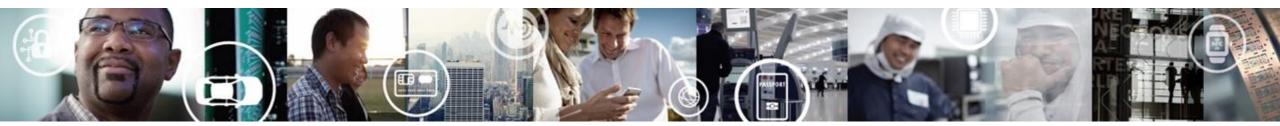
S32K146 EVB

QUICK START GUIDE

REV1.1

APPLIES FOR: S32K146 EVB (SCH-29844 REV B)





SECURE CONNECTIONS FOR A SMARTER WORLD



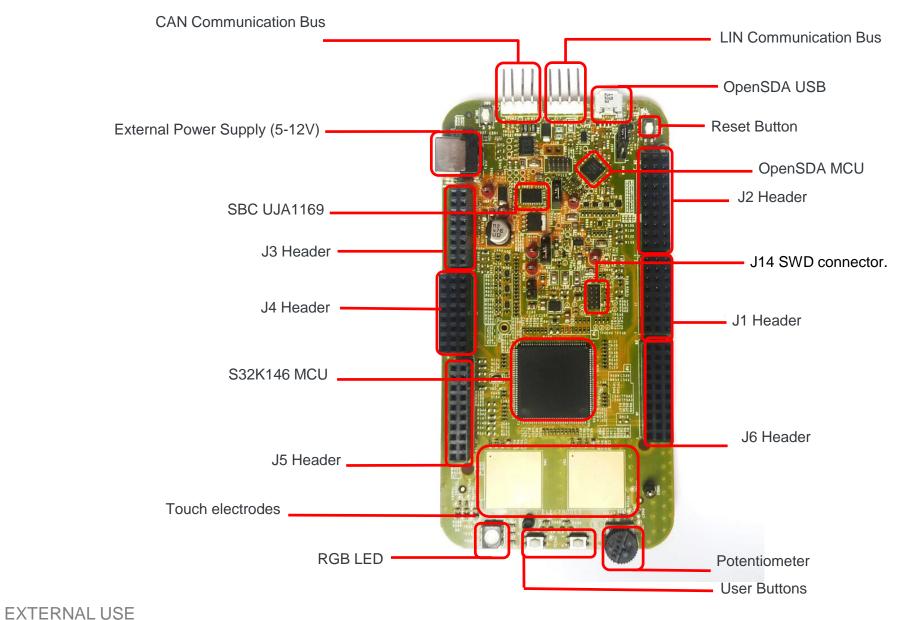
Contents:

- Get to Know S32K146 EVB
- JumpStart Setup
- JumpStart based on the FreeMASTER tool
- Introduction to OpenSDA
- Creating a new S32DS project for S32K146
- S32DS Debug basics
- Create a P&E debug configuration





Get to know S32K146-EVB





Downloaded from Arrow.com.

2

S32K146 EVB Features:

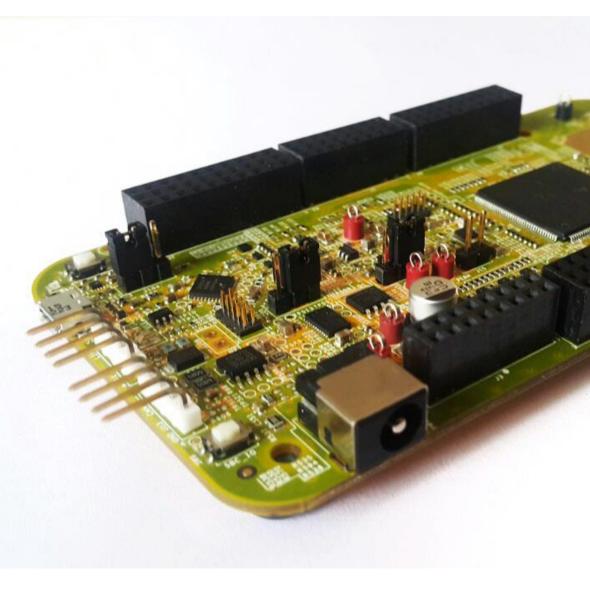
- Supports S32K146 144LQFP
- Small form factor size supports up to n" x n"
- Arduino[™] UNO footprint-compatible with expansion "shield" support
- Integrated open-standard serial and debug adapter (OpenSDA) with support for several industry-standard debug interfaces
- Easy access to the MCU I/O header pins for prototyping
- On-chip connectivity for CAN, LIN, UART/SCI.
- SBC UJA1169 and LIN phy TJA1027
- Potentiometer for precise voltage and analog measurement
- RGB LED
- Two push-button switches (SW2 and SW3) and two touch electrodes
- Flexible power supply options
 - microUSB or
 - external 12V power supply





Jumper Settings

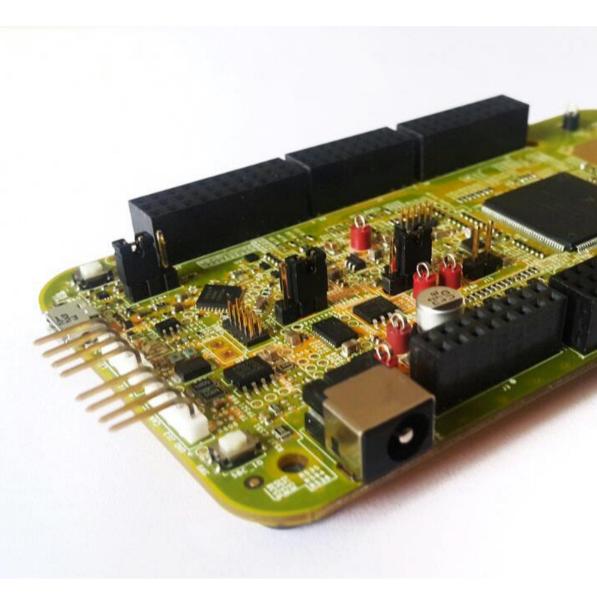
Jumper	Configuration	Description
J104	1-2	Reset signal to OpenSDA, use to enter into OpenSDA Bootloader mode
	2-3 (Default)	Reset signal direct to the MCU, use to reset S32K146.
J107	1-2	S32K146 powered by 12V power source.
	2-3 (Default)	S32K146 powered by USB micro connector.
J10	1-2 2-3 (Default)	VDD voltage is connected to 3.3 V VDD voltage is connected to 5 V





HMI mapping

Component	S32K146
Red LED	PTD15 (FTM0 CH0)
Blue LED	PTD0 (FTM0 CH2)
Green LED	PTD16 (FTM0 CH1)
Potentiometer	PTC14 (ADC0_SE12)
SW2	PTC12
SW3	PTC13
OpenSDA UART TX	PTC7 (LPUART1_TX)
OpenSDA UART RX	PTC6 (LPUART1_RX)
CAN TX	PTE5 (CAN0_TX)
CAN RX	PTE4 (CAN0_RX)
LIN TX	PTD7 (LPUART2_TX)
LIN RX	PTD6 (LPUART2_RX)
SBC_SCK	PTB14 (LPSPI1_SCK)
SBC_MISO	PTB15 (LPSPI1_SIN)
SBC_MOSI	PTB16 (LPSPI1_SOUT)
SBC_CS	PTB17 (LPSPI1_PCS3)



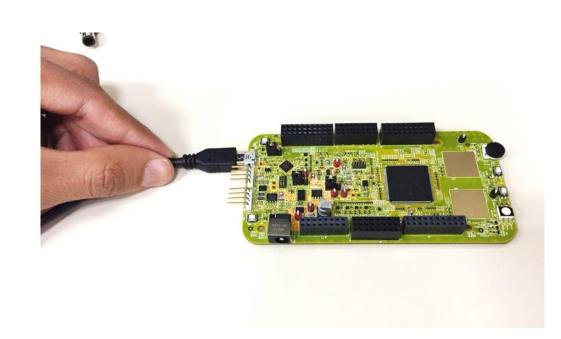


S32K146 EVB JUNPSTART



Step 1: Power up the Board – EVB Power Supplies

- The S32K146-EVB evaluation board powers from a USB or external 12V power supply. By default USB power is enabled with J107 (check slide 5)
- Connect the USB cable to a PC using supplied USB cable .
- Connect other end of USB cable (microUSB) to mini-B port on S32K146 at J7
- Allow the PC to automatically configure the USB drivers if needed
- Debug is done using OpenSDA through J7

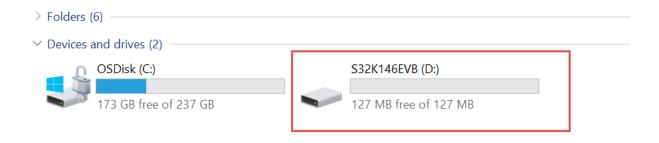




7 EXTERNAL USE

Step 1: Power up the Board – Is it powered on correctly?

- When powered through USB, LEDs D2 and D3 should light green
- Once the board is recognized, it should appear as a mass storage device in your PC with the name S32K146EVB.







Step 1: Power up the Board – Is it powered on correctly?

 Board is preloaded with a software, in which the red, blue and green LEDs will toggle at different rates.



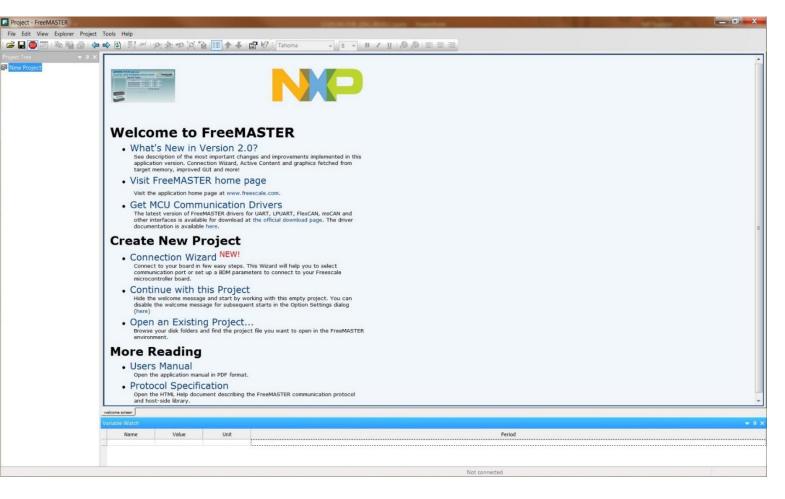


S32K146 EVB JUMPSTART BASED ON THE FREEMASTER



Install the FreeMASTER tool

- Download and install the FreeMASTER PC application <u>www.nxp.com/FreeMASTER</u>.
- Open the FreeMASTER application on your PC. You should see Welcome page:

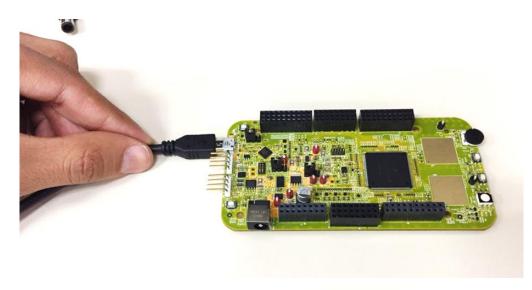




11

Power up the EVB board

- Powers the S32K146EVB evaluation board from a USB. By default, the USB power is enabled by J107 jumper (2-3 closed).
- Connect the USB cable to a PC and connect micro USB connector of the USB cable to micro-B port J7 on the S32K146EVB.
- Allow the PC to automatically configure the USB drivers if needed.
- When EVB is powered from USB, LEDs D2 and D3 should light green.
- The EVB board is preloaded with a software toggling the RGB LED colours periodically between RED-GREEN-BLUE.







12 EXTERNAL USE

Setup serial connection in the FreeMASTER tool

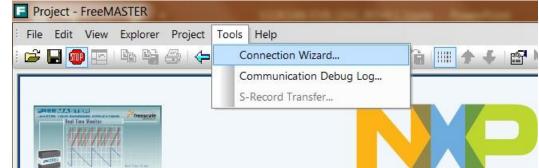
Setup communication port to "opensda" and speed to 115200 b/s:

 Setup communication manualy: "Project > Options > Comm"

-	110		
RS232:	Port	opensda 🗾 OpenSDA - CDC Serial Port (htt	p://www.pem
	Speed	: 115200 - Timeouts	
	opeed		
C Plug-in	Module:		*

OR

 Setup communication automatically: "Tools > Connection Wizard"







The JumpStart project will be automaticaly downloaded from <u>www.nxp.com</u>

Once the FreeMASTER application detects the web address stored as an TSA active content in the flash memory of the S32K146 MCU, the download of the FreeMASTER project from <u>www.nxp.com</u> will be initiated.



Explore the Board NEW!

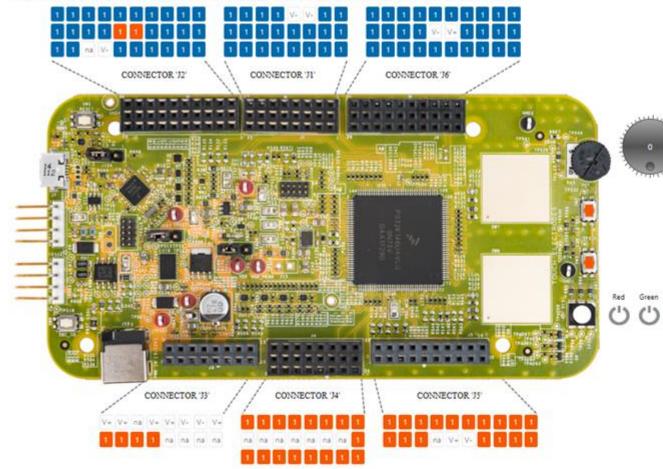
The attached board contains a graphical content you may want to see first.

(opens in 0 sec loading project, please wait...).



The FreeMASTER JumpStart project is loaded

Pins of the J2, J1 and J6 connectors are configured as outputs (except pins 14 and 17 on the J2 connector).

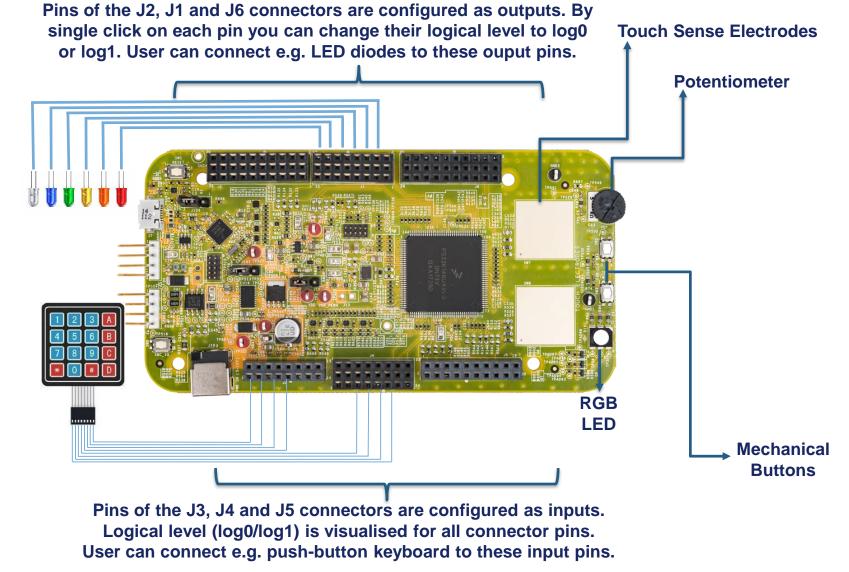


Pins of the J3, J4 and J5 connectors are configured as inputs.

S32K14x Web Links:

- >> S32K Overview
- >> S32K146 Evaluation Board:
 - > Getting Started
 - > S32K146EVB Quick Start Guide
 - > S32K146EVB-Q144 Schematic
 - > S32K1xx Fact Sheet
 - > S32K1xx Data Sheet
 - > S32K14x Reference Manual
 - > S32K1xx Product Brief
- >> SW Tools:
 - > FreeMASTER Run-Time Debugging Tool
 - > S32 Design Studio IDE
 - >> S32K146 EVB JumpStart Sources:
 - > S32K146 EVB JumpStart PC Host Project
 - > S32K146 EVB JumpStart Firmware

The FreeMASTER JumpStart project description

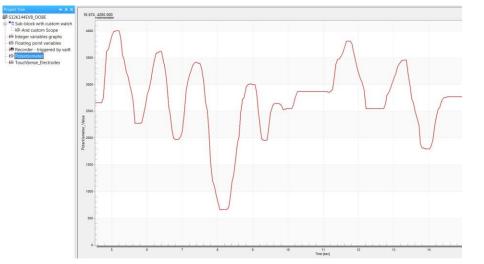




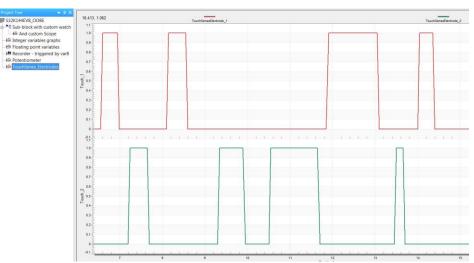
The FreeMASTER JumpStart oscilloscope feature examples

View Explorer Project Tools Help Display main project panel "Project > View > Project Tree". Project Tree Ctrl++ Variable Watch Ctrl+E Application Commands Ctrl+š Variable Stimulus Ctrl+C Show Control Page as a Bar Ctrl+R All App. Commands Visible 1 View Toolbar View Variable Watch Toolbar View Status Line





Analog values from potentiometer.



Application Layout

Responses from touch sense electrodes.





INTRODUCTION TO OPENSDA



Introduction to OpenSDA: 1 of 2

OpenSDA is an open-standard serial and debug adapter. It bridges serial and debug communications between a USB host and an embedded target processor. OpenSDA software includes a flash-resident USB mass-storage device (MSD) bootloader and a collection of OpenSDA Applications. S32K146 EVB comes with the MSD Flash Programmer OpenSDA Application preinstalled. Follow these instructions to run the OpenSDA Bootloader and update or change the installed OpenSDA Application.

Enter OpenSDA Bootloader Mode

- 1. Unplug the USB cable if attached
- 2. Set J104 on position 1-2.
- 3. Press and hold the Reset button (SW5)
- 4. Plug in a USB cable (not included) between a USB host and the OpenSDA USB connector (labeled "SDA")
- 5. Release the Reset button

A removable drive should now be visible in the host file system with a volume label of BOOTLOADER. You are now in OpenSDA Bootloader mode.

IMPORTANT NOTE: Follow the "Load an OpenSDA Application" instructions to update the MSD Flash Programmer on your S32K146 EVB to the latest version.

Load an OpenSDA Application

- 1. While in OpenSDA Bootloader mode, double-click **SDA_INFO.HTML** in the **BOOTLOADER** drive. A web browser will open the OpenSDA homepage containing the name and version of the installed Application. This information can also be read as text directly from **SDA_INFO.HTML**
- 2. Locate the **OpenSDA Applications**
- 3. Copy & paste or drag & drop the MSD Flash Programmer Application *to the BOOTLOADER* drive
- Unplug the USB cable and plug it in again. The new OpenSDA Application should now be running and a S32K146 EVB drive should be visible in the host file system

You are now running the latest version of the MSD Flash Programmer. Use this same procedure to load other OpenSDA Applications.



Introduction to OpenSDA: 2 of 2

The MSD Flash Programmer is a composite USB application that provides a virtual serial port and an easy and convenient way to program applications into the S32K146 MCU. It emulates a FAT16 file system, appearing as a removable drive in the host file system with a volume label of S32K146EVB. Raw binary and Motorola S-record files that are copied to the drive are programmed directly into the flash of the S32K146 and executed automatically. The virtual serial port enumerates as a standard serial port device that can be opened with standard serial terminal applications.

Using the MSD Flash Programmer

- 1. Locate the .srec file of your project , file is under the Debug folder of the S32DS project.
- 2. Copy & paste or drag & drop one of the .srec files to the S32K146EVB drive

The new application should now be running on the S32K146 EVB. Starting with v1.03 of the MSD Flash Programmer, you can program repeatedly without the need to unplug and reattach the USB cable before reprogramming.

Drag one of the .srec code for the S32K146 EVB board over USB to reprogram the preloaded code example to another example.

NOTE: Flash programming with the MSD Flash Programmer is currently only supported on Windows operating systems. However, the virtual serial port has been successfully tested on Windows, Linux and Mac operating systems.

Using the Virtual Serial Port

- Determine the symbolic name assigned to the EVB-S32K146 virtual serial port. In Windows open Device Manager and look for the COM port named "PEMicro/Freescale – CDC Serial Port".
- 2. Open the serial terminal emulation program of your choice. Examples for Windows include <u>Tera Term</u>, <u>PuTTY</u>, and <u>HyperTerminal</u>
- 3. Press and release the Reset button (SW0) at anytime to restart the example application. Resetting the embedded application will not affect the connection of the virtual serial port to the terminal program.
- 4. It is possible to debug and communicate with the serial port at the same time, no need to stop the debug.

NOTE: Refer to the OpenSDA User's Guide for a description of a known Windows issue when disconnecting a virtual serial port while the COM port is in use.



INSTALLING S32DS





Download S32DS from ARM based MCUs from:

S32DS for ARM



22 EXTERNAL USE

CREATE A NEW PROJECT IN S32 DESIGN STUDIO



Create New Project: First Time – Select a Workspace

- Start program: Click on "S32 Design Studio for ARM" icon
- Select workspace:
 - Choose default (see below example) or specify new one
 - Suggestion: Uncheck the box "Use this as the default and do not ask again"
 - Click OK

Select a dire	tory as workspace		
S32 Design St	udio for ARM uses the workspace directory to st	ore its preferences and development artifa	cts.
Workspace:	C:\Users\B52321\workspace	- Brow	/se
🔲 Use this as	the default and do not ask again		
Recent Wo	rkspaces		
		ОК	Cancel





Create New Project: Top Menu Selection

• File – New – S32DS Application Project

	Search Project	Run Processor Expert Window	Help							
New		Makefile Project with Existin		() v () v					Quick Access	
Open File		C++ Project	5							
Open Projects from File System		C Project						 🐉 Outline 🕮 🐵 Build Targ		- 0
Close	Cel.W	C/C++ Project						An outline is not available	s	
	Chill Chilth W	S32DS Application Project								
Close All	Ctrl+Shift+W	S32DS Library Project								
Save	Ctrl+S	S32DS Project from Examp	le							
Save As		Project								
Save All	Ctrl+Shift+S	Convert to a C/C++ Project	(Adds C/C++ Nature)							
Revert		Source Folder								
Move		Solder								
Rename	F2	Source File								
Refresh	F5	Header File								
Convert Line Delimiters To	3	 File from Template Class 								
Print		Class								
Switch Workspace		C Other	Ctrl+N							
Restart		G Outer	Cal+N							
Import										
Export										
Properties	Alt+Enter									
Exit										
EAR										
Project Creation S32DS Application Project S32DS Library Project Build/Debug Build (All) Clean (All) Debug Settings Project settings										
liscellaneous			🗈 Problems 🛛 🖉 Tasks 🖻 Conso	ole 🗏 Properties						~ - 0
Getting Started			0 items							
Quick access			Description	Resource	Path	Location	Туре			
								-		
									Ipdates Available	×
									Jpdates are available for your sof	
									-	



Create New Project: S32DS Project

- Project Name:
 - Example: FirstProject
- Project Type:
 - Select from inside executable or library folder
- Next

Project name: FirstProject				
Use default location				
Location: C:\Users\b50982\workspaceS32DS.	ARMv2_0\FirstProject			Brows
Processors :	ToolChain Selec	ction:		
type filter text	Core Kind	Name	Toolchain	
 Family KEA Family MAC57D5xx Family S32V Family S32K1xx S32K144 S32K142 S32K146 	M4	Cortex-M4F	Standard S32DS toolchain for ARM	
532K148	Description :			
	GCC toolchain	is selected		





Create New Project: S32DS Project

- Select Debugger Support and Library/SDKs Support if needed.
- Click Finish

New S32DS Application Project		
New S32DS Project for S32K146		
Select required cores and parameters for them.		
Project Name	FirstProject	
Core	Cortex-M4F	
Library	EWL	•
I/O Support	No I/O	•
FPU Support	Toolchain Default	-
Language	С	-
SDKs		
Debugger	PE Micro GDB server	•
?	< <u>B</u> ack <u>Finish</u>	Cancel





OpenSDA Configuration

- To Debug your project with OpenSDA, it is necessary to select the OpenSDA in the Debug Configuration.
- · Select your project, and click on debug configuration

Edit Source Refactor Navigate Search Project Bun Processor Expert Window			
□ 0 8 • % • B # C # × & \$ @ • 8 • 2 • 6 •			Quick Access
Project Explorer 💥 🙂 😁 🗖	(no launch history)	° 🗖	BE Outline 12 (1) Build Targets **
eCos	Debug As + se 1/ ist and the property polication[□ 1% R x ² = #
A Definition Debug	Debug Configurations		S32K146.h
> 🔊 Includes	Organize Favorites		e main(void) : int
B Project_Settings	organize revolution. In the company of the company		# COUNTER LIMIT
> 😆 include	9= int main(void)		
4 🖽 src	10 (
(c) main.c	11 #define COUNTER_LIMIT 100		
Ilesh_partitioning_s32k142	12 5		
flash partitioning s32k144	13int counter = 0; 1 14 1		
Ipspi dma s32k144	15 for(1;) {		
Duart, echo s32k146	16 counter++15		
MPU	17 1		
1 532/(342_EVB_CSEc	<pre>18 if(counter > COUNTER_LIMIT) { </pre>		
S32K142_EV8_Csc S32K142_EV8_FlashPartition	19 counter = 0;1		
S32K142_EVB_MPU	20 ··········}{21 ········}		
S32K142_EVB_NP0	22 1 Sector 10		
S32K144_EVB_CoreMark_Benchmark_S32D5	21 /* to avoid the warning message for GPG and IAR: statement is unreachable*/*		
S32K144_EVB_COREMARX_BENCHMARX_S32LIS	24 #if defined (ghs)"		
	25 #pragma ghs nowarning 111		
S32K144_EVB_EEEPROM_FlexRAM_not_loaded_at_nst	25 #endif		
S32K144_EVB_FlexCAN_GatewayNode	22 #if defined (_ICCARM_)		
S32K144_EVB_LED_D-Flash	28 #pragma diag_suppress-Pell1 29 #endif		
S32K144_EVB_SPI_SlaveSDK	30 · return 0;		
S32K144_FOTA_Bootloader	31 35		
S32K144_FOTA_CSEc	32		
S32K144_Wippers			
\$32K146_CMP_VLP5_trigger_mode			
1 \$32K346_CSEc			
\$32K146_EVB_CSEc_Test			
S32K146_EVB_EEEPROM			
S32K146 LED test			
32X146_LP5P1_ERR011089			
S32K146 Serial CAN Bridge SDK			
1 S32KJ48_CSEc			
S32K148_EVB_EEEPROM_QuickWrites			
i Status ski ski			
S32014x_EEE			
S32K14, EEEPROM_Test		15	
S32K14k_DD_Codes			
SZ2KI4k_SPLLVLD_Bit test	🕐 Problems: 🖉 Tasts 😨 Console 12 🗇 Properties: 😨 Debugger Console 🍃 Call Hierarchy 🕫 Programs: 🛷 Search		B. E. B D
	Processor Expert		
	Feb 2, 2018 11:57:31 AM Starting Processor Expert service		
	System directory = C:\nxp\S32D5 ARM_v2.0\eclipse\ProcessorExpert		
	Internal cache directory = C:\ProgramData\Processor Expert\PECache\2c21a405		
	Processor Expert license file - not used (no license file)		
	Feb 2, 2018 11:57:32 AM Successfully started Processor Expert service		
	4		
		Writable Smart Insert 2:1	

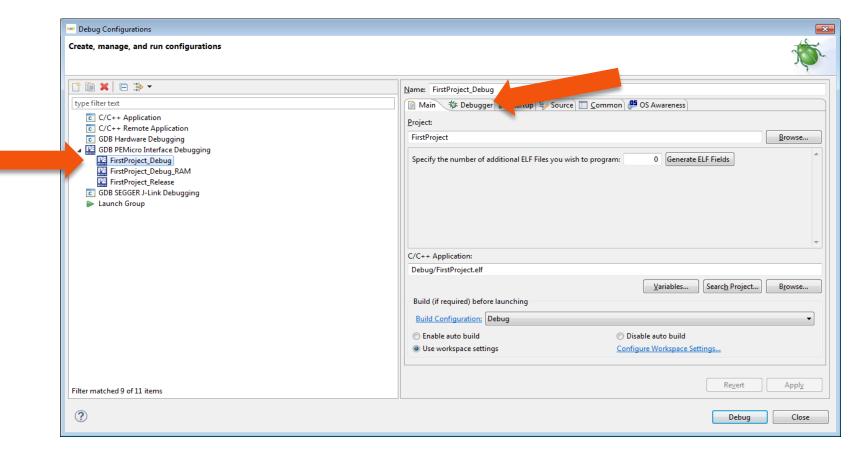


EXTERNAL USE

28

OpenSDA Configuration

- Select the Debug configuration under GDB PEMicro Interface Debugging
- Click on Debugger tab

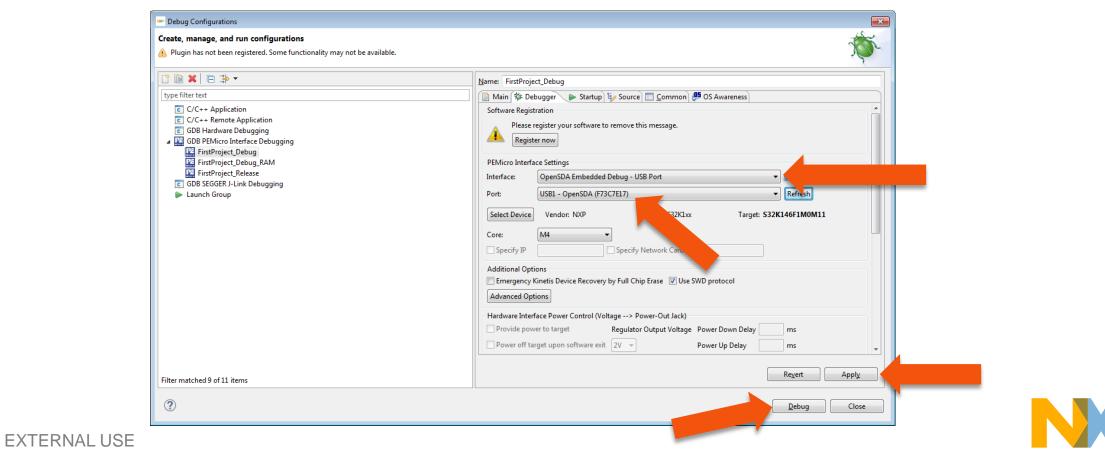






OpenSDA Configuration

- Select OpenSDA as the interface, if your board is plugged should appear in the Port field.
- Click Apply and debug to finish.



30

CREATE AN EXAMPLE FROM SDK



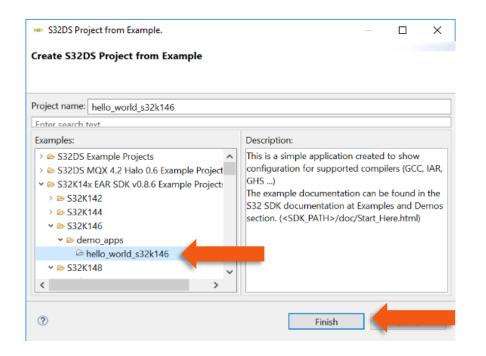
- The S32 Design Studio IDE already includes the Software Development Kit for quickly develop applications on S32K1xx devices.
- To create a project using an example go to File New S32DS Project from

New		Processor Expert Window H								
		Makefile Project with Existing	Code							Quick Access
Open File	-	C++ Project C Project							🔍 🗖 🐉 Outline 🕮 🐵 Build Ta	fargets
Open Projects from File System		CICLI Destant							An outline is not availal	ble.
Close	Ctrl+W	S32DS Application Project								
Close All	Ctrl+Shift+W	S32DS Library Project								
Save Save		⁷ S32DS Project from Example								
Save As		9 Project								
Save All		Convert to a C/C++ Project (A	dds C/C++ Nature)							
Revert		Source Folder Folder								
Move		Source File								
 Rename Refresh 		Header File								
Convert Line Delimiters To		File from Template								
		Class								
Print		8 Example								
Switch Workspace	> =	9 Other	Ctrl+N							
Restart										
🔤 Import										
La Export										
Properties	Alt+Enter									
Exit										
 Project Creation 										
L ^o S32DS Application Project S32DS Library Project Build/Debug Sauid (All) Clean (All) Debug Settings										
# S32DS Library Project • Build/Debug & Build (All) & Debug • Settings & Project settings										
# S32DS Library Project • Build/Debug \$ build (All) ¢ Clean (All) ¢ Debug • Settings % Project settings % build settings										
Image: S2DS Library Project Build/Debug Studie (All) Clean (All) Debug Settings % huld settings % buld settings % bulg settings			은 Problems 회 은 Tasks @ Cor	sole 🗏 Properties						
S2DS Library Project Suid/Debug Gen (Al) Clean (Al) Clean (Al) Stitug Stitug		0) items	sole □ Properties				-		
Image: S2DS Library Project Build/Debug Studie (All) Clean (All) Debug Settings % huld settings % buld settings % bulg settings		0		sole □ Properties	Resource	Path	Location	Туре		
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole III Properties	Resource	Path	Location	Туре		
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole □ Properties	Resource	Path	Location	Туре		
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole □ Properties	Resource	Path	Location	Туре		x
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole 🛛 Properties	Resource	Path	Location	Туре		
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole 🗆 Properties	Resource	Path	Location	Type		
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole III Properties	Resource	Path	Location	Туре		×
S2DS Library Project Build/Debug Studi (Ad) Clean (Ad) Clean (Ad) Oblog Settings Build settings Build settings Build settings Miscellaneous Getting Started		0) items	sole 🛛 Properties	Resource	Path	Location	Туре		



32

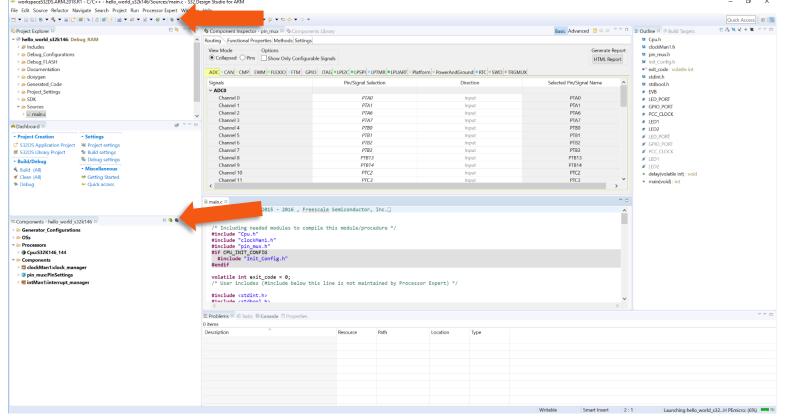
- Go to the S32K14x EAR SDK v0.8.6 Example Projects section and select the example that wants to be used.
- In this example the hello_world is selected







 A new project would be created in the workspace. Then click on generate code icon and then on debug, as indicated.



• If run correctly, the LED should start blinking red and green.



- The complete documentation of the SDK can be found in: C:\NXP\S32DS_ARM_v2018.R1\S32DS\S32SDK_S32K14x_EAR_0.8.
 6\doc\Start_here.html
- For more information about the use of the SDK go click on the following link for an SDK training (add hyperlink once if is online)



DEBUG BASICS



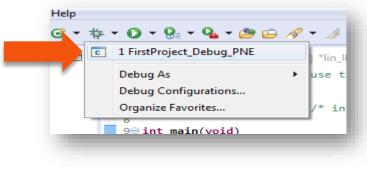
Debug Basics: Starting the Debugger

- Debug configuration is only required once. Subsequent starting of debugger does not require those steps.
- Three options to start debugger:
 - If the "Debug Configuration" has not been closed, click on "Debug" button on bottom right
 - Select Run Debug (or hit F11)



Note: This method currently selects the desktop target (*project.elf*) and gives an error. Do not use until this is changed.

<u>Recommended Method</u>: Click on pull down arrow for bug icon and select ..._debug.elf target

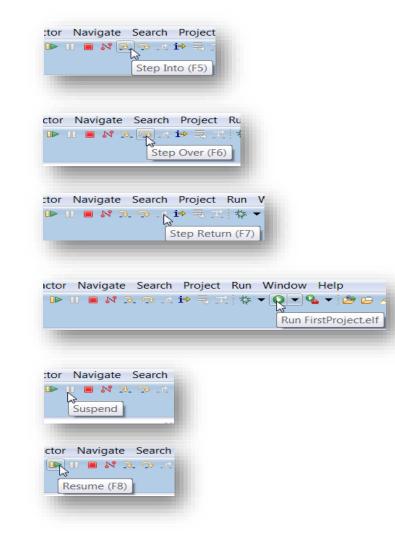




37

Debug Basics: Step, Run, Suspend, Resume

- Step Into (F5)
- Step Over (F6)
- Step Return (F7)
- Run
- Suspend
- Resume (F8)





Debug Basics: View & Alter Variables

- View variables in "Variables" tab.
- Click on a value to allow typing in a different value.

Name	Туре	Value
⇔= counter	int	8
		5





Debug Basics: View & Alter Registers

- View CPU registers in the "Registers" tab
- Click on a value to allow typing in a different value
- View peripheral registers in the EmbSys Registers tab

(x)= Variables 💁 Breakpoir	its its registers is	- Wiodales
Name		Value
General Registers		
1919 rO		3
1919 r1		5
1111 r2		536866944
1919 r3		8
1919 -4		0

		1 Sec.		I washing	to read values	
Register	Hex	Bin	Reset	Access	Address	Description
CSE_PRAM						CSE_PRAM
AIPS						AIPS-Lite Bridge
MSCM						MSCM
a 🗁 DMA						Enhanced Direct Memory Access
A nn DMA						Enhanced Direct Memory Access
⊳ lit CR	0x000000x0	000000000000000000000000000000000000000	0x0000000x0	RW	0x40008000	Control Register
> 1999 ES			0x0000000x0	RO	0x40008004	Error Status Register
▷ iiii ERQ			0x0000000x0	RW	0x4000800C	Enable Request Register
> 1000 EEI			0x0000000x0	RW	0x40008014	Enable Error Interrupt Register
▷ 1989 CEEI	- write only -		0x00	WO	0x40008018	Clear Enable Error Interrupt Regis
b 1888 SEEI	- write only -		0x00	WO	0x40008019	Set Enable Error Interrupt Registe
> 1818 CERQ	- write only -		0x00	WO	0x4000801A	Clear Enable Request Register
▷ bill SERQ	- write only -		0x00	WO	0x4000801B	Set Enable Request Register
▶ 1818 CDNE	- write only -		0x00	WO	0x4000801C	Clear DONE Status Bit Register



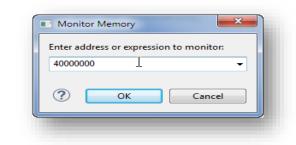
40 EXTERNAL USE

Debug Basics: View & Alter Memory

Add Memory Monitor



Select Base Address
 to Start at : 40000000



View Memory

Monitors	🕂 🗙 🔆	40000000 : 0x2	625A00 <hex></hex>	🖾 🛛 🕂 New	/ Renderings	
40000000		Address	0 - 3	4 - 7	8 - B	C - F
	02625A00	00000000	00000000	00000000	00000000	
	02625A10	00000000	00000000	00000000	00000000	
	02625A20	00000000	00000000	00000000	00000000	
	02625A30	00000000	00000000	00000000	00000000	
	02625A40	00000000	00000000	00000000	00000000	
	02625A50	00000000	00000000	00000000	00000000	
	02625A60	00000000	00000000	00000000	00000000	

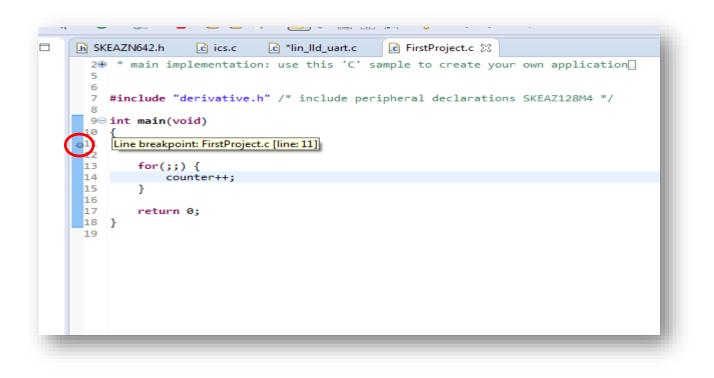




Debug Basics: Breakpoints

Add Breakpoint: Point and Click

light blue dot represents debugger breakpoint





42 EXTERNAL USE

Debug Basics: Reset & Terminate Debug Session

- Reset program counter
- Terminate Ctl+F2()





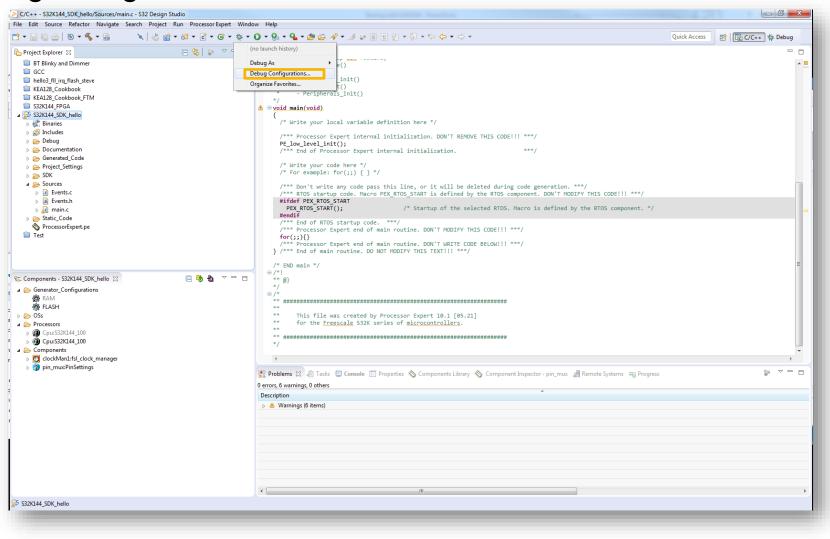


CREATE A P&E DEBUG CONFIGURATION (OPTIONAL)



New P&E debug configuration

Click in debug configurations







New P&E debug configuration

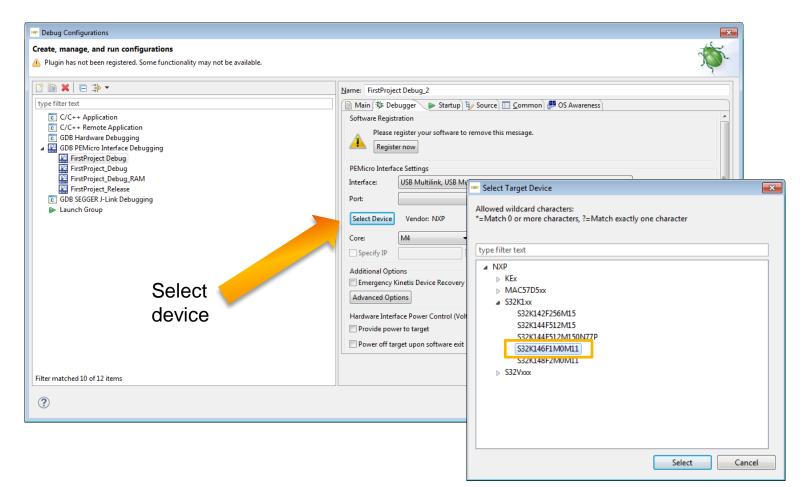
	Debug Configurations				
	Create, manage, and run configurations				
		Name: S32K144_SDK_h ug			
	type filter text	📄 Main 🕸 Debugger 🕨 Startup 🧤 Source 🔲 C	Common		
	GDB Hardware Debugging	Project			
	GDB PEMicro Interface Debugging S32K144_SDK_hello Debug	532K144_SDK_hello			
	GDB SEGGER J-Link Debugging	C/C++ Application:			
	Launch Group	Debug\S32K144_SDK_hello.elf			
			Variables Search Project		
Click to create a new		Build (if required) before launching			
&E launch		Build configuration: Select Automatically			
		 Enable auto build Use workspace settings 	 Disable auto build Configure Workspace Settings 		
		© ose workspace searings	compare workspace seeingsn		
	Filter matched 5 of 15 items		Apply Debug		



46 EXTERNAL USE

New P&E debug configuration

Select S32K146 device



- Click Apply and debug your application
- 47 EXTERNAL USE



USEFUL LINKS



Useful Links

- <u>Cookbook application note</u>. This application note contains a bunch of simple examples of how to use different peripherals.
- <u>S32K1xx community</u>. Visit this site for request support on the S32K1xx products, you can also look for threads that may contain the answer that you are looking for.





SECURE CONNECTIONS FOR A SMARTER WORLD

