S32G-VNP-GLDBOX SOFTWARE ENABLEMENT GUIDE

V0.2



NXP and the NXP logo are trademarks of NXP B.V. All other product or service names are the property of their respective owners. © 2017 NXP B.V. **PUBLIC**



CONTENTS SOFTWARE ENABLEMENT GUIDE

Prerequisites: Get "S32G Standard Software" Install Software Development Tool Light Up RGB LED Based On Real Time Drivers Run Linux BSP On Cortex-A53 Core





01.

PRE: Get "S32G Standard Software"





PREREQUISITES: GET "S32G STANDARD SOFTWARE"

 Please go to: <u>S32G Processors for Vehicle</u> <u>Networking</u> and use your NXP account to sign in.

Sign In
Email Address or NXP Company ID
Password
Sign in
Forgot your password?
Don't have an account? Register Now

Please download the S32G Standard Software for enablement

Production	Install Packages
S32 Design Studio for S32	SW32G2_S32DS_3.4.0_D2012.zip
Platform	S32DS.3.4_b201217_win32.x86_64.exe
S32G2 - Real Time Drivers	SW32_RTD_4_4_1.0.0_D2012_DS_updatesite.zip
S22C2 Linux BSD	binaries_auto_linux_bsp27.0_s32g274_pfe.tgz
332G2 - LINUX BSP	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

File Description	File Size 🌲	File Name
S32 Design Studio 3.4 development packages for offline use	3.7 GB	♣ SW32_S32DS_3.4.0_D2012.zip
S32 Design Studio 3.4 Release Notes	72.5 KB	S32DS_Release_Notes_3.4.0.pdf
S32 Design Studio 3.4 S32K1 Development Package Release Notes	47.5 KB	S32K1xx_Development_Package_Release_Notes_3
S32 Design Studio 3.4 S32S2TV Development Package Release Notes	49.6 KB	S32S2xxTV_Development_Package_Release_Notes_3.4.0.pdf
S32 Design Studio 3.4 S32V2 Development Package Release Notes	46 KB	S32V2xx_Development_Package_Release_Notes_3.4.0.pdf
S32 Design Studio Installation Guide	1.2 MB	S32DS_Installation_Guide_3.4.0.pdf
S32 Design Studio v3.4 Linux installer	1.1 GB	♣ S32DS.3.4_b201217_linux.x86_64.bin
S32 Design Studio v3.4 Windows installer	1.5 GB	

Click on "License Keys" to get Activation Code
 Product Download

S32 Design Studio for S32 Platform v.3.4 with support for S32G2 devices

			1	tem Description		S32G2 Standard Software (pre
Files	License Keys	Notes	(Order Number	3	SW32G2-STDSW-D_ULADX78731
			F	Purchase Order Number		-
_			1	Total Number of Licenses:		1
5	EXTERNAL	USE	/	Activation Code		ŧ

· Click .exe file to start installation

S32DS 3.4 b201217 win32.x86	InstallAnywhe	re	
••••••••••••••••••••••••••••••••••••••	S32 DS	InstallAnywhere is preparing to install	
4			
			Cancel

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

🔀 S32 Design Studio for S32 Plati	form 3.4 Installer	- X	
We	elcome to the S32 Design Studio f	or S32 Platform 3.4 Setun Wizard	
Welcome to the S32 Desig License Agreement Choose Install Location	This wizard will guide you through the in: N	(P Software Activation ×	
Choose Shortcut Folder	NOTE! Installer was unable to find Syno for manual installation please refer Gettir	?	
install Complete	It is recommended that you close all othe it possible to update relevant system file: Be ready to enter the activation code to a	Ok Cancel	
	was sent to your email when you accepted the code is also available in your NXP account on	elicense terms on the website. The activation	
	Slick Next to continue.	NXP Software Activation	×
L]	Choose activation type	
		Online Offline	
InstallAnywhere Cancel		Previous Next	

STEP 2: INSTALL DEVELOPMENT PACKAGES FOR S32G2

Click on S32 Design Studio version V3.4



Download development packages for your machine

+	File Description 🗘	File Size 🗢 🗘	File Name 🗘
+	S32 Design Studio 3.4 development packages for offline use	3.7 GB	<u>■</u> SW32 S32DS 3.4.0 D2012.zip
+	S32 Design Studio 3.4 development packages for offline use, support for S32G2 family	1.3 GB	± SW32G2 S32DS 3.4.0 D2012.zip 2
+	S32 Design Studio 3.4 Release Notes	72.5 KB	₹ S32DS Release Notes 3.4.0.pdf
+	S32 Design Studio 3.4 S32G2 Development Package Release Notes	48 KB	± S32G2xx Development Package Release Notes 3.4.0.pdf
+	S32 Design Studio Installation Guide	1.2 MB	₽ S32DS Installation Guide 3.4.0.pdf
+	S32 Design Studio v3.4 Linux installer	1.1 GB	± \$32D\$.3.4 b201217 linux.x86 64.bin
+	S32 Design Studio v3.4 Windows installer	1.5 GB	₽ S32DS.3.4 b201217 win32.x86 64.exe

 Select a directory as workspace and open "Install New Software ..."



6

- 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

88 Install				\times
Available Software				
Check the items that you wish to install.				
Work with: jar:file:/C:/Users/nxf47543/Downloads/SW32G2_S32DS_3.4.0_D2012.zip!/	~ Add		Manag	je
type filter text			Select	All
Name	Version		Deselec	t All
S32 Design Studio S32G2xx development package 3.4.0 S EP 3. INSTALL S32G2 REAL-TI				
5				
<		>		
Details				
				¢]
Show only the latest versions of available software Hide items that are a	lready installed			
Group items by category What is <u>already install</u>	ed?			
Show only software applicable to target environment				
Contact all update sites during install to find required software				
	6			
? < Back Next >	Finish		Cancel	



STEP 3: INSTALL S32G2 REAL-TIME DRIVERS

- Click on S32G2 Real Time Drivers 4.4 version 1.0.0
 - 1.0.0 S32G2 Real Time Drivers 4.4 version 1.0.0 This is the NXP S32 Real Time Drivers AUTOSAR 4.4 Version 1.0.0 release for the S32G274 platform. It can be used standarone or should be installed on top of S52 Design studio De v3.4. All software included in this release has RTM quality status in terms of testing and quality documentation RTM releases contain all planned features implemented and tested. RTM releases are candidates that can be used in production. For more details check the Release notes document.
- Download SW32_RTD_4_4_1.0.0_D2012_DS_updatesite.zip package

+	File Description 🗘	File Size 🗘	File Name	\$
+	S32_RTD_4.4_QualityPackage.zip	38.7 MB	♣ S32 RTD 4.4 QualityPackage.zip	
+	S32_RTD_4.4_Safety_Package.zip	1.2 MB		
+	SW32_RTD_4.4_1.0.0_D2012.exe	38.4 MB	<u>↓</u> SW32 RTD 4.4 1 .0.0 D2 012.exe	
+	SW32_RTD_4.4_1.0.0_D2012.pdf	495.7 KB	₽ SW32 RTD 4.4 1.0.0 D2012.pdf	
+	SW32_RTD_4.4_1.0.0_D2012.txt	1.7 KB	业 SW32 RTD 4.4 1.0.0 D2012.txt	
Ŧ	SW32_RTD_4_4_1.0.0_D2012_DS_updatesite.zip	66.6 MB	₿ SW32 RTD 4 4 1.0.0 D2012 DS updatesite.zip	

 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

🔀 Install				\times
Available Software				
Check the items that you wish to install.				9
Work with: :/Users/nxf47543/Downloads/SW32_RTD_4_4_1.0.0_D20	12_DS_updatesite (1).zip!/ ~	Add	Manag	e
type filter text			Select	All
Name	Ve	ersion	Deselec	t All
✓ ✓ S32 Design Studio S32CC SDK O ✓ ✓ <	1. 1.	0.0.202012221644 0.0.202012221644		
<		>		
2 items selected				
Details				
S32 Design Studio S32CC SDK 1.0.0.4cJJSch7E7E7AEEAUMM				÷.
Show only the latest versions of available software	Hide items that are alread	dy installed		
Group items by category	What is <u>already installed</u> ?			
Show only software applicable to target environment				
Contact all update sites during install to find required software				
	6			
?	k Next >	Finish	Cancel	
				D

9 EXTERNAL USE



03.

Light Up RGB LED Based On Real Time Drivers



10 EXTERNAL USE

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

File	Edit Source Refactor Navigate Search	Project ConfigTools Run Window Help	
	New Al	+Shift+N > 🗳 S32DS Project from Example	Ctrl+Alt+E
	Open File	🛎 S32DS Library Project	Ctrl+Alt+L
	Open Projects from File System	S32DS Application Project	Ctrl+Alt+A
	open rojects nom nie bystem		

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

					2
roject name: RGB LED M7 0					_
Use default location					
ocation: C:\Users\NXF65386\workspaceS32D	5.3.4			Browse	
Processors:	ToolChain Sel	ection:			
type filter text	Core Kind	Name	Toolchain	oit Bare-Meta	~
S32G Cortex-A53 Linux S32G233A Cortex-M7	M7 M7	Cortex-M7_1 Cortex-M7_2	NXP GCC 9.2 for Arm 32-1 NXP GCC 9.2 for Arm 32-1 NXP GCC 9.2 for Arm 32-1	oit Bare-Meta oit Bare-Meta	×
 S32G233A Cortex-A53 (decoupled) S32G233A Cortex-A53 (lockstep model) S32G234M Cortex-M7 					
S32G254A Cortex-M7	<				>
S32G254A Cortex-A53 (decoupled)	Description:				
S32G254A Cortex-A53 (lockstep mo	GNU 9.2 Too	Ichain is selected	ł		^
 \$32G274A, Rev2 Cortex-M7 \$32G274A, Rev1 Cortex-A53 (decould s32G274A, Rev2 Cortex-A53 (decould s32G274A, Rev2 Cortex-A53 (decould s32G274A, Rev2 Cortex-A53 (dockst) 					
< >>>					~

- 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

New S32DS Project for S32G274A_Rev2 Cortex-M7



Select required cores and parameters for them.

	ds M/ U		ds M/ 1	_	Is M7 2	
lore	Cortex-M	7_0	Cortex-M7_1		Cortex-M7_2	
ibrary	NewLib	~	NewLib	\sim	NewLib	
/O Support	No I/O	~	✓ No I/O		No I/O	
PU Support	Toolchain D	efault ~	Toolchain Defau	ult <u>~</u> 1	Toolchain Default	
anguage	C		C	~ (C	_
DKs						
ebugger	S32 Debugo	ger /				-
						1
Name	Ve	sion Device	(s) Device C	Core(s)	Edit/Show info	1
✓ Platform	SDK_S32G_20 1.0).0 S32G2	74A S32G274			
					Clone Rom git	
					Reload	
				ОК	Cancel	
					Curreer	

LIGHT UP RGB LED: PINS CONFIGURATION

• Select the created project and open pins tool



 Select peripheral Signals page and find out SIUL2_0 box

■ Pins Peripheral Signals ≅ ♥ Power Groups	
┺╋╋ ●●●● \$ ∅	
> 🗹 POST	^
> 🗹 PowerAndGround	
> QuadSPI	
<u>→ 🛛 RTC</u> 2	
> SIUL2_0	
> SIUL2_1	

- 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



Add gpio dio driver
 a. Click on "+" option

b. Select "Siul2_Dio" and click on "ok"

Components 🛛	Peripherals		🔀 Select component		— 🗆	×
type filter text		• 11	Show only components	in toolchain project 🗹 Show only latest co	mponent vers	ions
	MCAL	0	Configuration component	Component description	Category	Require ^
		20	Sd_Emmc_lp	SD Configuration	Drivers	platfor
	Drivers	ao	▲SEMA42	Semaphores2 Ip Driver	Drivers	platfor
osif_1		Siul2_Port_1	[▲] Siul2_Dio		Drivers	plat or
			[●] Siul2_lcu	SIUL2 Driver	Drivers	platfor
	OS	0	Siul2_Port		Drivers	platfor
			≜Spi_lp	Spi Ip Configuration	Drivers	platfor
			≜Stm	STM IPL Configuration	Drivers	platfor
			≜Swt_lp	Swt Configuration	Drivers	platfor
			System_lp	IP configuration 3	Drivers	platfor
			≜Tmu_lp	TMU Configuration	Drivers	platfor
			≜Usdhc_Ip	Usdhc Configuration	Drivers	platfor 🗸
			<			>
				ОК	Cai	ncel

Select Components to find out Drivers option





LIGHT UP RGB LED: CHECK CONFIGURATION AND UPDATE CODE

open pins tool to check configuration

Routed Pi	ns for BOA	ARD 3	0 🛛 🗠	
#	Periph	Signal	Route to	Label
≜ Y9	SIUL2_0	gpio, 6	PA_06	
🏝 Y11	SIUL2_0	gpio, 7	PA_07	1
💩 U8	SIUL2_0	gpio, 95	PF_15	

• open Peripherals tool to check configuration



2

Click on "Update Code"

15

F	ile E	dit	Source	Refactor	Navigate	Search	Proj	ect Cor	nfigTools	Periphera	als
	<u>n</u> → 1		a 📑	RDB2_LED	_1_M7_0			~	🔺 🖻 U	pdate Coo	de
	a . ∎Con	ł] ◄ npor	nents 🛛	▼ 🗢 ▼ ∦ Periphe	🖬 erals	6	• 🗆	ය Siul2	2_Dio_1 🛛		
	type f	filter	text			0	† ↓	Siul	2_Dio	[Drivers]	
				MCAL			0	Name	Siul2_Di	o_1	3
				Drivers			0	Mode	Siul2_Di	0	
	os	if_1	Si	ul2_Dio_1	Siu	l2_Port_1		DioG	eneral		
				OS			•	Dio	Develop	ment Erro	r D
(†	FF	RN		JSF							

• 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



ЪТ 32 /* Including necessary configuration files. */ 33 #include "Mcal.h" 34 35 #include "Clock Ip.h" 36 #include "Siul2 Port Ip.h" 37 #include "Siul2 Dio Ip.h" 38

Initialize clocks

~

> 🗁 include	Build Targets
Y 🗁 src	Clock_lp_Private.h
Clock_Ip_Divider.c	clockConfig : const Clock_Ip_ClockConfigType*
 Is Clock_Ip_Divider Irigger.c Clock_Ip_EvtOsc.c 	dockTreeIsConsumingPII : boolean
Clock Ip FracDiv.c	Clock_lp_Init(const Clock_lp_ClockConfigType*) : Clock_lp_StatusType
> Clock_Ip_Gate.c	 Clock_lp_InitClock(const Clock_lp_ClockConfigType*) : void
Clock_Ip_IntOsc.c	Clock_Ip_GetPllStatus(void) : Clock_Ip_PllStatusType
 Clock_lp_Monitor.c Clock_lp_Pll_c 	 Clock_lp_DistributePll(void) : void
 Clock Ip ProgFregSwitch.c 	 Clock_Ip_DisableClockMonitor(Clock_Ip_NameType) : void
> Clock_Ip_S32G2XX.c	 Clock_Ip_GetClockMonitorStatus(Clock_Ip_NameType) : Clock_Ip_CmuStatusType
Clock_Ip_S32R45.c	 Clock_lp_ClearClockMonitorStatus(Clock_lp_NameType) : void
Clock In Selector c	 Clock Ip UpdateFrequencies(power modes t) ; void
> Det stub.c	 Clock Ip DisableModuleClock(Clock Ip NameType) : void
> Det.c	 Clock Ip EnableModuleClock(Clock Ip NameType) : void
> Oslf_Timer_System.c	 Clock Ip GetClockFrequency(Clock Ip NameType) ; uint32
 Oslf_Timer.c Sinda Dia In c 	Clock Ip TimeDelav(void) : void
> iii Siul2_Dio_p.c	
🛩 🐸 board	
> Clock_lp_Cfg.c	# MCU_START_SEC_CONFIG_DATA_UNSPECIFIED
> 🖻 Clock_lp_Cfg.h	Mcu_MemMap.h
Clock Ip PBcfa.c	^e Mcu_aClockConfigPB : const Clock_Ip_ClockConfigType[]
> Clock_Ip_PBcfg.h	
> iul2_Port_lp_Cfg.c	
> B Siul2_Port_lp_Cfg.h	
> 🤐 generate	64 */
Y 🙆 STC	65@int main(void)
> 🖻 main.c	
> Debug_RAM	67 /* Write your coze here */
> 💋 include.bak	69 // Clocking
> 🐸 Linker_Files.bak	<pre>70 Clock_Ip_Init(Mcu_aClockConfigPB);</pre>
> 1× src bak	71







LIGHT UP RGB LED: APPLICATION CODE

Initialize pins

Y ≝ RTD			Project Explorer 😂	
 include isrc Clock_lp_Divider.c Clock_lp_ExtOsc.c Clock_lp_fracDiv.c Clock_lp_fracDiv.c Clock_lp_fracDiv.c Clock_lp_fracDiv.c Clock_lp_Nonitor.c Clock_lp_Pll.c Clock_lp_Pll.c Clock_lp_S32G2XX.c Clock_lp_S32R45.c Clock_lp_Selector.c Clock_lp_Selector.c Clock_lp_Selector.c Det_stub.c Det_stub.c Det.c Oslf_Timer.c Siul2_Port_lp.c 	 © Outline Solution Content of the second s	E I ^a k × void _input_mux) : void	 ✓ In RTD > include > in	 E Outline [©] Siul2
 Project Explorer S board Clock_lp_Cfg.c Clock_lp_Cfg.h Sdk_project_config.h Siul2_Port_lp_Cfg.c Siul2_Port_lp_Cfg.h 	Build Targets # _SIUL2_PORT_IP_CFG_H_ S326274A_SIUL2.h Siul2_Port_Ip_Types.h # DEV_ASSERT0 # NUM_OF_CONFIGURED_PINS0 g_pin_mux_InitConfigArr0 : pin_settings_config[]		 Project Explorer # RGB_LED_M7_0_M7_0: Debut Rinaries Includes Project_Settings RTD Board Generate Clock_Ip_Cfg_Defines.t Osif_Cfg.h Siul2_Dio_Ip_Cfg.h 	g_RAM
 > Includes > Project_Settings > RTD > Board > Generate > Imain.c > Debug_RAM > Minclude.bak > Minclude.bak > Sinclude.bak 	<pre>>>-/: 60 \brief The main function for the project. 61 \details The startup initialization sequence is the 62 * - startup asm routine 63 * - main() 64 */ 65=int main(void) 66 { 67 /* Write your code here */ 68 69 // Clocking 70 Clock_Ip_Init(Mcu_aClockConfigPB); 71 72 // 10 pads configuration 73 Siul2_Port_Ip_Init(NUM_OF_CONFIGURED_PINS0. g pin 73</pre>	e following: n_mux_InitConfigArr0)	 Siul2_Port_[p_Defines.h Stul2_Port_[p_Defines.h Stul2_Port_Polefues.h Stul2_Port_Polefues.h Stul2_Port_Polefues.h Includes Stul2_Port_Polefues.h Stude.bak 	64 65 66 67 78 68 69 70 71 72 73 75 76 77 77

· 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 μ SDHC 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 μ SDHC interface. BootROM also uses 4 KB of SRAM memory starting at 343FF000h for ADMA descriptors in case of μ SDHC boot. The Application boot image header should not point to this location in case of μ SDHC boot.

	1
SGB_LED_M7_0_M7_0: Debug_RAM	39 MEMORY
> 🗱 Binaries	40 (
> 🔊 Includes	41 int_itcm : ORIGIN = 0x00000000, LENGTH = 0x00000000 /* 0KB - Not Supported */
✓	42 int_dtcm : ORIGIN = 0x20000000, LENGTH = 0x00010000 /* 64K */
> 🖻 Startup Code	43 int snam shareable · ORIGIN = 0x2200000 LENGTH = 0x00000000 /* 15KR */
> 🖉 Debugger	44 int_sram : ORIGIN = 0x340000000, LENGTH = 0x00200000 /* 2MB */
> 💕 Linker Files	45 Int_sram_stack_c0 : ORIGIN = 0.542002000, LENGTH = 0.5000022000 / 5KB */
> 🥔 RTD	47 int_sram_stack_c2 : ORIGIN = 0x34204000, LENGTH = 0x00002000 /* 8KB */
> 😂 board	48 int_sram_no_cacheable : ORIGIN = 0x34206000, LENGTH = 0x00100000 /* 1MB, needs to include int_results *
> @ generate	49 ram_rsvd2 : ORIGIN = 0x34800000, LENGTH = 0 /* End of SRAM */
Y 🐸 SIC	50 LLCE CAN SHAREDMEMORY : ORIGIN = 0x43800000 LENGTH = 0x30000
> 🗟 main.c	52 LLCE LIN SHAREDMEMORY : ORIGIN = 0x4383D000 LENGTH = 0x3000
> 🖻 Debug RAM	53 LLCE_BOOT_END : ORIGIN = 0x43840000 LENGTH = 0x50
> 🖉 include.bak	v 54}

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



lb sources.mk

18 EXTERNAL USE

• 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

 Open Pins Open Clocks Open Peripherals Open DCD
 Open Clocks Open Peripherals Open DCD
Open Peripherals Open DCD
Open DCD
Open IVT
🖆 Open QuadSPI
🧭 Open DDR
Manage SDK Components

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

	Boot Configuration
Boot Targe M7_0 +	
BOOTSEQ: Secured boot mode	
Boot Target Watchdog	
L	Life Cycle
Life Cycle Keep existing configuration	2
	Interface selection
Boot device type SD/MMC/eMMC +	Interface selection
Boot device type SD/MMC/eMMC	Interface selection

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

Reserved Start address 0x100 Size in bytes Reserved Self-Test DCD (backup) N/A Start address 0x108 Size in bytes DCD N/A Start address 0x108 Size in bytes DCD N/A Start address 0x110 Size in bytes DCD (backup) N/A Start address 0x118 Size in bytes N/A Start address 0x118 Size in bytes N/A Start address 0x118 Size in bytes A Pi/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes Pi/A Start address 0x120 Size in bytes Pi/A N/A Start address 0x120 Size in bytes Pi/A N/A N/A N/A Size in bytes Pi/A N/A Size in bytes Size in bytes	Image Table	
Self-Test DCD N/A Image: Start address 0x100 Start address 0x108 Size in bytes 4 Image: Start address 0x108 Size in bytes 0 DCD N/A Image: Start address 0x100 Size in bytes 0 DCD N/A Image: Start address 0x110 Size in bytes 0 DCD N/A Image: Start address 0x110 Size in bytes 0 DCD (backup) N/A Image: Start address 0x118 Size in bytes 4 Image: Start address 0x118 Size in bytes 4 Image: Start address 0x120 Size in bytes 4 Image: Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x120 Image: Start addre	Reserved	۲
N/A Start address DCD N/A Start address Ox108 Size in bytes A N/A Start address Ox108 Size in bytes DCD N/A Start address Ox100 Size in bytes DCD N/A Start address Ox110 Size in bytes CO (backup) N/A Start address Ox118 Size in bytes HSE N/A Start address Ox120 Size in bytes HSE (backup) N/A Start address Ox120 Size in bytes HSE (backup) N/A Start address Ox120 Size in bytes HSE (backup) N/A Start address Ox128 Size in bytes	Self-Test DCD	
Start address 0x100 Size in bytes 4	N/A	
Reserved Self-Test DCD (backup) N/A Start address 0x108 Size in bytes A CO N/A Start address 0x110 Size in bytes A CO N/A Start address 0x118 Size in bytes CO N/A Start address 0x118 Size in bytes A CO N/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x128 Size in bytes A	Start address 0x100 Size in bytes 4	
Self-Test DCD (backup) N/A Image: Size in bytes 4 Start address 0x108 Size in bytes 4 N/A Image: Size in bytes 4 N/A Image: Size in bytes 4 Reserved Image: Size in bytes 4 Image: N/A Image: Size in bytes 4	Reserved	8
N/A Start address 0x108 Size in bytes A N/A Start address 0x110 Size in bytes A B DCD N/A Start address 0x118 Size in bytes A B B CD N/A Start address 0x118 Size in bytes A B B N/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes A N/A Start address 0x120 Size in bytes A N/A Start address 0x120 Size in bytes A Size in bytes A Size in bytes	Self-Test DCD (backup)	
Start address 0x108 Size in bytes 4	N/A	
Reserved N/A Start address 0x110 Size in bytes A DCD (backup) N/A Start address 0x118 Size in bytes A N/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x128 Size in bytes	Start address 0x108 Size in bytes 4	
DCD N/A Start address 0x110 Size in bytes A DCD (backup) N/A Start address 0x118 Size in bytes A Reserved HSE N/A Start address 0x120 Size in bytes HSE N/A Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x120 Size in bytes HSE (backup) N/A Start address 0x128 Size in bytes	Reserved	0
N/A Start address 0x110 Size in bytes A DCD (backup) N/A Start address 0x118 Size in bytes A N/A Start address 0x120 Size in bytes HSE N/A Size in bytes HSE N/A N/A Size in bytes N/A N/A Size in bytes HSE (backup) N/A Start address Ox128 Size in bytes	DCD	
Start address Ox110 Size in bytes 4 Reserved DCD (backup) N/A Start address Ox118 Size in bytes 4 Reserved Reserved HSE N/A Start address Ox120 Size in bytes 4 NA Start address Ox120 Size in bytes 4 NA Start address Ox120 Size in bytes 4 Start address Ox120 Size in bytes 4	N/A 3	
Reserved DCD (backup) N/A Start address 0x118 Size in bytes 4 HSE N/A Start address 0x120 Size in bytes HSE FW Configuration HSE (backup) N/A Start address 0x120 Size in bytes MA Size in bytes N/A Size in bytes Size in bytes N/A Size in bytes	Start address 0x110 🕒 Size in bytes 4	
DCD (backup) N/A N/A Start address 0x118 Size in bytes 4 N/A HSE N/A Start address 0x120 Size in bytes 4 HSE FW Configuration HSE (backup) N/A Start address 0x128 Size in bytes 4	Reserved	8
N/A Start address 0x118 Size in bytes 4 Reserved Reserved KIA Start address 0x120 Size in bytes 4 KIA Start address 0x120 Size in bytes 4 KIE Start address 0x120 Size in bytes 4 Start address 0x120 Size in bytes 4 Start address 0x128 Size in bytes 4	DCD (backup)	
Start address 0x118 Size in bytes 4	N/A	
Reserved HSE N/A Start address 0x120 Reserved HSE (backup) N/A Start address 0x128 Size in bytes 4	Start address 0x118 Size in bytes 4	
HSE N/A Start address 0x120 Size in bytes 4 HSE FW Configuration Reserved HSE (backup) N/A Start address 0x128 Size in bytes 4	Reserved	8
N/A Start address 0x120 Size in bytes 4 HSE FW Configuration Reserved HSE (backup) N/A Start address 0x128 Size in bytes 4	HSE	
Start address 0x120 Size in bytes 4 HSE FW Configuration Keserved KSE (backup) N/A Start address 0x128 Size in bytes 4	N/A	
Start address Ux120 Size in bytes 4 Size in bytes 4 Size in bytes 4 Size in bytes 4		
HSE FW Configuration Reserved HSE (backup) N/A Start address 0x128 Size in bytes 4	Start address Ux12U Size in bytes 4	
Reserved HSE (backup) N/A Start address 0x128 Size in bytes 4	► HSE FW Configuration	
HSE (backup)	Reserved	8
N/A Start address 0x128 Size in bytes 4	HSE (backup)	
Start address 0x128	N/A	
	Start address 0x128	



LIGHT UP RGB LED: MAKE IMAGE BY IVT TOOL

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

									Page 20
Project Explorer 88		Binker_ram.ld							
V PRGB_LED_M7_0_M7_0: Debug_RAN	/1	40 {	1 * * * *					6	
> 🐝 Binaries		41 int_	_itcm		ORIGIN = 0X	2000000000,	LENGTH = 0x00000000		
> 🔊 Includes		42 Int_	_uccm _snam_shaneah	: ام	ORIGIN = 0x	200000000,	LENGTH = 0x00010000		Application bootloader
✓ [™] Project_Settings		44 int	sram		ORIGIN = 0x	34000000	$I = 0 \times 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0$	1	
> 🗁 Startup Code		45 int	sram stack c	a :	ORIGIN = 0x	34200000.	$LENGTH = 0 \times 00002008$		C:\Users\NXF65386\workspaceS32DS.3.4\RGB_LED_M7_0\RGB_LED_M7_0_N
> A Debugger		46 int	sram stack c	1 :	ORIGIN = 0x	34202000,	LENGTH = 0x00002000		
V Cebugger		47 int	sram_stack_c	2 :	ORIGIN = 0x	34204000,	LENGTH = 0x00002000		Start address UX130 Size in bytes 2359296
 Linker_Files 		48 int	sram_no_cach	eable :	ORIGIN = 0x	34206000,	LENGTH = 0x00100000	N	
Mainker_ram.id		49 ram _	rsvd2	:	ORIGIN = 0x	34800000,	LENGTH = 0		Application Boot Code Image
> 🛎 RTD		50							
> 🐸 board		51 LLCE	_CAN_SHAREDM	EMORY :	ORIGIN = 0x	43800000 L	$ENGTH = 0 \times 3D000$		Application Boot Image
> 🐸 generate		52 LLCE	LIN_SHAREDM	EMORY :	ORIGIN = 0x	4383D000 L	$ENGTH = 0 \times 3000$		
> 🖴 src		53 LLCE	E_BOOT_END	:	ORIGIN = 0X	43840000 L	ENGIH = 0x50		RAM start pointer
S C Dabua BAM		54 j							
									Address 0x34000000
🕒 Project Explorer 🛛 🗆 😫	▶ [×] [□] □ B RGB_	LED_M7_0_M7_0.m	nap 🖾						
SGB_LED_M7_0_M7_0: Debug_RAM	 310 		0x34004f90		. = ALIGN	(0x4)			RAM entry pointer
> 🖑 Binaries	311		0x34004f90		sram_bss	_end = .			Address 0::24207000
> 🔊 Includes	312	non cacheable	9x34296999	8x3a888					Address 0x34207000
> Beroject_Settings	314	hon_cacheaoire	0x34206000	0,50000	. = ALIGN	(0x4)			
> 🐸 RTD	315	*(.int_result/	s)						
> 🖴 board	316		0x34206100		. = (. + 0	×100)			Code length 2359296
> 🐸 generate	317	*†111*	0x34206000	0×100		(0,1000)			
> 🐸 src	319	*fill*	0x34206100	0xf00	. = ALIGN	(971999)			
✓	320		0x34207000		interrup	ts_ram_star	t = .		Evport Image
> 🗁 board	321	*(.intc vector	r)						
> 😑 Project_Settings	322	.intc_vector	0x34207000	0x408	./Project_Sett	ings/Startu	p_Code/Vector_Table.o		7 Set the backup to be reserved
> 🗁 RTD	323		0x34207000 0x34207408			(0x4)			
> 🗁 src	325		0x34207408		interrup	ts ram end	= .		Reserved 8
> \$ RGB_LED_M7_0_M7_0.elf - [arm/le]	326		0x34207410		. = ALIGN	(0x10)			
🗟 makefile	327	*fill*	0x34207408	0×8					Application bootloader (backup)
🗟 objects.mk	328	*(hr= _	0x34207410		non_cach	eable_bss_s	tart = .		N/A 📭 🖻 🗂
RGB_LED_M7_0_M7_0.args	329	(.mcai_bss_h	0x34207410		= ALTON	(0x4)			
RGB_LED_M7_0_M7_0.bin	331		0x34207410		non cach	eable bss e	nd = .		
RGB_LED_M7_0_M7_0.map	332		0x34207410		. = ALIGN	(0x4)			
🗟 sources.mk	× 333	*(.mcal_data_r	no_cacheable)						

20 EXTERNAL USE

Click browse to select .bin files generated from



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.



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

\$ cat	/proc/	/partitions	5	
major	minor	#blocks	name	win-mounts
	0	500107000		
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

<pre>\$ dd if=/dev/zero of=/dev/sdb bs=512 count=1</pre>	&& 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

N	ote: the following prompt appears, please follow the steps below
	d: error writing '/dev/sdb': Permission denied
1	+0 records in
e)+0 records out) 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

Product Information

Automotive SW - S32G2 - Linux BSP

To register a New Product please click on the button belo

Current	Pre	evious
Version	—	Description
27.0.0	-	S32G274_LinuxBSP27.0.0 Adds SCMI Clock support, ADC, LLCE lo U-boot v2020.04 Ubuntu target: 16.04.6 v2.3 Xen v4.12 OP-TEE V3.9 Compatible package (version 0.1.1_0.8.5)
	-	S32G274A Linux BSP 'List Of Known Def Report is updated weekly for active pro-
V24.0	_	SW32G2 Linux EAR V24.0 Used for integration into Platform Softw

Product Download

S32G274_LinuxBSP27.0.0

Files License Keys Notes		O Download Help
Show All Files		8 Files
+ File Description	File Size	lame 🗘
+ binaries_auto_linux_bsp27.0_s32g274.tgz	758.6 MB 上 bir	aries auto linux bsp27.0 s32g274.tgz
+ binaries_auto_linux_bsp27.0_s32g274_pfe.tgz	787.2 MB 上 bir	aries auto linux bsp27.0 s32g274 pfe.tgz
+ S32G274_LinuxBSP27.0.0_Benchmark.pdf	111.6 KB 👤 S3	2G274 LinuxBSP27.0.0 Benchmark.pdf
+ S32G274_LinuxBSP27.0.0_license.manifest	59.1 KB 📕 S3	2G274 LinuxBSP27.0.0 license.manifest
+ S32G274_LinuxBSP27.0.0_PFE_license.manifest	59.2 KB 📕 S3	2G274 LinuxBSP27.0.0 PFE license.manifest
+ S32G274_LinuxBSP27.0.0_Quick_Start.pdf	2.7 MB 📕 S3	2G274 LinuxBSP27.0.0 Quick Start.pdf
+ S32G274_LinuxBSP27.0.0_Release_Notes.pdf	101.4 KB 📕 S3	2G274 LinuxBSP27.0.0 Release Notes.pdf
+ S32G274_LinuxBSP27.0.0_User_Manual.pdf	3.5 MB 🛓 S3	2G274 LinuxBSP27.0.0 User Manual.pdf

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

Register



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

fsl-image-auto-s32g274ardb2.sdcard

- fsl-image-auto-s32g274ardb2.tar.gz
- fsl-image-base-s32g274ardb2.cpio.gz.u-boot
- fsl-image-flash-s32g274ardb2.flashimage

🗋 fsl-s32g274a-rdb2.dtb

- 🗋 lmage
- u-boot-s32g274ardb2.s32
- 🗋 u-boot-s32g274ardb2.s32-qspi

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.



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



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

27 EXTERNAL USE

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

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

- 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







SECURE CONNECTIONS FOR A SMARTER WORLD

NXP and the NXP logo are trademarks of NXP B.V. All other product or service names are the property of their respective owners. © 2017 NXP B.V.