PTX Tunneling Library v1.4.1 for Renesas e2studio

The PTX Tunneling library can be used to evaluate and optimize the performance (antenna matching, system/RF configuration, etc.) of any custom-made device using a PTX1xxR device via SPI serial interface.

Embedding this library into the firmware enables the translation of communication from **UART to SPI**, so that the full functionality of **PTX1xxR * Config Tool** can be used in a custom environment. This document also provides instructions on how to create a sample application using a Renesas EK-RA4M2 development board.

Contents

1. Requirement....2

2. Sample Firmware....2

2.1 Creating the Project....2

2.2 Preparing Peripherals....5

2.3 Importing the Library....16

2.3.1. Adding the Include Path....18

2.3.2. Setting the Language Standard....19

2.3.3. Adding the Library File....20

2.4 Implement the HAL....21

2.5 Implementing the Main Function....27

2.6 Building the Firmware....27

3. Prepare the Hardware....28

4. Using the Tunneling Feature....29

5. Revision History....31

1. Requirement

The footprint of the library is ~13kB Flash and 10kB RAM. Moreover, a hardware abstraction layer must be implemented by the user for the particular uC/Board, which executes the low-level commands requested by the library.

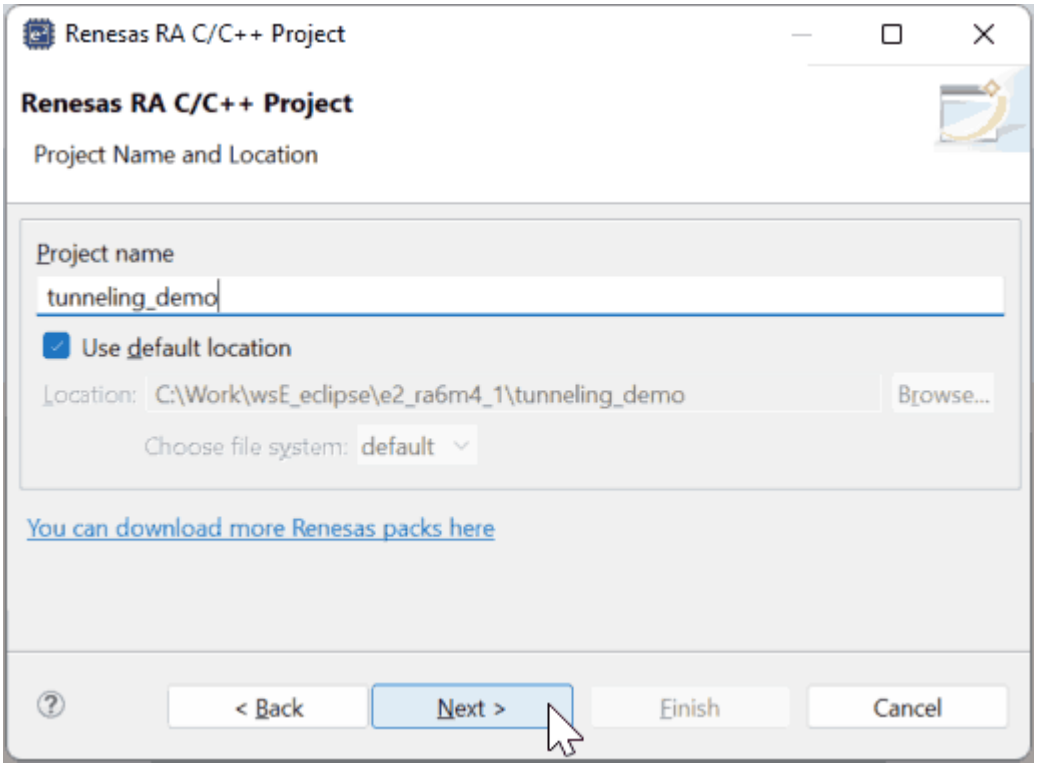
The library can be seamlessly integrated into a CMAKE project as well, but the e2studio is used in this document (for more information, see e2studio).

2. Sample Firmware

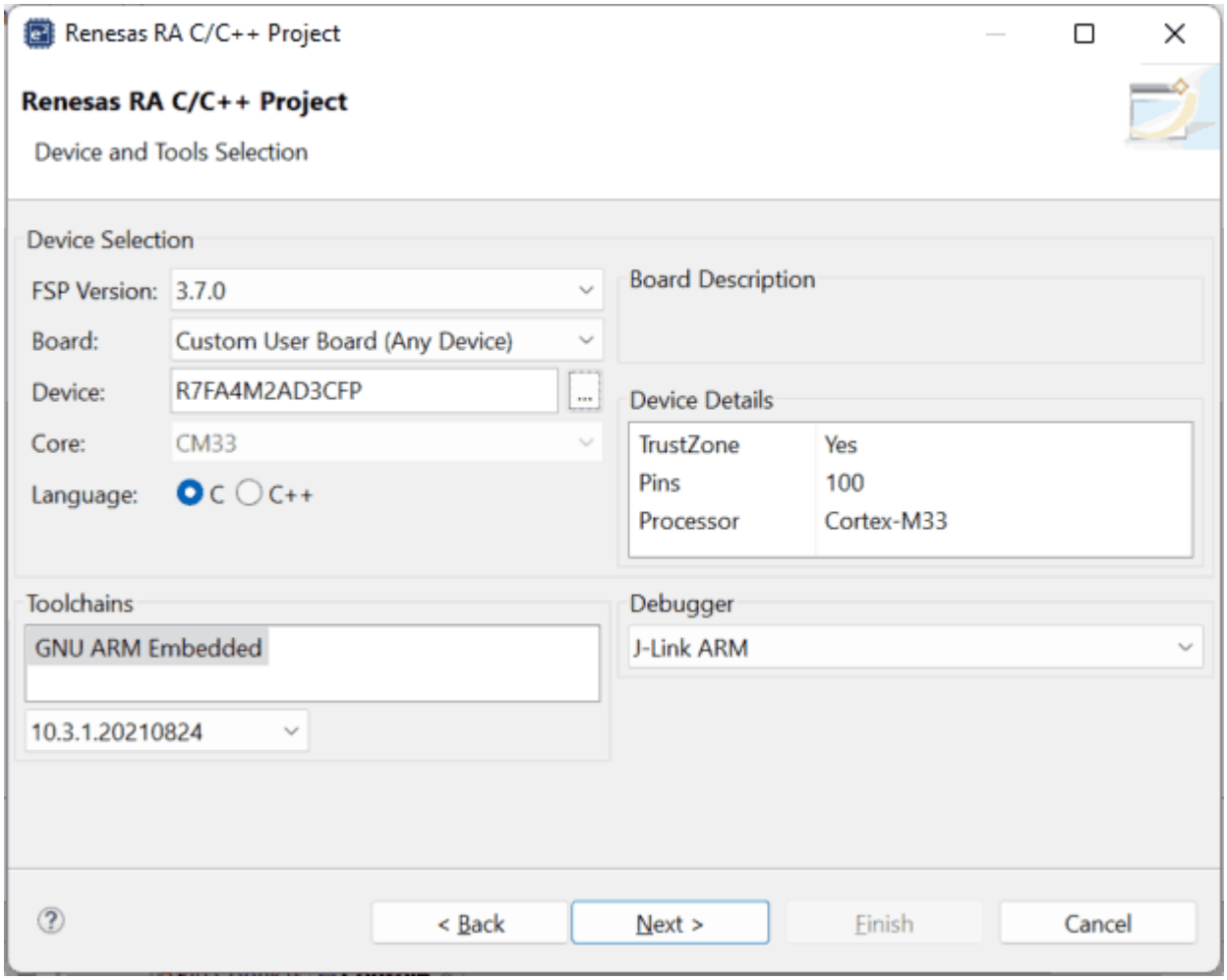
The sample application is used for creating and serving the tunnel between the host PC UART interface and the PTX1xxR chip connected by SPI. The library can be used either as a precompiled static library or as a source-library – most steps are the same for both cases.

2.1 Creating the Project

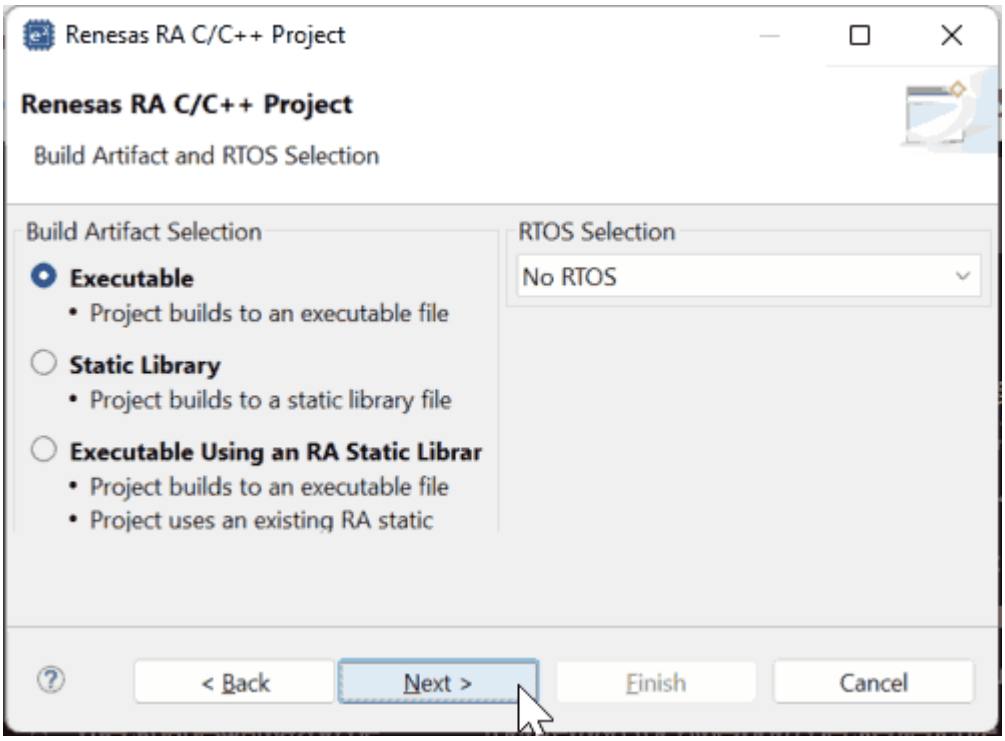
To create a new project in *e2studio* from Menu, select **File > New > Renesas C/C++ Project > Renesas RA** and choose Renesas RA C/C++ Project.



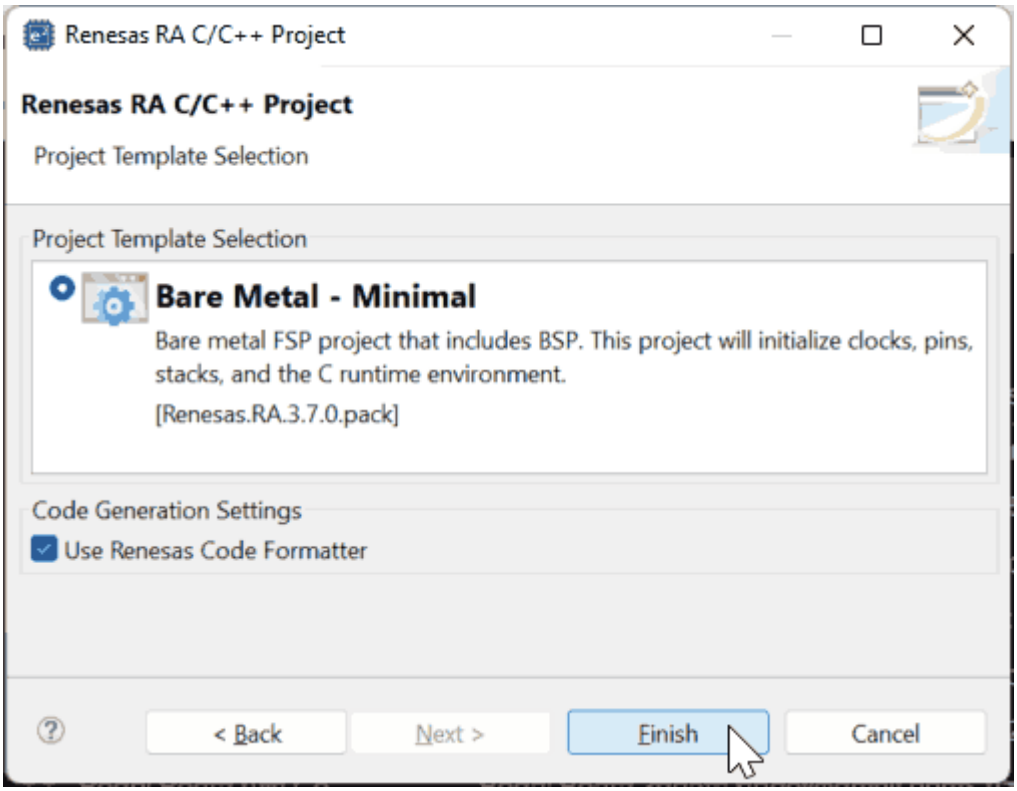
The following figure shows how to select the device and board type.



When prompted, the project should be a **Flat (Non-TrustZone) Project**. The project will have no operating system and will have a build artifact type of **Executable**.



Finally, the **Bare Metal** template project can be selected in order to create a minimal project.



2.2 Preparing Peripherals

The generated empty project does not contain the required components from the SDK by default. For the demonstration, several peripherals and their drivers, like SPI and Timer, must be added to the project. The easiest way to do this is to open the file `configuration.xml` in a text editor and update its content to the following.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<raConfiguration version="7">
  <generalSettings>
    <option key="#Board#" value="board.custom"/>
    <option key="CPU" value="RA4M2"/>
    <option key="Core" value="CM33"/>
    <option key="#TargetName#" value="R7FA4M2AD3CFP"/>
    <option key="#TargetARCHITECTURE#" value="cortex-m33"/>
    <option key="#DeviceCommand#" value="R7FA4M2AD"/>
    <option key="#RTOS#" value="_none"/>
    <option key="#pinconfiguration#" value="R7FA4M2AD3CFP.pincfg"/>
    <option key="#FSPVersion#" value="3.7.0"/>
    <option key="#SELECTED_TOOLCHAIN#" value="gcc-arm-embedded"/>
    <option key="#ToolchainVersion#" value="10.3.1.20210824"/>
  </generalSettings>
  <raBspConfiguration>
    <config id="config.bsp.ra4m2.R7FA4M2AD3CFP">
      <property id="config.bsp.part_number" value="config.bsp.part_number.value"/>
      <property id="config.bsp.rom_size_bytes" value="config.bsp.rom_size_bytes.value"/>
      <property id="config.bsp.rom_size_bytes_hidden" value="524288"/>
      <property id="config.bsp.ram_size_bytes" value="config.bsp.ram_size_bytes.value"/>
      <property id="config.bsp.data_flash_size_bytes"
value="config.bsp.data_flash_size_bytes.value"/>
      <property id="config.bsp.package_style" value="config.bsp.package_style.value"/>
      <property id="config.bsp.package_pins" value="config.bsp.package_pins.value"/>
      <property id="config.bsp.irq_count_hidden" value="96"/>
    </config>
    <config id="config.bsp.ra4m2">
      <property id="config.bsp.series" value="config.bsp.series.value"/>
    </config>
    <config id="config.bsp.ra4m2.fsp">
      <property id="config.bsp.fsp.tz.exception_response"
value="config.bsp.fsp.tz.exception_response.nmi"/>
      <property id="config.bsp.fsp.tz.cmsis.bfhfnmins"
value="config.bsp.fsp.tz.cmsis.bfhfnmins.secure"/>
      <property id="config.bsp.fsp.tz.cmsis.sysresetreqs"
value="config.bsp.fsp.tz.cmsis.sysresetreqs.secure_only"/>
      <property id="config.bsp.fsp.tz.cmsis.s_priority_boost"
value="config.bsp.fsp.tz.cmsis.s_priority_boost.disabled"/>
      <property id="config.bsp.fsp.tz.csar" value="config.bsp.fsp.tz.csar.both"/>
      <property id="config.bsp.fsp.tz.rstsar" value="config.bsp.fsp.tz.rstsar.both"/>
      <property id="config.bsp.fsp.tz.bbfsar" value="config.bsp.fsp.tz.bbfsar.both"/>
      <property id="config.bsp.fsp.tz.sramsar.sramprcr"
value="config.bsp.fsp.tz.sramsar.sramprcr.both"/>
      <property id="config.bsp.fsp.tz.sramsar.sramecc"
value="config.bsp.fsp.tz.sramsar.sramecc.both"/>
      <property id="config.bsp.fsp.tz.stbramsar"
value="config.bsp.fsp.tz.stbramsar.both"/>
      <property id="config.bsp.fsp.tz.bussara" value="config.bsp.fsp.tz.bussara.both"/>
      <property id="config.bsp.fsp.tz.bussarb" value="config.bsp.fsp.tz.bussarb.both"/>
      <property id="config.bsp.fsp.cache_line_size"
value="config.bsp.fsp.cache_line_size.32"/>
      <property id="config.bsp.fsp.OFS0.iwdt_start_mode"
value="config.bsp.fsp.OFS0.iwdt_start_mode.disabled"/>
```

```
      <property id="config.bsp.fsp.OFS0.iwdt_timeout"
value="config.bsp.fsp.OFS0.iwdt_timeout.2048"/>
      <property id="config.bsp.fsp.OFS0.iwdt_divisor"
value="config.bsp.fsp.OFS0.iwdt_divisor.128"/>
      <property id="config.bsp.fsp.OFS0.iwdt_window_end"
value="config.bsp.fsp.OFS0.iwdt_window_end.0"/>
      <property id="config.bsp.fsp.OFS0.iwdt_window_start"
value="config.bsp.fsp.OFS0.iwdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.iwdt_reset_interrupt"
value="config.bsp.fsp.OFS0.iwdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.iwdt_stop_control"
value="config.bsp.fsp.OFS0.iwdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS0.wdt_start_mode"
value="config.bsp.fsp.OFS0.wdt_start_mode.register"/>
      <property id="config.bsp.fsp.OFS0.wdt_timeout"
value="config.bsp.fsp.OFS0.wdt_timeout.16384"/>
      <property id="config.bsp.fsp.OFS0.wdt_divisor"
value="config.bsp.fsp.OFS0.wdt_divisor.128"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_end"
value="config.bsp.fsp.OFS0.wdt_window_end.0"/>
      <property id="config.bsp.fsp.OFS0.wdt_window_start"
value="config.bsp.fsp.OFS0.wdt_window_start.100"/>
      <property id="config.bsp.fsp.OFS0.wdt_reset_interrupt"
value="config.bsp.fsp.OFS0.wdt_reset_interrupt.Reset"/>
      <property id="config.bsp.fsp.OFS0.wdt_stop_control"
value="config.bsp.fsp.OFS0.wdt_stop_control.stops"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0.start"
value="config.bsp.fsp.OFS1.voltage_detection0.start.disabled"/>
      <property id="config.bsp.fsp.OFS1.voltage_detection0_level"
value="config.bsp.fsp.OFS1.voltage_detection0_level.280"/>
      <property id="config.bsp.fsp.OFS1.hoco_osc"
value="config.bsp.fsp.OFS1.hoco_osc.disabled"/>
      <property id="config.bsp.fsp.BPS.BPS0" value=""/>
      <property id="config.bsp.fsp.PBPS.PBPS0" value=""/>
      <property id="config.bsp.fsp.hoco_fll" value="config.bsp.fsp.hoco_fll.disabled"/>
      <property id="config.bsp.common.main_osc_wait"
value="config.bsp.common.main_osc_wait.wait_8163"/>
      <property id="config.bsp.fsp.mcu.adc.max_freq_hz" value="50000000"/>
      <property id="config.bsp.fsp.mcu.sci_uart.max_baud" value="20000000"/>
      <property id="config.bsp.fsp.mcu.adc.sample_and_hold" value="0"/>
      <property id="config.bsp.fsp.mcu.sci_spi.max_bitrate" value="25000000"/>
      <property id="config.bsp.fsp.mcu.spi.max_bitrate" value="50000000"/>
      <property id="config.bsp.fsp.mcu.iic_master.rate.rate_fastplus" value="1"/>
      <property id="config.bsp.fsp.mcu.sci_uart.cstpen_channels" value="0x0219"/>
      <property id="config.bsp.fsp.mcu.gpt.pin_count_source_channels" value="0xFFFF"/>
    </config>
    <config id="config.bsp.ra">
      <property id="config.bsp.common.main" value="0x400"/>
      <property id="config.bsp.common.heap" value="0"/>
      <property id="config.bsp.common.vcc" value="3300"/>
      <property id="config.bsp.common.checking"
value="config.bsp.common.checking.disabled"/>
      <property id="config.bsp.common.assert" value="config.bsp.common.assert.none"/>
      <property id="config.bsp.common.error_log"
value="config.bsp.common.error_log.none"/>
      <property id="config.bsp.common.soft_reset"
value="config.bsp.common.soft_reset.disabled"/>
      <property id="config.bsp.common.main_osc_populated"
value="config.bsp.common.main_osc_populated.enabled"/>
      <property id="config.bsp.common.pfs_protect"
value="config.bsp.common.pfs_protect.enabled"/>
```

```
      <property id="config.bsp.common.c_runtime_init"
value="config.bsp.common.c_runtime_init.enabled"/>
      <property id="config.bsp.common.early_init"
value="config.bsp.common.early_init.disabled"/>
      <property id="config.bsp.common.main_osc_clock_source"
value="config.bsp.common.main_osc_clock_source.crystal"/>
      <property id="config.bsp.common.subclock_populated"
value="config.bsp.common.subclock_populated.enabled"/>
      <property id="config.bsp.common.subclock_drive"
value="config.bsp.common.subclock_drive.standard"/>
      <property id="config.bsp.common.subclock_stabilization_ms" value="1000"/>
    </config>
  </raBspConfiguration>
  <raClockConfiguration>
    <node id="board.clock.xtal.freq" mul="24000000" option="_edit"/>
    <node id="board.clock.hoco.freq" option="board.clock.hoco.freq.20m"/>
    <node id="board.clock.loco.freq" option="board.clock.loco.freq.32768"/>
    <node id="board.clock.moco.freq" option="board.clock.moco.freq.8m"/>
    <node id="board.clock.subclk.freq" option="board.clock.subclk.freq.32768"/>
    <node id="board.clock.pll.source" option="board.clock.pll.source.xtal"/>
    <node id="board.clock.pll.div" option="board.clock.pll.div.3"/>
    <node id="board.clock.pll.mul" option="board.clock.pll.mul.250"/>
    <node id="board.clock.pll.display" option="board.clock.pll.display.value"/>
    <node id="board.clock.pll2.source" option="board.clock.pll2.source.disabled"/>
    <node id="board.clock.pll2.div" option="board.clock.pll2.div.2"/>
    <node id="board.clock.pll2.mul" option="board.clock.pll2.mul.200"/>
    <node id="board.clock.pll2.display" option="board.clock.pll2.display.value"/>
    <node id="board.clock.clock.source" option="board.clock.clock.source.pll"/>
    <node id="board.clock.clkout.source" option="board.clock.clkout.source.disabled"/>
    <node id="board.clock.uclk.source" option="board.clock.uclk.source.disabled"/>
    <node id="board.clock.iclk.div" option="board.clock.iclk.div.2"/>
    <node id="board.clock.pclka.div" option="board.clock.pclka.div.2"/>
    <node id="board.clock.pclkb.div" option="board.clock.pclkb.div.4"/>
    <node id="board.clock.pclkc.div" option="board.clock.pclkc.div.4"/>
    <node id="board.clock.pclkd.div" option="board.clock.pclkd.div.2"/>
    <node id="board.clock.fclk.div" option="board.clock.fclk.div.4"/>
    <node id="board.clock.clkout.div" option="board.clock.clkout.div.1"/>
    <node id="board.clock.uclk.div" option="board.clock.uclk.div.5"/>
    <node id="board.clock.iclk.display" option="board.clock.iclk.display.value"/>
    <node id="board.clock.pclka.display" option="board.clock.pclka.display.value"/>
    <node id="board.clock.pclkb.display" option="board.clock.pclkb.display.value"/>
    <node id="board.clock.pclkc.display" option="board.clock.pclkc.display.value"/>
    <node id="board.clock.pclkd.display" option="board.clock.pclkd.display.value"/>
    <node id="board.clock.fclk.display" option="board.clock.fclk.display.value"/>
    <node id="board.clock.clkout.display" option="board.clock.clkout.display.value"/>
    <node id="board.clock.uclk.display" option="board.clock.uclk.display.value"/>
  </raClockConfiguration>
  <raComponentSelection>
    <component apiversion="" class="Common" condition="" group="all"
subgroup="fsp_common" variant="" vendor="Renesas" version="3.7.0">
      <description>Board Support Package Common Files</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_ioport" variant="" vendor="Renesas" version="3.7.0">
      <description>I/O Port</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="CMSIS" condition="" group="CMSIS5" subgroup="CoreM"
variant="" vendor="Arm" version="5.8.0+renesas.0.fsp.3.7.0">
      <description>Arm CMSIS Version 5 - Core (M)</description>
```

```
      <originalPack>Arm.CMSIS5.5.8.0+renesas.0.fsp.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="Board" subgroup="custom"
variant="" vendor="Renesas" version="3.7.0">
      <description>Custom Board Support Files</description>
      <originalPack>Renesas.RA_board_custom.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra4m2" subgroup="device"
variant="R7FA4M2AD3CFP" vendor="Renesas" version="3.7.0">
      <description>Board support package for R7FA4M2AD3CFP</description>
      <originalPack>Renesas.RA_mcu_ra4m2.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra4m2" subgroup="device"
variant="" vendor="Renesas" version="3.7.0">
      <description>Board support package for RA4M2</description>
      <originalPack>Renesas.RA_mcu_ra4m2.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="BSP" condition="" group="ra4m2" subgroup="fsp"
variant="" vendor="Renesas" version="3.7.0">
      <description>Board support package for RA4M2 - FSP Data</description>
      <originalPack>Renesas.RA_mcu_ra4m2.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_dtc" variant="" vendor="Renesas" version="3.7.0">
      <description>Data Transfer Controller</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_gpt" variant="" vendor="Renesas" version="3.7.0">
      <description>General PWM Timer</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_sci_spi" variant="" vendor="Renesas" version="3.7.0">
      <description>Serial Peripheral Interface on Serial Communications
Interface</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
    <component apiversion="" class="HAL Drivers" condition="" group="all"
subgroup="r_sci_uart" variant="" vendor="Renesas" version="3.7.0">
      <description>SCI UART</description>
      <originalPack>Renesas.RA.3.7.0.pack</originalPack>
    </component>
  </raComponentSelection>
  <raElcConfiguration/>
  <raIcuConfiguration/>
  <raModuleConfiguration>
    <module id="module.driver.ioport_on_ioport.0">
      <property id="module.driver.ioport.name" value="g_ioport"/>
      <property id="module.driver.ioport.elc_trigger_ioport1" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioport2" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioport3" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioport4" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioportb" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioportc" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioportd" value="_disabled"/>
      <property id="module.driver.ioport.elc_trigger_ioporte" value="_disabled"/>
      <property id="module.driver.ioport.pincfg" value="g_bsp_pin_cfg"/>
    </module>
    <module id="module.driver.uart_on_sci_uart.728447659">
      <property id="module.driver.uart.name" value="g_uart0"/>
```

```
      <property id="module.driver.uart.channel" value="2"/>
      <property id="module.driver.uart.data_bits"
value="module.driver.uart.data_bits.data_bits_8"/>
      <property id="module.driver.uart.parity"
value="module.driver.uart.parity.parity_off"/>
      <property id="module.driver.uart.stop_bits"
value="module.driver.uart.stop_bits.stop_bits_1"/>
      <property id="module.driver.uart.baud" value="115200"/>
      <property id="module.driver.uart.baudrate_modulation"
value="module.driver.uart.baudrate_modulation.disabled"/>
      <property id="module.driver.uart.baudrate_max_err" value="5"/>
      <property id="module.driver.uart.flow_control"
value="module.driver.uart.flow_control.rts"/>
      <property id="module.driver.uart.pin_control_port"
value="module.driver.uart.pin_control_port.PORT_DISABLE"/>
      <property id="module.driver.uart.pin_control_pin"
value="module.driver.uart.pin_control_pin.PIN_DISABLE"/>
      <property id="module.driver.uart.clk_src"
value="module.driver.uart.clk_src.int_clk"/>
      <property id="module.driver.uart.rx_edge_start"
value="module.driver.uart.rx_edge_start.falling_edge"/>
      <property id="module.driver.uart.noisecancel_en"
value="module.driver.uart.noisecancel_en.disabled"/>
      <property id="module.driver.uart.rx_fifo_trigger"
value="module.driver.uart.rx_fifo_trigger.max"/>
      <property id="module.driver.uart.rs485.de_enable"
value="module.driver.uart.rs485.de_enable.disabled"/>
      <property id="module.driver.uart.rs485.de_polarity"
value="module.driver.uart.rs485.de_polarity.high"/>
      <property id="module.driver.uart.rs485.de_port_number"
value="module.driver.uart.rs485.de_port_number.PORT_DISABLE"/>
      <property id="module.driver.uart.rs485.de_pin_number"
value="module.driver.uart.rs485.de_pin_number.PIN_DISABLE"/>
      <property id="module.driver.uart.callback" value="uart_callback"/>
      <property id="module.driver.uart.rxi_ipl"
value="board.icu.common.irq.priority12"/>
      <property id="module.driver.uart.txi_ipl"
value="board.icu.common.irq.priority12"/>
      <property id="module.driver.uart.tei_ipl"
value="board.icu.common.irq.priority12"/>
      <property id="module.driver.uart.eri_ipl"
value="board.icu.common.irq.priority12"/>
    </module>
    <module id="module.driver.spi_on_sci_spi.1549399989">
      <property id="module.driver.spi.name" value="g_spi0"/>
      <property id="module.driver.spi.channel" value="0"/>
      <property id="module.driver.spi.operating_mode"
value="module.driver.spi.operating_mode.mode_master"/>
      <property id="module.driver.spi.clk_phase"
value="module.driver.spi.clk_phase.clk_phase_edge_odd"/>
      <property id="module.driver.spi.clk_polarity"
value="module.driver.spi.clk_polarity.clk_polarity_low"/>
      <property id="module.driver.spi.mode_fault"
value="module.driver.spi.mode_fault.mode_fault_error_disable"/>
      <property id="module.driver.spi.bit_order"
value="module.driver.spi.bit_order.bit_order_msb_first"/>
      <property id="module.driver.spi.p_callback" value="sci_spi_callback"/>
      <property id="module.driver.spi.rxi_ipl" value="board.icu.common.irq.priority12"/>
      <property id="module.driver.spi.txi_ipl" value="board.icu.common.irq.priority12"/>
      <property id="module.driver.spi.tei_ipl" value="board.icu.common.irq.priority12"/>
      <property id="module.driver.spi.eri_ipl" value="board.icu.common.irq.priority12"/>
```

```
      <property id="module.driver.spi.bitrate" value="8000000"/>
      <property id="module.driver.spi.bitrate_modulation"
value="module.driver.spi.bitrate_modulation.disabled"/>
    </module>
    <module id="module.driver.timer_on_gpt.453360959">
      <property id="module.driver.timer.name" value="g_timer0"/>
      <property id="module.driver.timer.channel" value="0"/>
      <property id="module.driver.timer.mode"
value="module.driver.timer.mode.mode_periodic"/>
      <property id="module.driver.timer.period" value="10"/>
      <property id="module.driver.timer.unit"
value="module.driver.timer.unit.unit_period_msec"/>
      <property id="module.driver.timer.gtior.gtioa.initial_output_level"
value="module.driver.timer.gtior.gtioa.initial_output_level.low"/>
      <property id="module.driver.timer.gtior.gtioa.cycle_end_output_level"
value="module.driver.timer.gtior.gtioa.cycle_end_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtioa.compare_match_output_level"
value="module.driver.timer.gtior.gtioa.compare_match_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtioa.count_stop_retain"
value="module.driver.timer.gtior.gtioa.count_stop_retain.disabled"/>
      <property id="module.driver.timer.gtior.gtiob.initial_output_level"
value="module.driver.timer.gtior.gtiob.initial_output_level.low"/>
      <property id="module.driver.timer.gtior.gtiob.cycle_end_output_level"
value="module.driver.timer.gtior.gtiob.cycle_end_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtiob.compare_match_output_level"
value="module.driver.timer.gtior.gtiob.compare_match_output_level.retain"/>
      <property id="module.driver.timer.gtior.gtiob.count_stop_retain"
value="module.driver.timer.gtior.gtiob.count_stop_retain.disabled"/>
      <property id="module.driver.timer.gtior.custom_waveform_enable"
value="module.driver.timer.gtior.custom_waveform_enable.disabled"/>
      <property id="module.driver.timer.duty_cycle" value="50"/>
      <property id="module.driver.timer.gtioca_output_enabled"
value="module.driver.timer.gtioca_output_enabled.false"/>
      <property id="module.driver.timer.gtioca_stop_level"
value="module.driver.timer.gtioca_stop_level.pin_level_low"/>
      <property id="module.driver.timer.gtiocb_output_enabled"
value="module.driver.timer.gtiocb_output_enabled.false"/>
      <property id="module.driver.timer.gtiocb_stop_level"
value="module.driver.timer.gtiocb_stop_level.pin_level_low"/>
      <property id="module.driver.timer.count_up_source" value=""/>
      <property id="module.driver.timer.count_down_source" value=""/>
      <property id="module.driver.timer.start_source" value=""/>
      <property id="module.driver.timer.stop_source" value=""/>
      <property id="module.driver.timer.clear_source" value=""/>
      <property id="module.driver.timer.capture_a_source" value=""/>
      <property id="module.driver.timer.capture_b_source" value=""/>
      <property id="module.driver.timer.gtioca_filter"
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
      <property id="module.driver.timer.gtiocb_filter"
value="module.driver.timer.gtioc_filter.gtioc_filter_none"/>
      <property id="module.driver.timer.p_callback" value="NULL"/>
      <property id="module.driver.timer.ipl" value="_disabled"/>
      <property id="module.driver.timer.capture_a_ipl" value="_disabled"/>
      <property id="module.driver.timer.capture_b_ipl" value="_disabled"/>
      <property id="module.driver.timer.trough_ipl" value="_disabled"/>
      <property id="module.driver.timer.extra"
value="module.driver.timer.extra.disabled"/>
      <property id="module.driver.timer.poeg_link"
value="module.driver.timer.poeg_link.poeg_link_poeg0"/>
      <property id="module.driver.timer.output_disable" value=""/>
      <property id="module.driver.timer.adc_trigger" value=""/>
```

```
      <property id="module.driver.timer.dead_time_count_up" value="0"/>
      <property id="module.driver.timer.dead_time_count_down" value="0"/>
      <property id="module.driver.timer.adc_a_compare_match" value="0"/>
      <property id="module.driver.timer.adc_b_compare_match" value="0"/>
      <property id="module.driver.timer.interrupt_skip.source"
value="module.driver.timer.interrupt_skip.source.none"/>
      <property id="module.driver.timer.interrupt_skip.count"
value="module.driver.timer.interrupt_skip.count.count_0"/>
      <property id="module.driver.timer.interrupt_skip.adc"
value="module.driver.timer.interrupt_skip.adc.none"/>
      <property id="module.driver.timer.gtioca_disable_setting"
value="module.driver.timer.gtioca_disable_setting.gtioc_disable_prohibited"/>
      <property id="module.driver.timer.gtiocb_disable_setting"
value="module.driver.timer.gtiocb_disable_setting.gtioc_disable_prohibited"/>
    </module>
    <module id="module.driver.transfer_on_dtc.1741066903">
      <property id="module.driver.transfer.name" value="g_transfer0"/>
      <property id="module.driver.transfer.mode"
value="module.driver.transfer.mode.mode_normal"/>
      <property id="module.driver.transfer.size"
value="module.driver.transfer.size.size_2_byte"/>
      <property id="module.driver.transfer.dest_addr_mode"
value="module.driver.transfer.dest_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.src_addr_mode"
value="module.driver.transfer.src_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.repeat_area"
value="module.driver.transfer.repeat_area.repeat_area_source"/>
      <property id="module.driver.transfer.p_dest" value="NULL"/>
      <property id="module.driver.transfer.p_src" value="NULL"/>
      <property id="module.driver.transfer.interrupt"
value="module.driver.transfer.interrupt.interrupt_end"/>
      <property id="module.driver.transfer.length" value="0"/>
      <property id="module.driver.transfer.num_blocks" value="0"/>
      <property id="module.driver.transfer.activation_source" value="_disabled"/>
    </module>
    <module id="module.driver.transfer_on_dtc.730821984">
      <property id="module.driver.transfer.name" value="g_transfer1"/>
      <property id="module.driver.transfer.mode"
value="module.driver.transfer.mode.mode_normal"/>
      <property id="module.driver.transfer.size"
value="module.driver.transfer.size.size_2_byte"/>
      <property id="module.driver.transfer.dest_addr_mode"
value="module.driver.transfer.dest_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.src_addr_mode"
value="module.driver.transfer.src_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.repeat_area"
value="module.driver.transfer.repeat_area.repeat_area_source"/>
      <property id="module.driver.transfer.p_dest" value="NULL"/>
      <property id="module.driver.transfer.p_src" value="NULL"/>
      <property id="module.driver.transfer.interrupt"
value="module.driver.transfer.interrupt.interrupt_end"/>
      <property id="module.driver.transfer.length" value="0"/>
      <property id="module.driver.transfer.num_blocks" value="0"/>
      <property id="module.driver.transfer.activation_source" value="_disabled"/>
    </module>
    <module id="module.driver.transfer_on_dtc.1206570453">
      <property id="module.driver.transfer.name" value="g_transfer2"/>
      <property id="module.driver.transfer.mode"
value="module.driver.transfer.mode.mode_normal"/>
      <property id="module.driver.transfer.size"
value="module.driver.transfer.size.size_2_byte"/>
```

```
      <property id="module.driver.transfer.dest_addr_mode"
value="module.driver.transfer.dest_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.src_addr_mode"
value="module.driver.transfer.src_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.repeat_area"
value="module.driver.transfer.repeat_area.repeat_area_source"/>
      <property id="module.driver.transfer.p_dest" value="NULL"/>
      <property id="module.driver.transfer.p_src" value="NULL"/>
      <property id="module.driver.transfer.interrupt"
value="module.driver.transfer.interrupt.interrupt_end"/>
      <property id="module.driver.transfer.length" value="0"/>
      <property id="module.driver.transfer.num_blocks" value="0"/>
      <property id="module.driver.transfer.activation_source" value="_disabled"/>
    </module>
    <module id="module.driver.transfer_on_dtc.415632937">
      <property id="module.driver.transfer.name" value="g_transfer3"/>
      <property id="module.driver.transfer.mode"
value="module.driver.transfer.mode.mode_normal"/>
      <property id="module.driver.transfer.size"
value="module.driver.transfer.size.size_2_byte"/>
      <property id="module.driver.transfer.dest_addr_mode"
value="module.driver.transfer.dest_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.src_addr_mode"
value="module.driver.transfer.src_addr_mode.addr_mode_fixed"/>
      <property id="module.driver.transfer.repeat_area"
value="module.driver.transfer.repeat_area.repeat_area_source"/>
      <property id="module.driver.transfer.p_dest" value="NULL"/>
      <property id="module.driver.transfer.p_src" value="NULL"/>
      <property id="module.driver.transfer.interrupt"
value="module.driver.transfer.interrupt.interrupt_end"/>
      <property id="module.driver.transfer.length" value="0"/>
      <property id="module.driver.transfer.num_blocks" value="0"/>
      <property id="module.driver.transfer.activation_source" value="_disabled"/>
    </module>
    <context id="_hal.0">
      <stack module="module.driver.ioport_on_ioport.0"/>
      <stack module="module.driver.uart_on_sci_uart.728447659">
        <stack module="module.driver.transfer_on_dtc.1741066903"
requires="module.driver.uart_on_sci_uart.requires.transfer_tx"/>
        <stack module="module.driver.transfer_on_dtc.415632937"
requires="module.driver.uart_on_sci_uart.requires.transfer_rx"/>
      </stack>
      <stack module="module.driver.spi_on_sci_spi.1549399989">
        <stack module="module.driver.transfer_on_dtc.730821984"
requires="module.driver.spi_on_sci_spi.requires.transfer_tx"/>
        <stack module="module.driver.transfer_on_dtc.1206570453"
requires="module.driver.spi_on_sci_spi.requires.transfer_rx"/>
      </stack>
      <stack module="module.driver.timer_on_gpt.453360959"/>
    </context>
    <config id="config.driver.ioport">
      <property id="config.driver.ioport.checking"
value="config.driver.ioport.checking.system"/>
    </config>
    <config id="config.driver.sci_spi">
      <property id="config.driver.sci_spi.param_checking_enable"
value="config.driver.sci_spi.param_checking_enable.bsp"/>
      <property id="config.driver.sci_spi.dtc_support_enable"
value="config.driver.sci_spi.dtc_support_enable.enabled"/>
    </config>
    <config id="config.driver.dtc">
```

```
      <property id="config.driver.dtc.param_checking_enable"
value="config.driver.dtc.param_checking_enable.bsp"/>
      <property id="config.driver.dtc.vector_table" value=".fsp_dtc_vector_table"/>
    </config>
    <config id="config.driver.sci_uart">
      <property id="config.driver.sci_uart.param_checking_enable"
value="config.driver.sci_uart.param_checking_enable.bsp"/>
      <property id="config.driver.sci_uart.fifo_support"
value="config.driver.sci_uart.fifo_support.disabled"/>
      <property id="config.driver.sci_uart.dtc_support"
value="config.driver.sci_uart.dtc_support.enabled"/>
      <property id="config.driver.sci_uart.flow_control"
value="config.driver.sci_uart.flow_control.disabled"/>
      <property id="config.driver.sci_uart.rs485"
value="config.driver.sci_uart.rs485.disabled"/>
    </config>
    <config id="config.driver.gpt">
      <property id="config.driver.gpt.param_checking_enable"
value="config.driver.gpt.param_checking_enable.bsp"/>
      <property id="config.driver.gpt.output_support_enable"
value="config.driver.gpt.output_support_enable.disabled"/>
      <property id="config.driver.gpt.write_protect_enable"
value="config.driver.gpt.write_protect_enable.disabled"/>
      <property id="config.driver.gpt.gpt_core_clock"
value="module.driver.timer.gpt_core_clock.pclk"/>
    </config>
  </raModuleConfiguration>
  <raPinConfiguration>
    <symbolicName propertyId="p413.symbolic_name" value="PTX_PLAT_SPI_CS_PIN"/>
    <symbolicName propertyId="p414.symbolic_name" value="PTX_PLAT_SPI_IRQ_PIN"/>
    <pincfg active="true" name="R7FA4M2AD3CFP.pincfg" selected="true"
symbol="g_bsp_pin_cfg">
      <configSetting altId="debug0.mode.jtag" configurationId="debug0.mode"/>
      <configSetting altId="debug0.tck.p300" configurationId="debug0.tck"/>
      <configSetting altId="debug0.tdi.p110" configurationId="debug0.tdi"/>
      <configSetting altId="debug0.tdo.p109" configurationId="debug0.tdo"/>
      <configSetting altId="debug0.tms.p108" configurationId="debug0.tms"/>
      <configSetting altId="p108.debug0.tms" configurationId="p108"/>
      <configSetting altId="p108.gpio_mode.gpio_mode_peripheral"
configurationId="p108.gpio_mode"/>
      <configSetting altId="p109.debug0.tdo" configurationId="p109"/>
      <configSetting altId="p109.gpio_mode.gpio_mode_peripheral"
configurationId="p109.gpio_mode"/>
      <configSetting altId="p110.debug0.tdi" configurationId="p110"/>
      <configSetting altId="p110.gpio_mode.gpio_mode_peripheral"
configurationId="p110.gpio_mode"/>
      <configSetting altId="p112.sci2.txd" configurationId="p112"/>
      <configSetting altId="p112.gpio_mode.gpio_mode_peripheral"
configurationId="p112.gpio_mode"/>
      <configSetting altId="p113.sci2.rxd" configurationId="p113"/>
      <configSetting altId="p113.gpio_mode.gpio_mode_peripheral"
configurationId="p113.gpio_mode"/>
      <configSetting altId="p300.debug0.tck" configurationId="p300"/>
      <configSetting altId="p300.gpio_mode.gpio_mode_peripheral"
configurationId="p300.gpio_mode"/>
      <configSetting altId="p410.sci0.rxd" configurationId="p410"/>
      <configSetting altId="p410.gpio_mode.gpio_mode_peripheral"
configurationId="p410.gpio_mode"/>
      <configSetting altId="p411.sci0.txd" configurationId="p411"/>
      <configSetting altId="p411.gpio_mode.gpio_mode_peripheral"
configurationId="p411.gpio_mode"/>
```

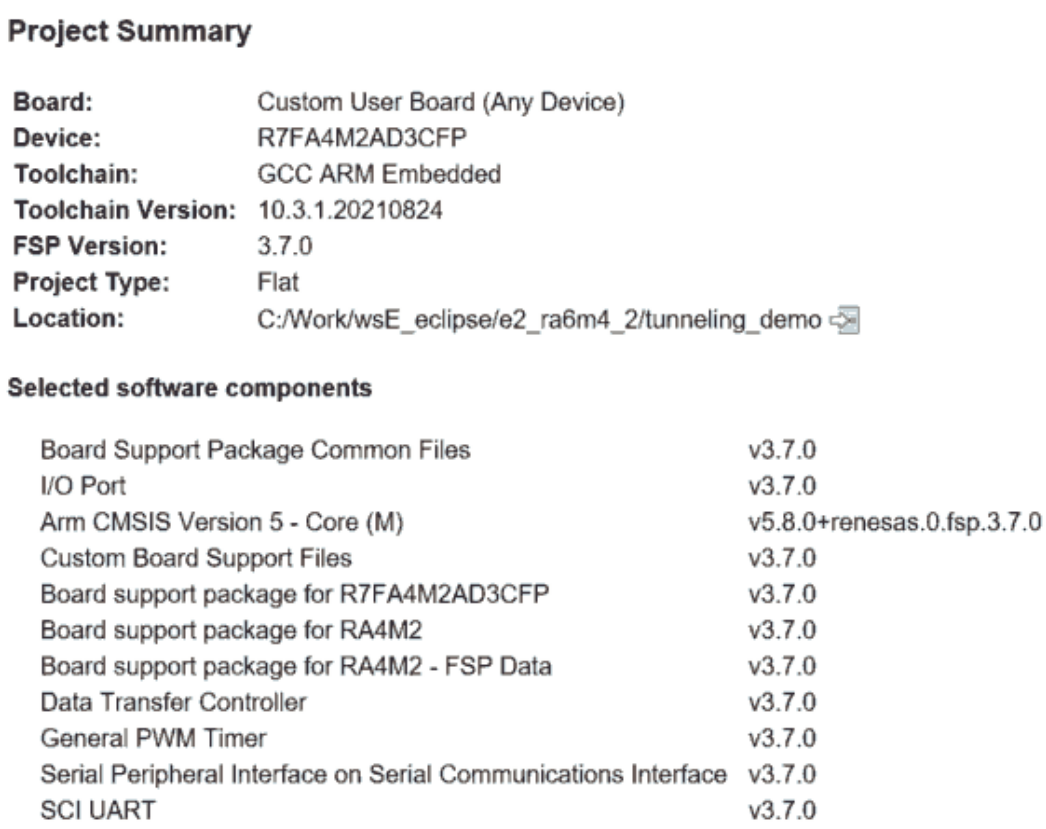
```
      <configSetting altId="p412.sci0.sck" configurationId="p412"/>
      <configSetting altId="p412.gpio_mode.gpio_mode_peripheral"
configurationId="p412.gpio_mode"/>
      <configSetting altId="p413.output.high" configurationId="p413"/>
      <configSetting altId="p413.gpio_mode.gpio_mode_out.high"
configurationId="p413.gpio_mode"/>
      <configSetting altId="p414.input" configurationId="p414"/>
      <configSetting altId="p414.gpio_mode.gpio_mode_in"
configurationId="p414.gpio_mode"/>
      <configSetting altId="sci0.mode.spi.b" configurationId="sci0.mode"/>
      <configSetting altId="sci0.pairing.b" configurationId="sci0.pairing"/>
      <configSetting altId="sci0.rxd.p410" configurationId="sci0.rxd"/>
      <configSetting altId="sci0.sck.p412" configurationId="sci0.sck"/>
      <configSetting altId="sci0.txd.p411" configurationId="sci0.txd"/>
      <configSetting altId="sci2.mode.asynchronous.free" configurationId="sci2.mode"/>
      <configSetting altId="sci2.rxd.p113" configurationId="sci2.rxd"/>
      <configSetting altId="sci2.txd.p112" configurationId="sci2.txd"/>
    </pincfg>
  </raPinConfiguration>
</raConfiguration>
```

In addition, the pin configuration file `R7FA4M2AD3CFP.pincfg` must be updated the same way with the following content.

```
<?xml version="1.0" encoding="utf-8"?>
<v1:pinSettings xmlns:v1="http://www.tasking.com/schema/pinsettings/v1.1">
  <v1:pinMappingsRef version="2.05" file="" />
  <v1:deviceSetting id="renesas.ra4m2_fp" pattern="R7FA4M2****FP">
    <v1:packageSetting id="renesas.100lqfp" />
  </v1:deviceSetting>
  <v1:configSetting configurationId="debug0.mode" altId="debug0.mode.jtag" />
  <v1:configSetting configurationId="p108.gpio_mode"
altId="p108.gpio_mode.gpio_mode_peripheral" />
  <v1:configSetting configurationId="p108" altId="p108.debug0.tms">
    <v1:connectionSetting altId="debug0.tms.p108" />
  </v1:configSetting>
  <v1:configSetting configurationId="debug0.tms" altId="debug0.tms.p108">
    <v1:connectionSetting altId="p108.debug0.tms" />
  </v1:configSetting>
  <v1:configSetting configurationId="p109.gpio_mode"
altId="p109.gpio_mode.gpio_mode_peripheral" />
  <v1:configSetting configurationId="p109" altId="p109.debug0.tdo">
    <v1:connectionSetting altId="debug0.tdo.p109" />
  </v1:configSetting>
  <v1:configSetting configurationId="debug0.tdo" altId="debug0.tdo.p109">
    <v1:connectionSetting altId="p109.debug0.tdo" />
  </v1:configSetting>
  <v1:configSetting configurationId="p110.gpio_mode"
altId="p110.gpio_mode.gpio_mode_peripheral" />
  <v1:configSetting configurationId="p110" altId="p110.debug0.tdi">
    <v1:connectionSetting altId="debug0.tdi.p110" />
  </v1:configSetting>
  <v1:configSetting configurationId="debug0.tdi" altId="debug0.tdi.p110">
    <v1:connectionSetting altId="p110.debug0.tdi" />
  </v1:configSetting>
  <v1:configSetting configurationId="p300.gpio_mode"
altId="p300.gpio_mode.gpio_mode_peripheral" />
  <v1:configSetting configurationId="p300" altId="p300.debug0.tck">
    <v1:connectionSetting altId="debug0.tck.p300" />
```

```
    </v1:configSetting>
    <v1:configSetting configurationId="debug0.tck" altId="debug0.tck.p300">
      <v1:connectionSetting altId="p300.debug0.tck" />
    </v1:configSetting>
  </v1:pinSettings>
```

After editing the files and opening the `configuration.xml` in the e2studio configuration editor, the following **Project Summary** should be seen:



Project Summary

Board:	Custom User Board (Any Device)
Device:	R7FA4M2AD3CFP
Toolchain:	GCC ARM Embedded
Toolchain Version:	10.3.1.20210824
FSP Version:	3.7.0
Project Type:	Flat
Location:	C:/Work/wsE_eclipse/e2_ra6m4_2/tunneling_demo

Selected software components

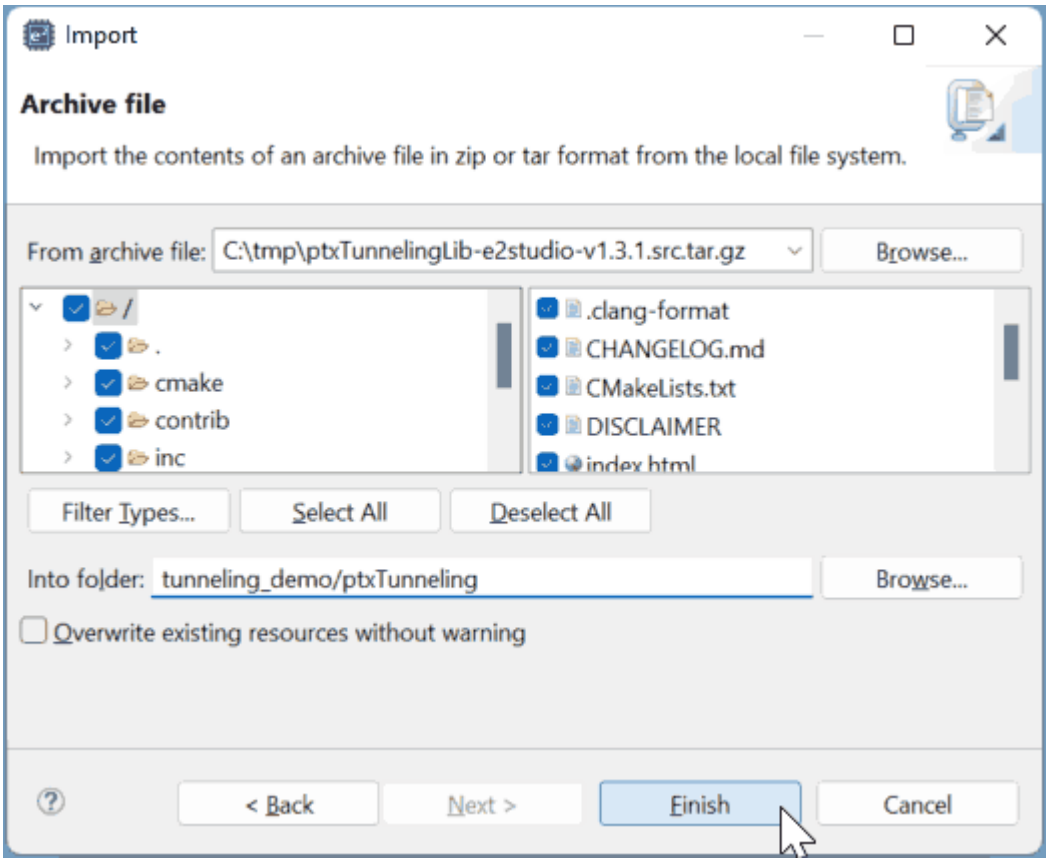
Board Support Package Common Files	v3.7.0
I/O Port	v3.7.0
Arm CMSIS Version 5 - Core (M)	v5.8.0+renesas.0.fsp.3.7.0
Custom Board Support Files	v3.7.0
Board support package for R7FA4M2AD3CFP	v3.7.0
Board support package for RA4M2	v3.7.0
Board support package for RA4M2 - FSP Data	v3.7.0
Data Transfer Controller	v3.7.0
General PWM Timer	v3.7.0
Serial Peripheral Interface on Serial Communications Interface	v3.7.0
SCI UART	v3.7.0

The source code describing the modules can now be generated by clicking on the **Generate Project Content** in the top-right corner.

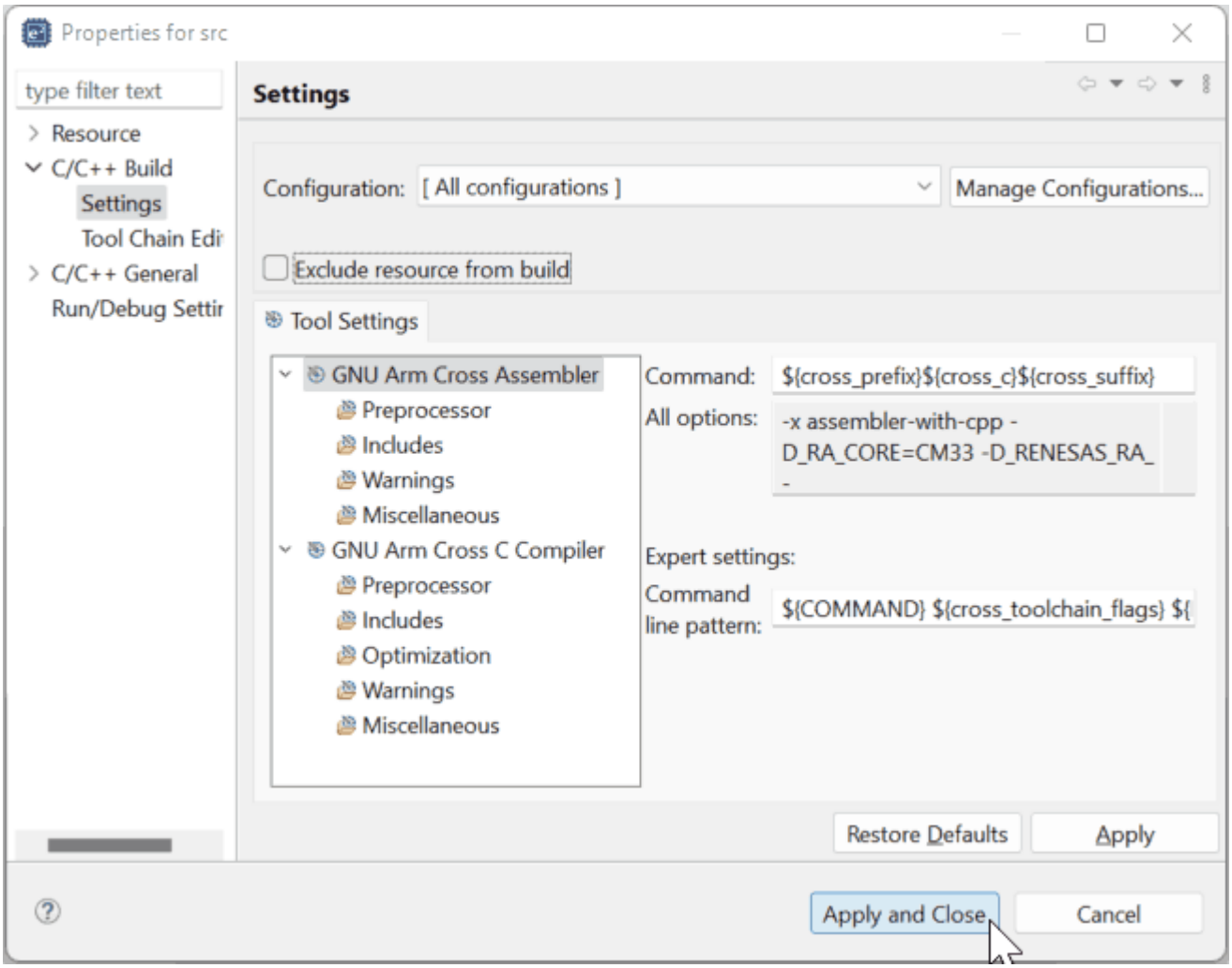
2.3 Importing the Library

There is no difference whether the source or the precompiled package is being used: the library archive must be imported to the project using **File > Import... > Archive File** option.

To keep the folder structure clean, the library will be imported to the `ptxTunneling` subfolder by appending the default location displayed in the following figure.

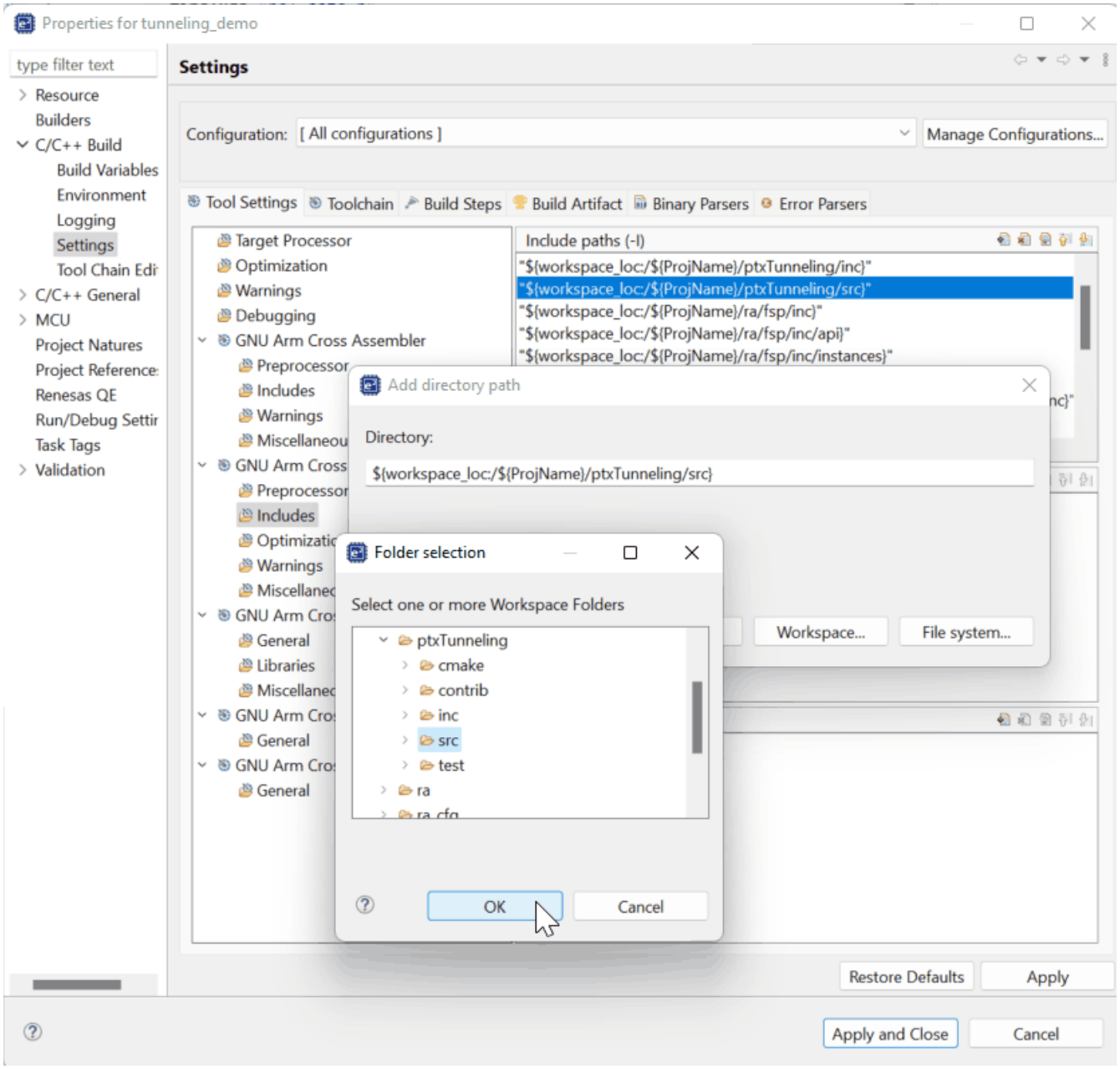


In the case where the library will be used in source form, its `src` folder must be included in the build. This can be done by opening the context menu with a right-mouse click on the folder name in the **Project Explorer** and selecting **Properties**. The checkbox **Exclude resource from build** found in *C/C++ Build* > *Settings* must be unchecked, as displayed in the following figure.



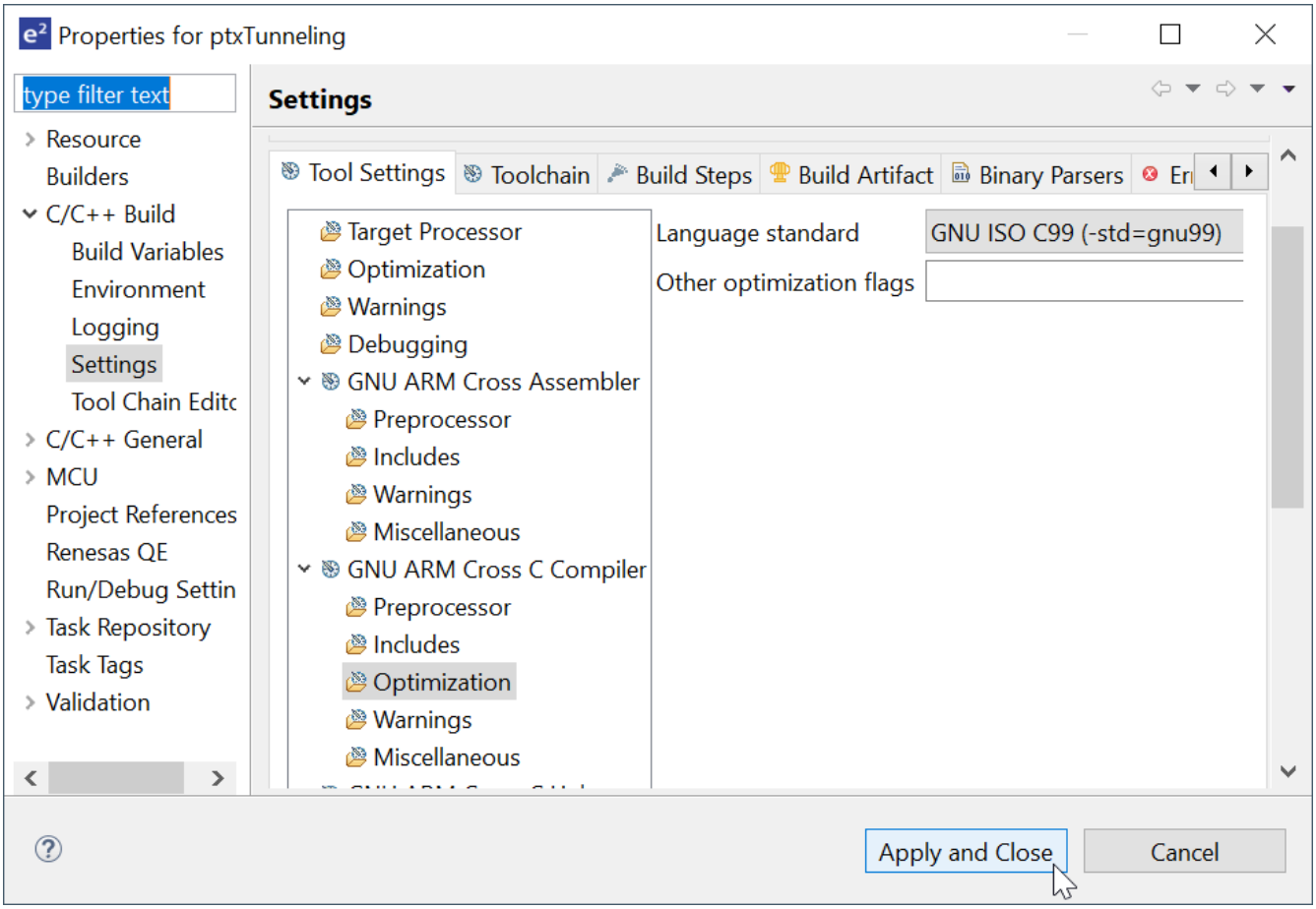
2.3.1. Adding the Include Path

In order for the compiler to find the header (.h) files containing the API functions, the library folder `inc` (and also `src` if using the source package variant) must be added to the list of user-defined include directories. This can be done by navigating to **Project > C/C++ Build > Settings > Tool Settings > GNU ARM Cross Compiler > Includes**. Then, click on the **Add...** button on the right side of the small toolbar and use the **Workspace** button in the popup window to locate the folder.



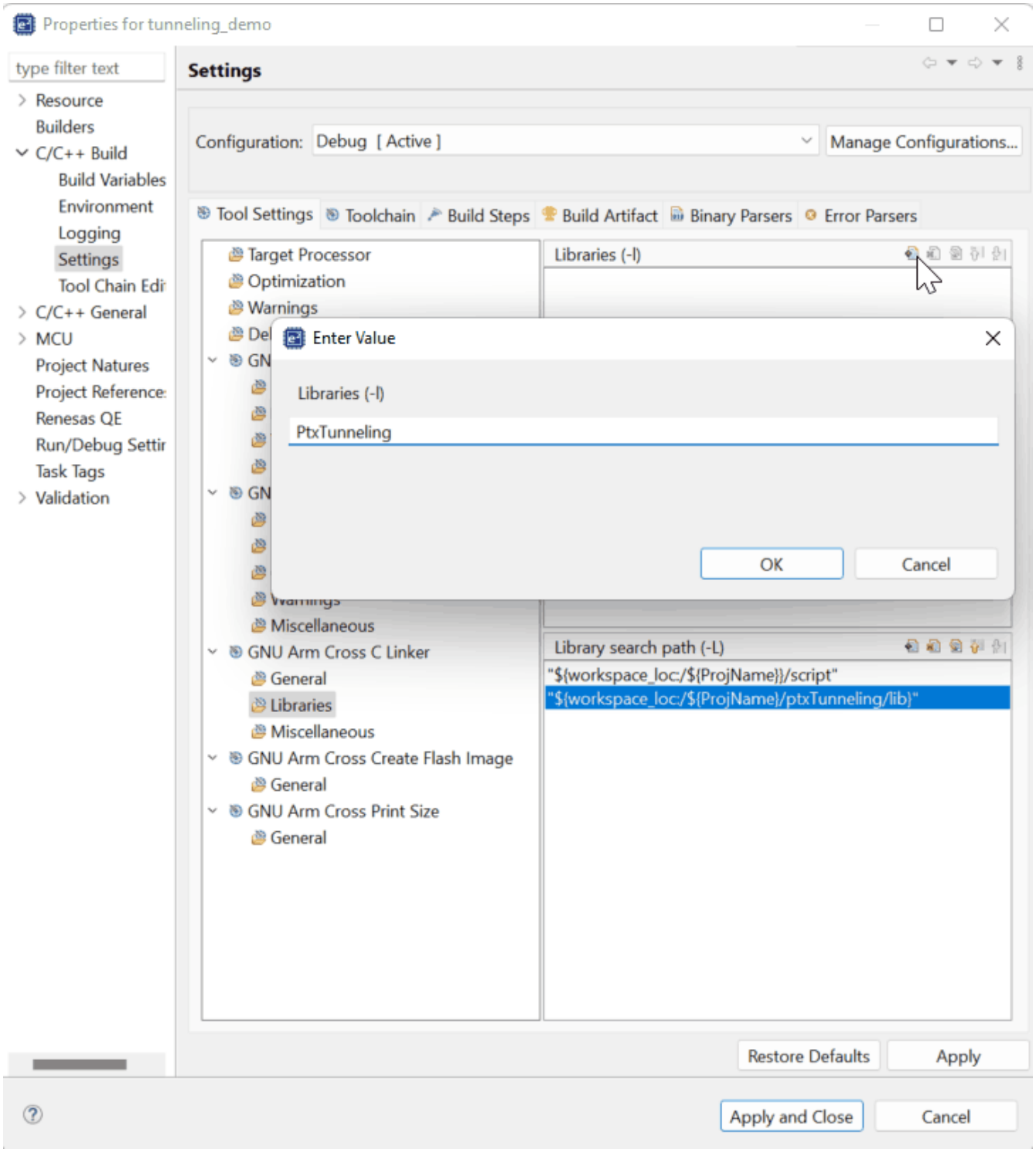
2.3.2. Setting the Language Standard

This step is required only if you use the source package. Ensure that the **GNU ISO C99** is selected among the **GNU ARM Cross C Compile > Optimization** settings. Since there is the `typeof()` statement used in the source code, the project must be compiled with the GNU extensions enabled; otherwise, compilation will fail.



2.3.3. Adding the Library File

This step is required only if you are working with the precompiled binary package. Since there is no source code to be compiled, the linker must be able to find the functions in the library. In the same dialog window, changing to **GNU ARM Cross C Linker** section, the `PtxTunneling/lib` folder can be added to the list of folders (lower pane) where the compiler is looking for external libraries. Additionally, the exact library needs also to be specified (upper pane) by its name `PtxTunneling`. From this the compiler will automatically find the static library file `libPtxTunneling.a`.



2.4 Implement the HAL

The library functions cannot access the underlying hardware or software resources. They require access to the Hardware Abstraction Layer (HAL), which then performs the requested action. Since this layer depends on the specific hardware configuration, it must be implemented for the exact setup.

The PtxTunneling library includes the header file ptx_tunneling_hal.h, which contains all the functions that must be provided by the host platform.

For the current case, there should be the file `src/ptx_tunneling_hal.c` created in the project folder with the following content.

```
/*
    ---------------------------------------------------------------
    SPDX-License-Identifier: BSD-3-Clause

    Copyright (c) 2024, Renesas Electronics Corporation and/or its affiliates


    Redistribution and use in source and binary forms, with or without modification,
    are permitted provided that the following conditions are met:

    1. Redistributions of source code must retain the above copyright notice, this
       list of conditions and the following disclaimer.

    2. Redistributions in binary form must reproduce the above copyright notice, this
list of
       conditions and the following disclaimer in the documentation and/or other
       materials provided with the distribution.

    3. Neither the name of Renesas nor the names of its
       contributors may be used to endorse or promote products derived from this
       software without specific prior written permission.



    THIS SOFTWARE IS PROVIDED BY Renesas "AS IS" AND ANY EXPRESS
    OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
    OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE ARE
    DISCLAIMED. IN NO EVENT SHALL RENESAS OR CONTRIBUTORS BE
    LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
    CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
    GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
    HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
    LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
    OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
    ---------------------------------------------------------------

    Project     : PtxTunneling
    Module      : HAL
    File        : ptx_tunneling_hal.c

    Description : Implementation of HAL for tunneling
*/

#include "ptx_tunneling_hal.h"
#include "ptx_tunneling.h"

#include "hal_data.h"

#define OFFSET_CMD_LENGTH_BYTE 1U
#define COMMS_MAX_MESSAGE_LENGTH 280U
#define COMMS_HEADER_SIZE 2U
#define CMD_CODE_TUNNELING_MSG 0x55

static uint8_t uartRxBuf[2048];
static uint16_t readPos = 0;
static uint16_t writePos = 0;
static uint8_t rx[COMMS_MAX_MESSAGE_LENGTH];
static uint8_t tx[COMMS_MAX_MESSAGE_LENGTH];
static uint16_t rxi = 0;
static volatile bool txDone;
static volatile bool uartComplete;
```

```
uint32_t hal_TimerGetTime(void);
void tunneling_init();
void tunneling_demo();

uint32_t hal_TimerGetTime(void)
{
    timer_status_t time;
    g_timer0.p_api->statusGet(g_timer0.p_ctrl, &time);
    return time.counter;
}

void uart_callback(uart_callback_args_t *arg)
{
    // read 1 byte
    if (arg->event == UART_EVENT_RX_CHAR)
    {
        // receive the header and the length byte
        rx[rxi] = (uint8_t)arg->data;
        if (rxi != 0 || rx[rxi] == CMD_CODE_TUNNELING_MSG)
        { // if there are some invalid data received while waiting for sync byte, just
discard them
            rxi++;
        }

        if (rxi >= COMMS_HEADER_SIZE)
        {
            uint16_t packLen = COMMS_HEADER_SIZE +
                               (rx[OFFSET_CMD_LENGTH_BYTE] == 0 ? 256 :
rx[OFFSET_CMD_LENGTH_BYTE]);
            if (rxi >= packLen)
            {
                // whole packet has been received
                memcpy(uartRxBuf + writePos, rx, rxi);
                writePos += rxi;
                rxi = 0;
            }
        }
    }
    else if (arg->event == UART_EVENT_TX_COMPLETE)
    {
        uartComplete = true;
    }
}

bool ptxTunneling_GPIO_IsIrqPinAsserted(ptxHal_t *context)
{
    FSP_PARAMETER_NOT_USED(context);

    bsp_io_level_t pinLevel = 0;

    g_ioport.p_api->pinRead(g_ioport.p_ctrl, PTX_PLAT_SPI_IRQ_PIN, &pinLevel);

    return (bool)pinLevel;
}

int ptxTunneling_UART_rxLength(ptxHal_t *context)
{
    FSP_PARAMETER_NOT_USED(context);
    __disable_irq();
    const uint16_t count = writePos - readPos;
```

```
    __enable_irq();
    return count;
}

int ptxTunneling_UART_read(ptxHal_t *context, uint8_t *buf, unsigned int len)
{
    FSP_PARAMETER_NOT_USED(context);
    assert(len < sizeof(uartRxBuf));
    __disable_irq();
    int readCount = writePos - readPos;
    if (readCount > (int)len)
        readCount = (int)len;

    memcpy(buf, uartRxBuf + readPos, (size_t)readCount);

    readPos = (uint16_t)(readPos + readCount);
    if (readPos == writePos)
    {
        readPos = 0;
        writePos = 0;
    }
    __enable_irq();
    return readCount;
}

int ptxTunneling_UART_write(ptxHal_t *context, const uint8_t *buf, unsigned int len)
{
    FSP_PARAMETER_NOT_USED(context);
    if (len)
    {
        while (!uartComplete)
        {
            R_BSP_SoftwareDelay(100, BSP_DELAY_UNITS_MICROSECONDS);
        }
        uartComplete = false;
        memcpy(tx, buf, len);

        g_uart0.p_api->write(g_uart0.p_ctrl, tx, len);
    }
    return 0;
}

void ptxTunneling_Timer_stopwatchStart(ptxHal_t *context, ptxTimeDiff_t *startVal)
{
    FSP_PARAMETER_NOT_USED(context);

    *startVal = hal_TimerGetTime();
}

void ptxTunneling_Timer_stopwatchStop(ptxHal_t *context, ptxTimeDiff_t *startStopVal)
{
    FSP_PARAMETER_NOT_USED(context);

    *startStopVal = hal_TimerGetTime() - *startStopVal;
}

void ptxTunneling_Timer_ThreadSleep(ptxHal_t *context, uint32_t msSleep)
{
    FSP_PARAMETER_NOT_USED(context);

    R_BSP_SoftwareDelay(msSleep, BSP_DELAY_UNITS_MILLISECONDS);
```

```
}

int ptxTunneling_SPI_trx(ptxHal_t *context, uint8_t *const txBuf[], const size_t
txLen[],
    size_t numBuffers, uint8_t *rxBuf, size_t *rxLen)
{
    FSP_PARAMETER_NOT_USED(context);

    bool status = true;
    if ((NULL != txBuf) && (NULL != txLen))
    {
        /* At this point the SPI transfer operation is triggered */
        g_ioport.p_api->pinWrite(g_ioport.p_ctrl, PTX_PLAT_SPI_CS_PIN,
BSP_IO_LEVEL_LOW);

        uint8_t index = 0;
        while ((true == status) && (index < numBuffers))
        {
            if ((txBuf[index] != NULL) && (txLen[index] > 0) && (rxBuf != NULL))
            {
                txDone = false;
                if (g_spi0.p_api->writeRead(
                        g_spi0.p_ctrl, txBuf[index], rxBuf, txLen[index],
SPI_BIT_WIDTH_8_BITS))
                {
                    status = false;
                }
                while (!txDone && status)
                    R_BSP_SoftwareDelay(100, BSP_DELAY_UNITS_MICROSECONDS);
            }
            else
            {
                status = false;
            }
            index++;
        }
        /* Tx part of the overall transaction. */
    }
    else
    {
        /* Let's see if there is something to read. */
        if ((NULL != rxBuf) && (NULL != rxLen))
        {
            txDone = false;
            if (g_spi0.p_api->read(g_spi0.p_ctrl, rxBuf, (uint32_t)*rxLen,
SPI_BIT_WIDTH_8_BITS))
            {
                status = false;
            }
            while (!txDone && status)
                R_BSP_SoftwareDelay(100, BSP_DELAY_UNITS_MICROSECONDS);
        }
        else
        {
            status = false;
        }
    }
    /* In any case, at this point the SPI transfer operation is finished */
    g_ioport.p_api->pinWrite(g_ioport.p_ctrl, PTX_PLAT_SPI_CS_PIN, BSP_IO_LEVEL_HIGH);
    return (!status);
}
```

```
void sci_spi_callback(spi_callback_args_t *p_args)
{
    if (p_args->event == SPI_EVENT_TRANSFER_COMPLETE)
        txDone = true;
}

void ptxTunneling_NVIC_disableInterrupts()
{
    // disabling interrupts not needed in this sample, since the SPI/UART communication
uses neither
    // interrupts nor multithreading, therefore no race conditions can occur during
queue operations
}

void ptxTunneling_NVIC_enableInterrupts()
{
    // enabling interrupts not needed in this sample, since the SPI/UART communication
uses neither
    // interrupts nor multithreading, therefore no race conditions can occur during
queue operations
}

void tunneling_init()
{
    g_spi0.p_api->open(g_spi0.p_ctrl, g_spi0.p_cfg);
    g_uart0.p_api->open(g_uart0.p_ctrl, g_uart0.p_cfg);
    g_timer0.p_api->open(g_timer0.p_ctrl, g_timer0.p_cfg);
    g_timer0.p_api->start(g_timer0.p_ctrl);
    uartComplete = true;
    __enable_irq();
}

void tunneling_demo()
{
    ptxTunneling_init();
    tunneling_init();
    while (true)
    {
        ptxTunneling_poll(NULL);
    }
}
```

This example implementation is specific to the Renesas EK-RA4M2 development board. It is not guaranteed to work on any other hardware.

2.5 Implementing the Main Function

The firmware will provide the tunneling functionality by calling the library's superloop function, the `ptxTunneling_poll()`, after initializing the modules. This function performs the data processing and translation and also drives the SPI communication. The e2studio has already generated the default user-entry source file `hal_entry.c`. Please update this file according to the following code.

```
void tunneling_demo();

void hal_entry(void)
{
    tunneling_demo();
}
```

This code change is responsible for involving the main loop of the tunneling firmware once the MCU initialization is finished and the user-code is started.

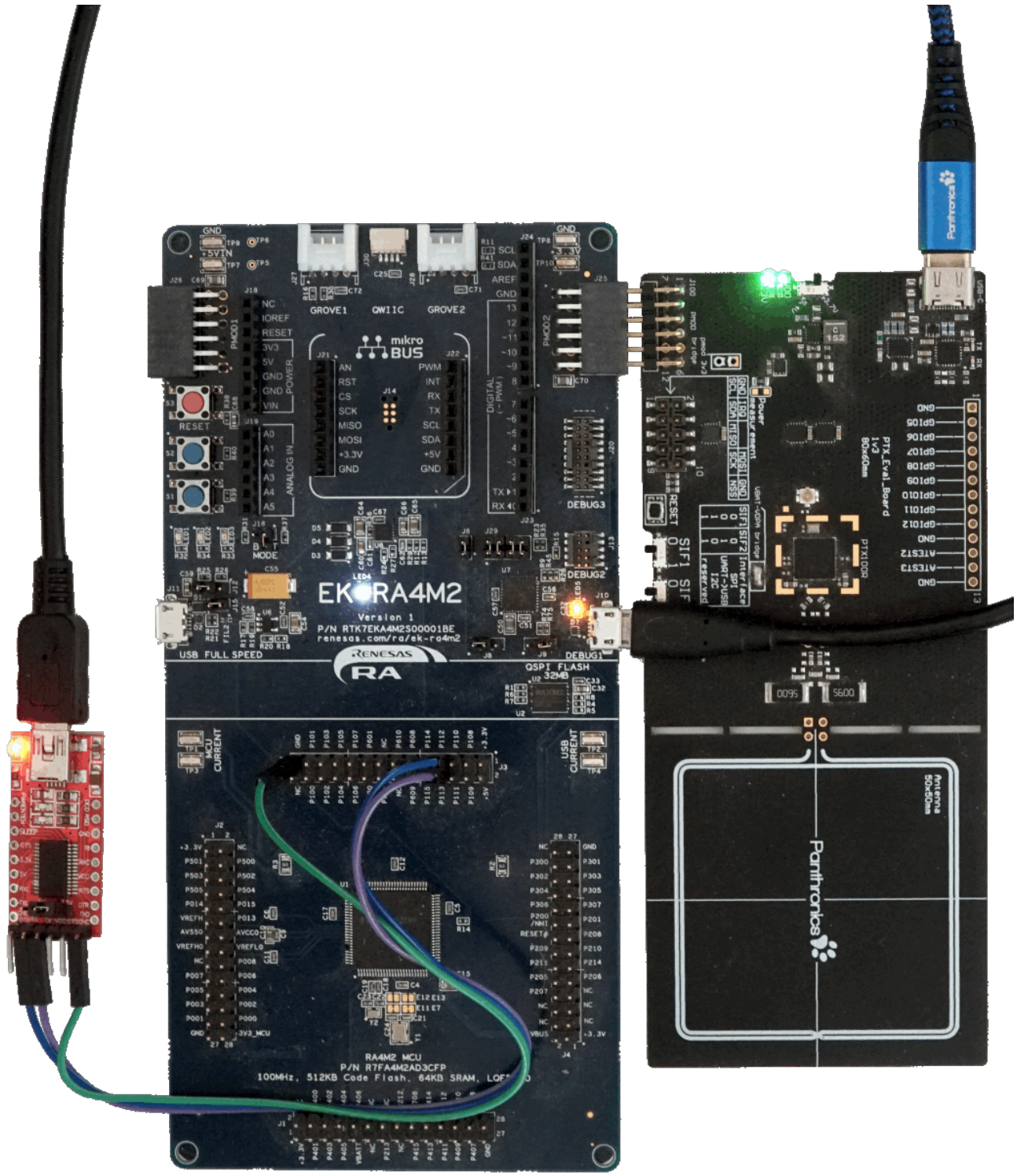
2.6 Building the Firmware

After all the source file is created, the project can be built with the **Project > Build Project**. Once the build process is finished successfully, a table similar to the following will display with the footprint sizes.

```
arm-none-eabi-size --format=berkeley "tunneling_demo.elf"
   text    data     bss     dec     hex filename
  14588      80    8048   22716    58bc tunneling_demo.elf
```

3. Prepare the Hardware

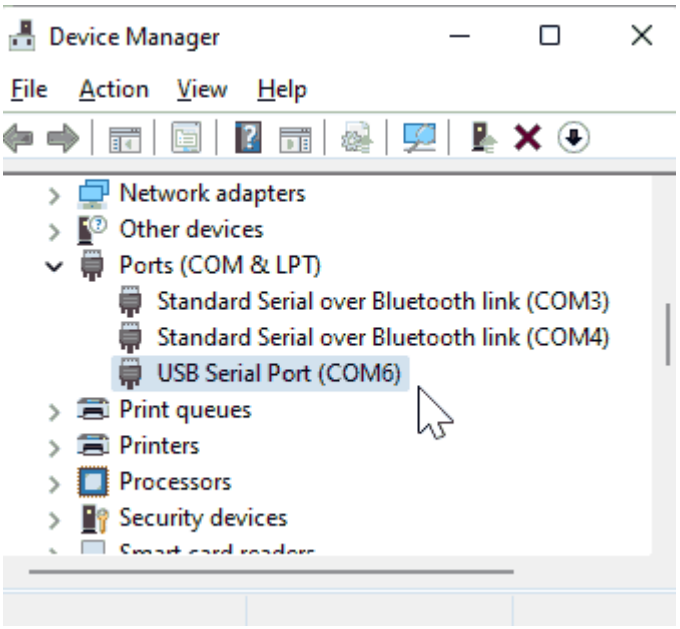
To demonstrate the usage of the tunneling functionality, a PTX100R evaluation board must be connected to the Renesas evaluation board through the **PMOD2** connector.



Before powering up from the USB-C port, it is important to remove the jumper "pmod 3v3 bridge" and set both of the **SIF1** and **SIF2** switches on the PTX100R evaluation board to **"0"** state to select the SPI communication protocol.

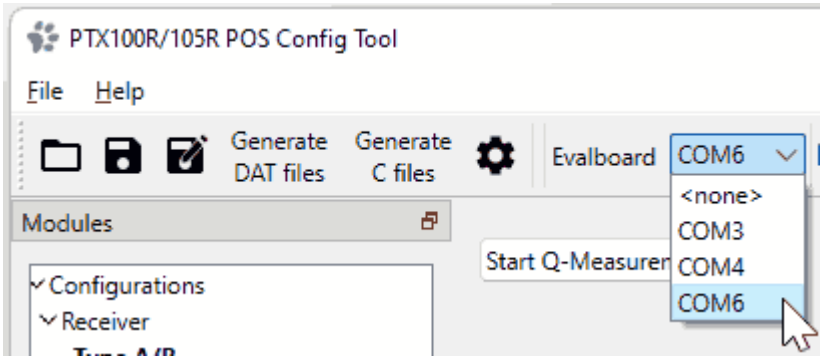
The Renesas board must be connected to the host PC by a micro-USB cable using the "Debug1" connector. This way, the firmware can be easily flashed and started with the **Run > Run** command from the menu.

Once the firmware starts successfully, it will listen for incoming commands on the on-chip UART using pins **P112** and **P113**. To use the tunneling functionality, an external USB/UART converter should be connected to those pins. Windows will show a notification about the connected USB/UART device with the communication port name indicated. If the notification is missing, opening the **Device Manager** will reveal the newly configured com port.

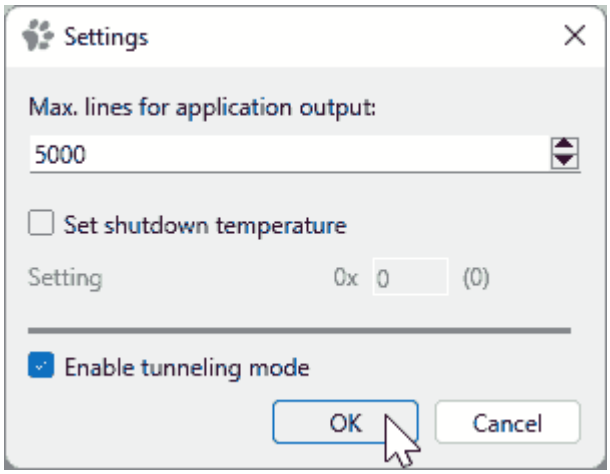


4. Using the Tunneling Feature

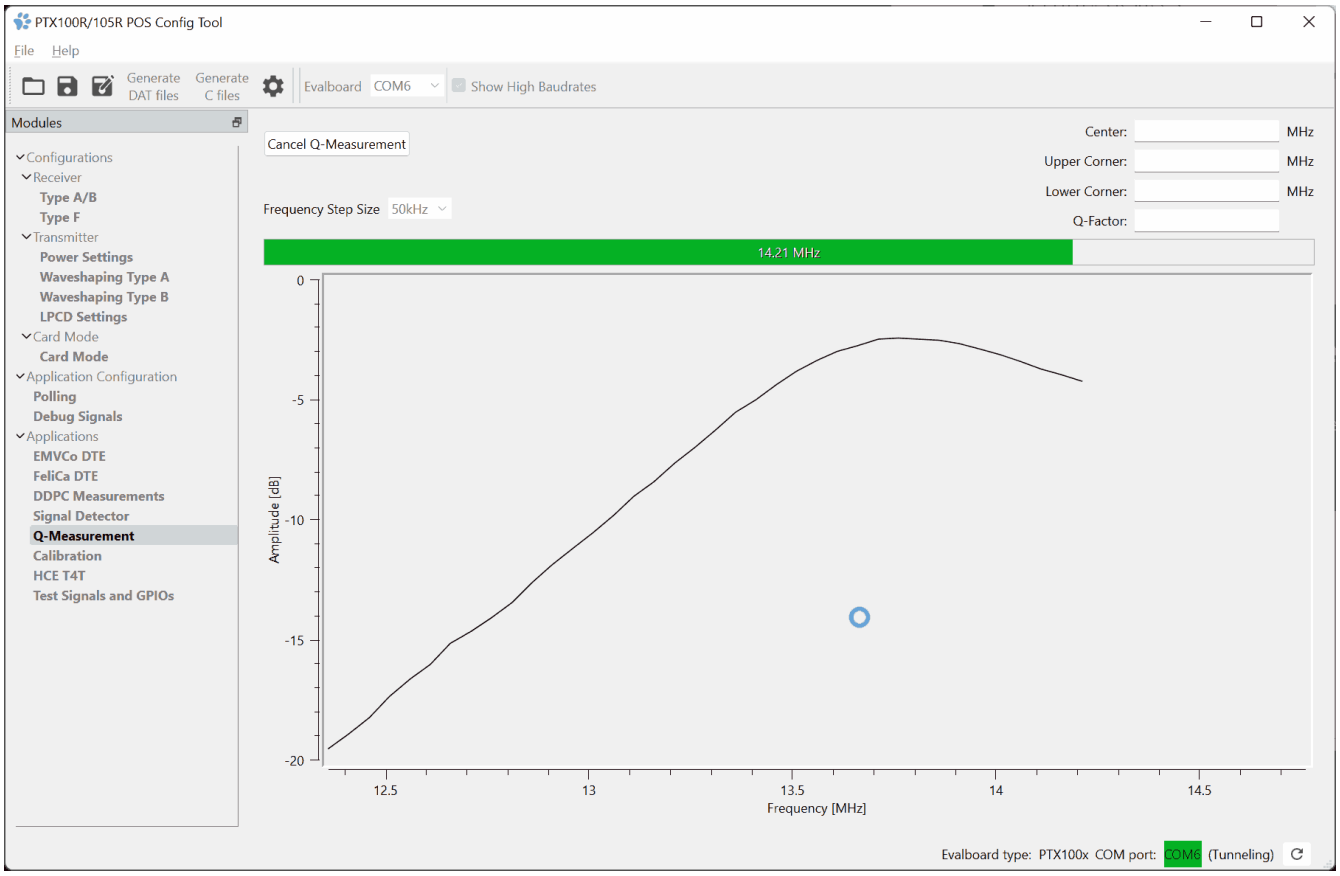
In order to use the tunneling functionality, the **PTX1xxR * Config Tool** must be started and configured to use the USB communication port by selecting the correct entry in the dropdown list in toolbar.



The config tool is using the hardware flow control functionality of the UART by default, which is not provided in this case. As a result, the **tunneling mode** should be enabled in the **File > Settings** menu.



The configuration is ready; any config tool application (for example, Q-Measurement) started will now communicate with the PTX100R via the tunneling firmware.



5. Revision History

Revision	Date	Description
1.01	Jun 18, 2024	Updated license text in ptx_tunneling_hal.c file.
1.00	Jan 16, 2024	Initial release.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.