How to use the AEK-MCU-SPC5LNK programmer and debugger for SPC5 automotive microcontrollers

Introduction

AEK-MCU-SPC5LNK is a debugger/programmer dongle for SPC58 (and SPC56 coming within year 2024) automotive microcontrollers.

It consists of a passive USB-to-JTAG debugger and programmer, which provides a cost-effective, small-size, and fast-prototyping solution for any vehicle application.

This dongle ensures short development time through easy-to-use hardware and simple software fully integrated into SPC5-Studio and AutoDevKit Studio IDEs.

The JTAG interface ensures an easy plug to any SPC5 family automotive microcontroller evaluation board (compliant with the IEEE 1149.1 JTAG protocol).

AEK-MCU-SPC5LNK provides a virtual COM port interface that enables the host PC to communicate with the microcontroller through UART.

Two different types of dongles are available:

- AEK-MCU-SPC5LNK, to be used with OpenOCD
- AEK-MCU-SPC5LNKU, to be used with UDE PLS

If you are using AEK-MCU-SPC5LNK, there is no need to install additional software. In case of issues, uninstall and reinstall the drivers as described in Section 3.2: Using the dongle.

If you are using AEK-MCU-SPC5LNKU and you want to continue using UDE PLS as debugger, there is no need to install additional software. Otherwise, to use OpenOCD, refer to Section 3.2: Using the dongle.

The OpenOCD drivers are freely downloadable at the following link: STSW-AEK-SPC5LNK.

Warning: *The AEK-MCU-SPC5LNK evaluation dongle has not to be used in a vehicle as it is designed for R&D laboratory use only.*

Figure 1. AEK-MCU-SPC5LNK top view

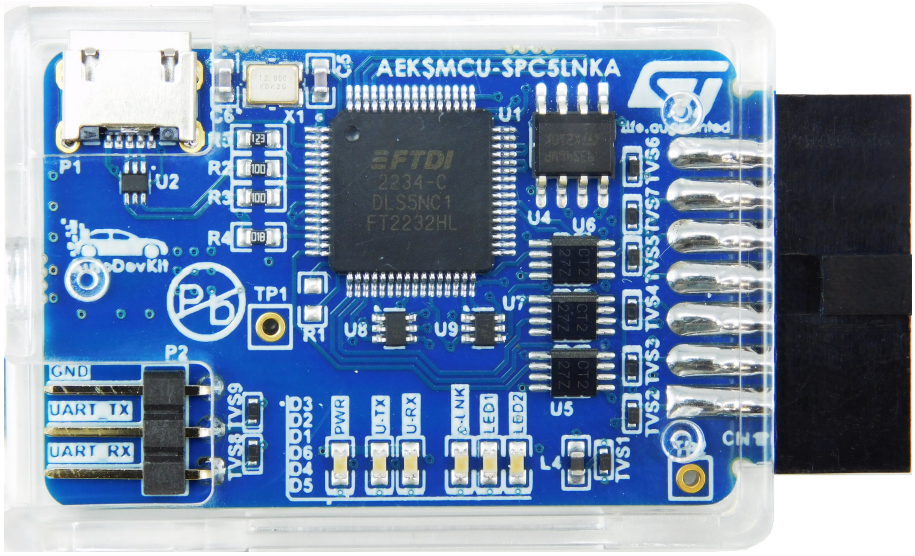
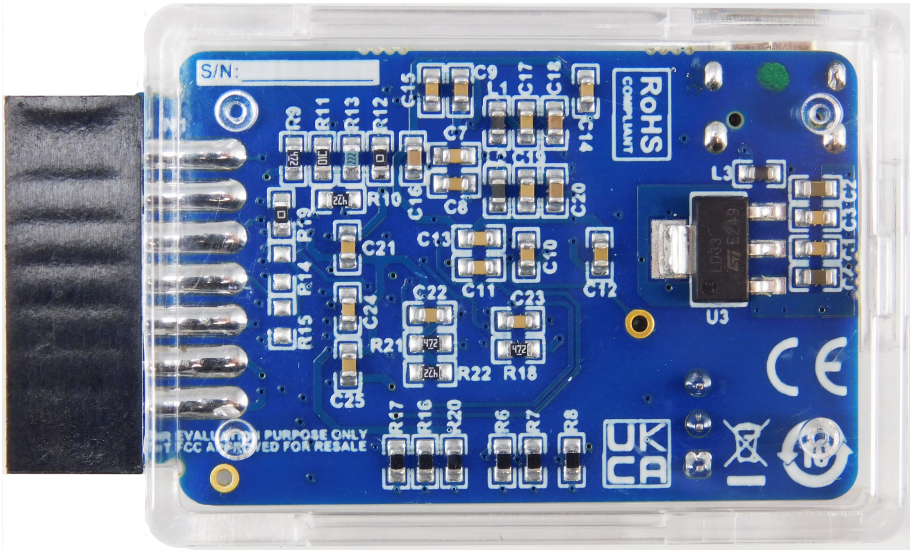


Figure 2. **AEK-MCU-SPC5LNK bottom view**



1 Hardware description

1.1 Hardware features

- USB/JTAG debugger dongle for SPC58 automotive MCUs
- USB connector for supply and data communication
- Download your application firmware in SPC5x device FLASH or RAM memory
- Run and debug your application firmware adding breakpoints and watching local and global variables
- Compliant with the IEEE 1149.1 JTAG protocol
- Integrates serial port connection via USB interface (virtual COM)
- Provides NVM programming (erase/program/verify)
- Connectors:
  - 14-pin header connector for JTAG interface
  - 3-pin header connector for UART interface
- Status LEDs to indicate the target IO voltage, connection state, and running state
- Operating temperature range: from 0°C to 50°C
- Included in the AutoDevKit ecosystem

2 Hardware configuration

The AEK-MCU-SPC5LNK is a USB adapter based on an FTDI FT2232HL interface chip. The user EEPROM is programmed with a unique serial number.

The debugger provides the interface between the PC USB port and the target device in-system debug/ programming circuitry.

The USB connector provides the power supply and data communication line.

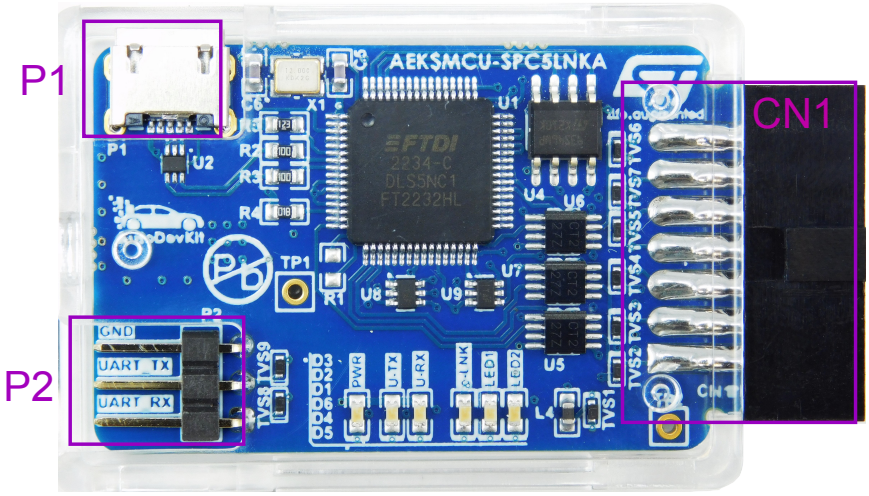
2.1 Connectors

The following table describes the connectors on the AEK-MCU-SPC5LNK board.

Table 1. Connectors

Connector	Description
P1	Micro-USB female connector
CN1	14-pin header connector for JTAG interface
P2	3-pin header connector for UART interface

Figure 3. AEK-MCU-SPC5LNK connectors



2.2 CN1

The following table describes the CN1 pinout.

Table 2. CN1 pinout

Pin	Description
1	TDI
2	GND
3	TDO
4	GND
5	TCK
6	GND
7	USERID 0
8	USERID 1
9	SRST#
10	TMS
11	VIN
12	N.C.
13	N.C.
14	TRST#

2.3 P2

The following table describes the CN2 pinout.

Table 3. CN2 pinout

Pin	Description
1	UART_RX
2	UART_TX
3	GND

2.4 LEDs

The following table describes the LEDs present in the AEK-MCU-SPC5LNK board.

Table 4. LEDs

Connector	Description
D1	U-RX (UART-RX)
D2	U-TX (UART-TX)
D3	PWR (Power on LED)
D4	LED1
D5	LED2
D6	VCIO (Target IO voltage)

3 How to use the dongle

3.1 Prerequisites

- Install AutoDevKit, which already embeds OpenOCD and sample projects for SPC58x evaluation boards, as well as already compiled elf files
- Basic knowledge of Eclipse development environments
- Administrator rights to install/uninstall SPC5-UDESTK USB drivers

3.2 Using the dongle

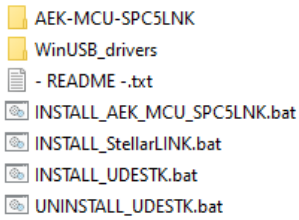
Uninstall previous PLS SPC5-UDESTK drivers by running “UninstallPlsUsbJtagDriver” from “C:\Program Files\pls\UDE Starterkit 2021\Driver\JtagUsbDriver” folder.

PLS UDESTK software will not work anymore. To restore it, uninstall WinUSBdrivers and reinstall PLS UDESTK USB drivers from the folder above.

WinUSBdrivers can be removed by running UNINSTALL_UDESTK (with Administrator rights) from the following folder: C:\{installation folder}\openocd\drivers.

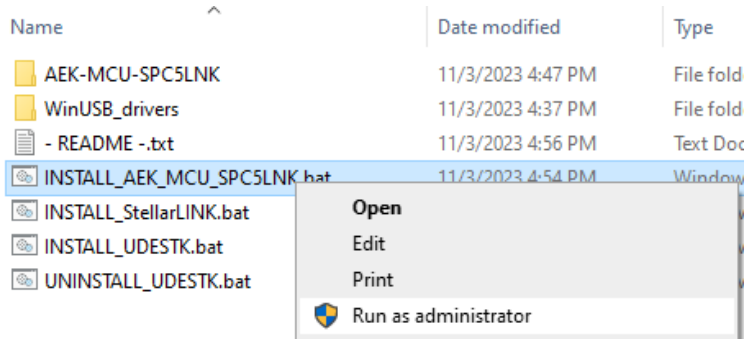
To install OpenOCD drivers, go to folder C:\{installation folder}\openocd\drivers.

Figure 4. Installing OpenOCD drivers (1 of 2)



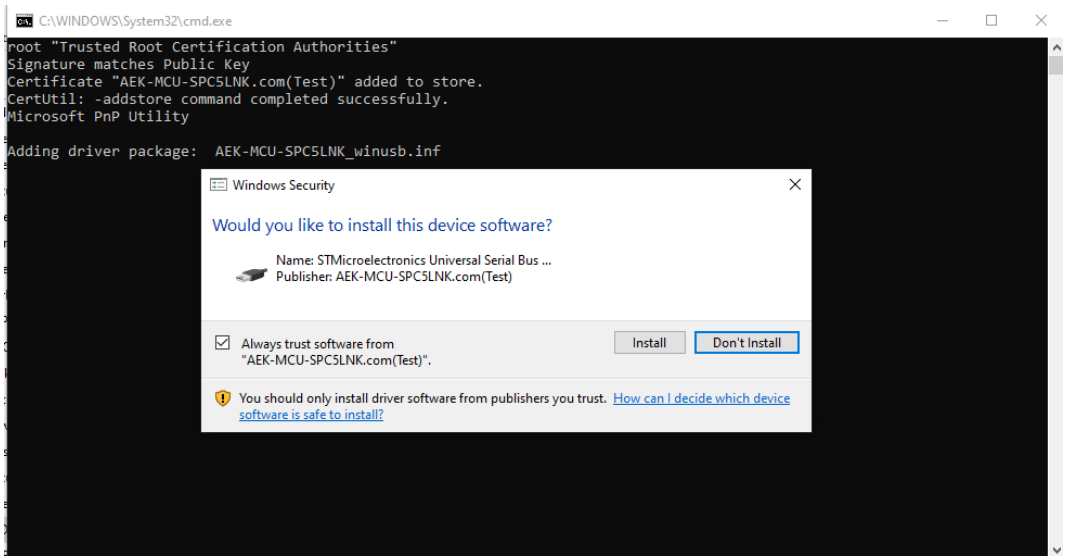
Right-click on INSTALL_AEK_MCU_SPC5LNK.bat and select “Run as administrator”.

Figure 5. Installing OpenOCD drivers (2 of 2)



A CMD window and a security message appear three times, as shown below.

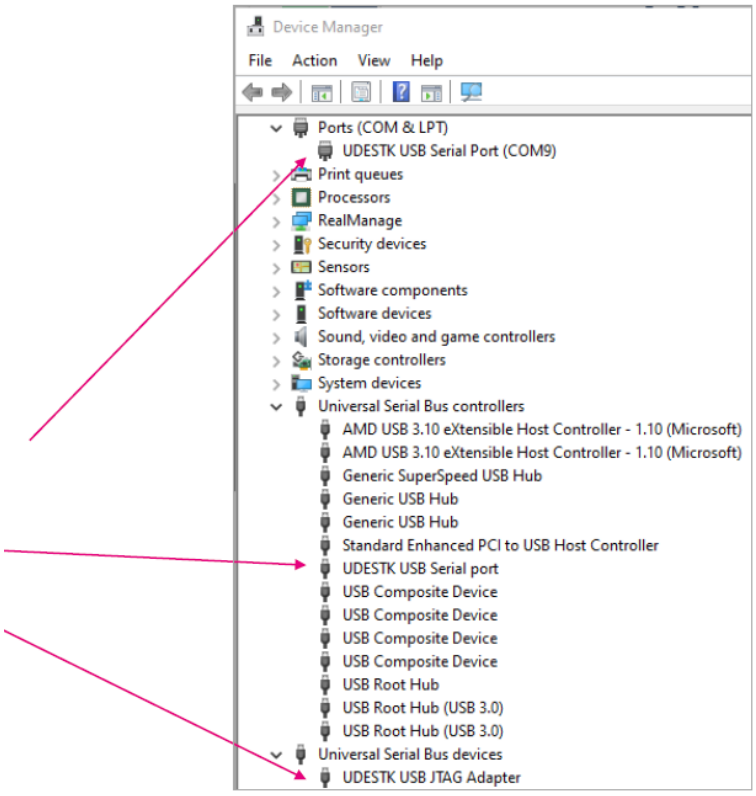
Figure 6. Security message



Uncheck “Always trust from “AEK-MCU-SPC5LNK.com(Test)””. Then, click on the “Install” button.

To check that the UDESTK driver has been installed correctly, connect the dongle to the PC via a micro-USB cable. Then, check the Device Manager. It should show: UDESTK USB Serial Port (COMx), UDESTK USB Serial port, and UDESTK USB JTAG Adapter.

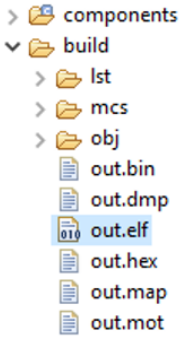
Figure 7. Device Manager when the UDESTK driver installation is successful



To use an existing debug configuration and modify it according to a pre-configuration, follow the procedure below.

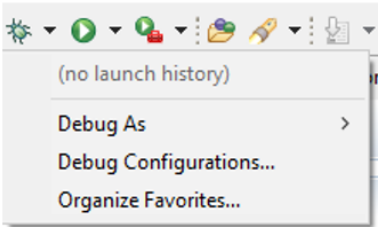
1. Generate and compile to get the out.elf binary.

Figure 8. Generating out.elf file



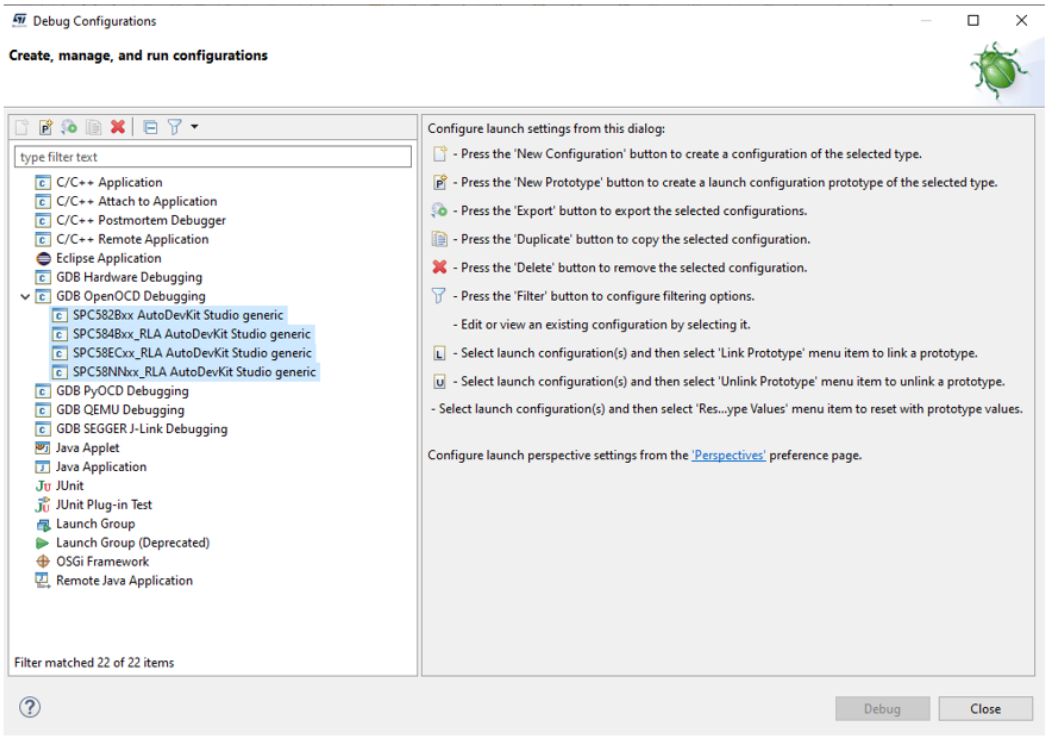
2. Switch to Debug perspective.
3. Select Debug Configurations.

Figure 9. Debug configurations



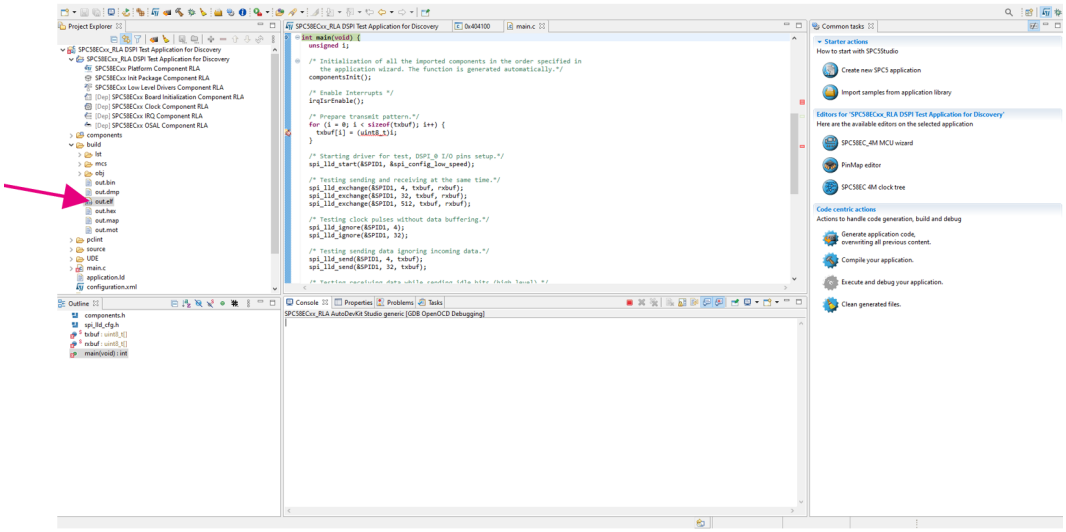
4. Select one of the four example pre-configurations available for the four platforms supported by AutoDevKit (SPC582B, SPC584B, SPC58EC, SPC58NN).

Figure 10. Selecting a platform



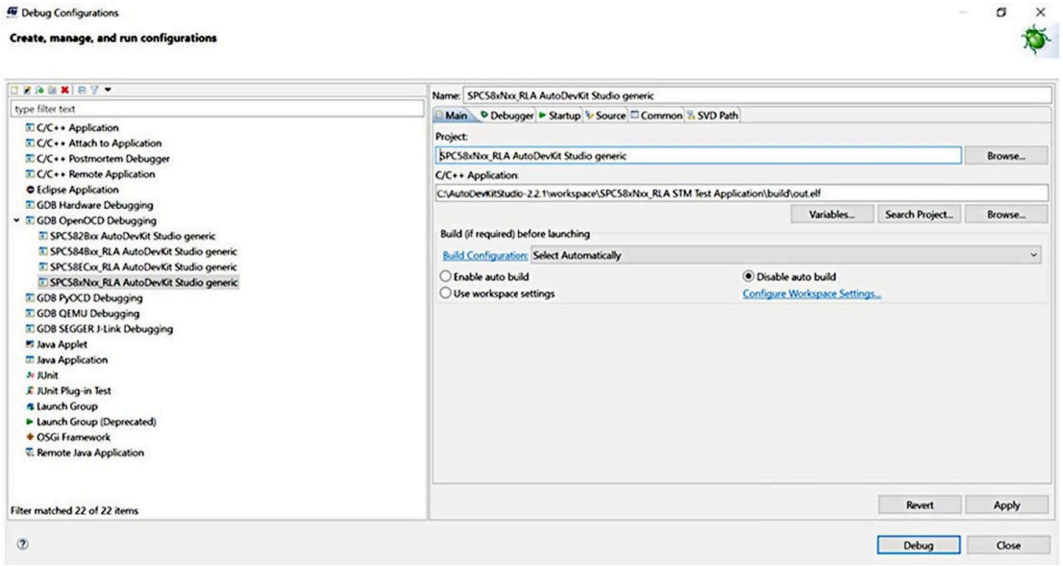
5. Specify the out.elf file, click on the "Browser" button, and select your application.

Figure 11. Specifying the out.elf file



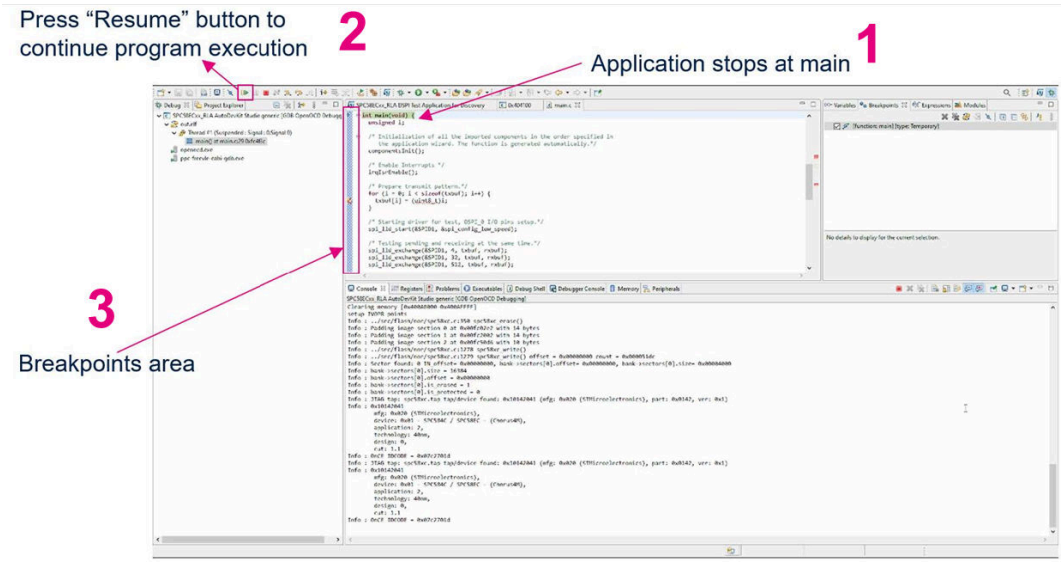
6. After selecting your configuration, insert the path of the elf file in the project field. Click on Debug.

Figure 12. Create, manage, and run configurations



7. The console opens up and shows your application code execution. The code execution stops at the main file. To continue executing the code, press resume. You can manually insert a breakpoint to stop code execution.

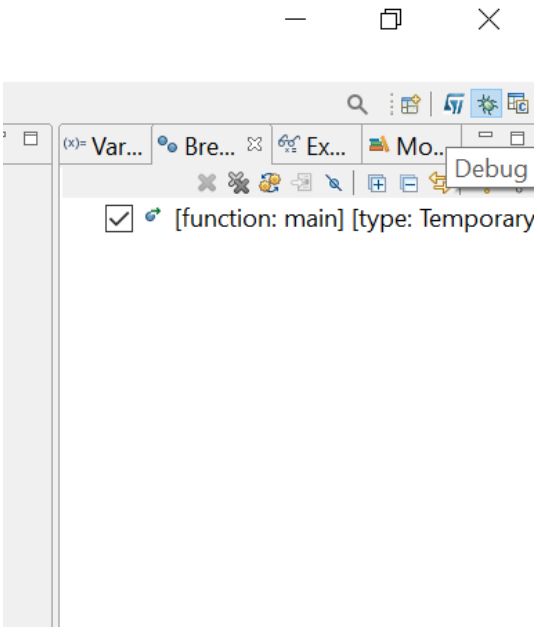
Figure 13. Application code execution



To create, from scratch, and debug a new configuration according to your own application, follow the procedure below:

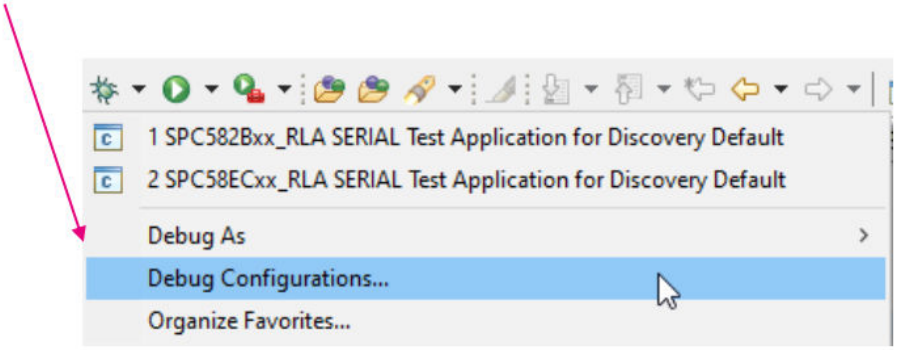
1. Switch to Debug perspective

Figure 14. Debug perspective



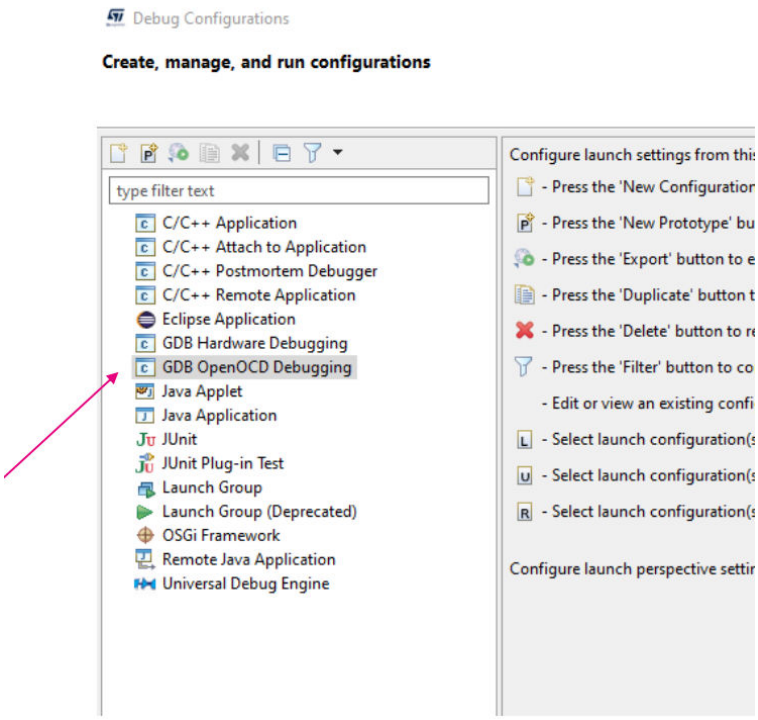
2. Select Debug Configurations

Figure 15. Debug configurations



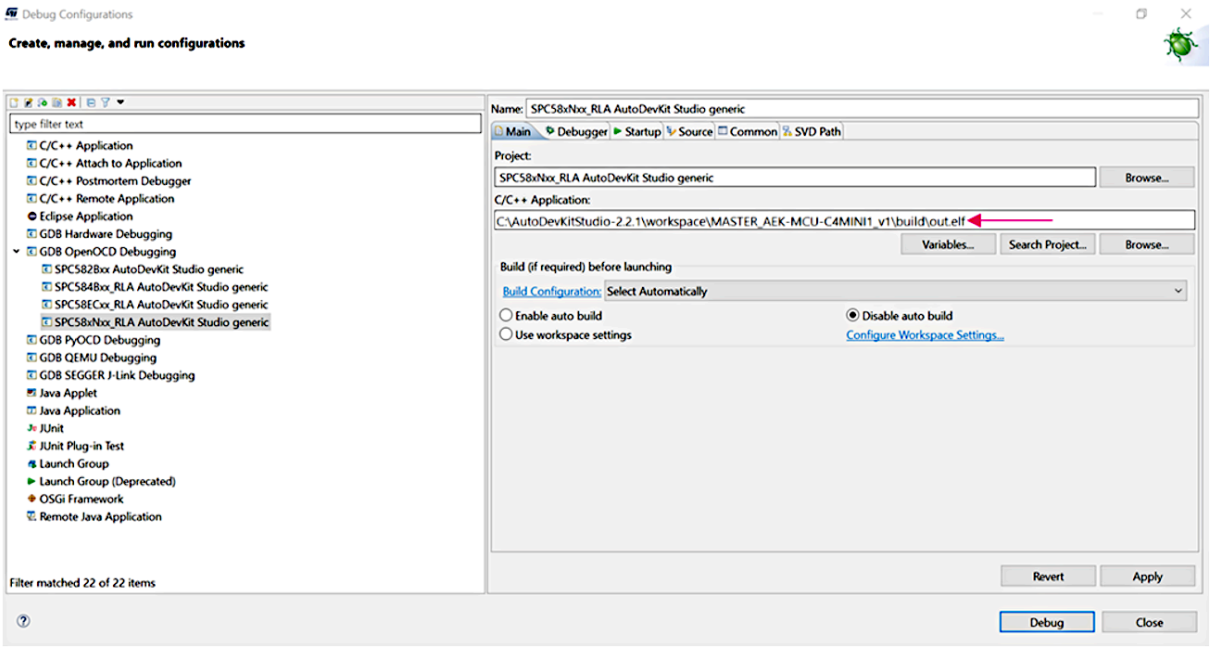
3. Double click on “GDB OpenOCD Debugging”

Figure 16. OpenOCD debugging



4. Configure and fill the "C/C++ Application” field with the elf file of your application.
   You can find the elf file in the build path directory of your project.
   The elf file will be created after generating and compiling your application

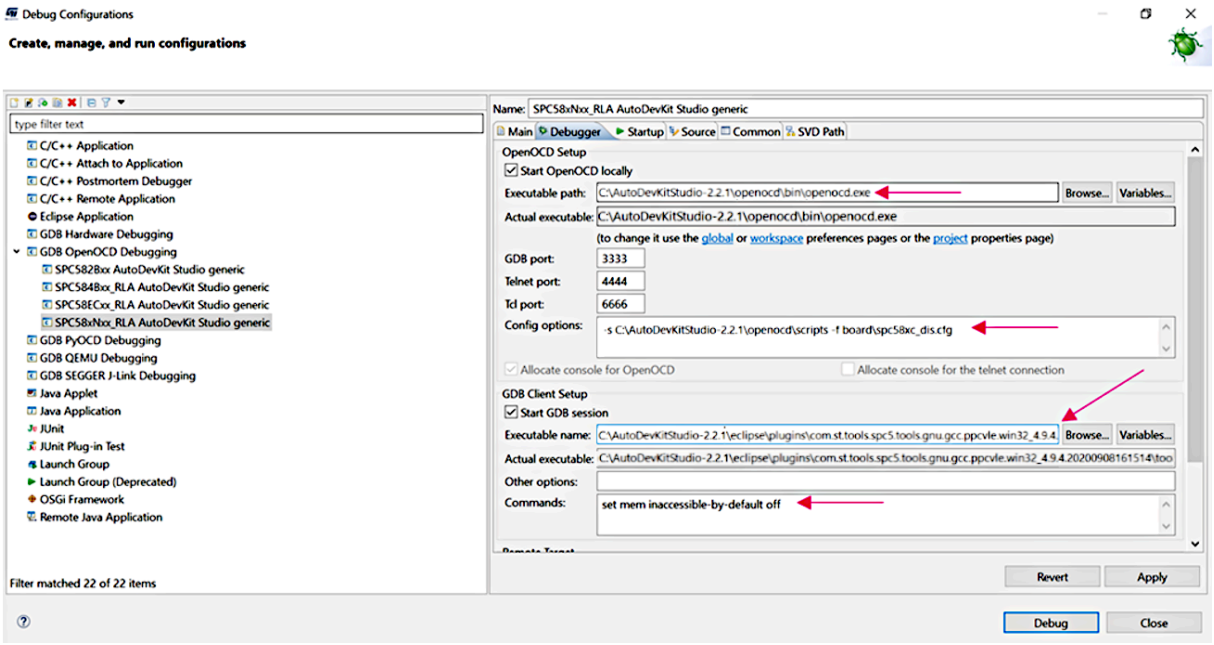
Figure 17. Generating and compiling the elf file



5. Fill the “Executable path” field with the path of the openocd executable available inside your AutoDevKitSstudio installation
6. Fill “Config option” section according to the board used in the configuration. To identify your configuration file go to C path:\AEKStudio-2.0\openocd\scripts\board. Here you can find the list of all configuration files. For example,if you use SPC58xNxx, put the " -s C:\ {installation folder} \openocd\scripts -f board\ spc58nn-disp.cfg”
7. Fill the "Executable name" field with your gdb client installed in AutoDevKit studio environment. For example, “C:\ {installation folder} \eclipse\plugins\com.st.tools.spc5.tools.gnu.gcc.ppcvle.win32_4.9.4.20200908161514\toolchain\bin\ppc-freevle-eabi-gdb.exe”

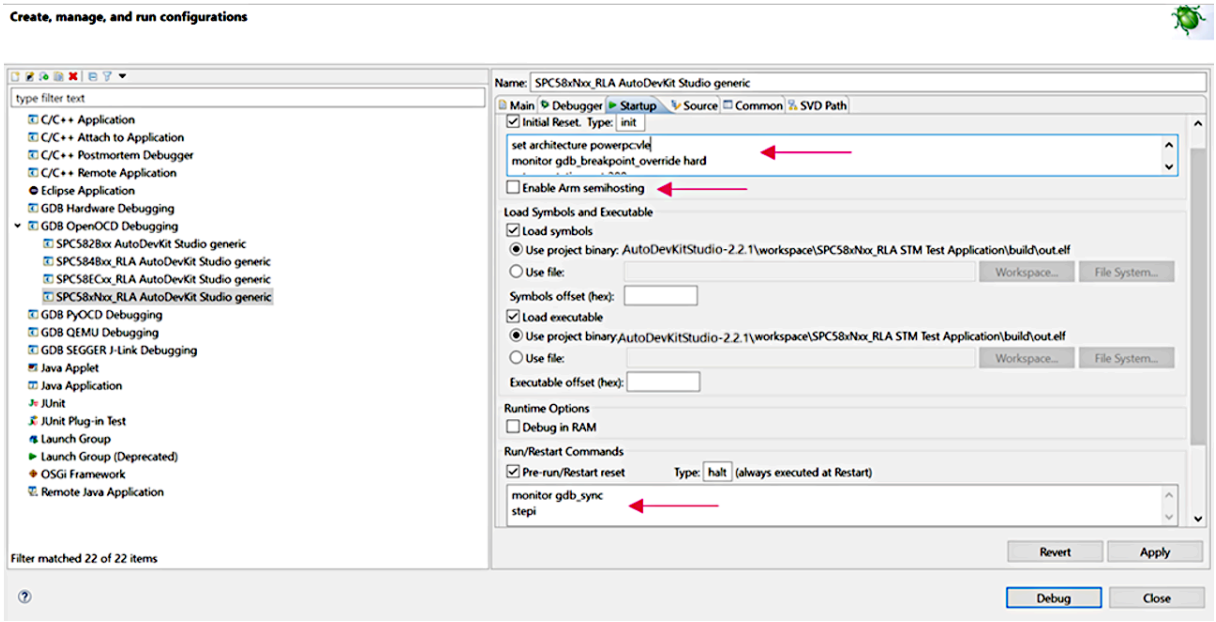
8. Fill "Commands" with “set mem inaccessible-by-default off”.

Figure 18. Setting configuration fields (1 of 2)



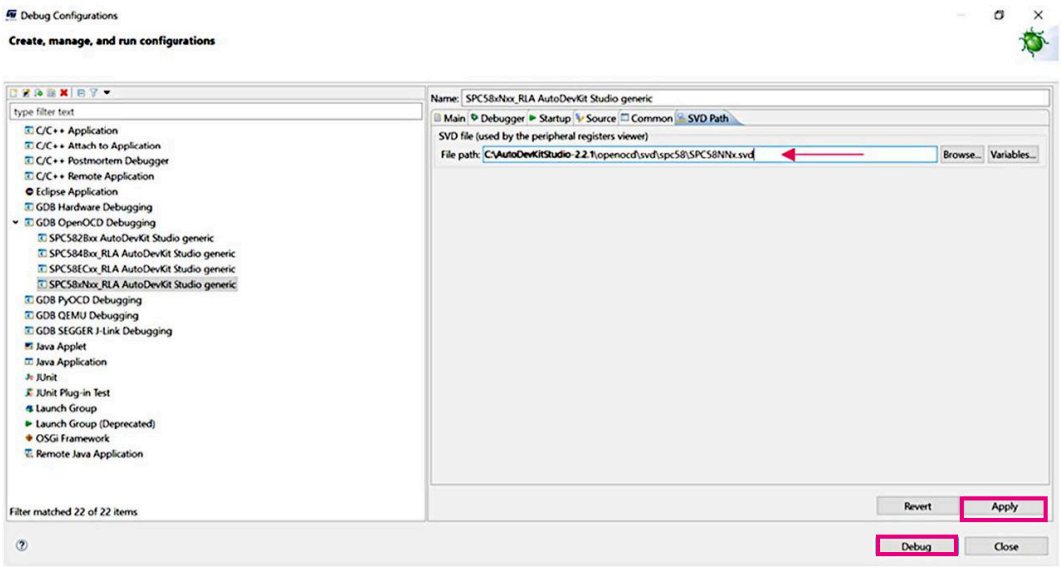
9. Fill Initialization “Commands” with:
   set architecture powerpc:vle
   monitor gdb_breakpoint_override hard
   set remotetimeout 200"
10. Uncheck “Enable Arm semihosting”
11. Fill “Run/Restart Commands” with:
    monitor gdb_sync
    stepi

Figure 19. Setting configuration fields (2 of 2)



12. Browse to the file path according to the board used in your application.
    To identify your file go to C:\AEKStudio-2.0\openocd\svd\spc58 . Here you can find the list of all files. For example: " C:\ {installation folder}\openocd\svd\spc58\SPC58NNx.svd "
13. Click on Apply and then Debug

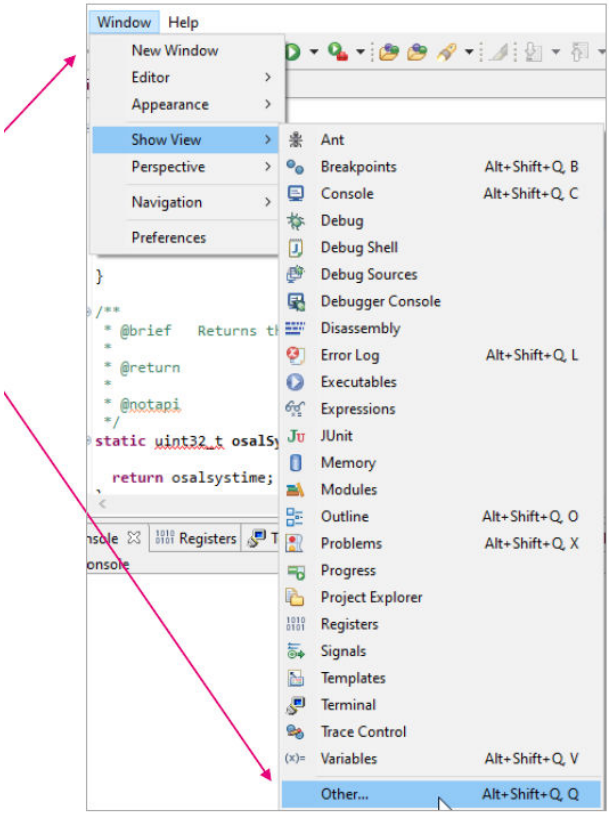
Figure 20. Debug



3.2.1 Peripherals view usage

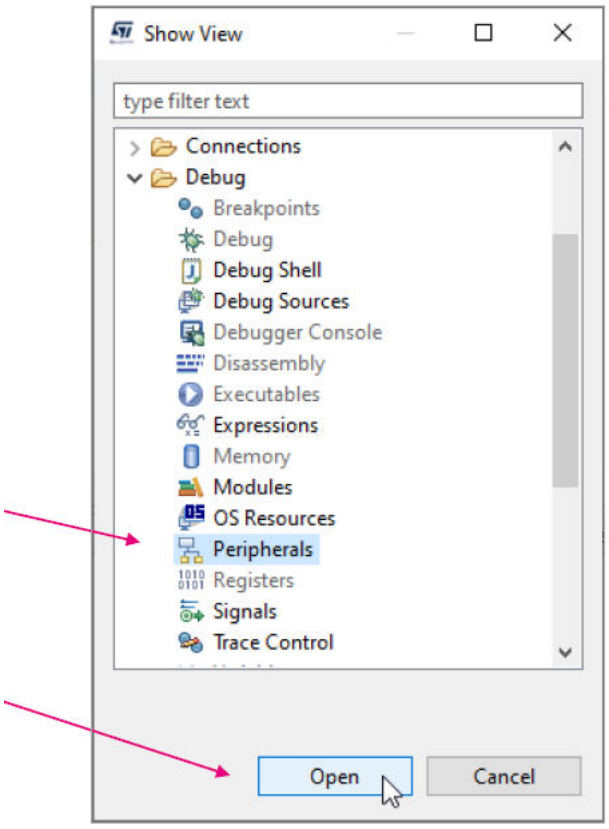
1. From the Window menu, select “Show View”, then “Other…”.

Figure 21. Selecting “Other…” view



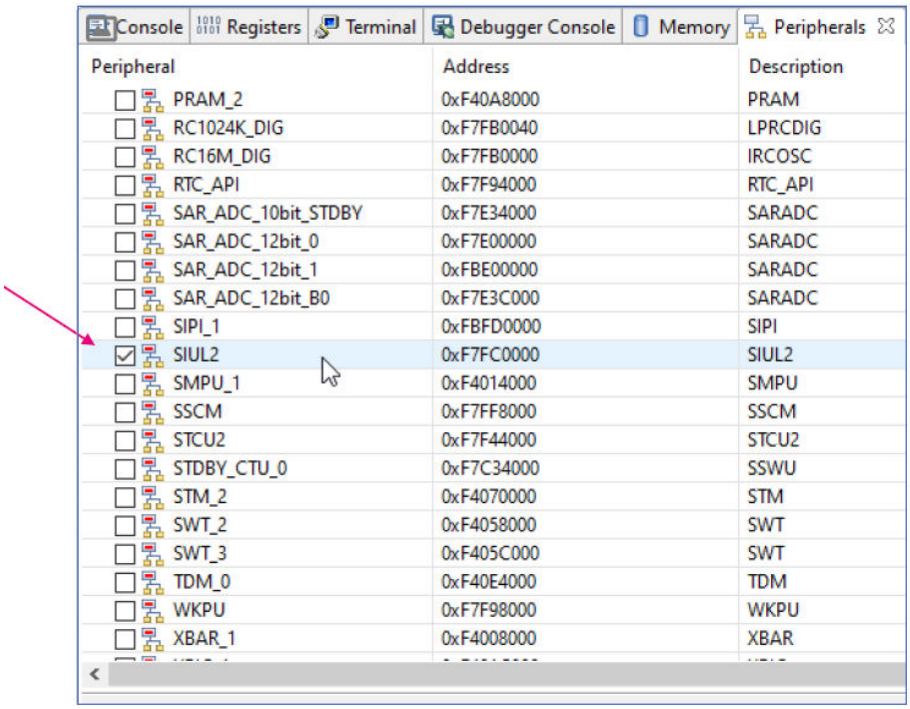
2. Select “Peripherals”.
3. Click on the “Open” button.

Figure 22. Opening peripherals



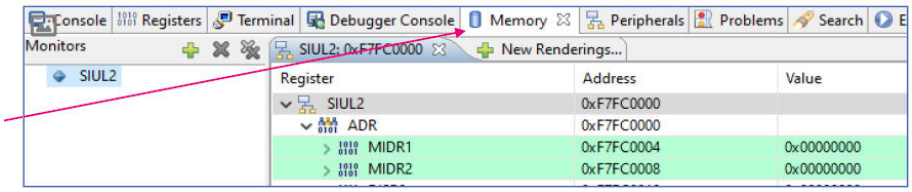
“Peripherals” view is populated only when debugging. For example, check the “SIUL2” and wait a few seconds.

Figure 23. **Peripheral view**



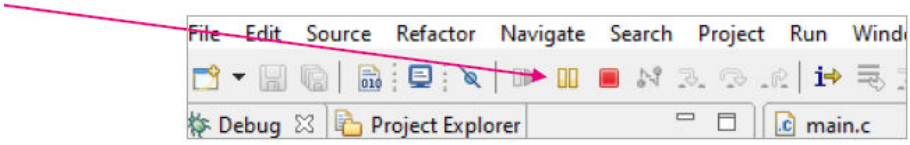
Memory View will open automatically.

Figure 24. **Memory view**



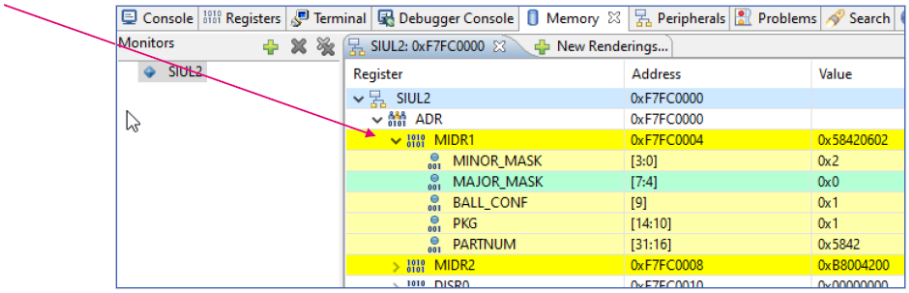
Pause debugging execution.

Figure 25. **Pausing debugging execution**



Expand “MIDR1”.

Figure 26. **Expanding MIDR1**



4 Schematic diagrams

Figure 27. **AEK-MCU-SPC5LNK circuit schematic (1 of 3)**

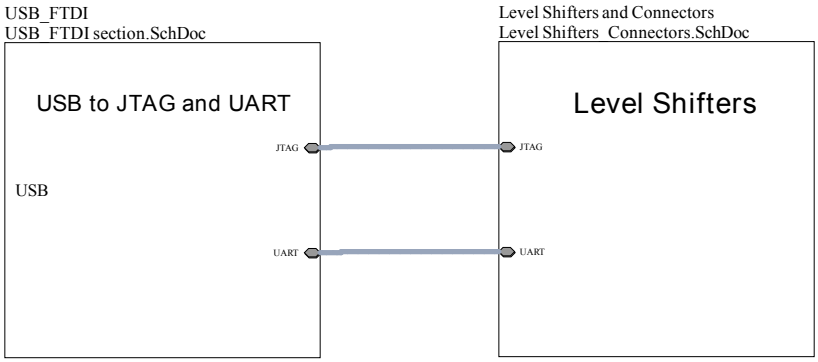


Figure 28. **AEK-MCU-SPC5LNK circuit schematic (2 of 3)**

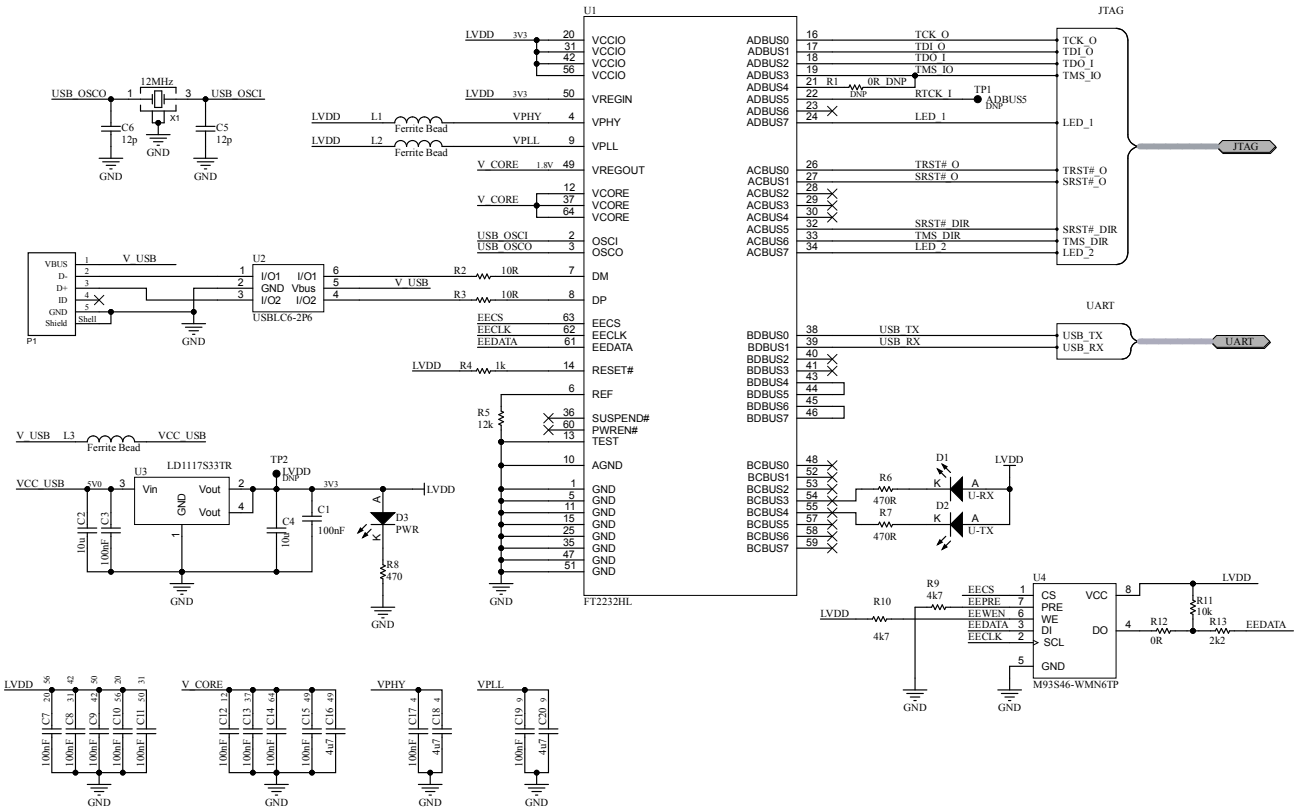
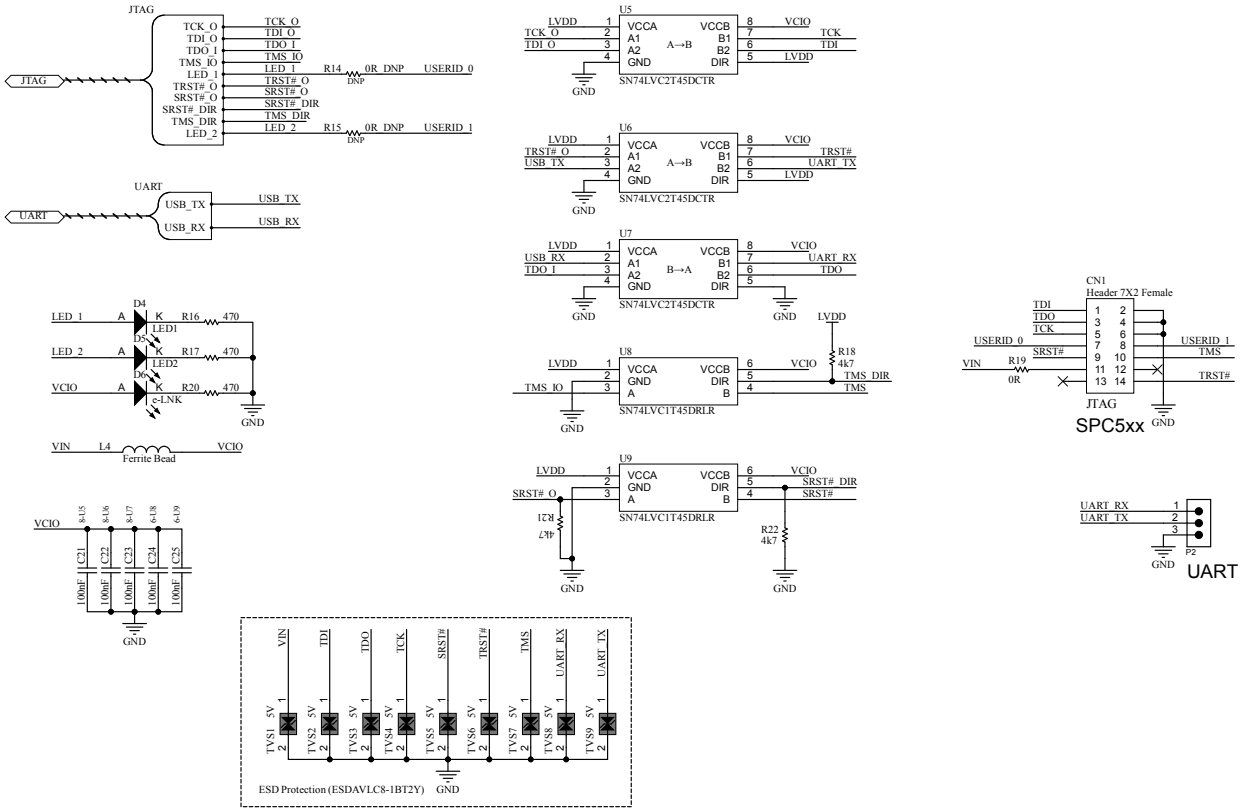


Figure 29. **AEK-MCU-SPC5LNK circuit schematic (3 of 3)**



5 Bill of materials

Table 5. **AEK-MCU-SPC5LNK bill of materials**

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
1	18	C1, C3, C7, C8, C9, C10, C11, C12, C13, C14, C15, C17, C19, C21, C22, C23, C24, C25	100nF	Capacitor X7R - 0603	WE	885012206095
2	2	C2, C4	10u	Capacitor X7R - 0603	WE	885012106031
3	2	C5, C6	18p	C0G Ceramic Multilayer Capacitor	TDK	C1608C0G1H180J080AA
4	3	C16, C18, C20	4u7	Capacitor X7R - 0603	WE	885012106012
5	1	CN1	Header 7X2 Female	Header, 7-Pin, Dual row (6+2.5+10mm)	Sullins	SFH11-PBPC-D07-ST-BK
6	6	D1, D2, D3, D4, D5, D6	150060GS55040	LED Green	WE	150060GS55040
7	4	L1, L2, L3, L4	74279267	Ferrite Bead 0603 60Ohm 500mA	WE	74279267
8	1	P1	629105150521	WR-COM Micro USB 2.0 SMT Type B Horizontal 5 Contacts High Current	WE	629105150521
9	1	P2	61300311021	2.54mm - WR-PHD Angled Pin Header, THT, pitch 2.54mm, Single Row, Vertical, 3p	WE	61300311021
10	3	R1, R14, R15	N.M.	Resistor 0603	N.A	N.A
11	2	R2, R3	10R	Resistor 0603	VISHAY	CRCW060310R0FKTA
12	1	R4	1k	Resistor 0603	BOURNS	CHP0603AFX-1001ELF.
13	1	R5	12k	Resistor 0603	PANASONIC	ERJU03F1202V
14	6	R6, R7,R8, R16, R17, R20	470	Resistor 0603	BOURNS	CR0603-JW-471ELF
15	5	R9, R10, R18, R21, R22	4k7	Resistor 0603	YAGEO	AC0603JR-074K7L
16	1	R11	10k	Resistor 0603	VISHAY	RCA060310K0FKEA
17	2	R12, R19	0R	Resistor 0603	PANASONIC	ERJH3G0R00V
18	1	R13	2k2	Resistor 0603	PANASONIC	ERJUP3J222V
19	2	TP1, TP2	N.M.	Headers & Wire Housings C-Grid SR Vt Pn 6.75 .75/2.90mm Au-F 1Ckt	N.M.	N.M.

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
20	9	TVS1, TVS2, TVS3, TVS4, TVS5, TVS6, TVS7, TVS8, TVS9	3.0V, SOD-882	ESD Suppressor WE-VE, VDC=5.0V	ST	ESDAVLC8-1BT2Y
21	1	U1	FT2232HL	FT2232HL	FTDI	FT2232HL-REEL
22	1	U2	USBLC6-2P6, SOT666	VERY LOW CAPACITANCE ESD PROTECTION	ST	USBLC6-2P6
23	1	U3	LD1117S33TR, SOT-223	Low drop positive voltage regulator	ST	LD1117S33TR
24	1	U4	M93S46XS, SO8	1K (x16) Serial Microwire Bus EEPROM With Block Protection	ST	M93S46-WMN6TP
25	3	U5, U6, U7	SN74LVC2T45 DCTR	Dual-Bit Dual-Supply Bus Transceiver	TI	SN74LVC2T45DCTR
26	2	U8, U9	SN74LVC1T45 DCK	Single-Bit Dual-Supply Bus Transceiver	TI	SN74LVC1T45DCK
27	1	X1	12MHz	WE-XTAL Quartz Crystal, SMT, CFPX-180, 12MHz, +/-10ppm	WE	830108206909

6 Order codes

Table 6. **Order codes**

Order code	Note
AEK-MCU-SPC5LNK	USB/JTAG debugger dongle for SPC5 automotive MCUs compatible with OpenOCD debugger
AEK-MCU-SPC5LNKU	USB/JTAG debugger dongle for SPC5 automotive MCUs compatible with PLS UDE debugger

7 Board versions

Table 7. **AEK-MCU-SPC5LNK versions**

Finished good	Schematic diagrams	Bill of materials
AEK$MCU-SPC5LNKA	AEK$MCU-SPC5LNKA schematic diagrams	AEK$MCU-SPC5LNKA bill of materials
AEK$MCU-SPC5LNKU	AEK$MCU-SPC5LNKU schematic diagrams	AEK$MCU-SPC5LNKU bill of materials

8 Regulatory compliance information

Notice for US Federal Communication Commission (FCC)

For evaluation only; not FCC approved for resale

FCC NOTICE - This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine

whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter 3.1.2.

Notice for Innovation, Science and Economic Development Canada (ISED)

For evaluation purposes only. This kit generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to Industry Canada (IC) rules.

À des fins d'évaluation uniquement. Ce kit génère, utilise et peut émettre de l'énergie radiofréquence et n'a pas été testé pour sa conformité aux limites des appareils informatiques conformément aux règles d'Industrie Canada (IC).

Notice for the European Union

This device is in conformity with the essential requirements of the Directive 2014/30/EU (EMC) and of the Directive 2015/863/EU (RoHS).

Notice for the United Kingdom

This device is in compliance with the UK Electromagnetic Compatibility Regulations 2016 (UK S.I. 2016 No. 1091) and with the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012 (UK S.I. 2012 No. 3032).

Revision history

Table 8. **Document revision history**

Date	Revision	Changes
08-Feb-2024	1	Initial release.
26-Apr-2024	2	Updated Section 3.2: Using the dongle.
26-Aug-2025	3	Updated Section 5: Bill of materials.

Contents

List of tables

List of figures

IMPORTANT NOTICE – READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice.

In the event of any conflict between the provisions of this document and the provisions of any contractual arrangement in force between the purchasers and ST, the provisions of such contractual arrangement shall prevail.

The purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

The purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of the purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

If the purchasers identify an ST product that meets their functional and performance requirements but that is not designated for the purchasers' market segment, the purchasers shall contact ST for more information.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.