# STM32 ReadOut Protection

This tutorial will introduce the STM32 Readout Protection feature.

- Readout protection is a global Flash memory protection allowing the embedded firmware code to be protected against copy, reverse engineering, dumping, using debug tools or code injection in SRAM.

Most STM32 have 3 levels of Readout protection:

- Level 0: no protection, factory default)
- Level 1: Memory protected (boot mode $\neq$ Flash memory)
- Level 2: Locked Device
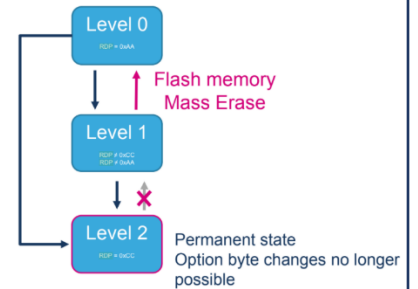
Hardware:

- Nucleo-L476RG board(64-pin),available at: www.st.com/en/evaluation-tools/nucleo-l476rg.html
- Standard-A -to- Mini USB cable

Literature:

- STM32L476xx Datasheet
- UM1724 User manual STM32 Nucleo-64 boards
- UM1884 Description of STM32L4/L4+ HAL and low-layer drivers
- UM1718 User manual STM32CubeMX for STM32 configuration and initialization C code generation
- RM0351 Reference Manual

Stages

- 1: Create and load a blinking project
- 2: Transition to RDP Level 1
- 3: Do an RDP regression to level 0
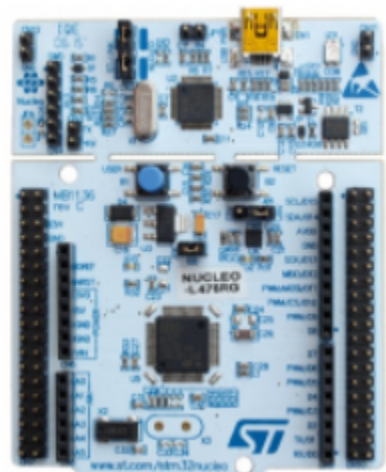
Summary of RDP levels:
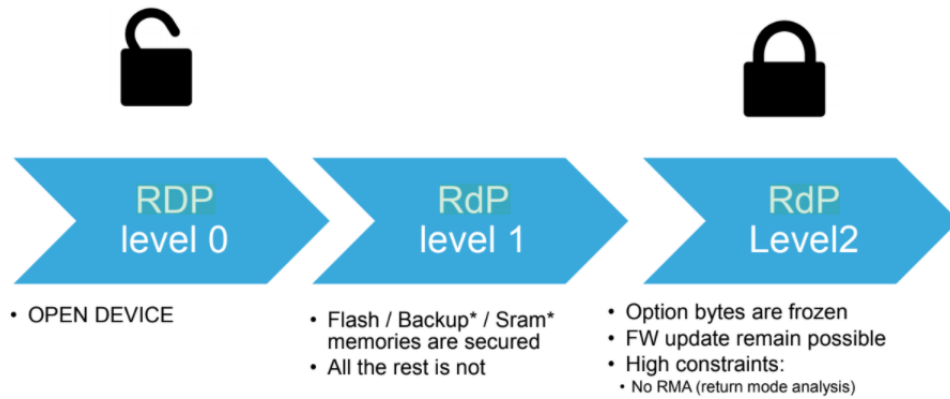
1: Create and load a blinking project

First we create a simple blinking project. Open STM32CubeIDE. Go to *File->New->STM32 Project.* Go to board selector and select Nucleo-L476RG. When prompted to initialize peripherals in their default mode, select yes.

Go to main.c and add the code shown in the image below to blink the led:

```
94
95      /* Infinite loop */
96      /* USER CODE BEGIN WHILE */
97      while (1)
98      {
99        /* USER CODE END WHILE */
100
101        /* USER CODE BEGIN 3 */
102          HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
103          HAL_Delay(500);
104      }
105      /* USER CODE END 3 */
106  }
107
```

Build and Debug the project. Resume the execution and verify that the led is toggling. Next, terminate the debug session.

2: Transition to RDP Level 1

Open STM32CubeProgrammer and select ST-Link and click connect. Navigate Option Byte and expand the Read Out Protection tab. We will move to RDP Level 1. Change "value" to BB and click apply.

▼ Read Out Protection		
Name	Value	Description
RDP	BB ▾	Read protection option byte The read protection is used to protect the software code stored in Flash memory. AA : Level 0, no protection BB : or any value other than 0xAA and 0xCC: Level 1, read protection CC : Level 2, chip protection
▶ BOR Level		
▶ User Configuration		
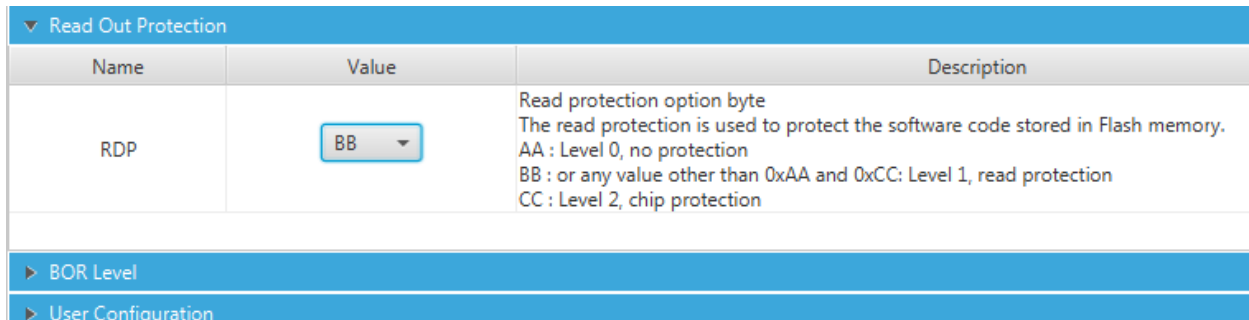
Click Disconnect and push the reset button on the Nucleo board. You will notice that the LED is still not blinking. Remove the Jumper from the pins labeled "IDD" and put it back in. The LED will begin to blink.

Now click Reconnect. An error will pop up that the data read failed. You will see you cannot read the Flash memory. This is because the Flash content is locked. If you click disconnect, the LED will not blink even after pushing reset. The only way for the LED to blink again is to remove the IDD jumper and put it back.

3: Do an RDP regression to level 0

Reconnect to the CubeProgrammer and change RDP to AA (level 0) and click apply. Now go to your Flash Content. Click read and you will the flash has been erased. On the transition from RDP level 1 to level 0, a flash mass erase is performed.

Memory & File edition

Device memory	Open file	+

Address 0x08000000 ▾ Size 0x400 Data width 32-bit ▾ Read ▾

Address	0	4	8	C	ASCII
0x08000000	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000010	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000020	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000030	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000040	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000050	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000060	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000070	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000080	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x08000090	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x080000A0	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x080000B0	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy
0x080000C0	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	yyyyyyyyyyyyyyyy