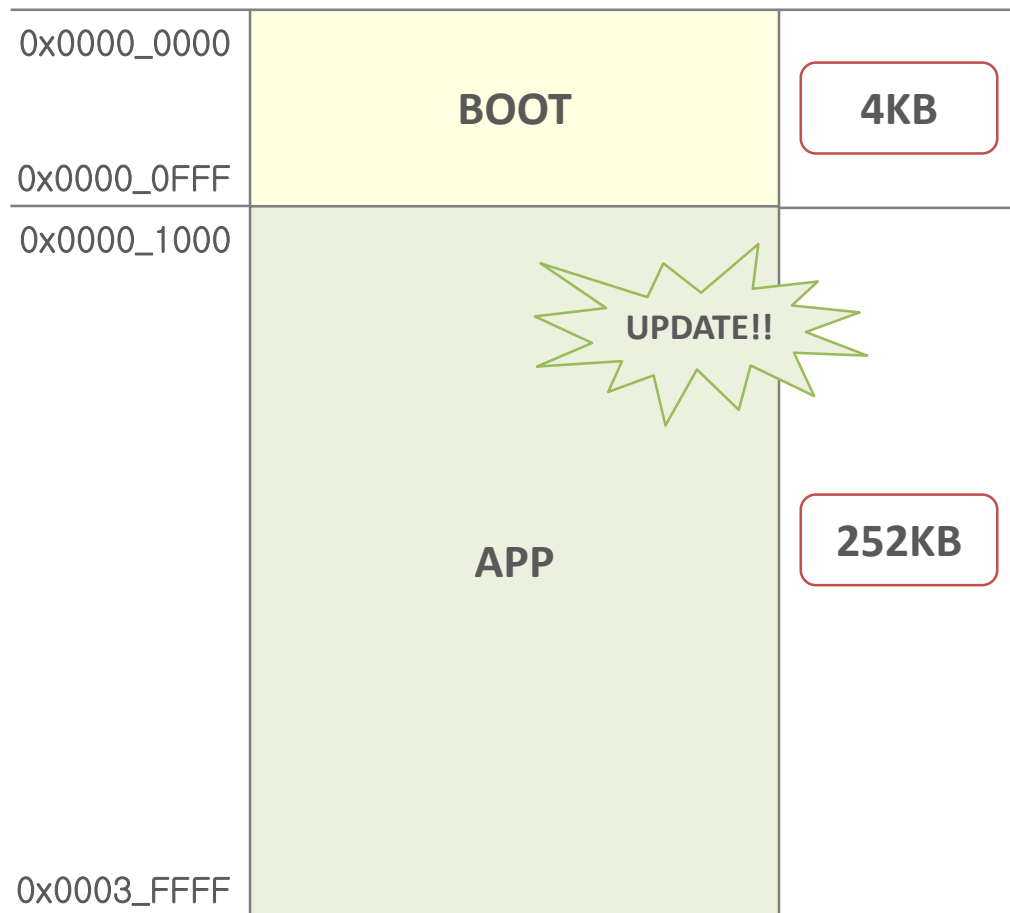
- » Application 영역을 re-write하기 위한 Boot code 저장 영역
- » Re-write 불가능
- » 필요한 만큼 할당하여 사용(현재 4KB)

• Application 영역

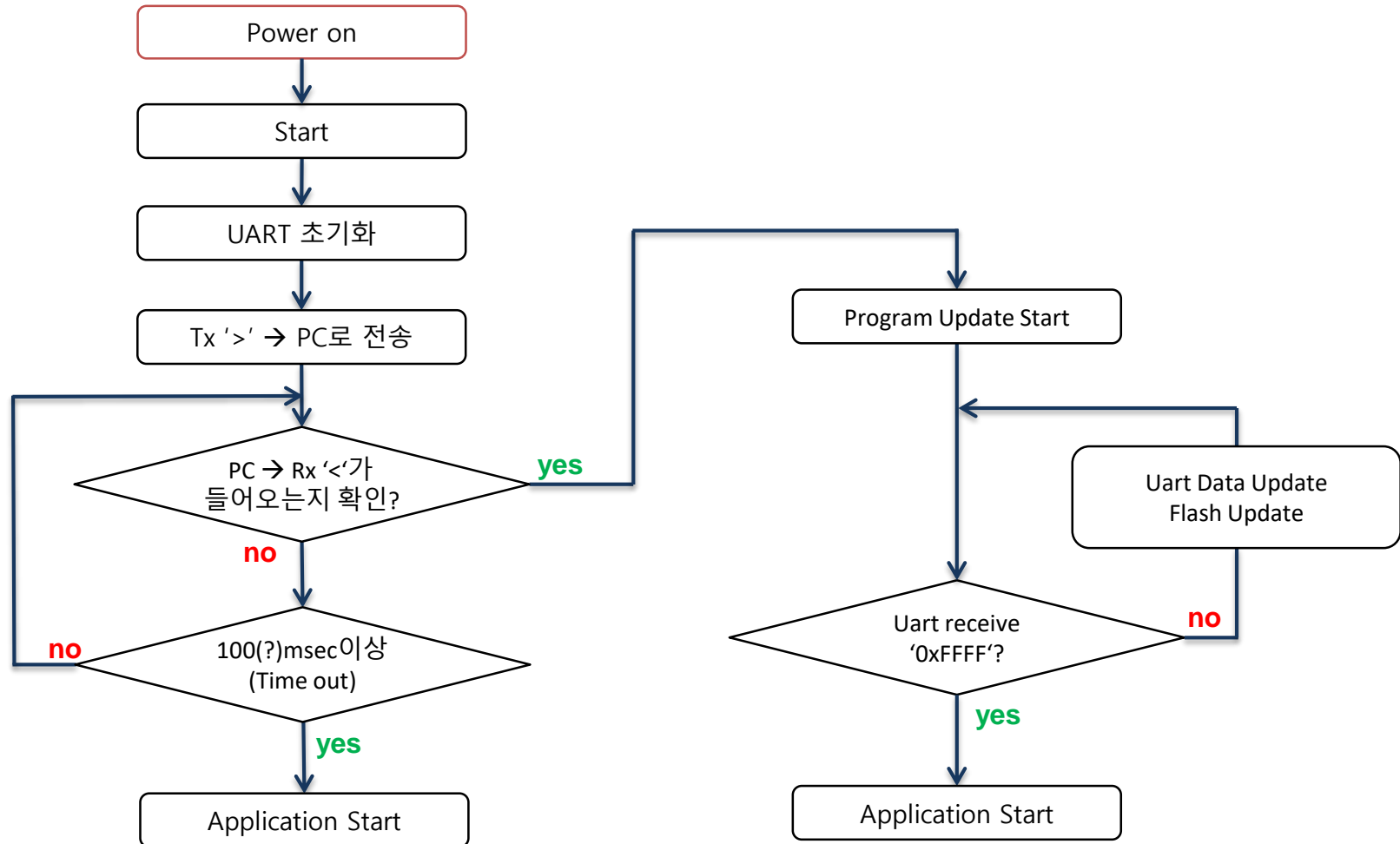
- » 사용자 코드 저장 영역
- » Boot 영역을 통해 re-write 가능
- » Boot 영역 이외 영역 사용 가능

• 옆 그림은 메모리의 사용 영역을 표시합니다.

- » 예) A33G526 Flash 구성



■ Bootloader 흐름도



※ Time out 시간은 system 에 따라 수정 가능.

- Directory : DeviceName_keil_bootcode → bootcode project

1. startup_(DeviceName)_bootkeil.s

- a. Application Start Address 정의 (Boot Area hard lock은 '0x0000 ~ 0x0FFF')
- b. _fjump 를 사용하여 Application 으로 점프하여 실행

```
AREA |.text|, CODE, READONLY
; jump application start
_fjump      PROC
            EXPORT      _fjump

            MOV         R0, #0x00001000 ;;application start address
            LDR         SP, [R0]
            LDR         PC, [R0, #4]
            NOP
            NOP
            NOP
            NOP

            ENDP
```

2. main.c

- a. “c” 파일에서 extern으로 정의하여 사용

```
//// Assembly jump code
extern void _fjump(void);
...
_fjump();
```

- Directory : DeviceName_keil_bootcode → bootcode project

3. def.h 혹은 defboot.h 내에 MCU/bootloader info 를 설정

설정은 해당 디바이스에 맞게 정의된 값을 지정한다. (MCU->PC로 전송되는 정보)

a. 예) A33G526 info

- BootSize : Boot4096 ('f')
- DeviceID : ABOV ('A')
- FlashSize : Flash256k ('q')
- PageSize : Page1024 ('W') <- 한번에 전송되는 erase/program size
- PageByte : 프로그램 내에서 사용될 Pagesize와 동일한 숫자 (1024)
- NSHIFTPAGE : 프로그램 내에서 사용될 어드레스 계산용 shift 숫자 ($1024 = 2^{10}$)

```
#define BootSize    Boot4096
#define DeviceID    ABOV
#define FlashSize   Flash256k
#define PageSize    Page1024
#define EEpromSize  EEprom64
#define PageByte    1024
#define NSHIFTPAGE  10
```

Def.h 내 A33G526 info 설정 예

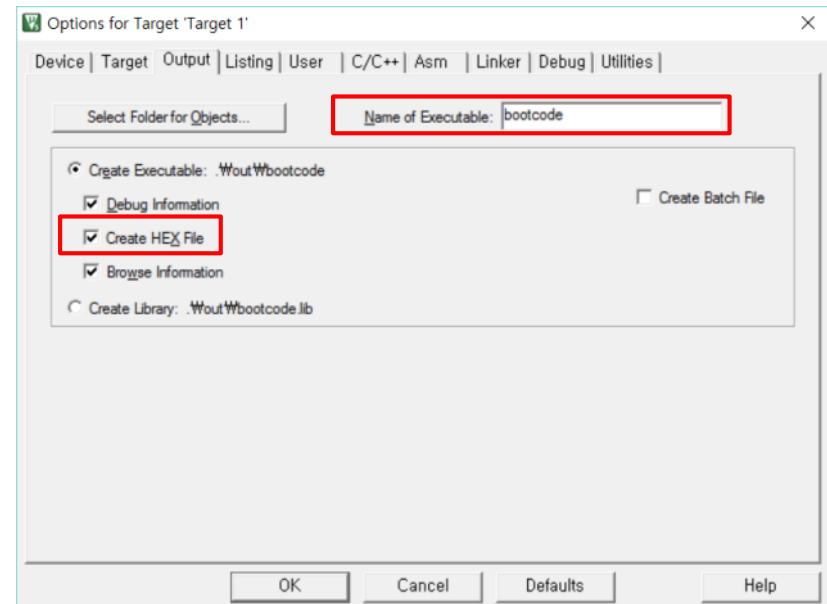
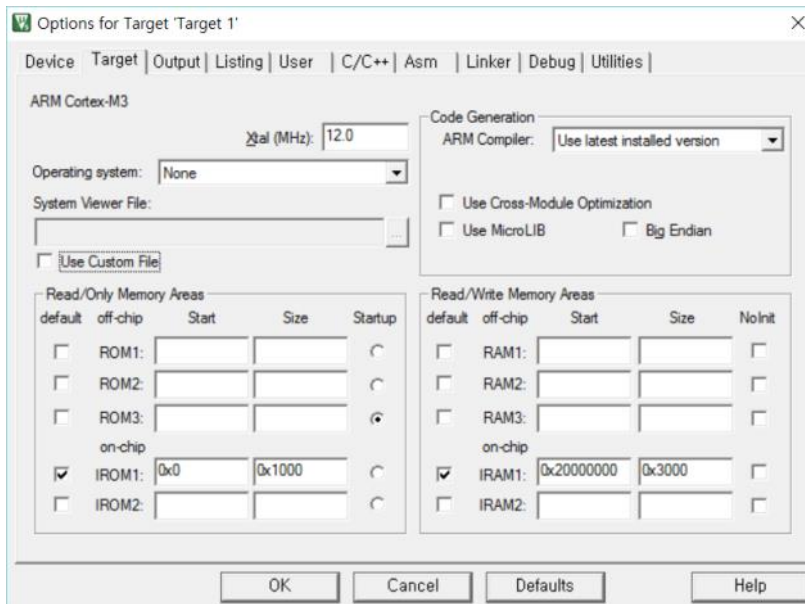
4. bootloader에서는 Interrupt 사용하지 않음

- a. UART polling 모드로 송수신
- b. Application project에서 Interrupt 사용 가능

- Directory : DeviceName_keil_bootcode → bootcode project

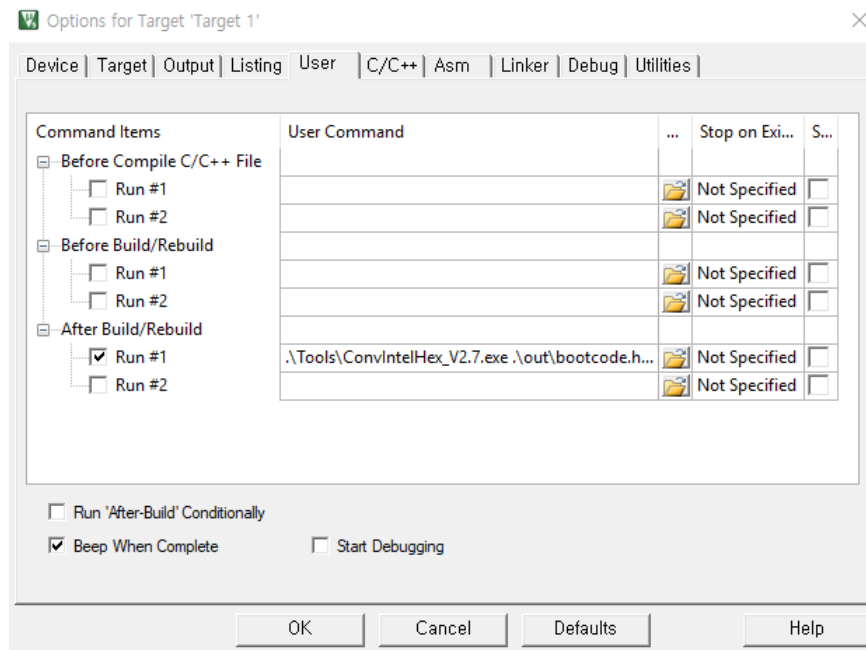
5. Compiler Setting

- 특별한 환경설정은 없으며 output file name(hex file)을 “bootcode”로 함
- file name을 “bootcode”로 한 것은 Application project에서 사용하기 위해 bootcode hex file을 Array로 변환하여 하나의 C file로 만드는데, 이 때 C file명이 hex file명과 같이 되며 Project 상에 bootcode.c로 include 하기 때문에 편의상 같은 이름을 사용



A33G526 keil option for Target 예

- Compile 된 hex file을 Array로 변환하는 방법
 - ConvIntelHex_V2.7.exe 를 사용하여 변환
 - 위치는 DeviceName_keil_bootcode\Examples\BOOT\bootcode\Keil\Tools\ConvIntelHex_V2.7.exe
 - Options for Target 의 “User”에 변환 프로그램과 출력 hex 파일을 설정
 - ✓ “ .\Tools\ConvIntelHex_V2.7.exe .\out\bootcode.hex -ed [0,1000] ”
 - 컴파일 후 등록된 ConvIntelHex_V2.7.exe 파일 실행됨
 - Compile된 hex file을 Array형태로 변경됨(out\bootcode.c 생성)
 - Array data가 C file로 생성되고 application project에 복사하여 사용



Keil option for Target Hex를 array C file로 설정

- Directory : A33G52x_keil_application → Examples → APP (APP code 영역의 code 작성)

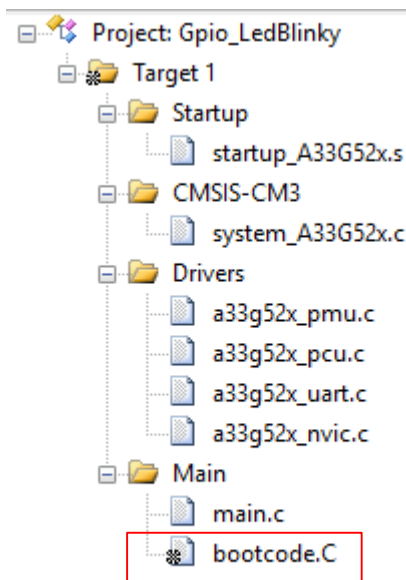
1. Main.c

- a. Application Vector Offset 재 정의

SCB->VTOR = 0x00001000; //Vector Table Offset Register

2. bootcode.c

- a. Hex file로 부터 변환된 C file을 Project에 Include 함 (예제 프로젝트에는 이미 추가 되어 있음)
- b. Bootloader project에서 변경이 있다면 업데이트 해야 함

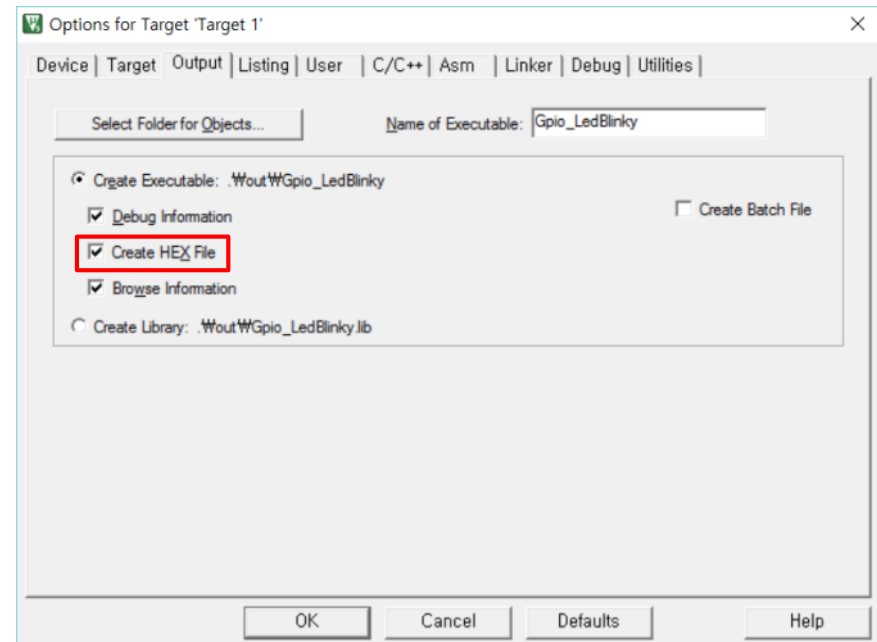
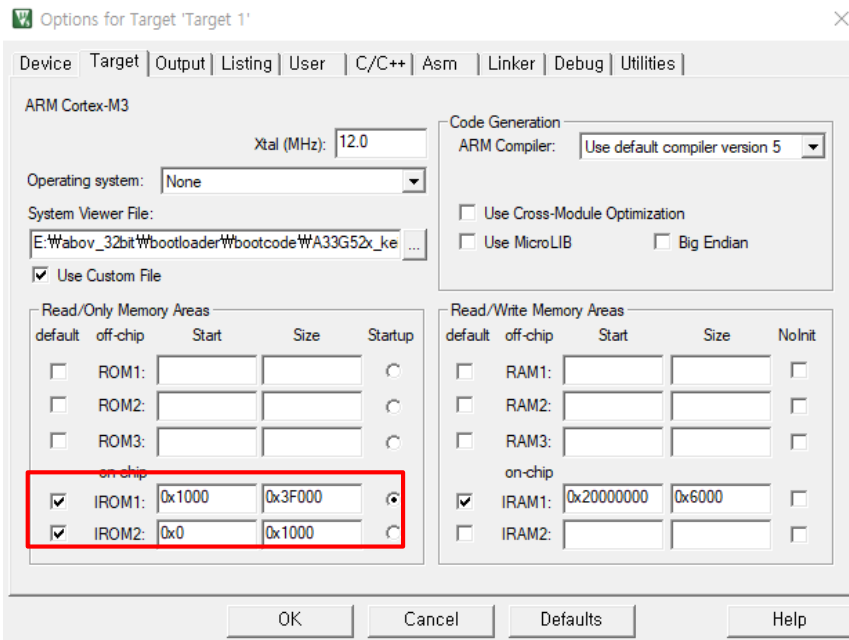


Application project에서 bootcode.c 추가 예 - A33G526

- Directory : A33G52x_keil_application → Examples → APP (APP code 영역의 code 작성)

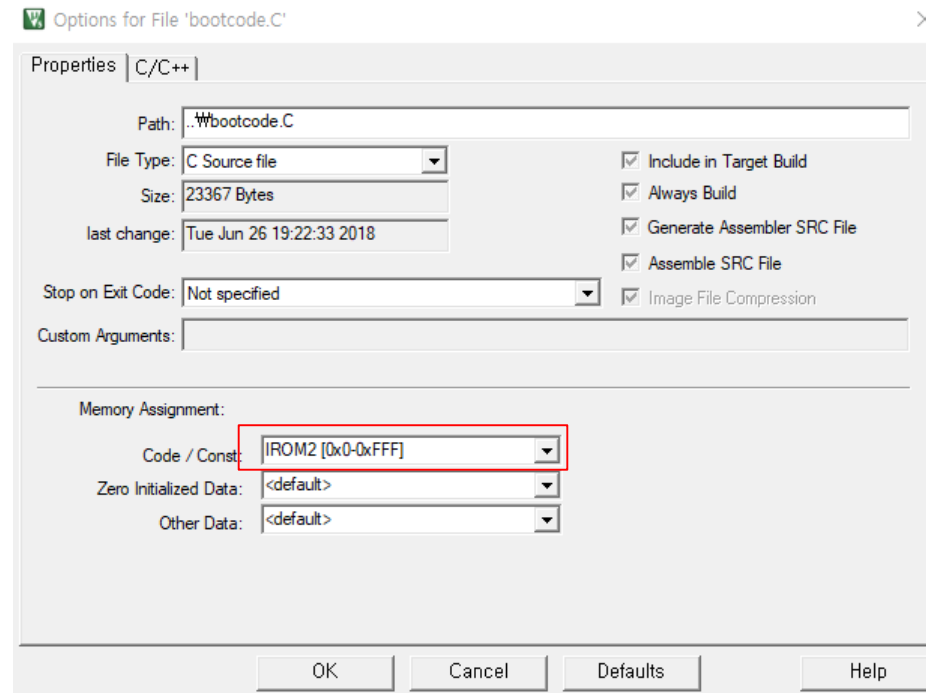
3. Compiler Setting

- Application 영역에 맞게 Compiler Setting을 해야 함
- 예: A33G526 application project
 - IROM1에 Application 시작 번지(0x1000 ~ 0x3F000)와 IROM2에 Bootcode 영역(0x0 ~ 0x1000)을 설정한다.
 - Hex file 생성 옵션을 선택한다. (UART 다운로드 프로그램에서 사용하기 위해)



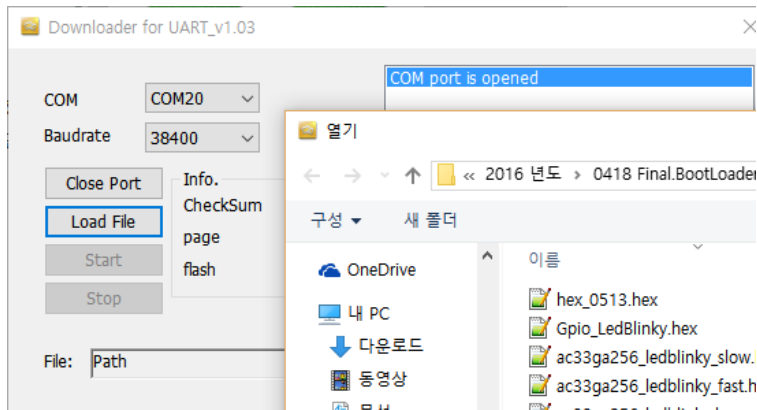
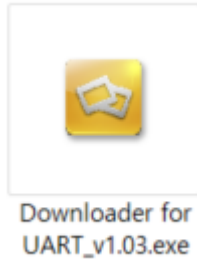
Application project에서 Options for target 설정 예 - A33G526

- Directory : A33G52x_keil_application → Examples → APP (APP code 영역의 code 작성)
- 3. Compiler Setting
 - c. Bootcode에 대한 특정 address 지정 (bootcode.c 파일 옵션)

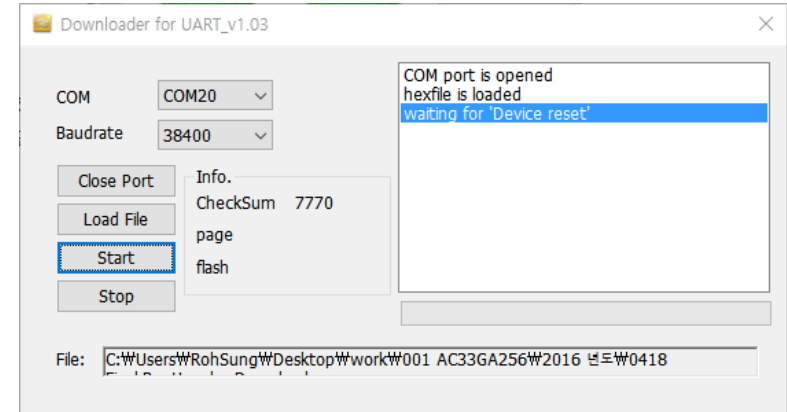


Application project에서 bootcode.c 파일 옵션 예 - A33G526

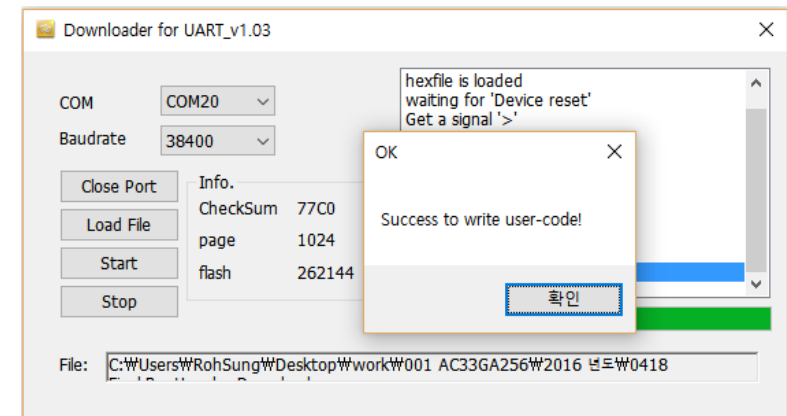
: Boot-Loader Test Tool



1. Downloader for UART_v1.03.exe 실행 후 Open Port.
2. Load File로 Update 할 Hex file 선택.

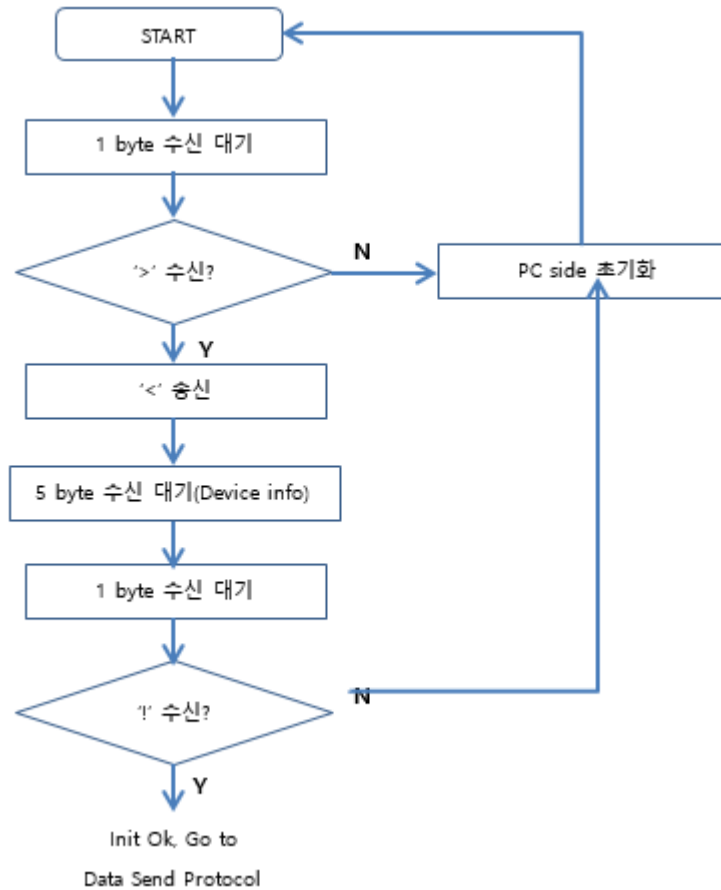


3. Start Button 을 누르면 '>'신호를 기다림
4. H/W Reset을 하면 A33G526에서 UART로 '>'신호를 보내면 '>'신호를 잘 받았다는 메시지와 함께 Update를 시작함



1. 완료되면 Success pop-up이 나오고 update가 완료 됨

- Init Protocol



■ Data Send Protocol

