STM32 RDP without POR

This tutorial will demonstrate how to change RDP level without doing a Power on Reset(POR).

- RDP Level 1: Debugger access detected or boot mode ≠user then R/W/Erase to Flash memory, SRAM, and Backup registers are blocked.

Problem: When protection level 1 is set through boot from RAM, bootloader, or debugger, a power-down or standby is required to execute the user code. But what if device has battery soldered to board?

Solution: We can instead activate RDP with embedded code.

Hardware:

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

Literature:

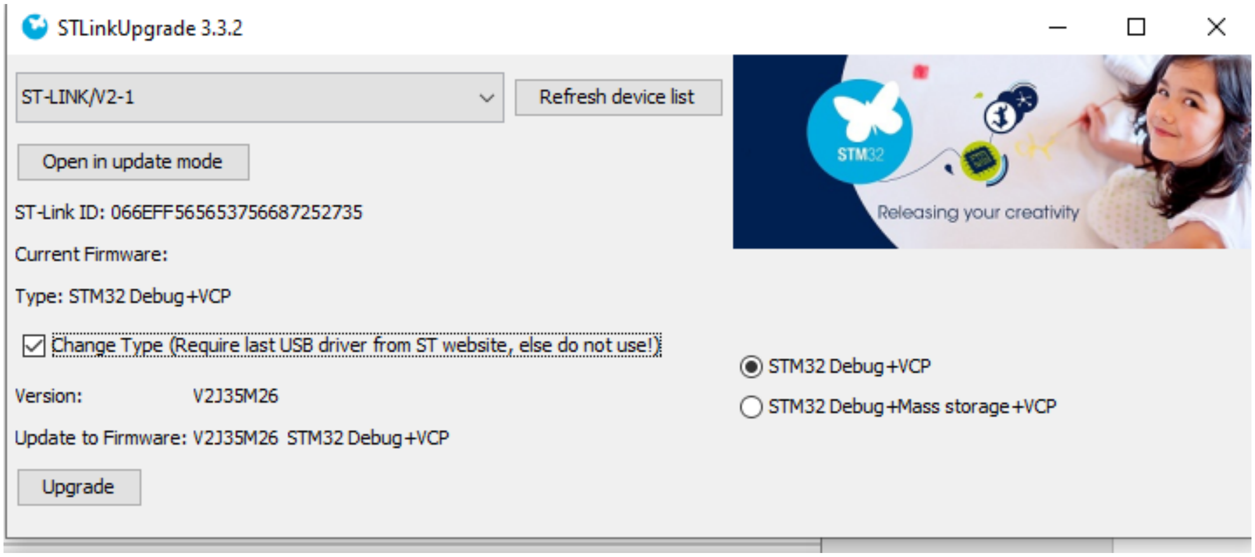
- STM32L476xx Datasheet
- UM1724 User manual STM32 Nucleo-64 boards
- UM1884 Description of STM32L4/L4+ HAL and low-layer drivers
- UM1718 User manual STM32CubeMX for STM32 configuration and initialization C code generation
- RM0351 Reference Manual

Stages

- 1: Remove mass storage capability
- 2: Create Project and modify main.c
- 3: Execute

1: Remove mass storage capability

First we will remove the mass storage capability of the STLink. Open STM32CubeProgrammer. With your Nucleo board connected to your PC, select Firmware Upgrade. Click “Open in update mode. Check the box labled “Change Type” and select STM32 Debug + VCP as shown below.



ST-Link ID: 066EFF565653756687252735

Current Firmware:

Type: STM32 Debug+VCP

☑ Change Type (Require last USB driver from ST website, else do not use!)

Version: V2J35M26

Update to Firmware: V2J35M26 STM32 Debug+VCP

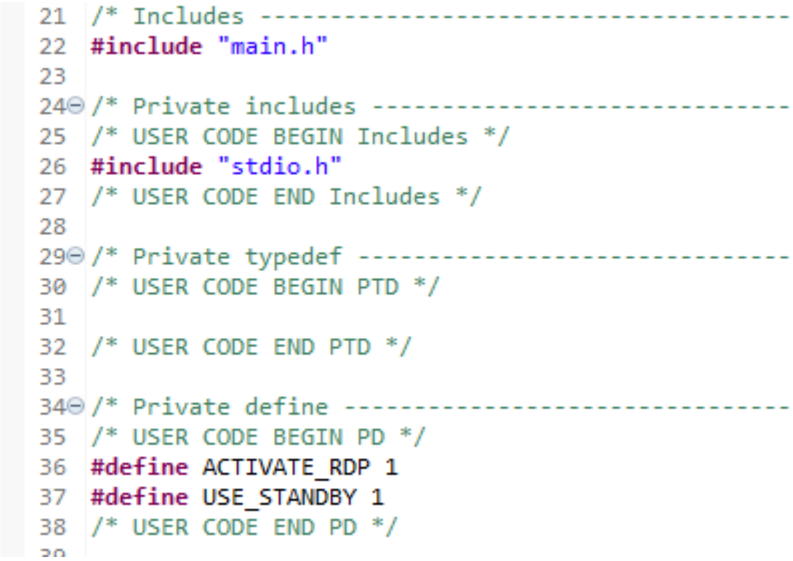
● STM32 Debug+VCP

○ STM32 Debug+Mass storage+VCP

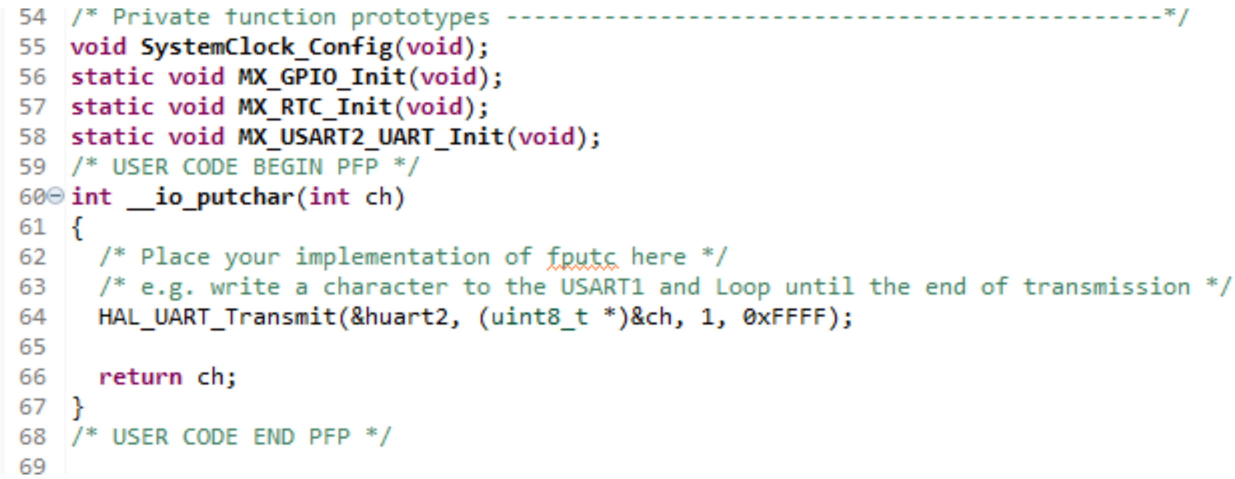
2: Create Project and modify main.c

Open STM32CubeIDE. Go to File->New->STM32Project. Go to board selector and select Nucleo-L476RG. When prompted to initialize peripherals in their default mode, select yes.

First add the code shown below above int main():



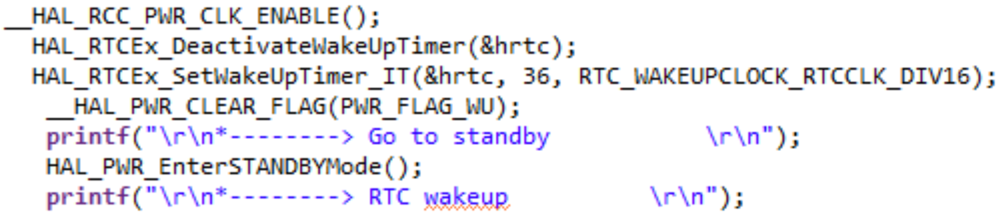
```
/* Includes ------------------------------------------------------------------
#include "main.h"

/* Private includes ----------------------------------------------------------
/* USER CODE BEGIN Includes */
#include "stdio.h"
/* USER CODE END Includes */

/* Private typedef -----------------------------------------------------------
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

/* Private define ------------------------------------------------------------
/* USER CODE BEGIN PD */
#define ACTIVATE_RDP 1
#define USE_STANDBY 1
/* USER CODE END PD */
```



```c
/* Private function prototypes -----------------------------------------------*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_RTC_Init(void);
static void MX_USART2_UART_Init(void);
/* USER CODE BEGIN PFP */
int __io_putchar(int ch)
{
  /* Place your implementation of fputc here */
  /* e.g. write a character to the USART1 and Loop until the end of transmission */
  HAL_UART_Transmit(&huart2, (uint8_t *)&ch, 1, 0xFFFF);

  return ch;
}
/* USER CODE END PFP */

```

Next we can activate RDP with code. We unlock the flash -> unlock the option byte -> get current configuration of option byte -> check current level and if it is different than 1 we program the option byte.

After programming the option byte we need to add a transition from StandBy state. This transition in code is show below:



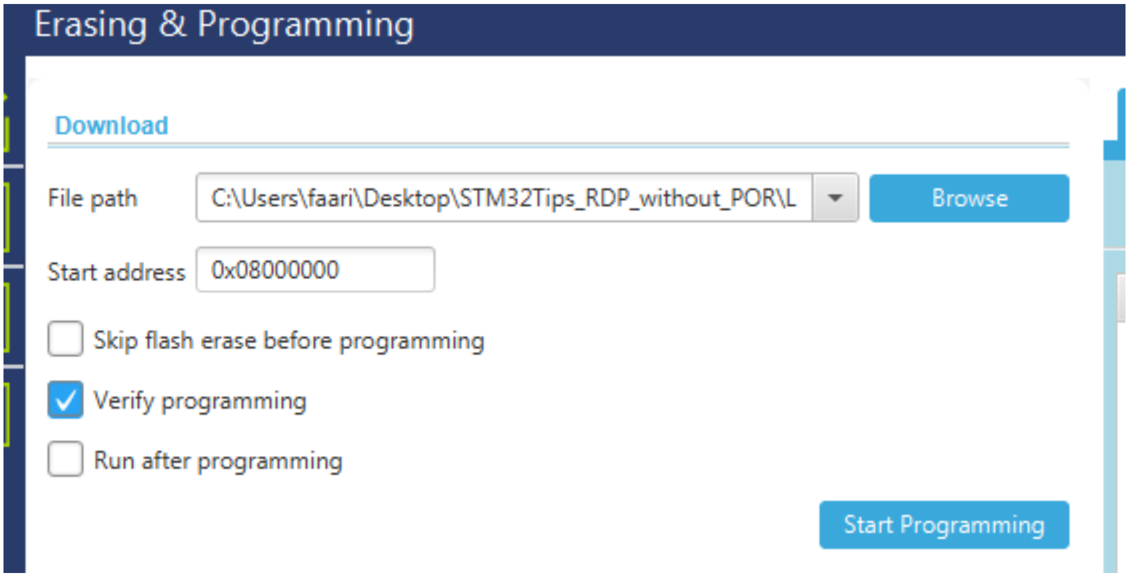
```c
__HAL_RCC_PWR_CLK_ENABLE();
  HAL_RTCEx_DeactivateWakeUpTimer(&hrtc);
  HAL_RTCEx_SetWakeUpTimer_IT(&hrtc, 36, RTC_WAKEUPCLOCK_RTCCLK_DIV16);
   __HAL_PWR_CLEAR_FLAG(PWR_FLAG_WU);
   printf("\r\n*--------> Go to standby           \r\n");
   HAL_PWR_EnterSTANDBYMode();
   printf("\r\n*--------> RTC wakeup          \r\n");
```

NOTE: We do a transition from Standby state because when connecting via debugging link or bootloader an internal flag is set indicating that debugging link/bootloader was used thus the flash is locked. The flag can only be reset by POR, or from a transition from StandBy state.

The code below shows the entire int main() code demonstrating RDP without POR.

```
int main(void)
{
  /* USER CODE BEGIN 1 */

  /* USER CODE END 1 */

  /* MCU Configuration--------------------------------------------------------*/

  /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
  HAL_Init();

  /* USER CODE BEGIN Init */
    HAL_DBGMCU_DisableDBGStandbyMode();
  /* USER CODE END Init */

  /* Configure the system clock */
  SystemClock_Config();

  /* USER CODE BEGIN SysInit */

  /* USER CODE END SysInit */

  /* Initialize all configured peripherals */
  MX_GPIO_Init();
  MX_RTC_Init();
  MX_USART2_UART_Init();
  /* USER CODE BEGIN 2 */
  printf("\r\n**********************************\r\n");
  printf("\r\n*       Appli started            *\r\n");
  printf("\r\n**********************************\r\n");

#ifdef ACTIVATE_RDP
  if(HAL_FLASH_Unlock() == HAL_OK) {
         if (HAL_FLASH_OB_Unlock() == HAL_OK) {
               FLASH_OBProgramInitTypeDef pOBInit;
               HAL_FLASHEx_OBGetConfig(&pOBInit);

               if (pOBInit.RDPLevel != OB_RDP_LEVEL_1) {
                      printf("\r\n*--------> RDP Level 0           \r\n");
                      pOBInit.OptionType = OPTIONBYTE_RDP;
                      pOBInit.RDPLevel = OB_RDP_LEVEL_1;
                      HAL_FLASHEx_OBProgram(&pOBInit);
#ifndef USE_STANDBY
                      printf("\r\n*--------> HAL_FLASH_OB_Launch           \r\n");
                   HAL_FLASH_OB_Launch();
                      printf("\r\n*--------> HAL_FLASH_OB_Launch finished
\r\n");

#else
                      __HAL_RCC_PWR_CLK_ENABLE();
                           HAL_RTCEx_DeactivateWakeUpTimer(&hrtc);
                           HAL_RTCEx_SetWakeUpTimer_IT(&hrtc, 36,
RTC_WAKEUPCLOCK_RTCCLK_DIV16);
                            __HAL_PWR_CLEAR_FLAG(PWR_FLAG_WU);
                            printf("\r\n*--------> Go to standby            \r\n");
                           HAL_PWR_EnterSTANDBYMode();
                           printf("\r\n*--------> RTC wakeup          \r\n");

#endif
               } else {
                      printf("\r\n*--------> RDP Level 1           \r\n");
               }
               HAL_FLASH_OB_Lock();
         }
    HAL_FLASH_Lock();
  }
#endif
  /* USER CODE END 2 */

  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  while (1)
  {
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
      HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
      printf("\r\n*--------> LED blinking          \r\n");
      HAL_Delay(500);
      if (HAL_GPIO_ReadPin(B1_GPIO_Port, B1_Pin) == GPIO_PIN_RESET)
      {
        if(HAL_FLASH_Unlock() == HAL_OK) {
               if (HAL_FLASH_OB_Unlock() == HAL_OK) {
                      FLASH_OBProgramInitTypeDef pOBInit;
                      HAL_FLASHEx_OBGetConfig(&pOBInit);

                      if (pOBInit.RDPLevel != OB_RDP_LEVEL_1) {
                             pOBInit.OptionType = OPTIONBYTE_RDP;
                             pOBInit.RDPLevel = OB_RDP_LEVEL_1;
                             HAL_FLASHEx_OBProgram(&pOBInit);
                             HAL_FLASH_OB_Launch();
                      }
                      HAL_FLASH_OB_Lock();
               }
          HAL_FLASH_Lock();
        }
      }
  }
  /* USER CODE END 3 */
}
```

After editing main.c build the project. Next open STM32CubeProgrammer and click connect. Go to erasing and programming and browse to the project directory to select the .bin file as shown below. The .bin file can be found in the Debug folder of the project.



Erasing & Programming

Download

File path: C:\Users\faari\Desktop\STM32Tips_RDP_without_POR\L

Start address: 0x08000000

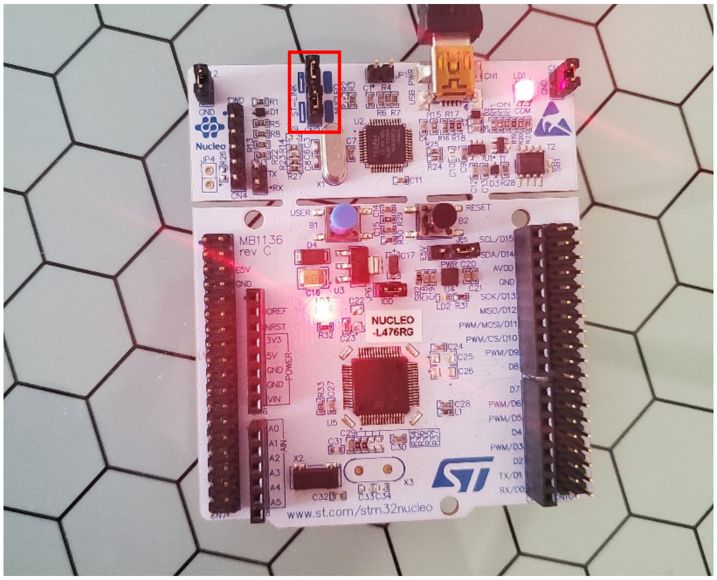
□ Skip flash erase before programming

☑ Verify programming

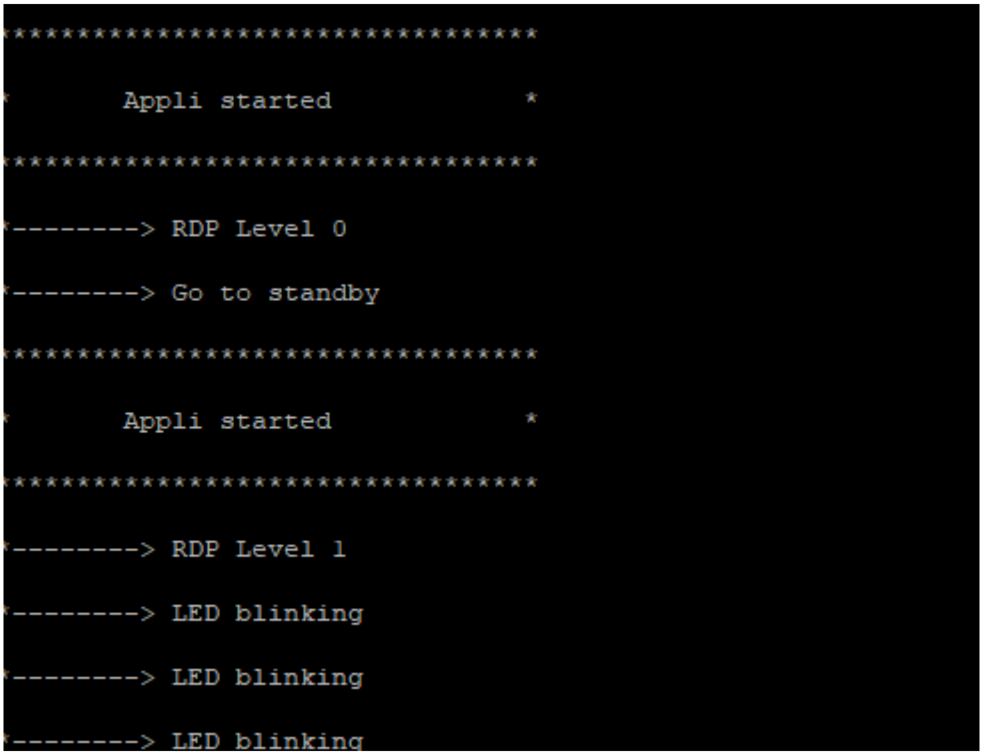
□ Run after programming

Click Start Programming. Open Putty or any terminal program and connect to the appropriate COM port and set baud rate to 115200.

Remove the ST-Link Jumpers and press the reset button. The jumpers are shown in the red box in the below image.



We can see on the terminal the transition from RDP Level 0 to Level 1 without the need for a POR.



```
**********************************
*      Appli started             *
**********************************
*--------> RDP Level 0
*--------> Go to standby
**********************************
*      Appli started             *
**********************************
*--------> RDP Level 1
*--------> LED blinking
*--------> LED blinking
*--------> LED blinking
```