S32G-VNP-GLDBOX
SOFTWARE ENABLEMENT GUIDE

V0.2

SECURE CONNECTIONS
FOR A SMARTER WORLD

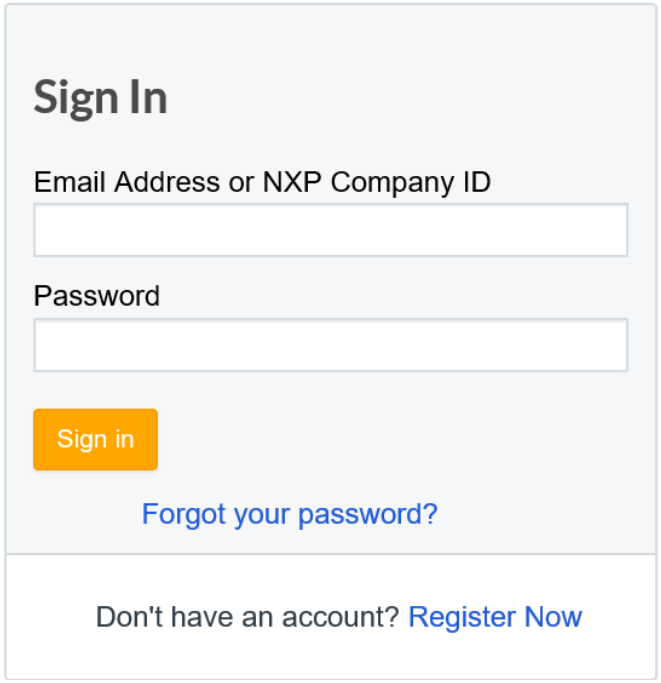
CONTENTS SOFTWARE ENABLEMENT GUIDE

01.

PRE: Get “S32G Standard Software”

PREREQUISITES: GET “S32G STANDARD SOFTWARE”

- Please go to: S32G Processors for Vehicle Networking and use your NXP account to sign in.



Please download the S32G Standard Software for enablement

Production	Install Packages
S32 Design Studio for S32 Platform	SW32G2_S32DS_3.4.0_D2012.zip
	S32DS.3.4_b201217_win32.x86_64.exe
S32G2 - Real Time Drivers	SW32_RTD_4_4_1.0.0_D2012_DS_updatesite.zip
S32G2 - Linux BSP	binaries_auto_linux_bsp27.0_s32g274_pfe.tgz
	S32G274_LinuxBSP27.0.0_User_Manual.pdf

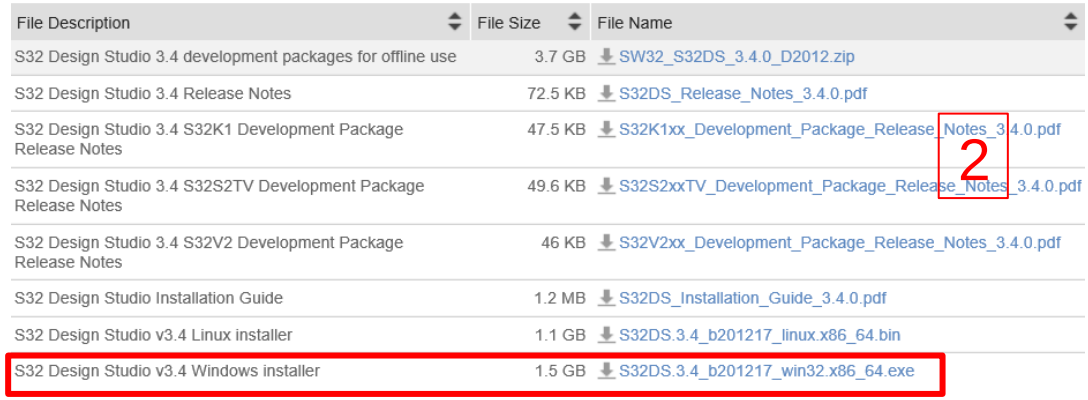
02.

Install Software Development Tool

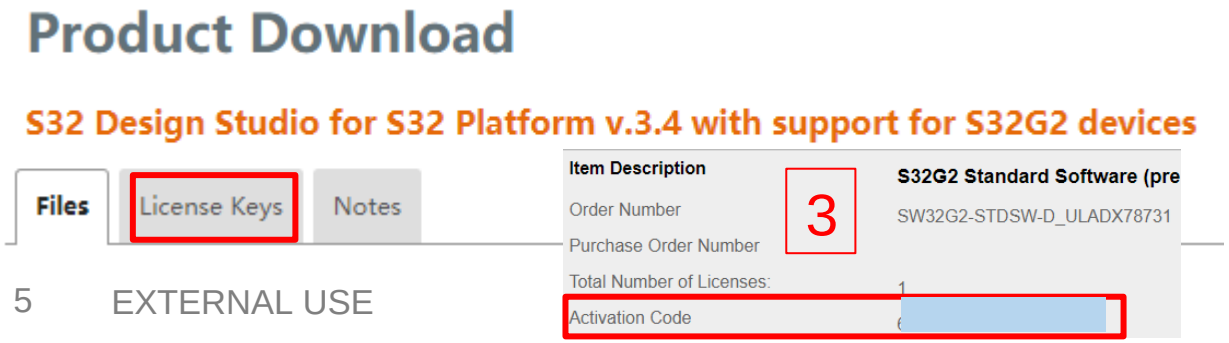
STEP 1: INSTALL S32 DESIGN STUDIO 3.4

- Click on S32 Design Studio version V3.4

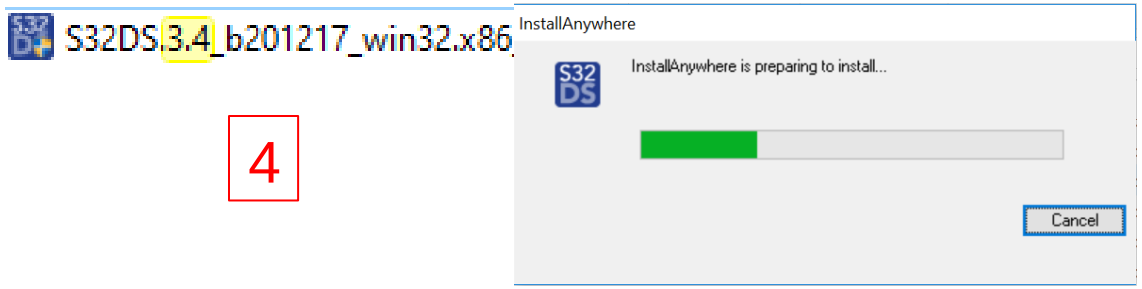
- Download installation package for your machine



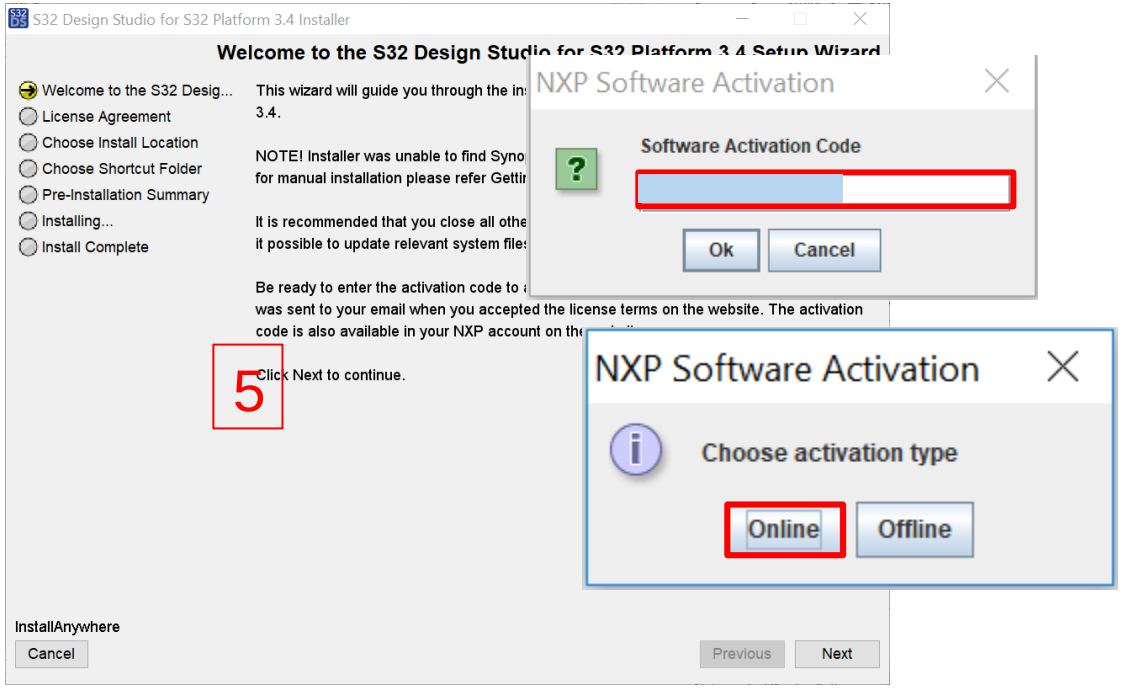
- Click on "License Keys" to get Activation Code



- Click .exe file to start installation



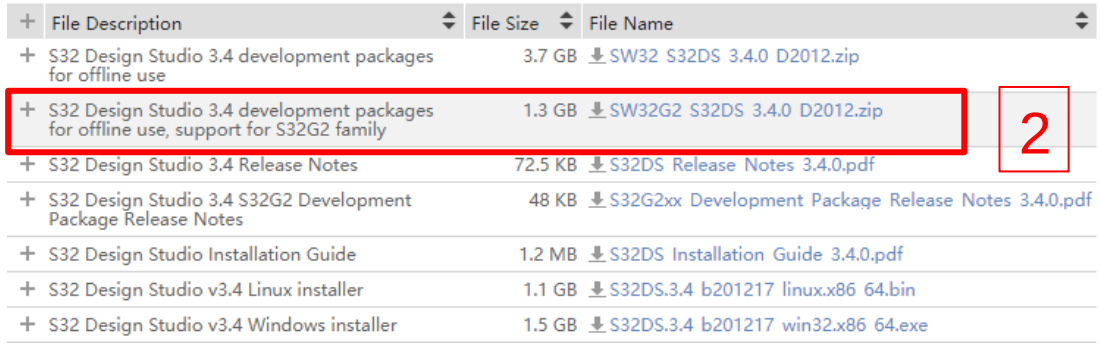
- Click “Next” to step by step install. Input the Activation code got in step3 when necessary and click on “Online”



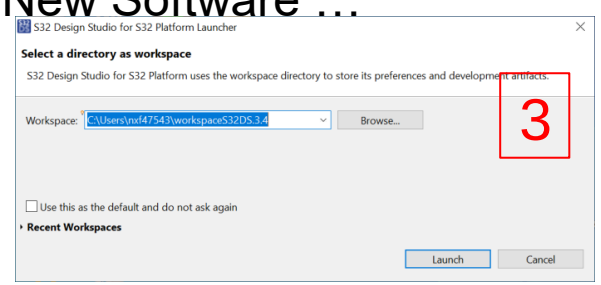
STEP 2: INSTALL DEVELOPMENT PACKAGES FOR S32G2

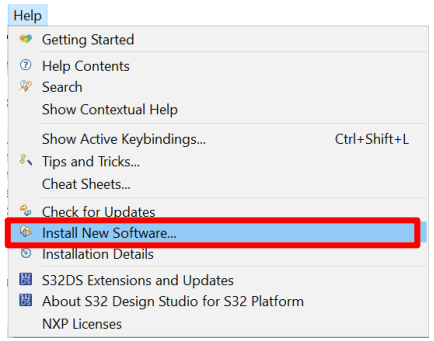
- Click on S32 Design Studio version V3.4

- Download development packages for your machine

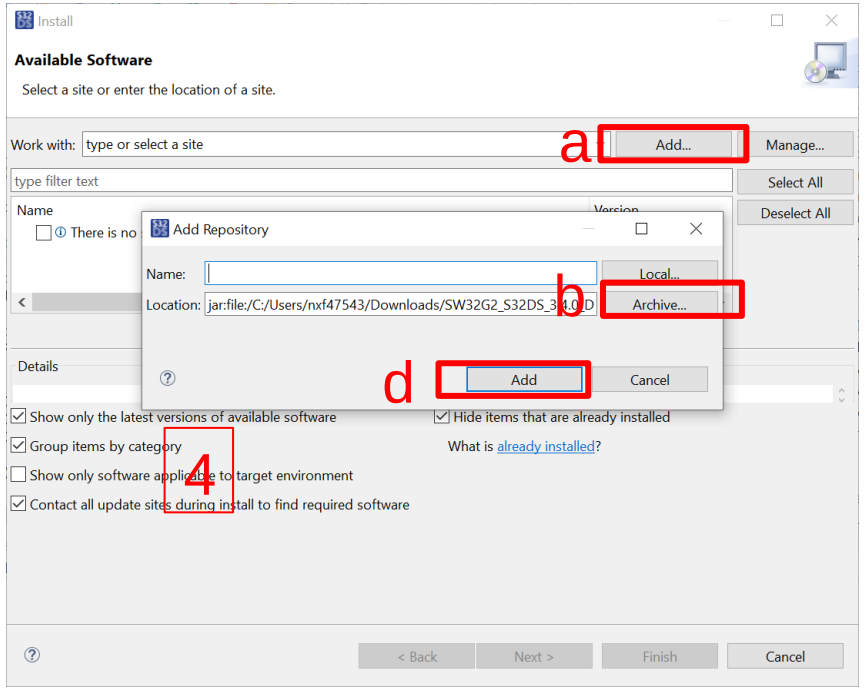


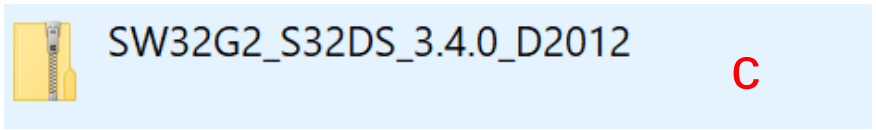
- Select a directory as workspace and open “Install New Software …”



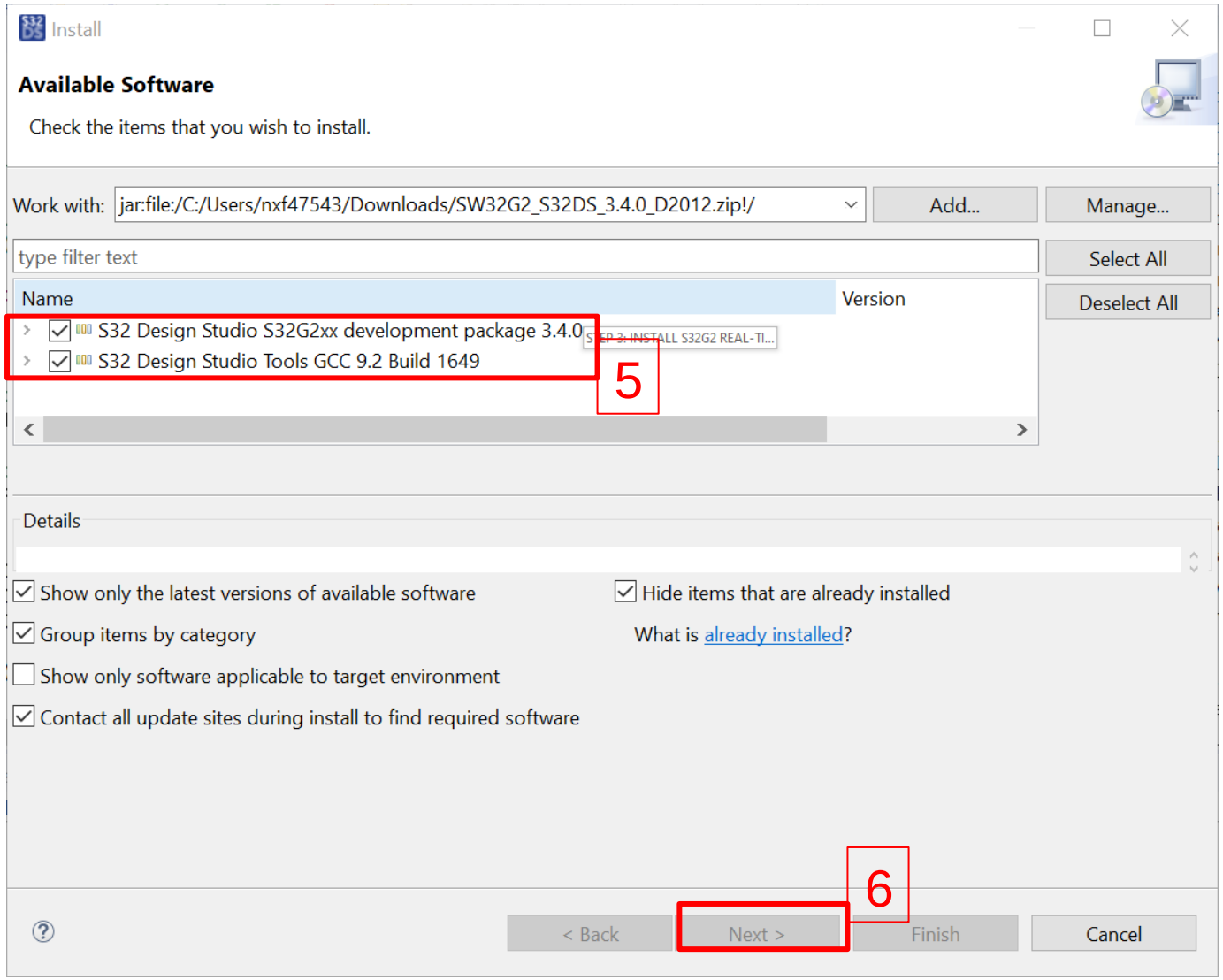


- Add update package of S32DS
  a. Click on “Add” button ”
  b. Click on “Archive” button in Add Site dialog
  c. Select .zip update file downloaded in step2 of current page and click on “open”
  d. Click on “Add” button in Add Site dialog



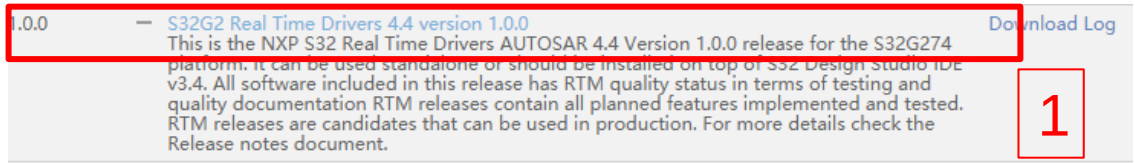


STEP 2: INSTALL UPDATE WITH SUPPORT FOR S32G2

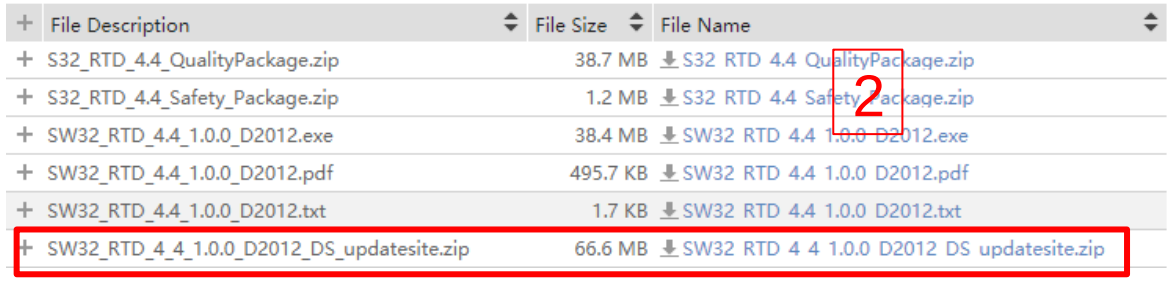


STEP 3: INSTALL S32G2 REAL-TIME DRIVERS

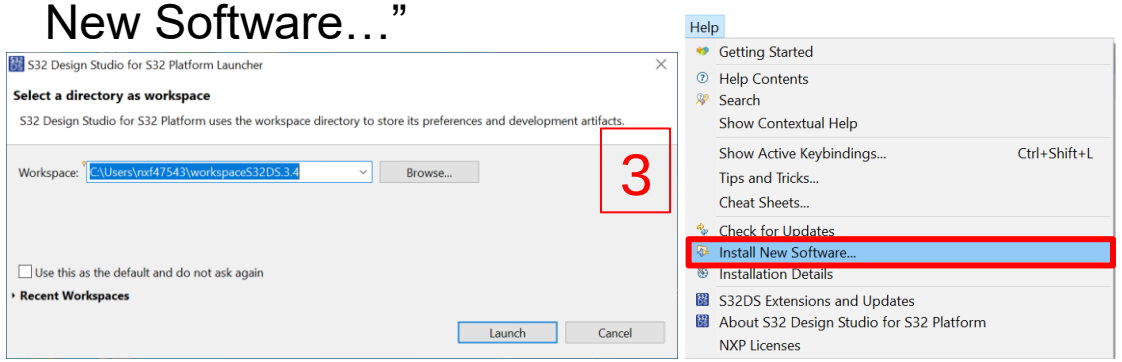
- Click on S32G2 Real Time Drivers 4.4 version 1.0.0



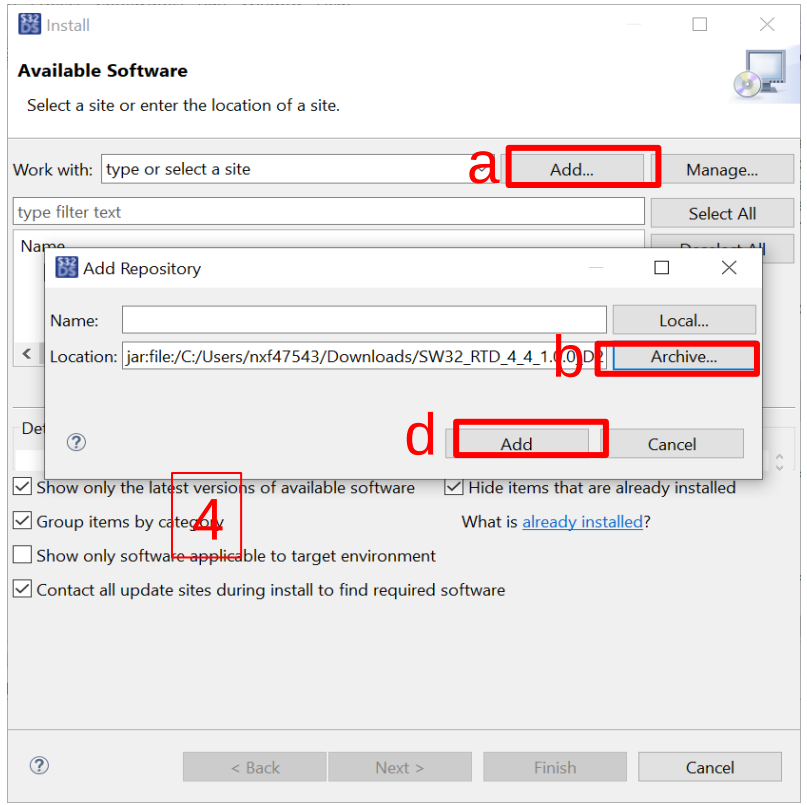
- Download SW32_RTD_4_4_1.0.0_D2012_DS_updatesite.zip package

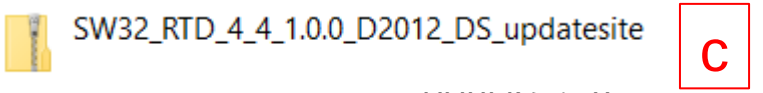


- Select a directory as workspace and open “Install New Software…”



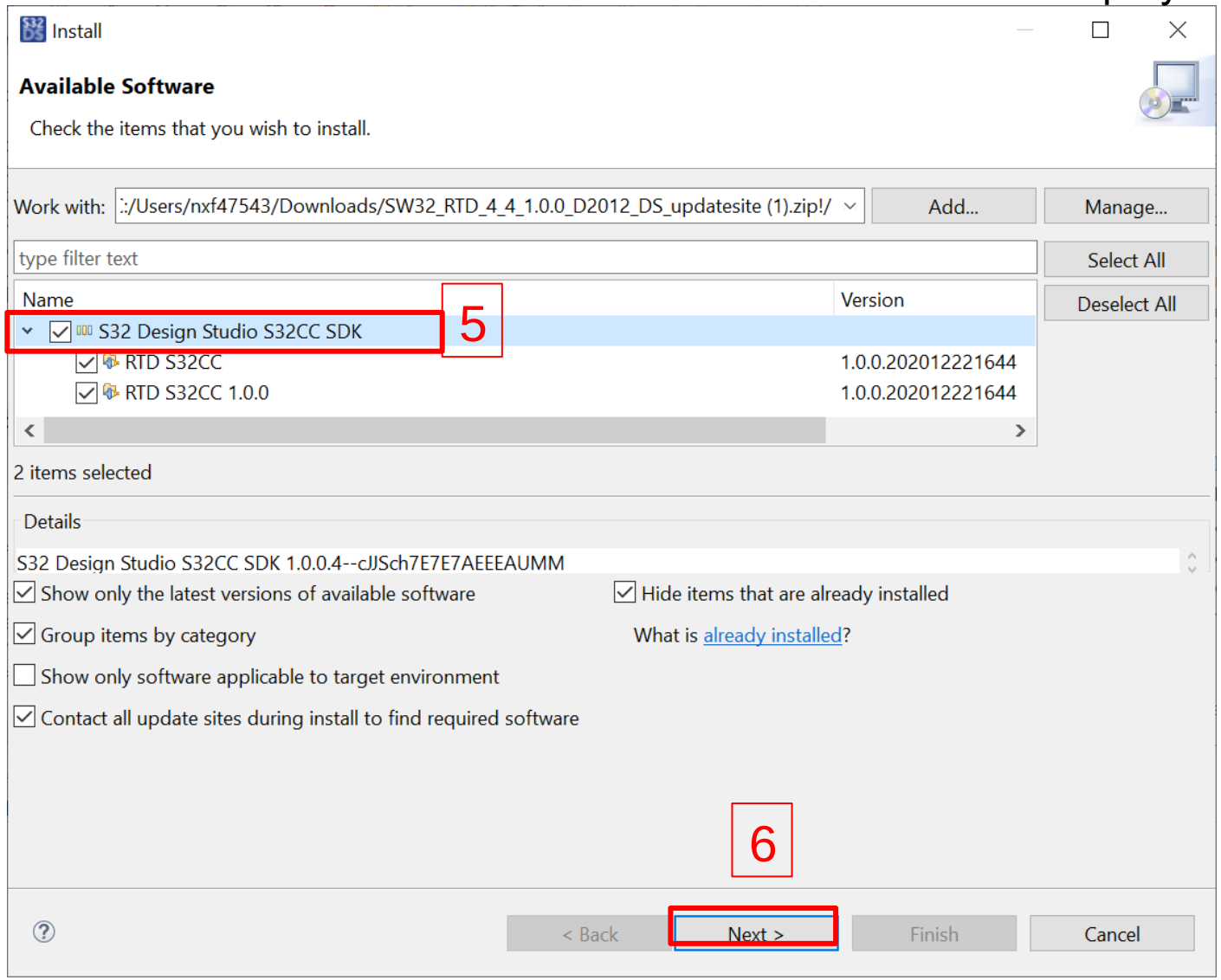
- Add update package of S32DS
  - a. Click on “Add…”
  - b. Click on “Archive” button in Add Repository dialog
  - c. Select .zip update file downloaded in step2 of current page and click on “open”
  - d. Click on ”ok” to go back “Available Software” dialog





STEP 3: INSTALL S32G2 REAL-TIME DRIVERS

Check the “RTD S32CC” box and click on “Next” to install step by step

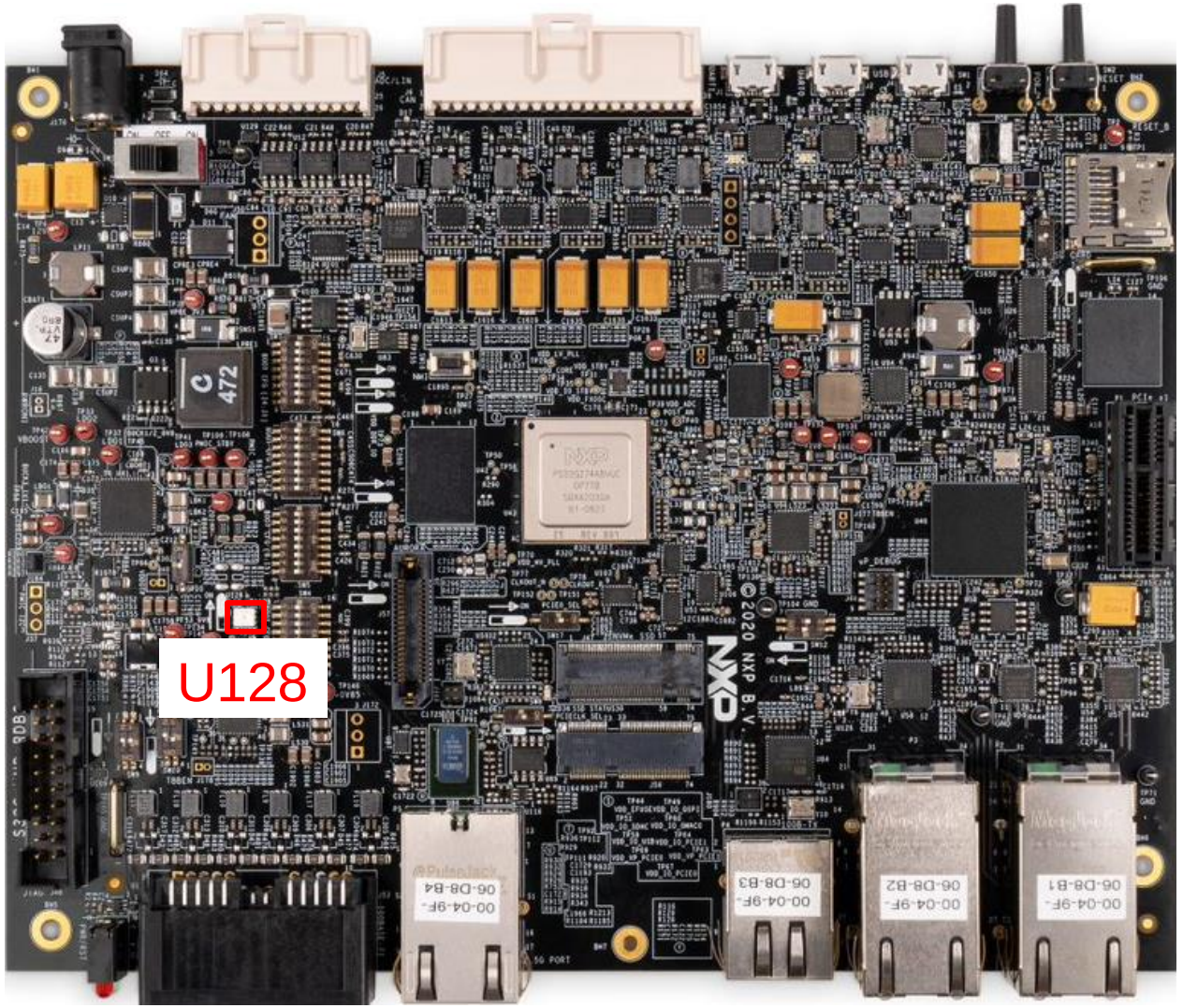


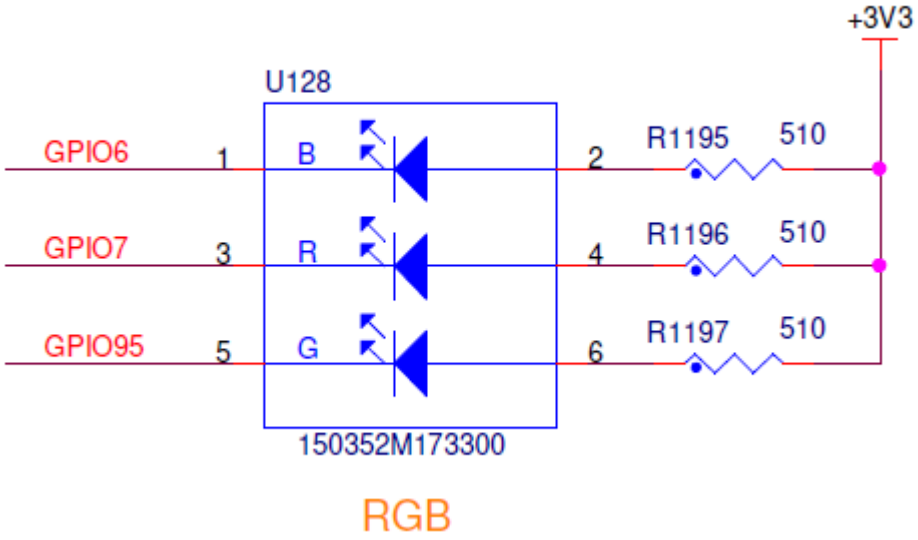
03.

Light Up RGB LED Based On Real Time Drivers

LIGHT UP RGB LED: HARDWARE RESOURCES

- Resources to be used: on-board RGB LED

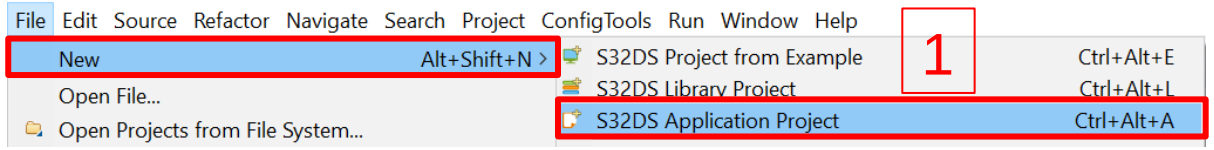




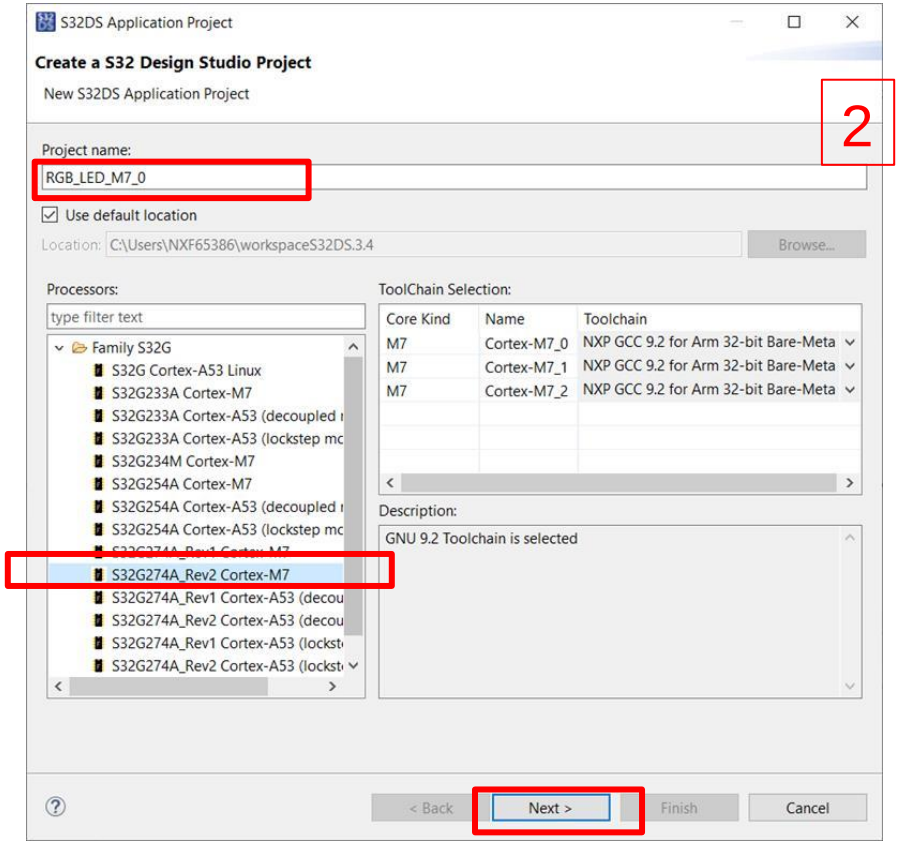
- An RGB LED is a combination of three LED in one package: 1x Blue LED, 1x Red LED and 1x Green LED.
- Because the LEDs are very close to each other, our eyes see the result of the combination of colors, rather than the three colors individually.

LIGHT UP RGB LED: CREATE PROJECT

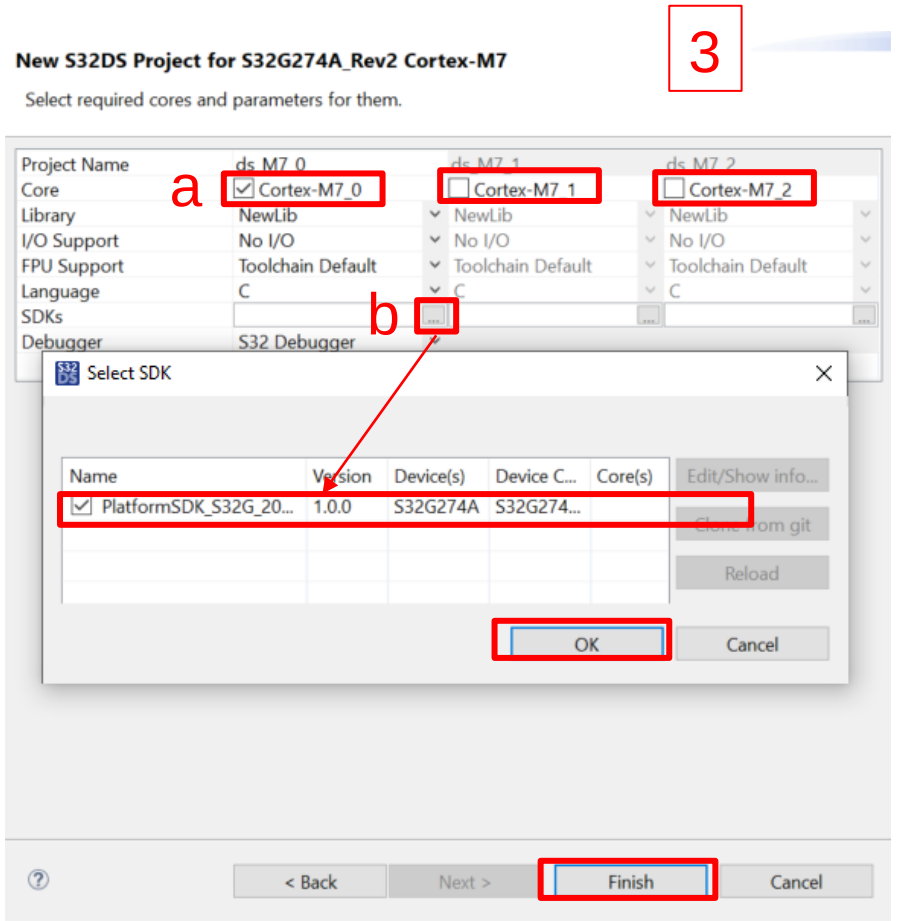
- Create a new S32DS Application Project



- Input project name and select S32G274A_REV2_Cortex-M7 as Processors, then click on “Next”

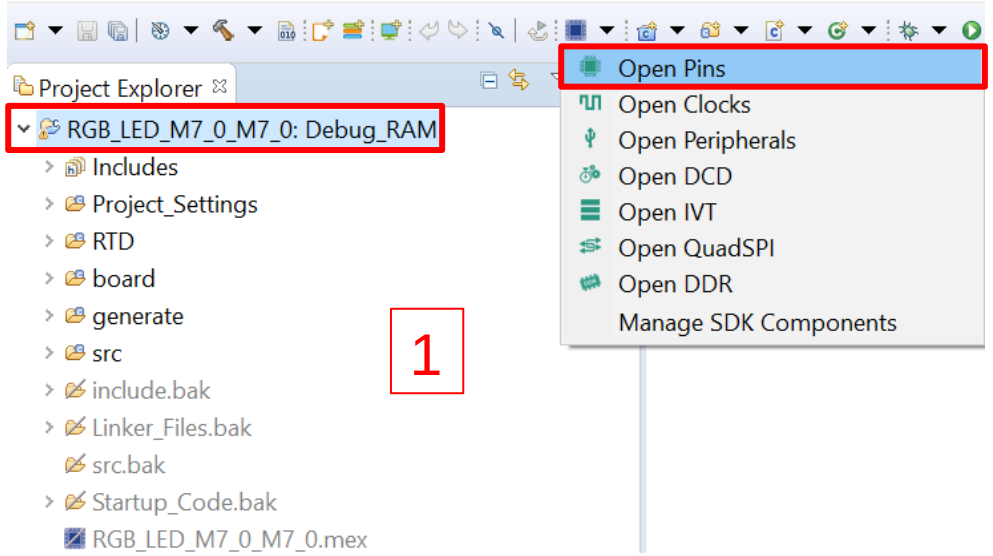


- Select required core and SDKs
  a. Check only Cortex-M7_0 core
  b. Check ‘RTD_CD01…’ as SDKs and click on “ok”
  c. Click on “Finish” to complete configuration

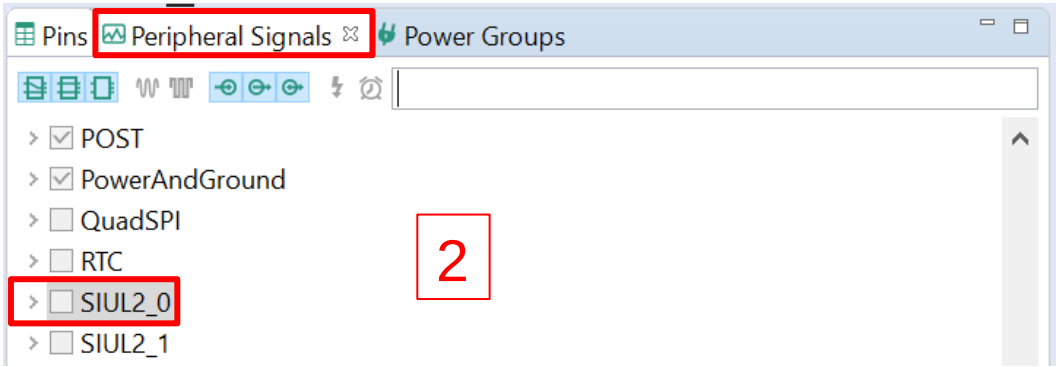


LIGHT UP RGB LED: PINS CONFIGURATION

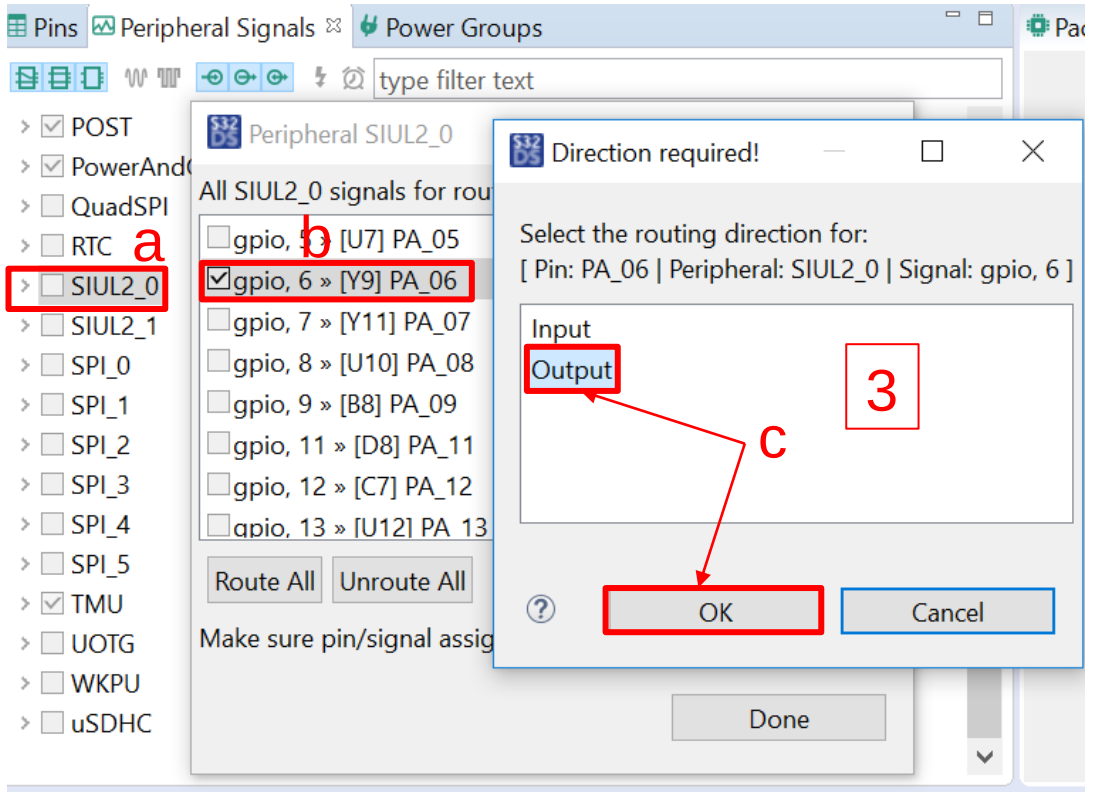
- Select the created project and open pins tool



- Select peripheral Signals page and find out SIUL2_0 box

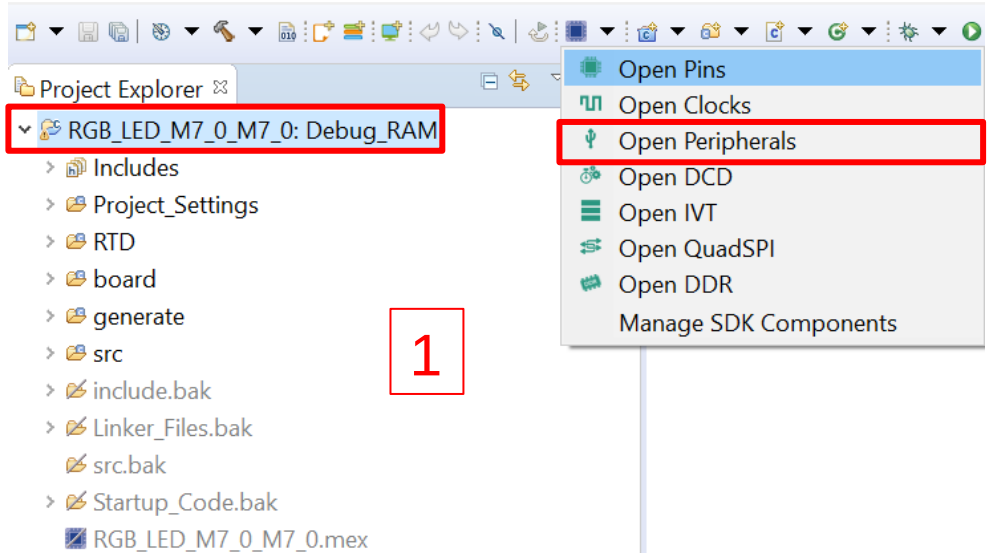


- Configure the corresponding gpio pins according to page 12.
  a. Click on “SIU2_0”
  b. Check gpio, 6 box
  c. Click on “Output” and ok to complete one pin configuration
  d. Follow b and c to configure gpio 7, 95, then click on “Done”

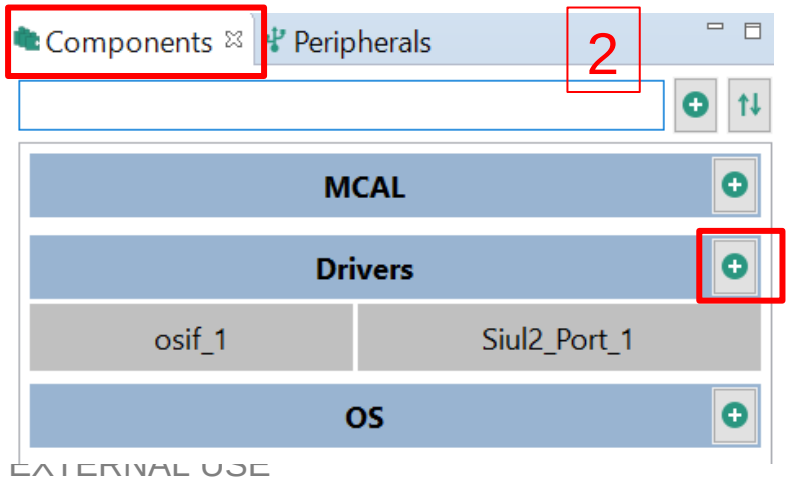


LIGHT UP RGB LED: ADD GPIO DRIVERS

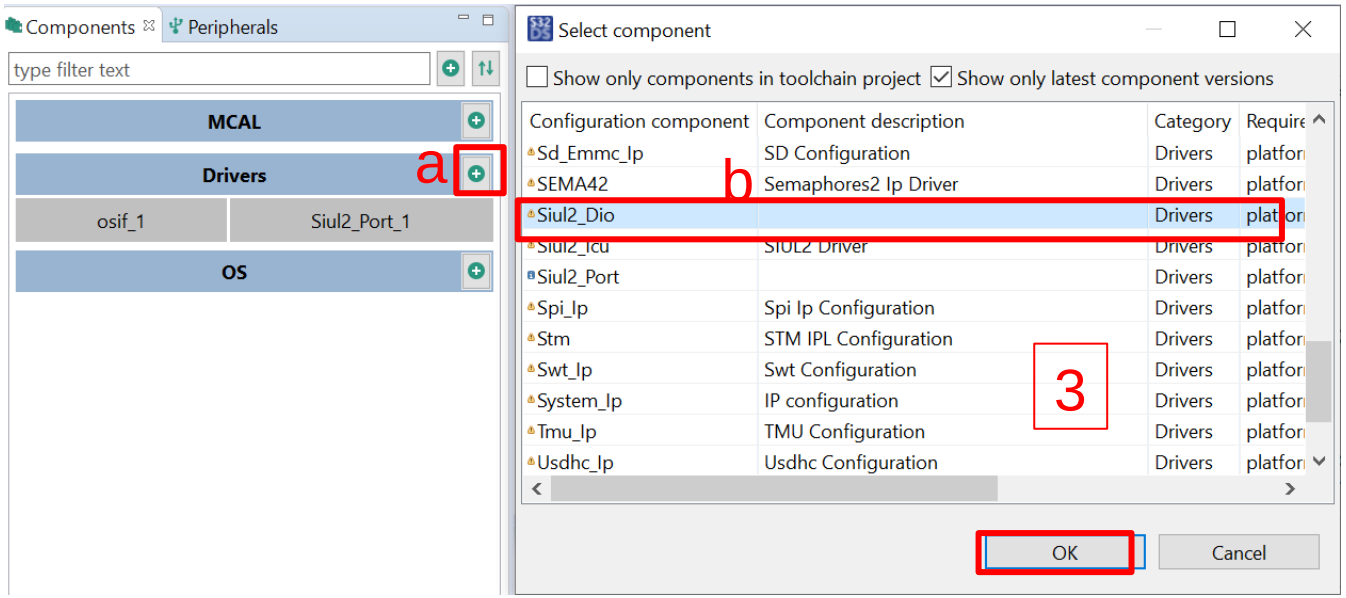
- Select the created project and open peripherals tool



- Select Components to find out Drivers option

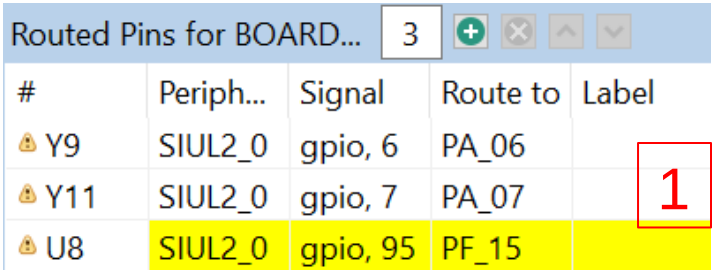


- Add gpio dio driver
  a. Click on “+” option
  b. Select “Siul2_Dio” and click on “ok”

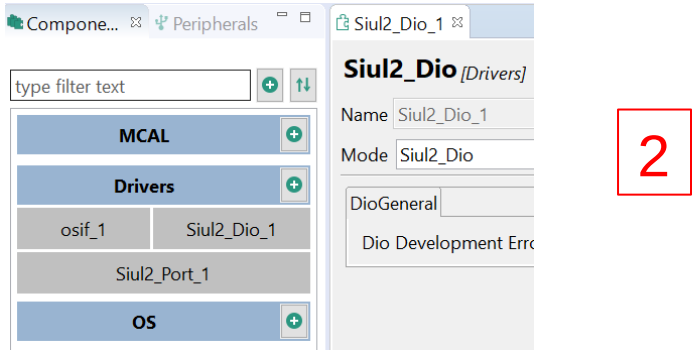


LIGHT UP RGB LED: CHECK CONFIGURATION AND UPDATE CODE

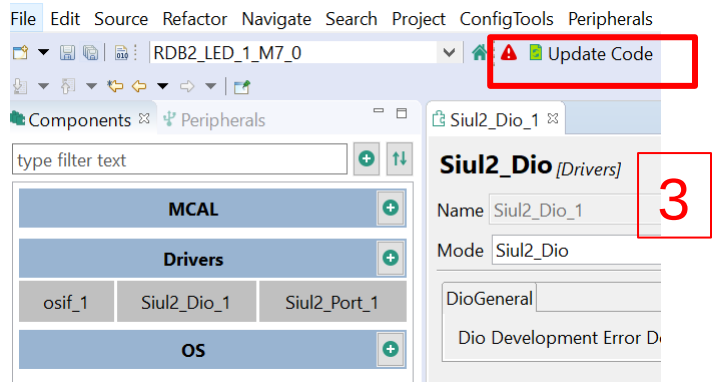
- open pins tool to check configuration



- open Peripherals tool to check configuration



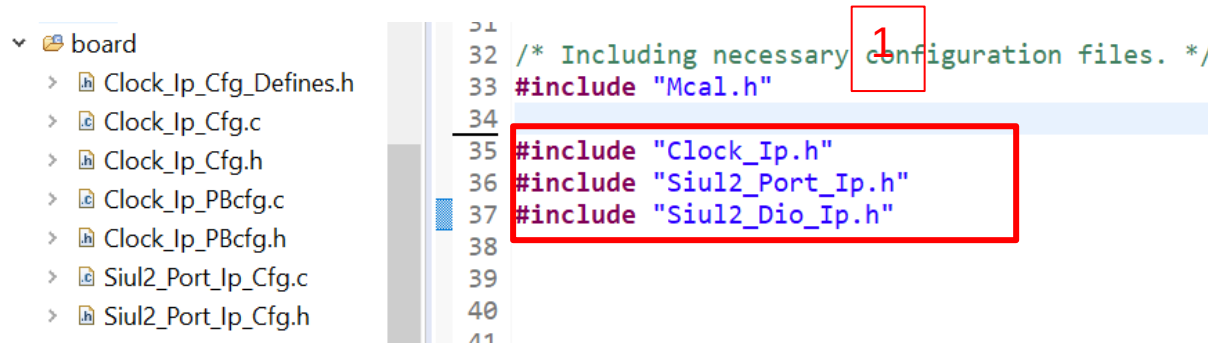
- Click on “Update Code”



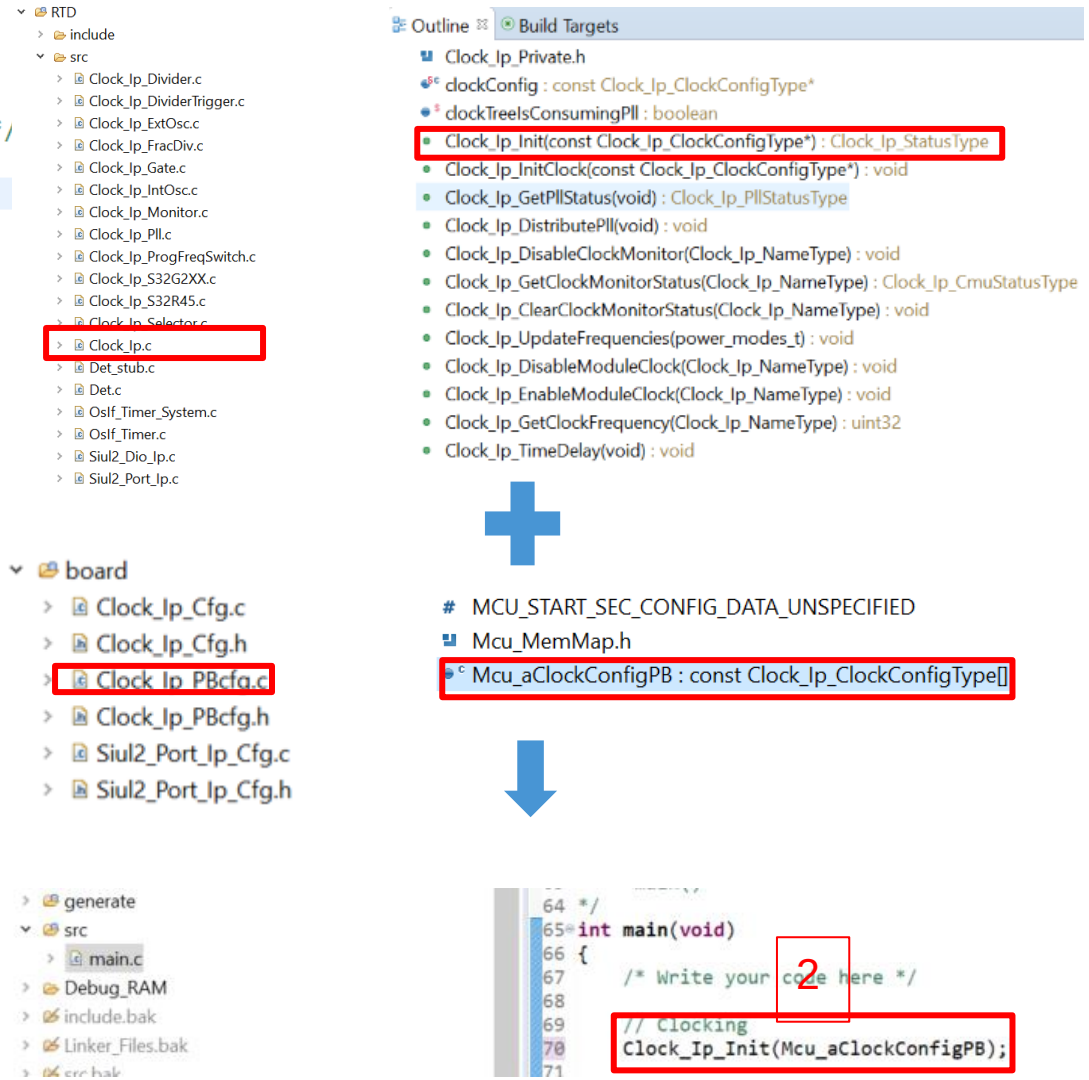
- Check the “Pins” and “Peripherals” box and click on “ok” to start update code, uncheck the other boxes

LIGHT UP RGB LED: APPLICATION CODE

- Add header files of project configuration and module drivers in main.c file

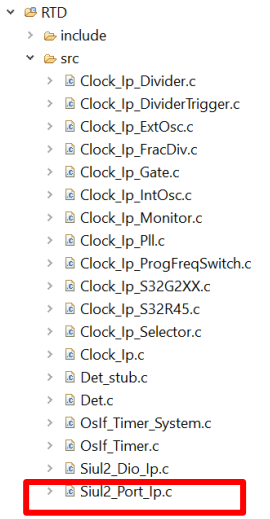


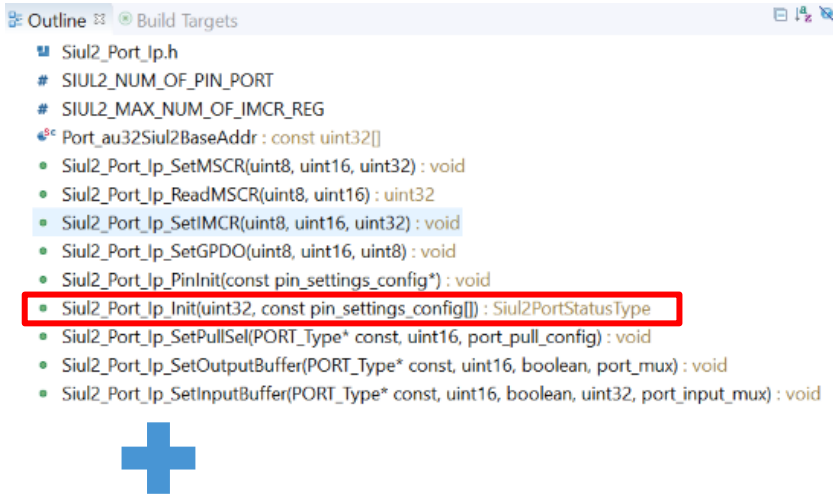
- Initialize clocks

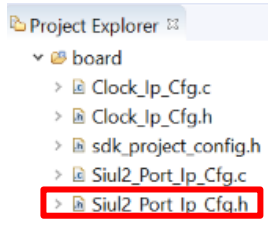


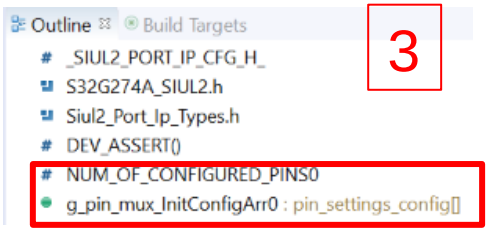
LIGHT UP RGB LED: APPLICATION CODE

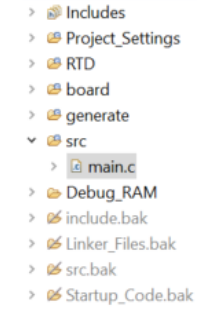
- Initialize pins

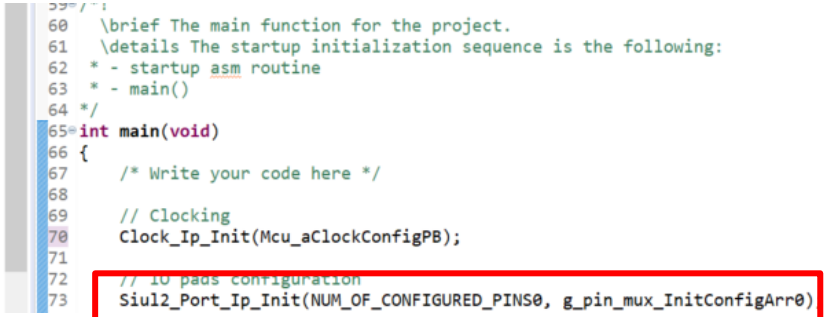




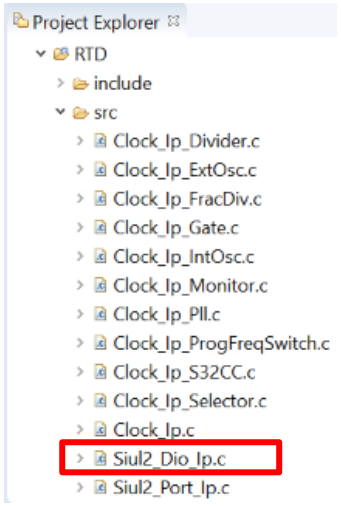


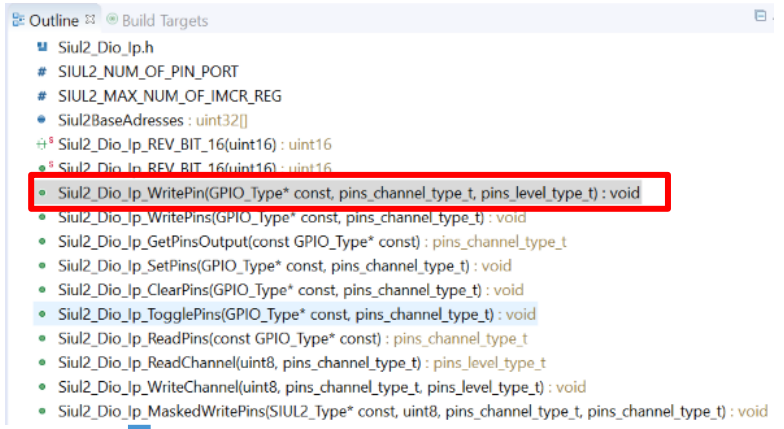


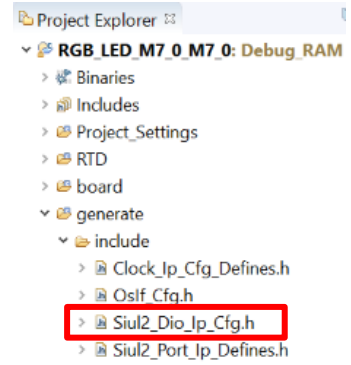


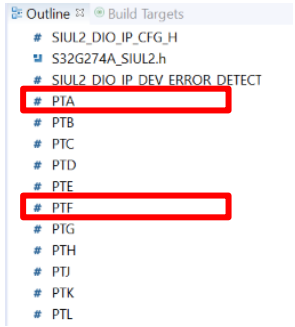


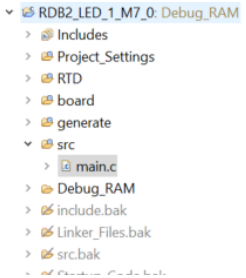
- Add the implementation of lighting up LED

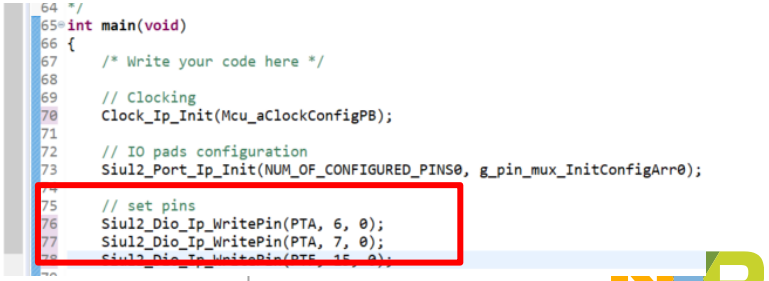










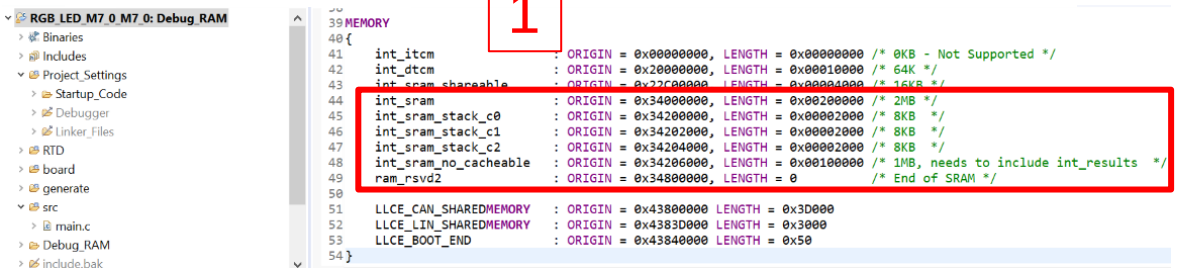


LIGHT UP RGB LED: BUILD PROJECT AND GENERATE .BIN FILE

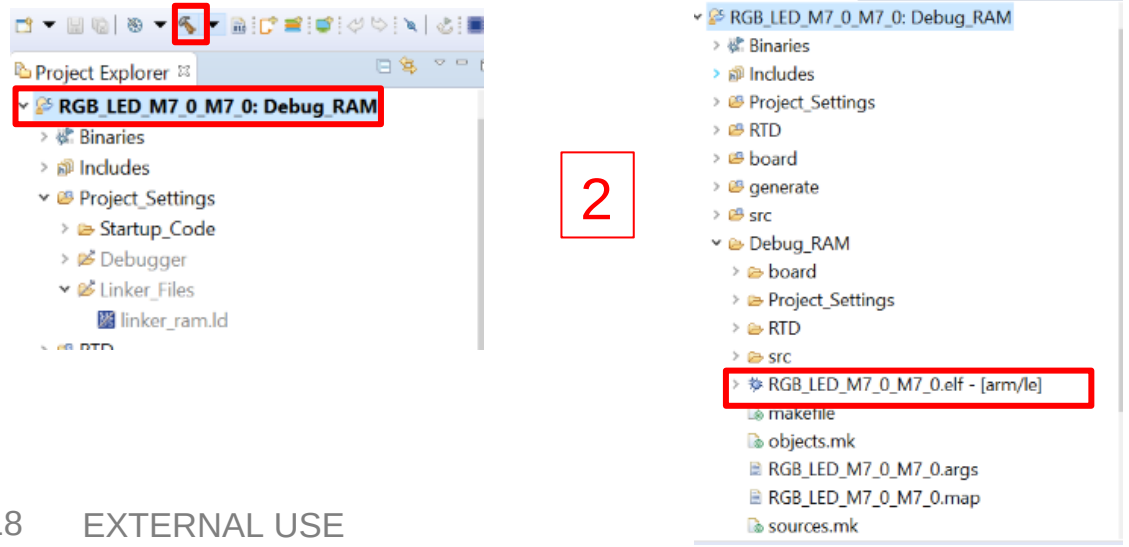
- Open and modify the link file according to the noted information which is from reference manual of S32G

NOTE

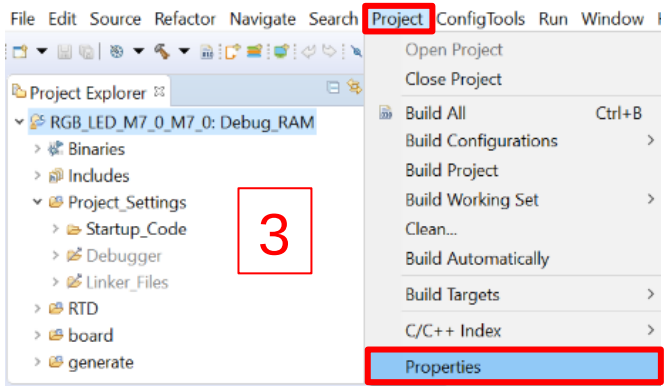
For application boot via the µSDHC interface, when BOOT_SEQ == 0, the RAM start pointer for the application should not point between 34008000h to 34078000h. This address range is used by BootROM for internal operation during boot via the µSDHC interface. BootROM also uses 4 KB of SRAM memory starting at 343FF000h for ADMA descriptors in case of µSDHC boot. The Application boot image header should not point to this location in case of µSDHC boot.



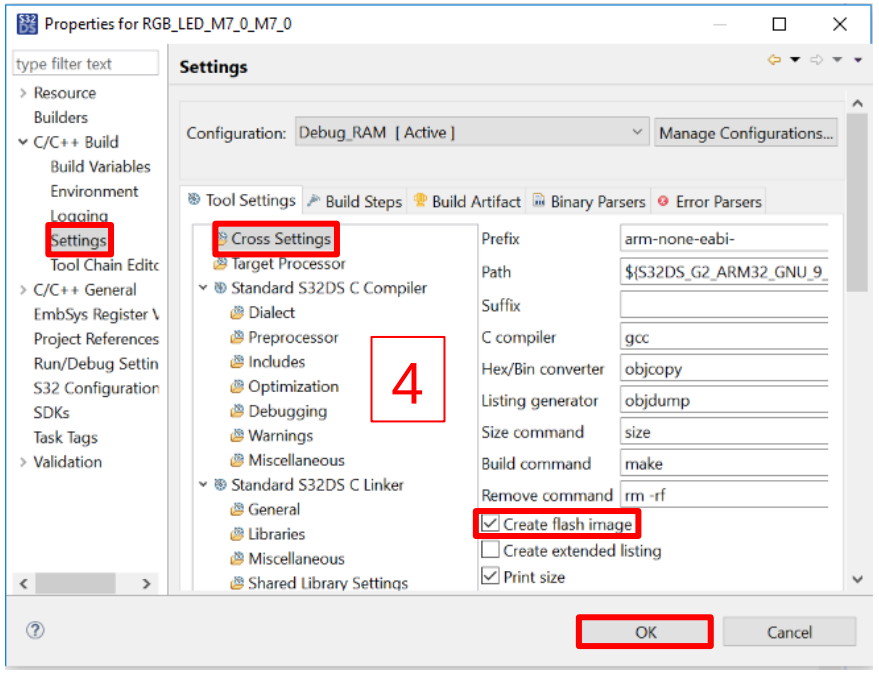
- Select and build project, .elf file will be generated



- Open the properties of project

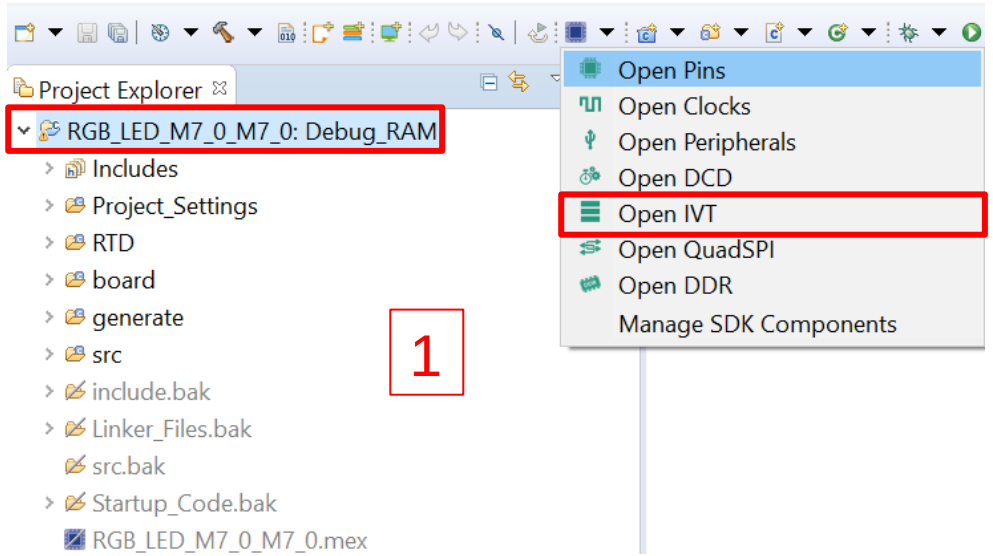


- Check the “Create flash image” box and click on “OK”

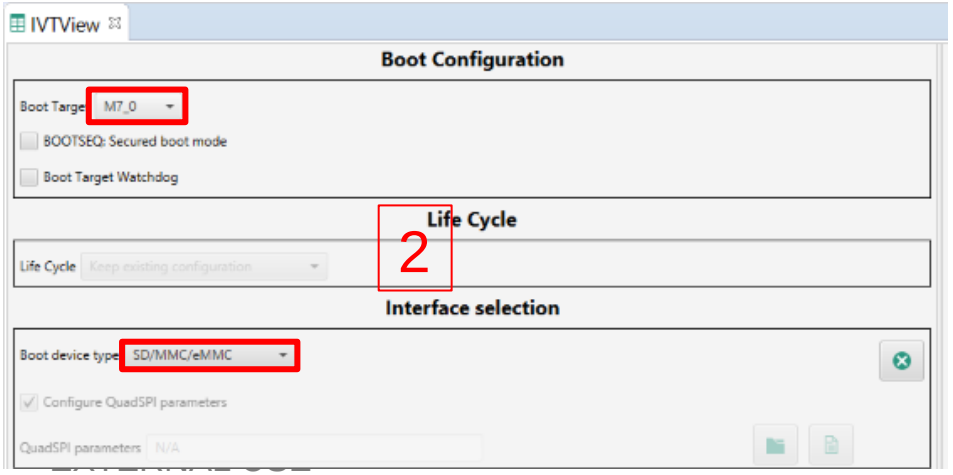


LIGHT UP RGB LED: MAKE IMAGE BY IVT TOOL

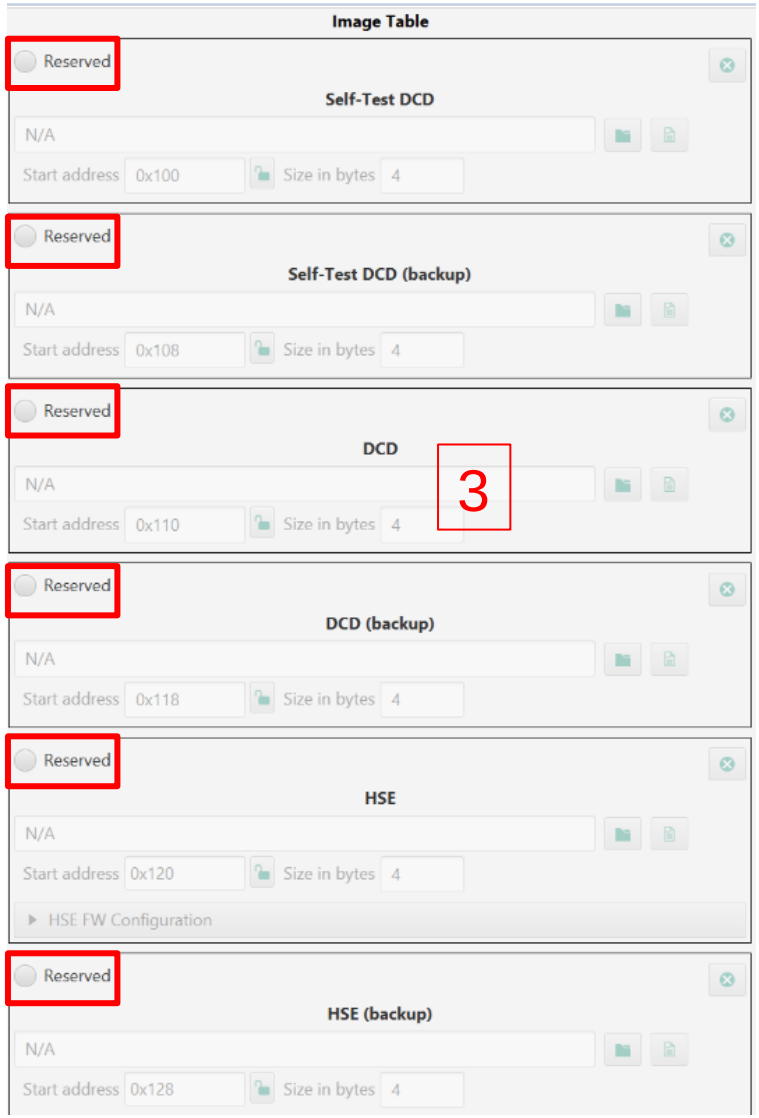
- Select the created project and open IVT tool



- Select M7_0 as Boot Target and select SD/MMC/eMMC as Boot device type

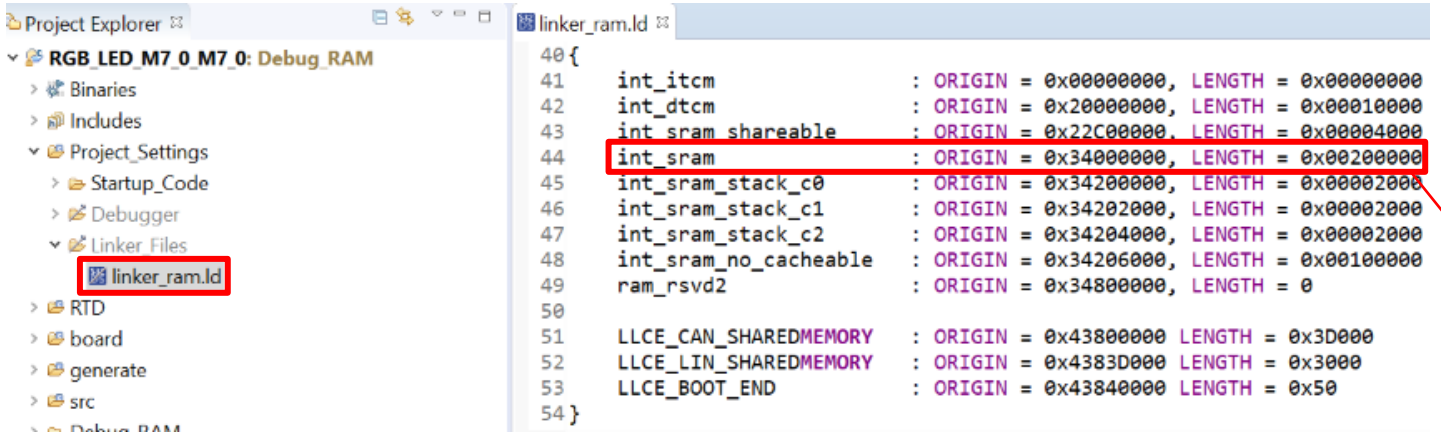


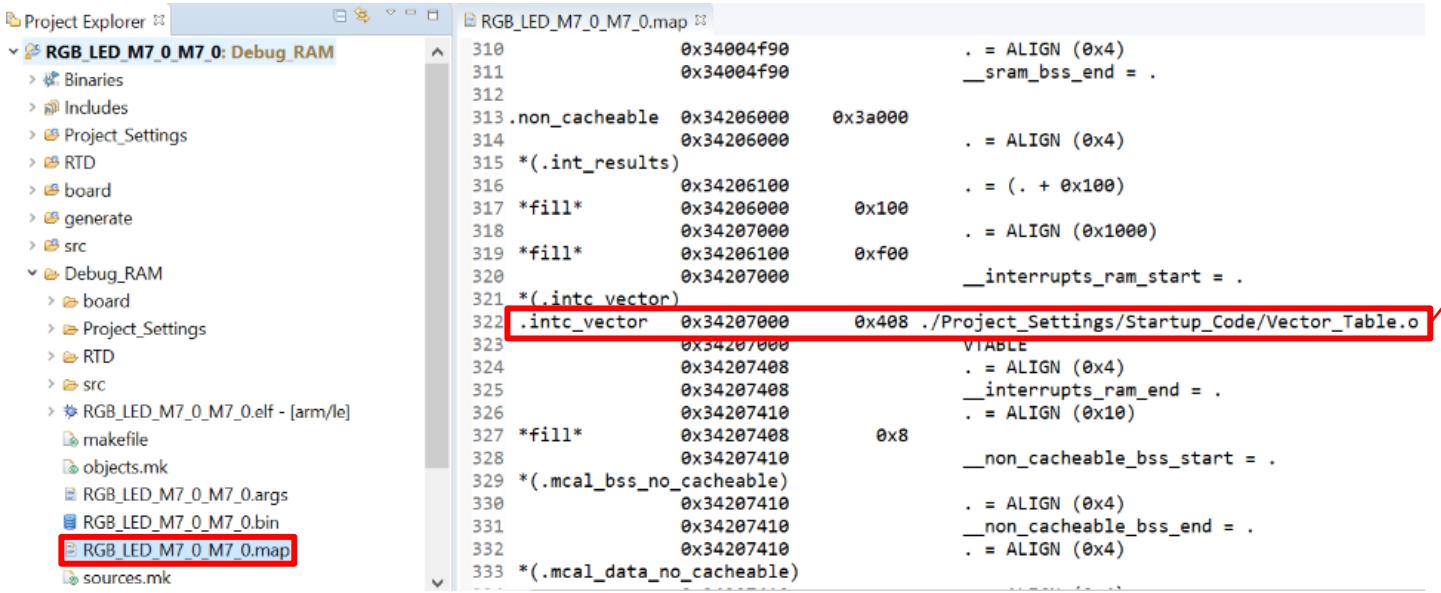
- Set Self-Test DCD, DCD and HSE to be reserved



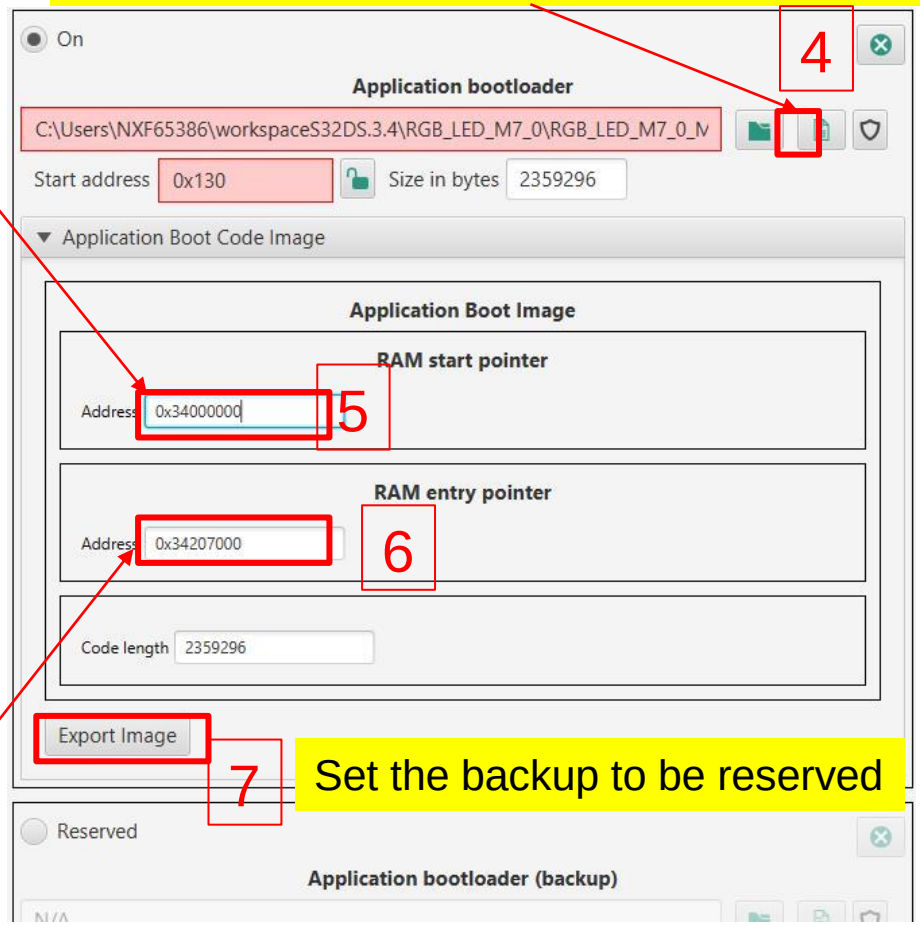
LIGHT UP RGB LED: MAKE IMAGE BY IVT TOOL

- Configure Application Boot Image according to .ld file and .map file





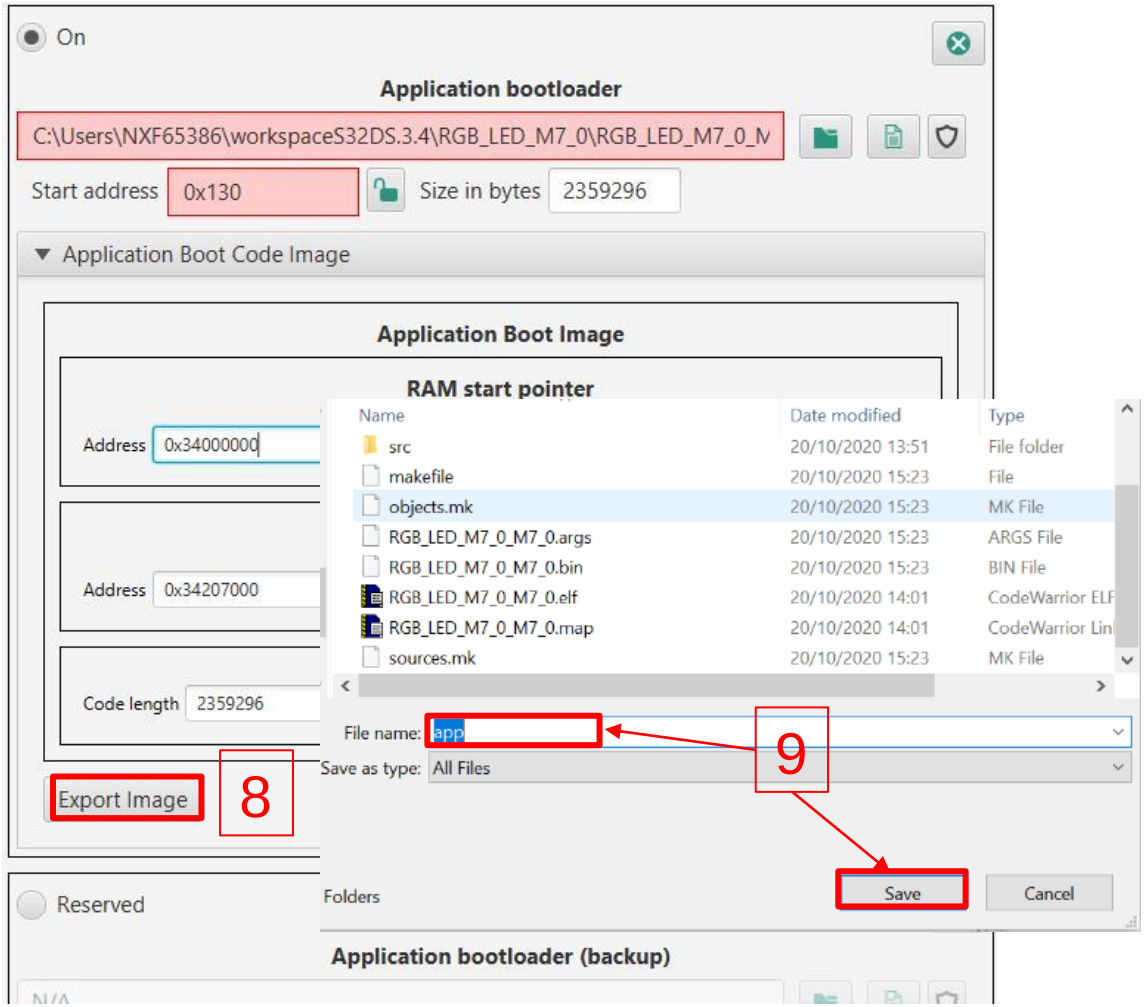
Click browse to select .bin files generated from Page 20



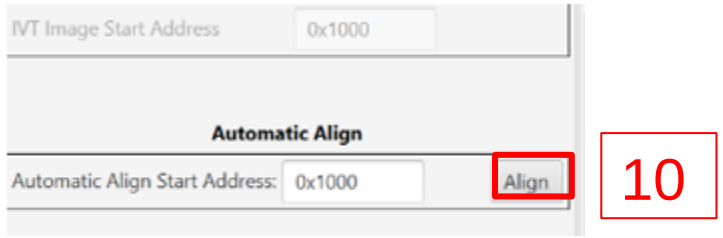
Set the backup to be reserved

LIGHT UP RGB LED: MAKE IMAGE BY IVT TOOL

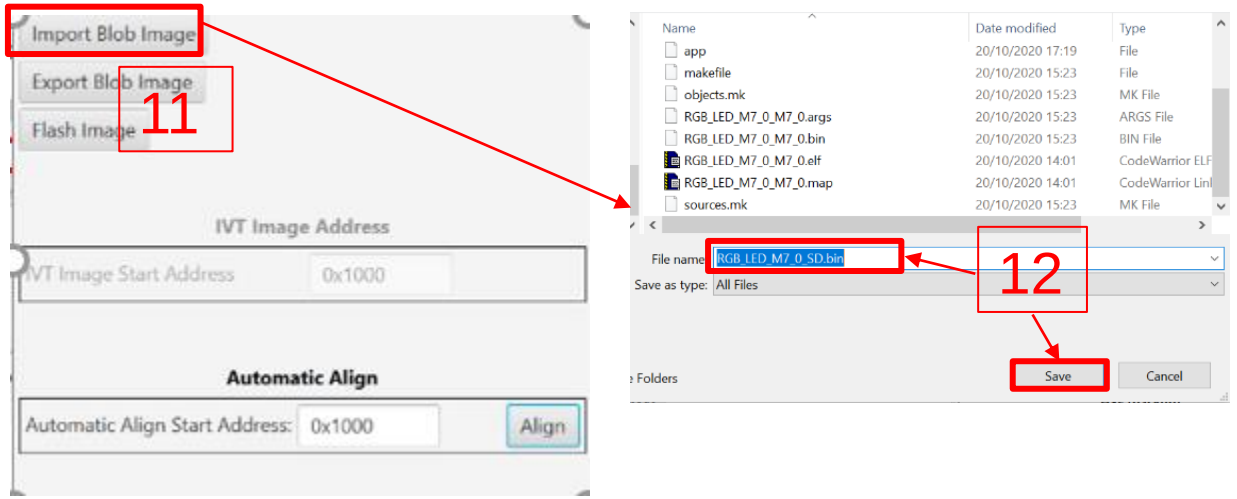
- Export and save image as any name



- Click on “Align” to resolve error



- Click on “Export Blob Image” to generate and save final image as any name



LIGHT UP RGB LED: DOWNLOAD IMAGE INTO SD CARD

1. Install and Run Cygwin as administrator.

2. Before inserting SD card into the slot, run "cat /proc/partitions" cmd and note the current devices.

```
$ cat /proc/partitions
major minor  #blocks  name   win-mounts

   8     0 500107608 sda
   8     1    307200 sda1
   8     2    524288 sda2
   8     3    131072 sda3
   8     4 499143680 sda4   C:\
```

3. After inserting SD card into the slot, run "cat /proc/partitions" cmd again and find out the SD card descriptor

```
$ cat /proc/partitions
major minor  #blocks  name   win-mounts

   8     0 500107608 sda
   8     1    307200 sda1
   8     2    524288 sda2
   8     3    131072 sda3
   8     4 499143680 sda4   C:\
   8    16  15224832 sdb
   8    17  15220736 sdb1   D:\
```

4. Erase the sub-partition info on the SD card

dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync

```
$ dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync
1+0 records in
1+0 records out
512 bytes copied, 0.0033774 s, 152 kB/s
```

5. Run "cd path of bin file" cmd to find out image generated in page 26, download image into the SD card

dd if=RGB_LED_M7_0_SD.bin of=/dev/sdb bs=1M count=4 && sync

```
$ dd if=RGB_LED_M7_0_SD.bin of=/dev/sdb bs=1M count=4 && sync
2+1 records in
2+1 records out
2363984 bytes (2.4 MB, 2.3 MiB) copied, 0.0649369 s, 36.4 MB/s
```

Note:
If the following prompt appears, please follow the steps below

```
dd: error writing '/dev/sdb': Permission denied
1+0 records in
0+0 records out
0 bytes copied, 0.0063647 s, 0.0 kB/s
```

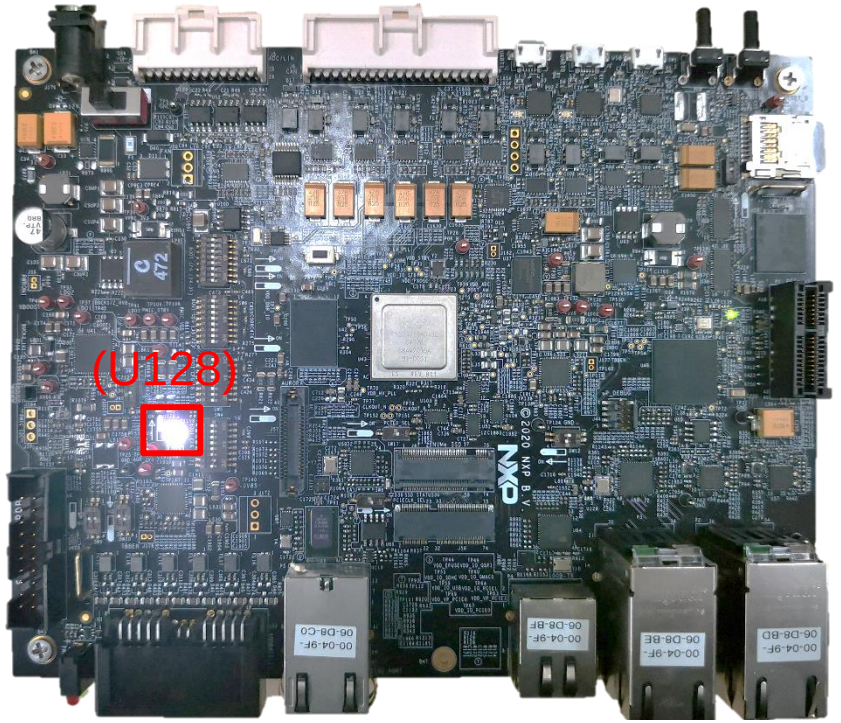
① Take out the SD card and insert it again
② Execute dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync
③ Take out the SD card and insert it again
④ Burn the image to the SD card

LIGHT UP RGB LED: SET RDB2 AND RUN APPLICATION

- Set SW3, 4, 9,10 to select SD card boot mode and set SW11 into “ON” to connect RGB LED with S32G pin(Refer to Quick Start Guide to find out the corresponding switch)

Part Reference Number	Setting
SW3	ON
SW4	7-ON, Other-OFF
SW9	1-OFF, 2-OFF
SW10	1-ON, 2-OFF
SW11	ON

- Power on RDB2, the RGB LED (U128) will be lighted in white color

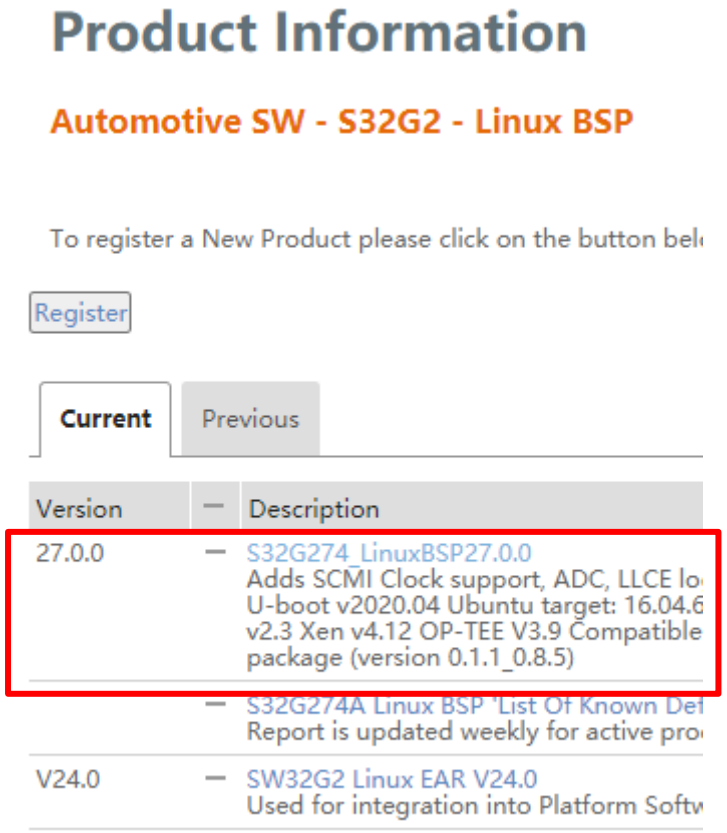


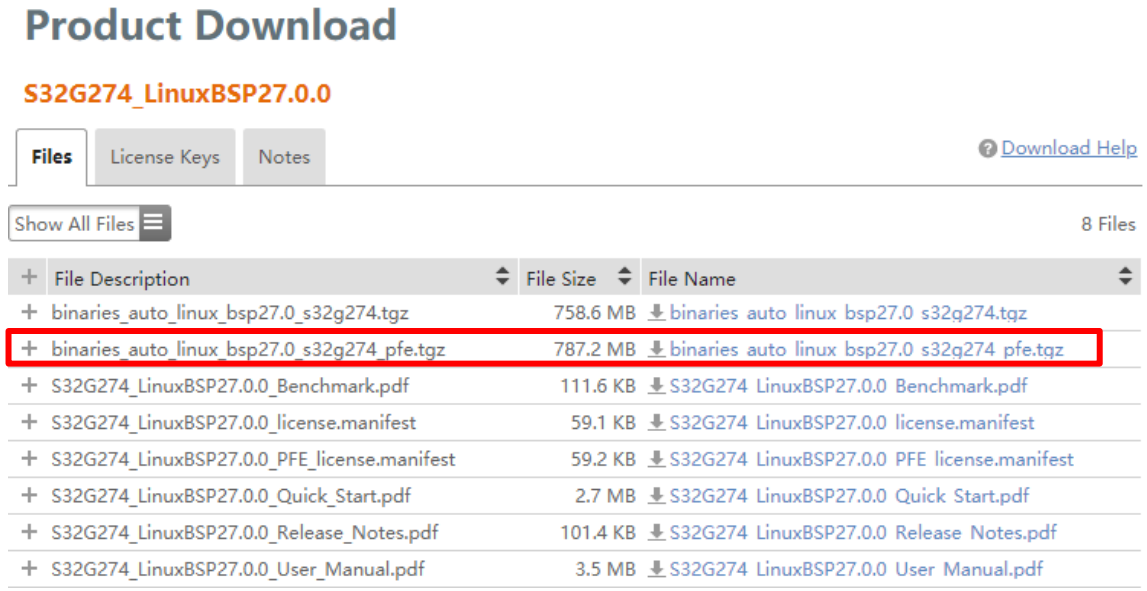
04.

RUN Linux BSP ON Cortex-A53 Core

STEP 1: DOWNLOAD LINUX BSP FROM SOFTWARE CENTER

- Click on “S32G274_LinuxBSP27.0.0”
- Download the file which included PFE



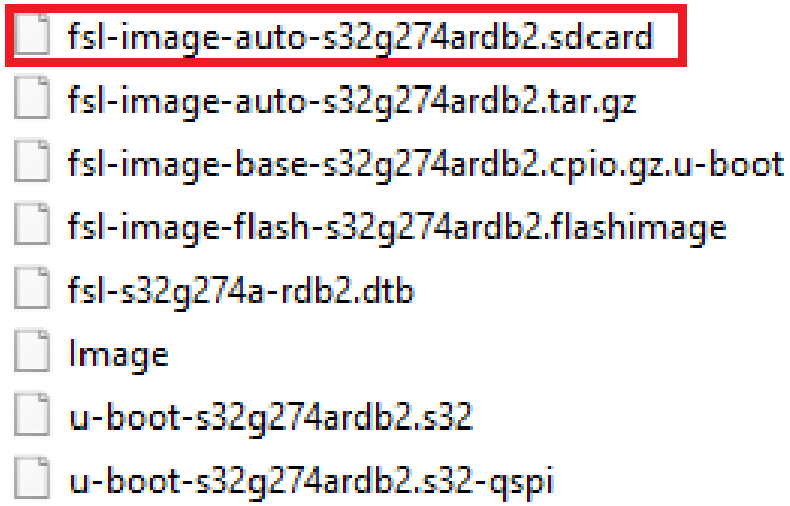


Note: If need more information for building BSP, refer to S32G274_Linux_BSP_27.0.0_User_Manual.pdf

STEP 1: DOWNLOAD LINUX BSP FROM SOFTWARE CENTRE

- Unzip and untar the download file by 7-zip

- Find the .sdcard file in the s32g274ardb2 folder



Note: The .sdcard file can be loaded into SD or eMMC. Refer to next step to know loading image

STEP 2: DOWNLOAD BSP IMAGE INTO SD CARD

1. Install and Run Cygwin as administrator

2. Before inserting SD card into the slot, run "cat /proc/partitions" cmd and note the current devices.

```
$ cat /proc/partitions
major minor  #blocks  name   win-mounts

   8     0 500107608 sda
   8     1    307200 sda1
   8     2    524288 sda2
   8     3    131072 sda3
   8     4 499143680 sda4   C:\
```

3. After inserting SD card into the slot, run cat /proc/partitions again and find out the SD card descriptor

```
$ cat /proc/partitions
major minor  #blocks  name   win-mounts

   8     0 500107608 sda
   8     1    307200 sda1
   8     2    524288 sda2
   8     3    131072 sda3
   8     4 499143680 sda4   C:\
   8    16  15224832 sdb
   8    17  15220736 sdb1   D:\
```

4. Erase the sub-partition info on the SD card

   dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync

```
$ dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync
1+0 records in
1+0 records out
512 bytes copied, 0.0033774 s, 152 kB/s
```

5. Burn all contents of the BSP image but the first four mega bytes into the SD card

   dd if=fsl-image-auto-s32g274ardb.sdcard of=/dev/sdb bs=1M skip=4 seek=4 && sync

```
$ dd if=fsl-image-auto-s32g274ardb2.sdcard of=/dev/sdb bs=1M skip=4 seek=4 && sy
nc
452+0 records in
452+0 records out
473956352 bytes (474 MB, 452 MiB) copied, 45.4739 s, 10.4 MB/s
```

6. Burn the first four mega bytes of the BSP image into the SD card

   dd if=fsl-image-auto-s32g274ardb.sdcard of=/dev/sdb bs=1M count=4 && sync

```
$ dd if=fsl-image-auto-s32g274ardb2.sdcard of=/dev/sdb bs=1M count=4 && sync
4+0 records in
4+0 records out
4194304 bytes (4.2 MB, 4.0 MiB) copied, 0.405748 s, 10.3 MB/s
```

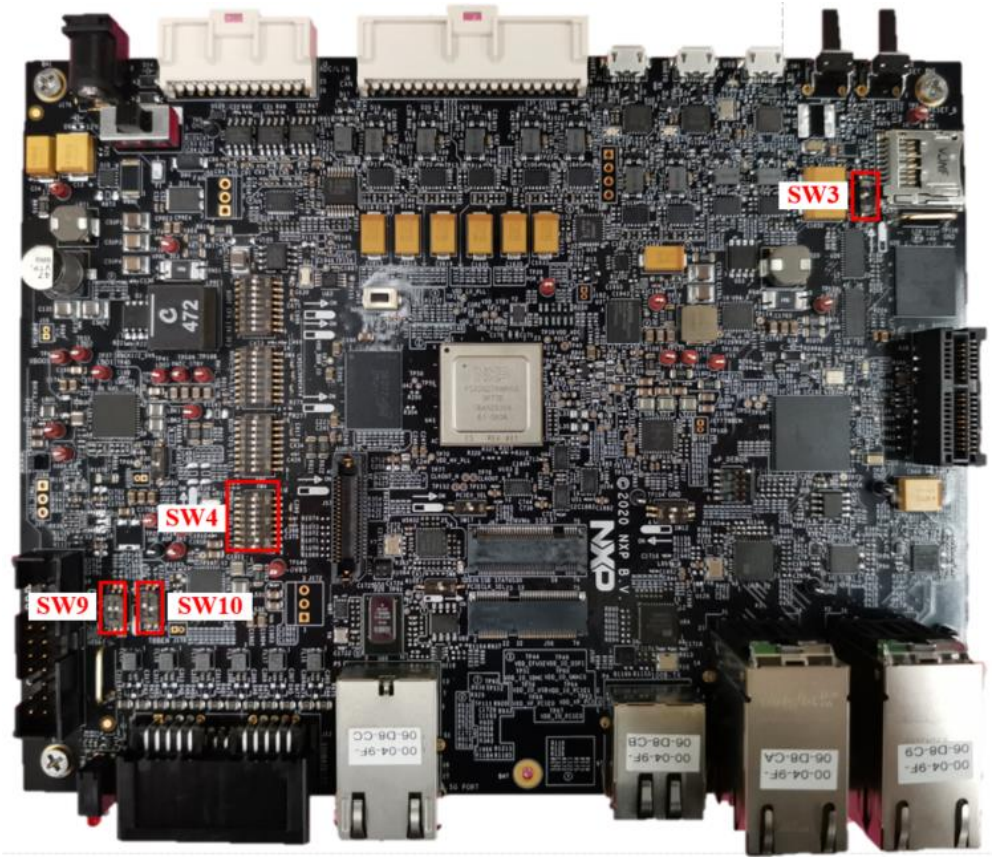
Note:
If the following prompt appears, please follow the steps below

```
dd: error writing '/dev/sdb': Permission denied
1+0 records in
0+0 records out
0 bytes copied, 0.0063647 s, 0.0 kB/s
```

① Take out the SD card and insert it again
② Execute dd if=/dev/zero of=/dev/sdb bs=512 count=1 && sync
③ Take out the SD card and insert it again
④ Burn the image to the SD card

STEP 3: SELECT BOOT MODE OF RDB2

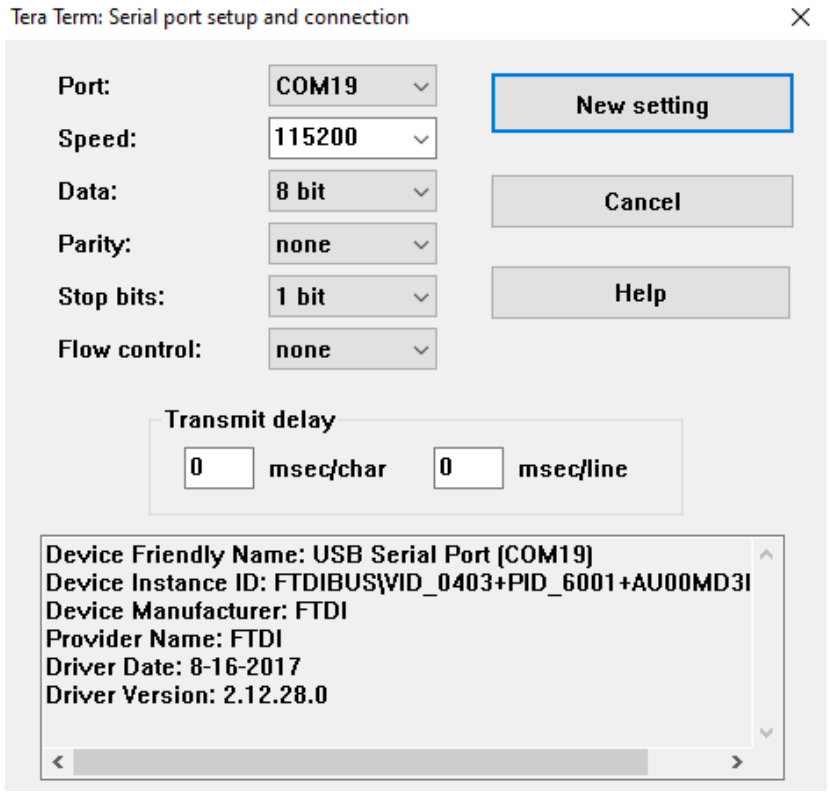
- Set RDB2 to SD card boot mode



Part Reference Number	Setting
SW3	ON
SW4	7-ON, Other-OFF
SW9	1-OFF, 2-OFF
SW10	1-ON, 2-OFF

Step 4: Run Linux bsp

- Connect UART cable to UART0. Then open serial terminal and configure COM port



- Power up the S32G-VNP-RDB and view print message in serial terminal

```
U-Boot 2020.04+g64825fa242 (Oct 09 2020 - 12:53:30 +0000)

CPU:    NXP S32G274A rev. 2.1.0
Reset cause: Power-On Reset
Model: NXP S32G27x
Board:  NXP S32G274A-RDB
DRAM:  4 GiB
CA53 core 1 running.
CA53 core 2 running.
CA53 core 3 running.
All (4) cores are up.
MMC:   FSL_SDHC: 0
Loading Environment from MMC... OK
Using external clock for PCIe0
Configuring PCIe0 as RootComplex(x2)
Using external clock for PCIe1
Frequency 125Mhz configured for PCIe1
Configuring PCIe1 as SGMII(x2) [XPCS0 2.5G, XPCS1 OFF]
PCIe0: Failed to get link up
Pcie0: LINK_DBG_1: 0x00000000, LINK_DBG_2: 0x00000800 (expected 0x000000d1)
DEBUG_R0: 0x00afb000, DEBUG_R1: 0x08200000
PCI: Failed autoconfig bar 1c
```

SECURE CONNECTIONS
FOR A SMARTER WORLD