

---

# **Silica JenNode - JN5168 demobaord Documentation**

*Release 0*

**Silica**

**Mar 20, 2017**



<b>1</b>	<b>INTRODUCTION</b>	<b>3</b>
1.1	Installing Jennic Development Suite	4
1.1.1	First step	4
1.1.2	Installing Jennic Toolchain JN-SW-4041	5
1.1.3	Installing JN-AN-4065 library	8
1.1.4	Starting Jennic Toolchain	9
1.1.5	Installing Flash GUI Tool Programmer	11
1.1.6	Upgrading Flash GUI tool (for JN5168)	16
1.2	<b>WINXP TROUBLESHOOTING</b>	17
1.2.1	INSTALLING DRIVER FOR USB DONGLE	17
1.2.2	CYGWIN1.DLL TROUBLES WITH WINXP	18
1.3	Quick start guide	19
1.3.1	Hardware requirements	19
1.3.2	Software requirements	19
1.3.3	Hardware setup	20
1.3.4	HyperTerminal settings	25
1.3.5	Rialto Board FW installation & setup	26
1.3.6	Programming Rialto Board with Flash GUI Tool	38
1.4	<b>“Ready to run” Rialto Kit</b>	46
1.4.1	Working with Rialto board	46
1.5	Firmware specification	50
1.5.1	Firmware overview	50
1.5.2	Firmware restrictions	51
1.5.3	Main firmware features	51
1.5.4	Project structure	52
1.5.5	Firmware & documents download	52
1.6	Serial Monitor guide	53
1.6.1	Using serial monitor	53
1.6.2	Coordinator Serial Monitor	53
1.6.3	End-Node Serial Monitor	64
1.7	Tip & Tricks	71
1.7.1	Stand-alone use of Rialto Boards	71
1.7.2	Serial channel details	71
1.7.3	Poll and Sleep settings	73
1.7.4	Led functions	73
1.7.5	Firmware debugging	74



**Version** 1.00

**Copyright** (C)2016 Avnet Silica company

**Date** 30 july 2013



# CHAPTER 1

---

## INTRODUCTION

---

Silica Rialto Board is useful system to evaluate a basic IEEE802.15.4 network using JN5168 wireless module.

Rialto board has a single built-in USB dongle. Simply switching serial channel from terminal application to programming software, you can perform two alternate functions:

- a - Evaluate network with simple serial monitor (using a PC terminal such as HyperTerminal)
- b - Program Firmware into JN5168 module (using JN-SW-4007-Flash-Programmer v1.8.9)

*If you want, you can use separate USB-TTL adapter connected with 6 pin strip on Rialto board. In this case, you can have two separate serial channel, first (USBdongle) for programming and second (6 pin strip) for Serial Monitor. See [Serial channel details](#)*

Rialto board is also designed as a plugin for SerizII board. **Rialto “ready to run” kit** is specially designed for quick use and fast evaluate network functionality. All you have to do is plug End-Node on SerizII board, plug Coordinator on PC USB port, start HyperTerminal and enjoy!!

With Rialto board you can evaluate the performance of new NXP SENS300/01 device. The SEN300/01 integrates one high-accuracy temperature sensor, four relative humidity sensors, six light sensors, a user-writable non-volatile memory and a 10-bit analog-to-digital converter. Rialto application reads out basic data from temperature, humidity and light sensors. For more information about this sensor, see at NXP official site [NXP official site](#)

Rialto board integrates a NTAG203F, NFC Forum Type 2 Tag compliant IC with 144 bytes user memory and field detection. This device is used on SerizII labs. See at official documentation page of NXP site [NTAG203F](#)

Firmware application was developed with JN-SW-4041-SDK-Toolchain-v.1.1 (Eclipse based).

You must also install JN-SW-4065-JN516x-JenNet-IP-SDK-v857 software library.

Firmware project is included in Rialto\_Jennic.zip file.

Reference guide are included into Install.zip and are:

- 1) JN-UG-3064 - Software Developer’s Kit Installation and User Guide
- 2) JN-UG-3007 - JN51xx Flash Programmer User Guide

Install.zip also include JN-SW-4041-SDK-Toolchain-v.1.1.exe and JN-SW-4065-JN516x-JenNet-IP-SDK-v857.exe to install developement suite.

For JN5168 modules programming, you must install FlashGUI programmer from JN-SW-4007 (included in Install.zip). Please, note that JN-UG-3007 refers to Flash GUI programmer version 1.8.4, and JN-SW-4047 contains new release 1.8.9 of Flash GUI programmer. See release note document included in JN-SW-4047 package. The release 1.8.9 must be installed for JN516x modules programming.

Install.zip and Rialto\_Jennic.zip files can be downloaded from Rialto section of Silica ArchiTech web page (registration needed)

**Installing Jennic Developement Suite** chapter will guide you through the basic steps of the installation procedure of Jennic Developement Suite

We suggest you to read first the Quick Start Guide to perform a correct Firmware setup.

*Quick start guide*

This guide explains how to use this application and provides an overview of on the structure of the project firmware

## Installing Jennic Developement Suite

**This small guide is intended to explain the main steps to properly install the Jennic SDK Toolchain.**

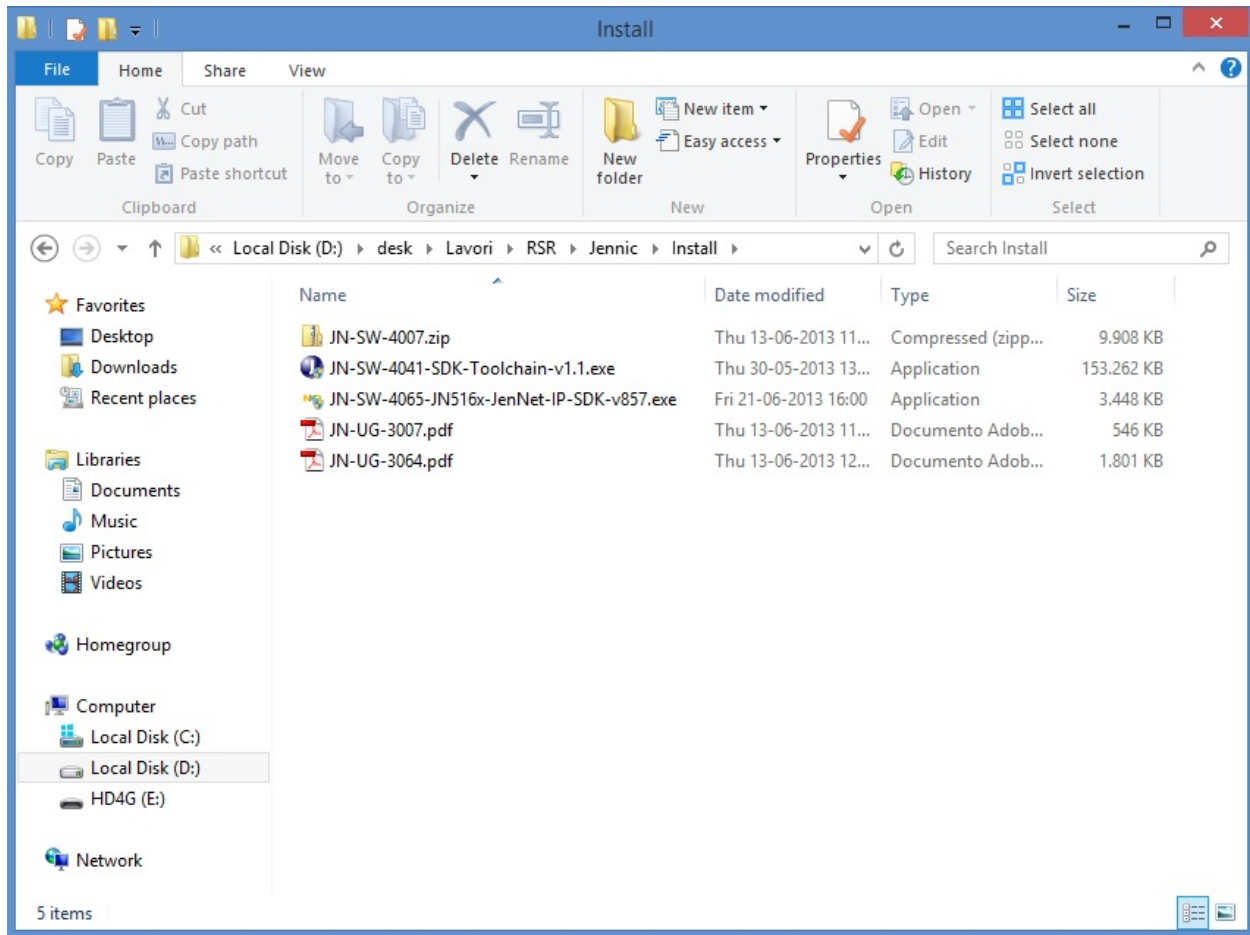
**As possibly not shown here, refer to the document JN-UG-3064.**

**All steps here described are performed with WINDOWS 7 OS**

### First step

Unpack the file 'Install.zip' that you have downloaded You must find these files:





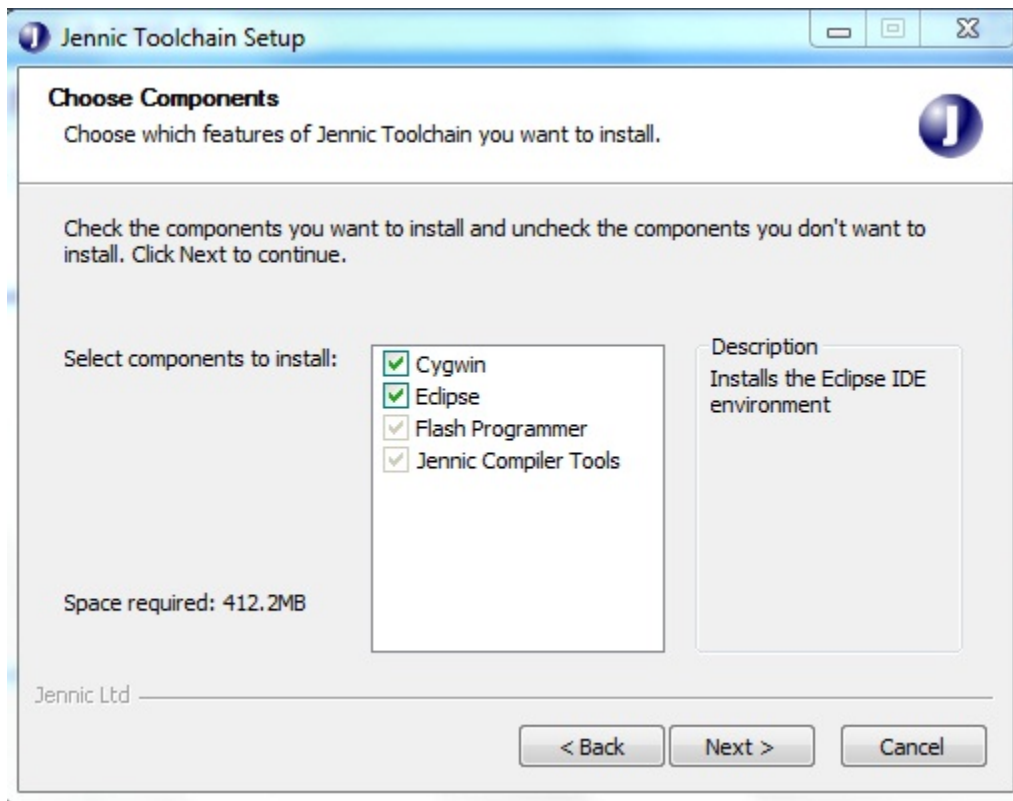
Then, you are ready for Jennic Development Suite setup.

## Installing Jennic Toolchain JN-SW-4041

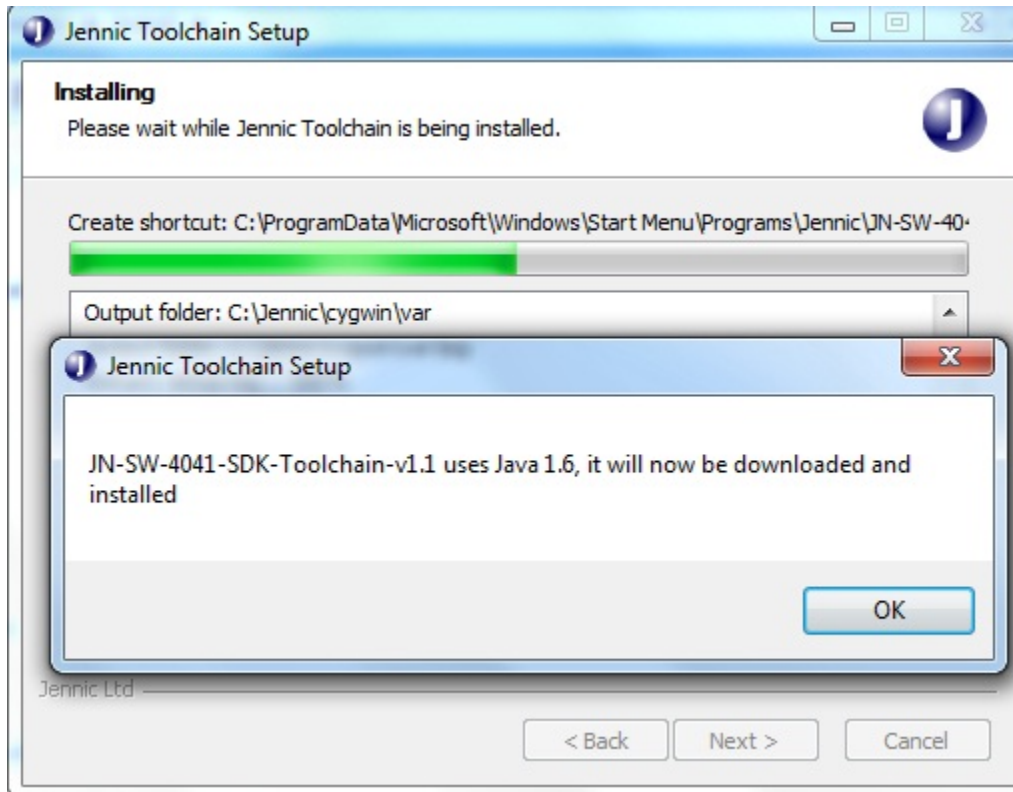
Double click on **JN-SW-4041-SDK-Toolchain-v1.1.exe**.

Leave unchanged all default setting in the following dialog box.

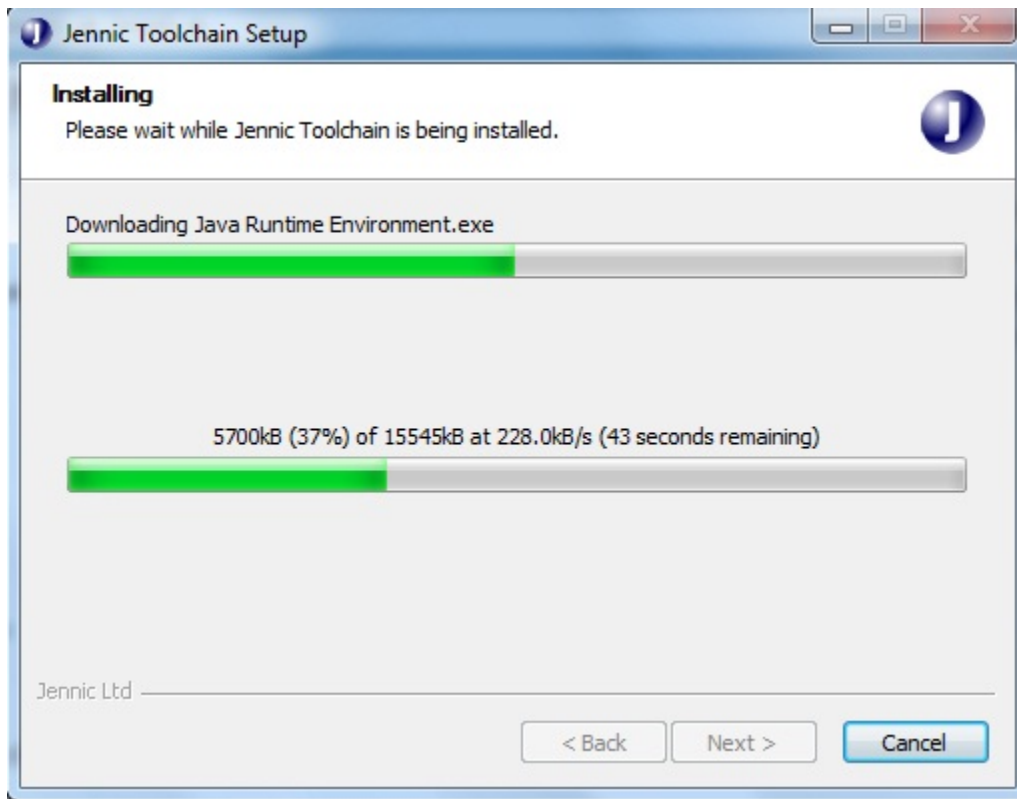
When appear verify checkbox selection as image below



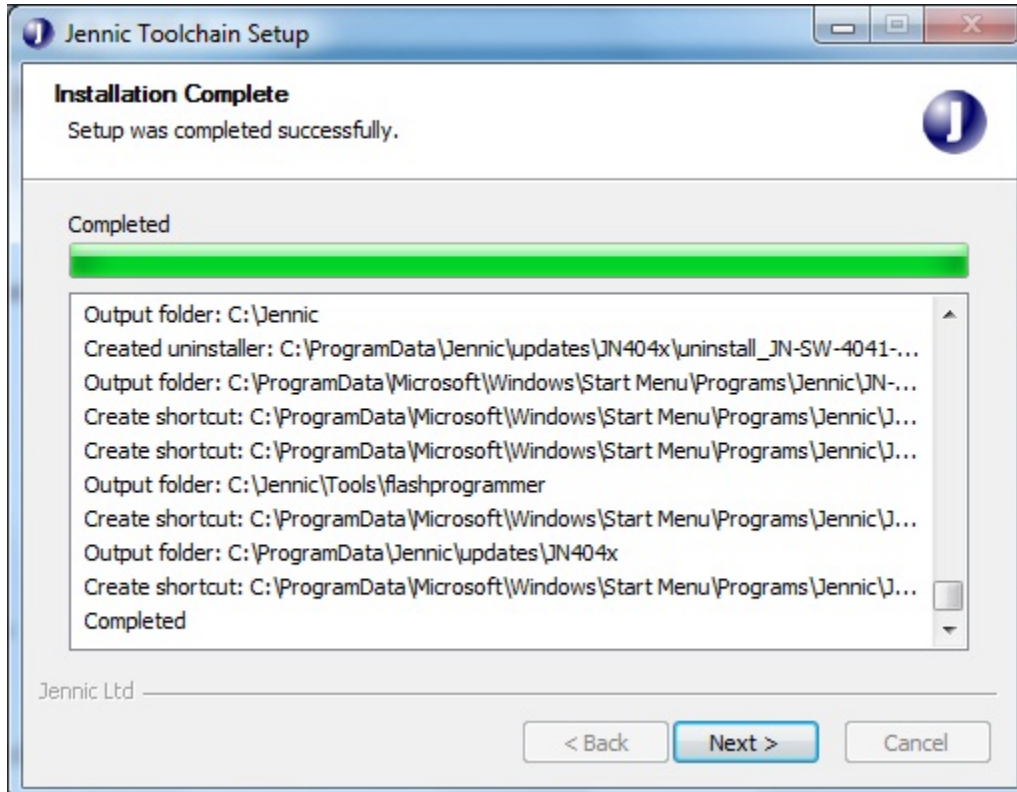
During installation, if you don't have Java Runtime on your PC, this popup will appear:



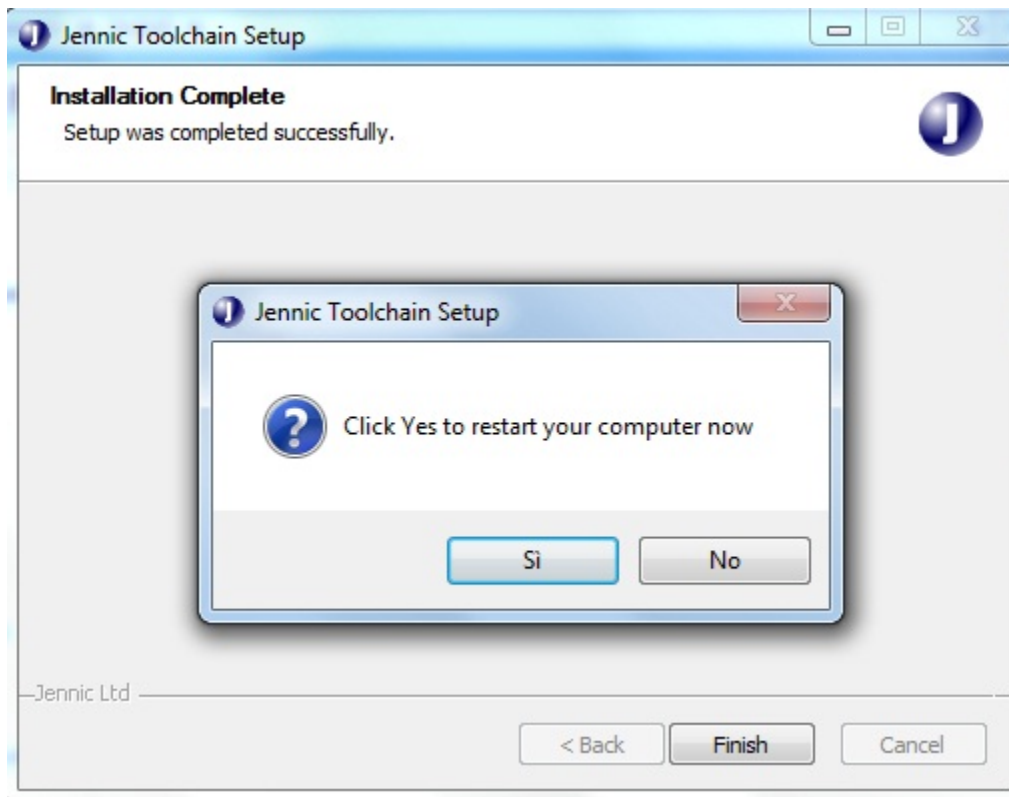
Click OK button and proceed following Java Installer instructions



When installation ends, the following will appear:



click “next” button, and setup will ask you to perform a system restart



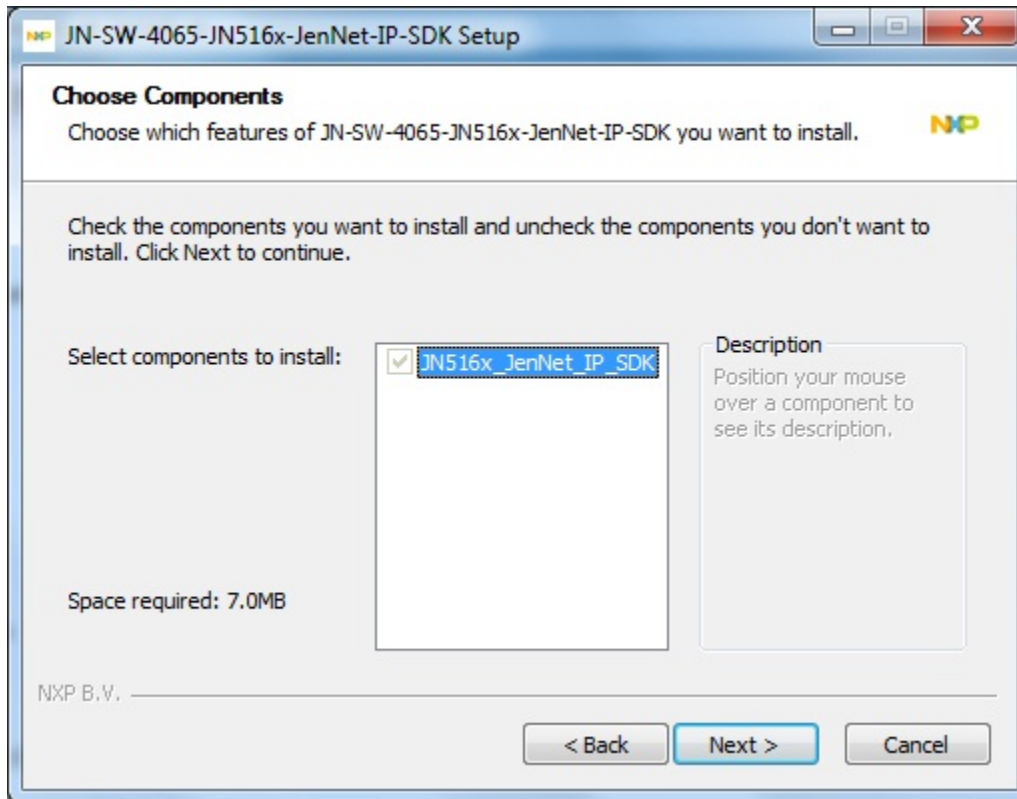
click “OK” and wait for restart

## Installing JN-AN-4065 library

Double click on **JN-SW-4065-JN516x-JenNet-IP-SDK-v857.exe**.

Leave unchanged all default setting in the following dialog box.

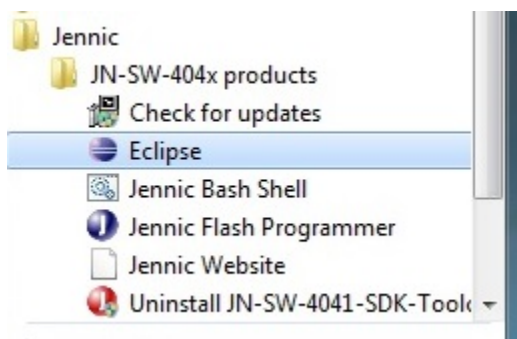
When appear verify checkbox selection as image below, then click “Next”



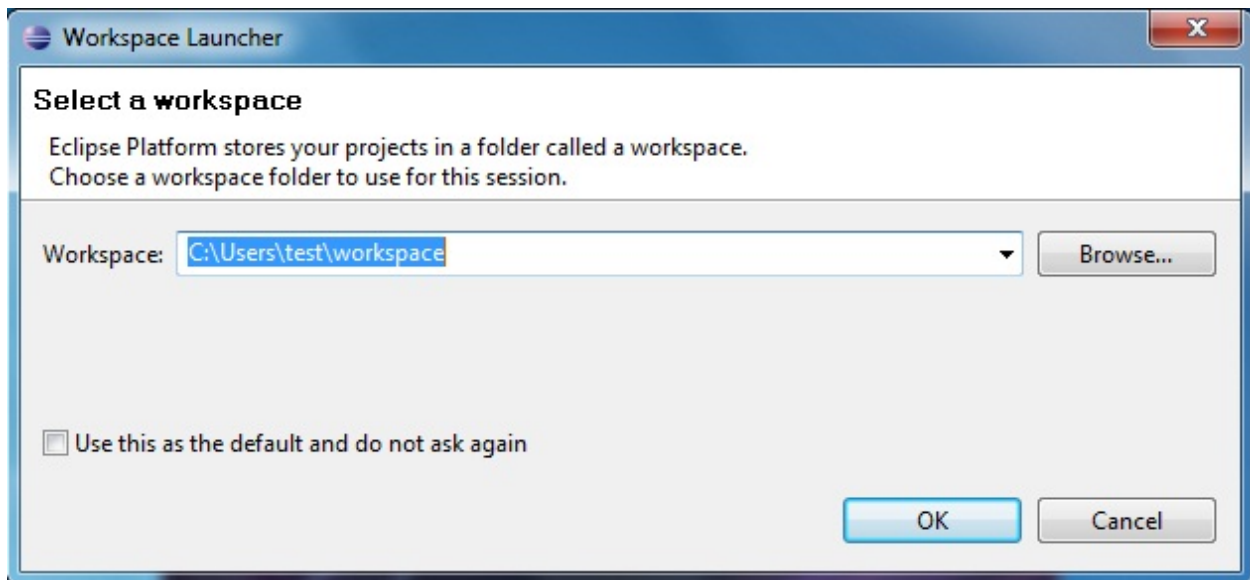
When library installation has finished, Jennic Toolchain is ready to start

## Starting Jennic Toolchain

Open Start and go to Jennic → Eclipse (as show in figure below)  
click on Eclipse to start program



Jennic Toolchain will ask for workspace. Leave unchanged and click “OK”

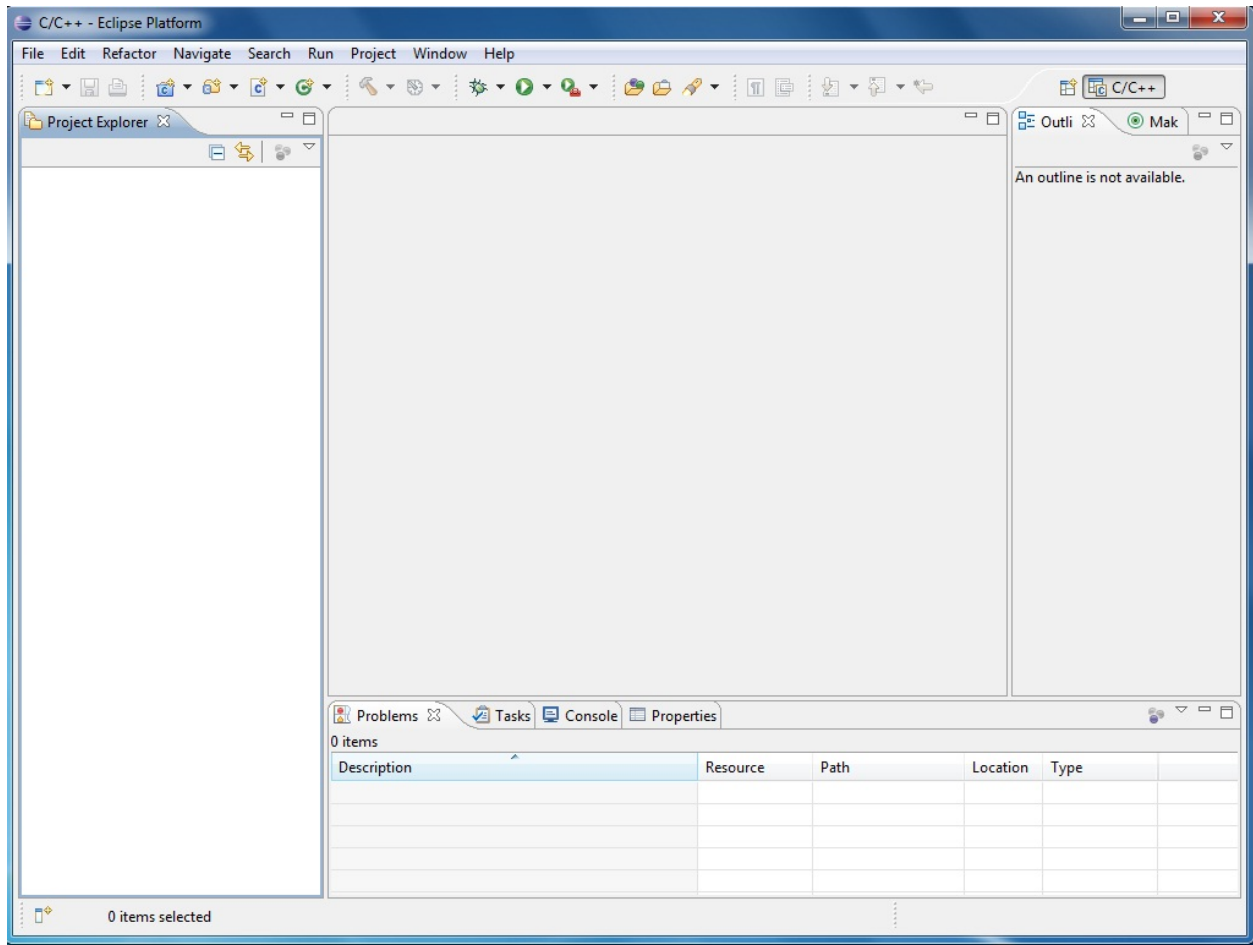


Jennic Eclipse will open Welcome tab.



Close Welcome tab by clicking on X red circled in figure above  
Now the Jennic Eclipse main window will appear

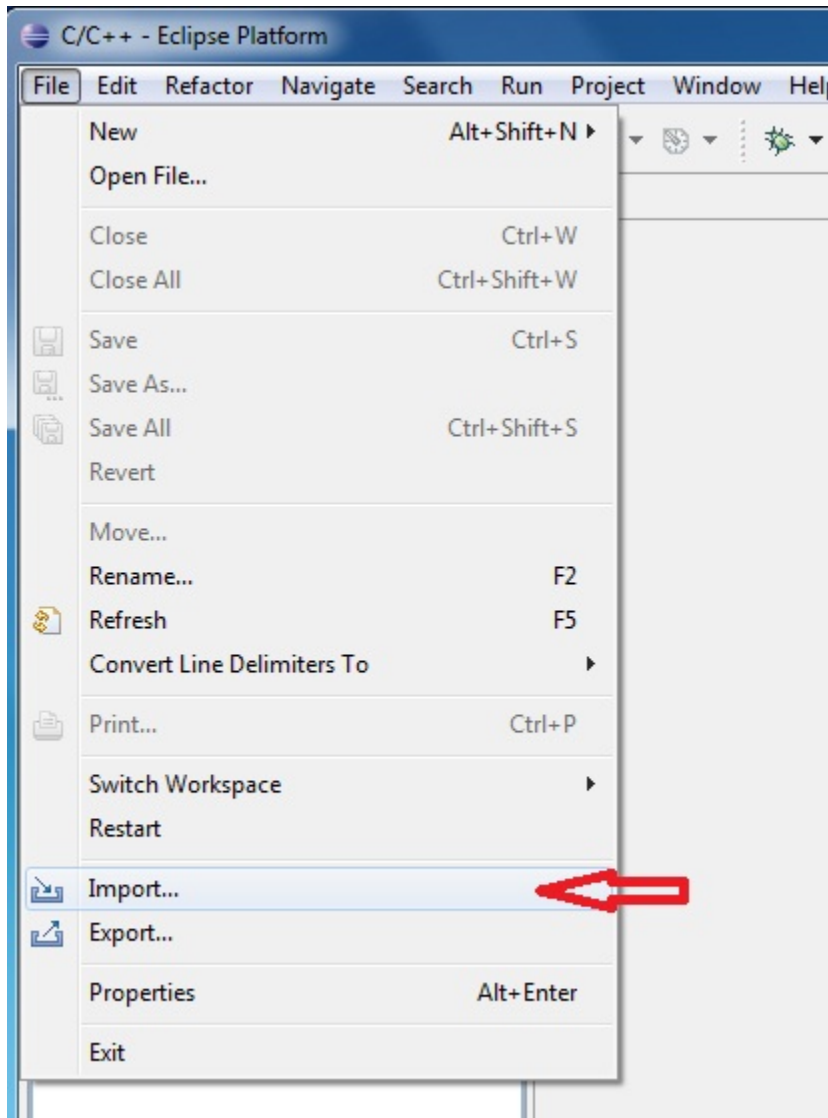




Close Eclipse and proceed with next steps.

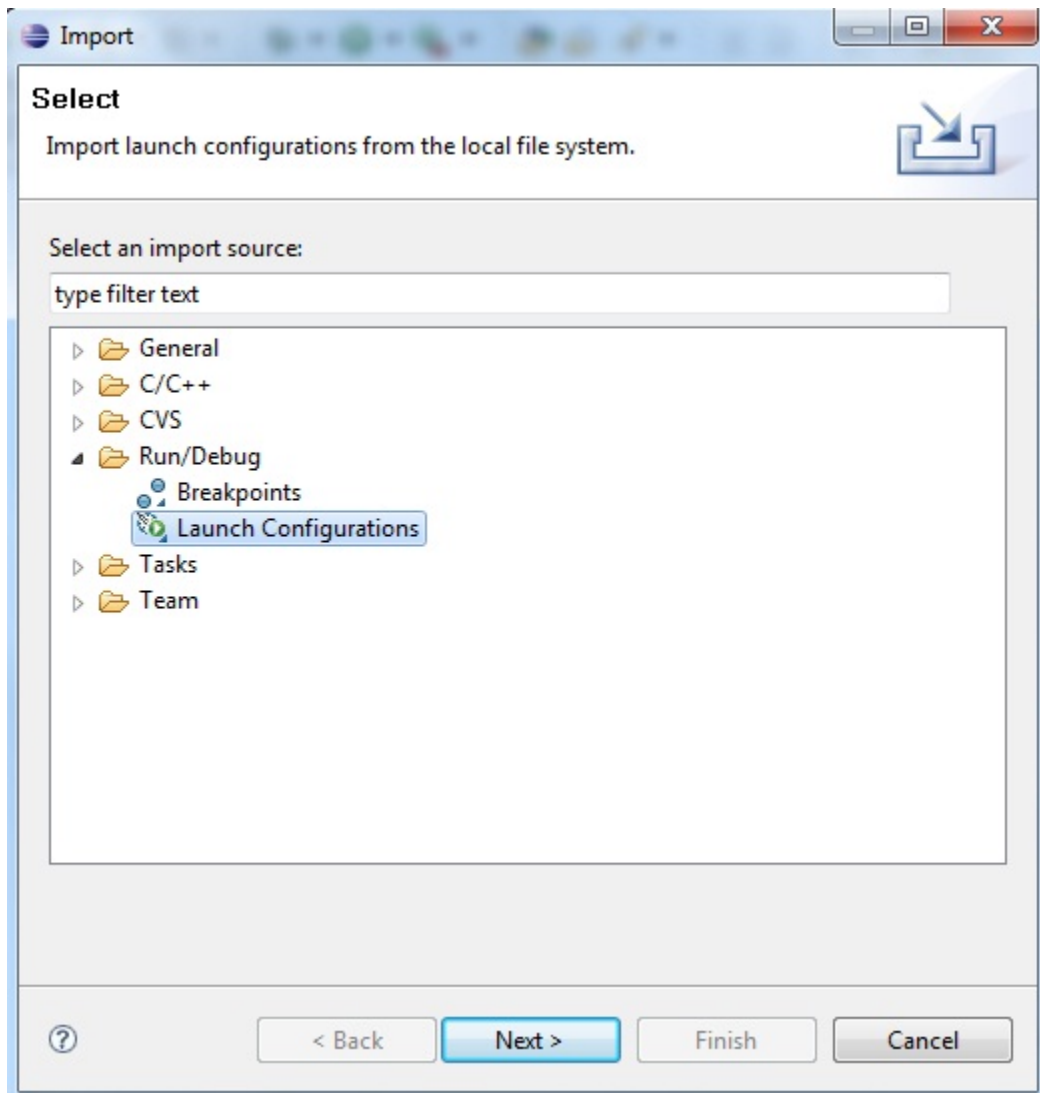
## Installing Flash GUI Tool Programmer

In order to complete development suite setup, you must install Flah Programmer as a external tool.  
Open file menu, and select "Import"

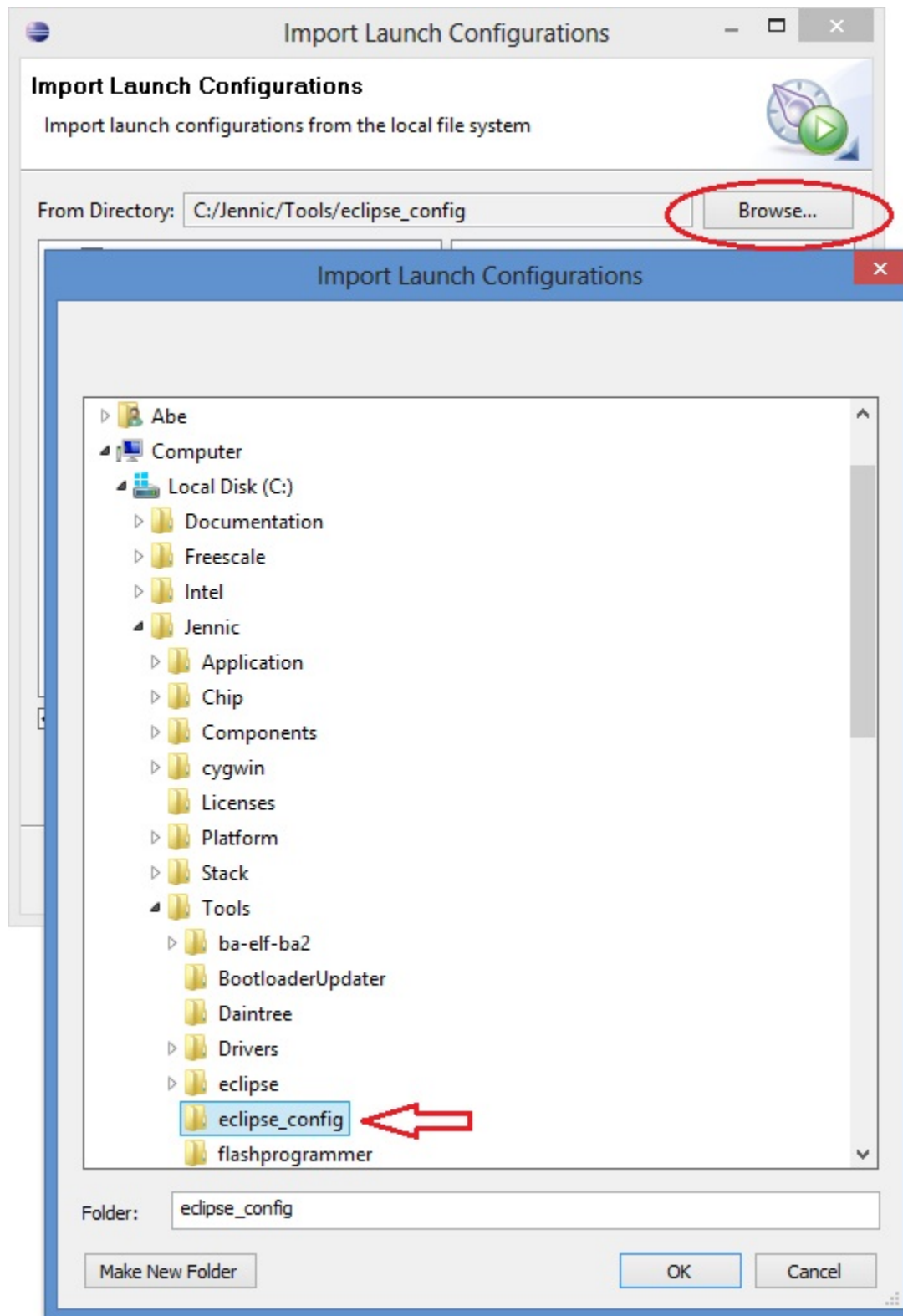


chose “Run/Debug –> Launch Configurations and click “Next” button





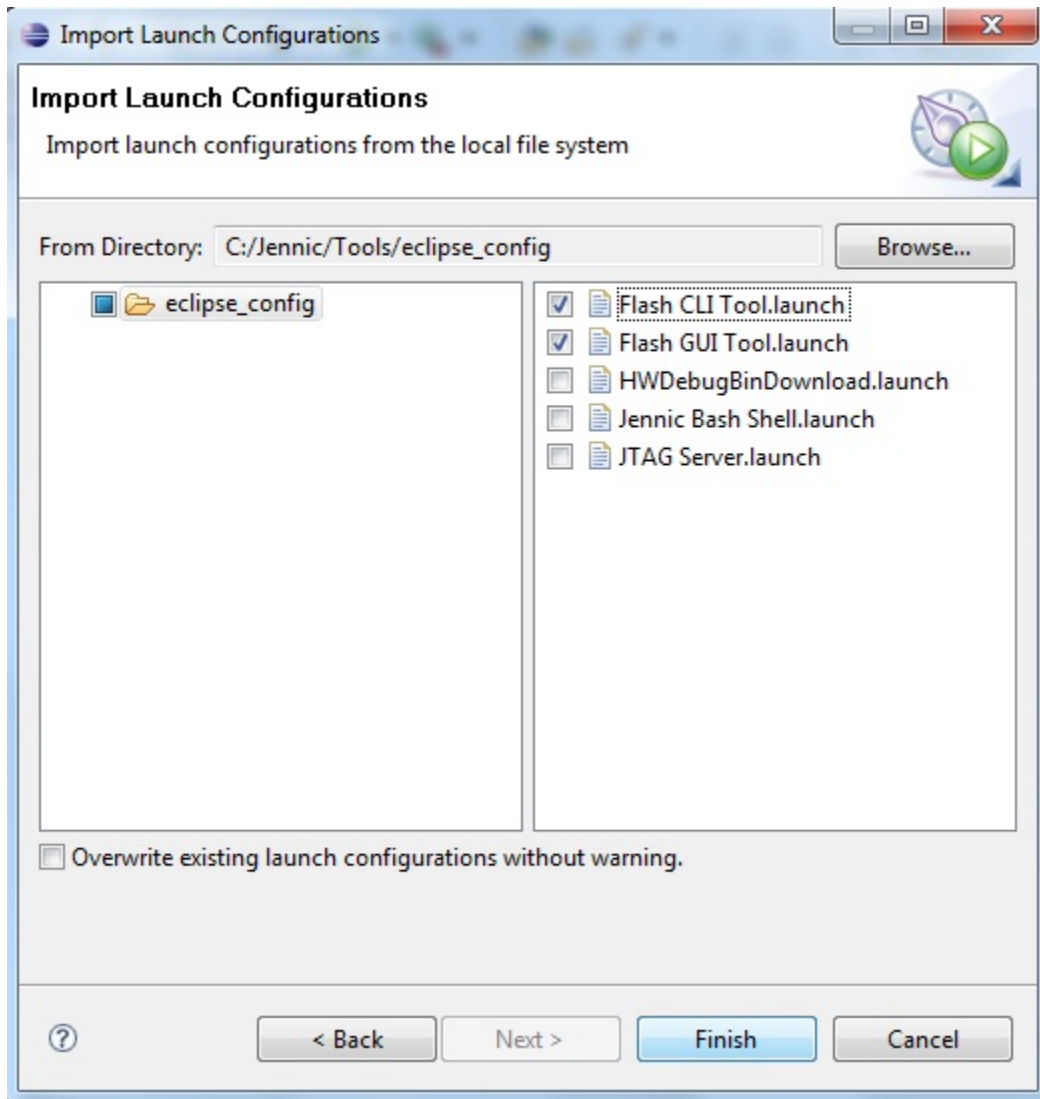
click “Browse” button and navigate to **C:\Jennic\Tools\eclipse\_config** and then click “OK”



Sign “Flash GUI Tool” and “Flash CLI Tool” checkbox

Please, note that Eclipse default is all tool checked, but as actual for JN5168 application only the two Flash tool are

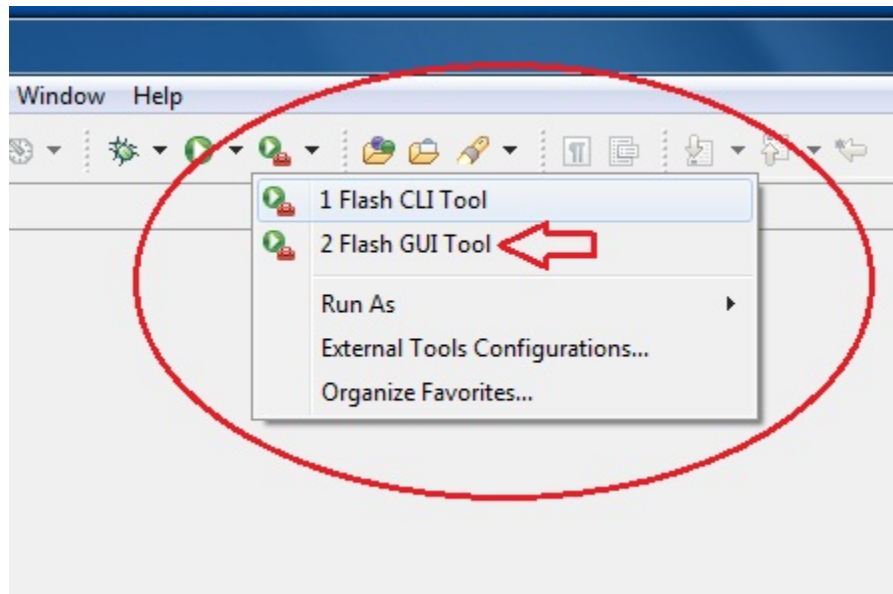
needed.



Click “Finish” button to perform Flash Tools setup.

When finished, you can find the tools in the Run->External Tools menu

They can also be accessed from the drop-down arrow next to the tools symbol, as show in figure below

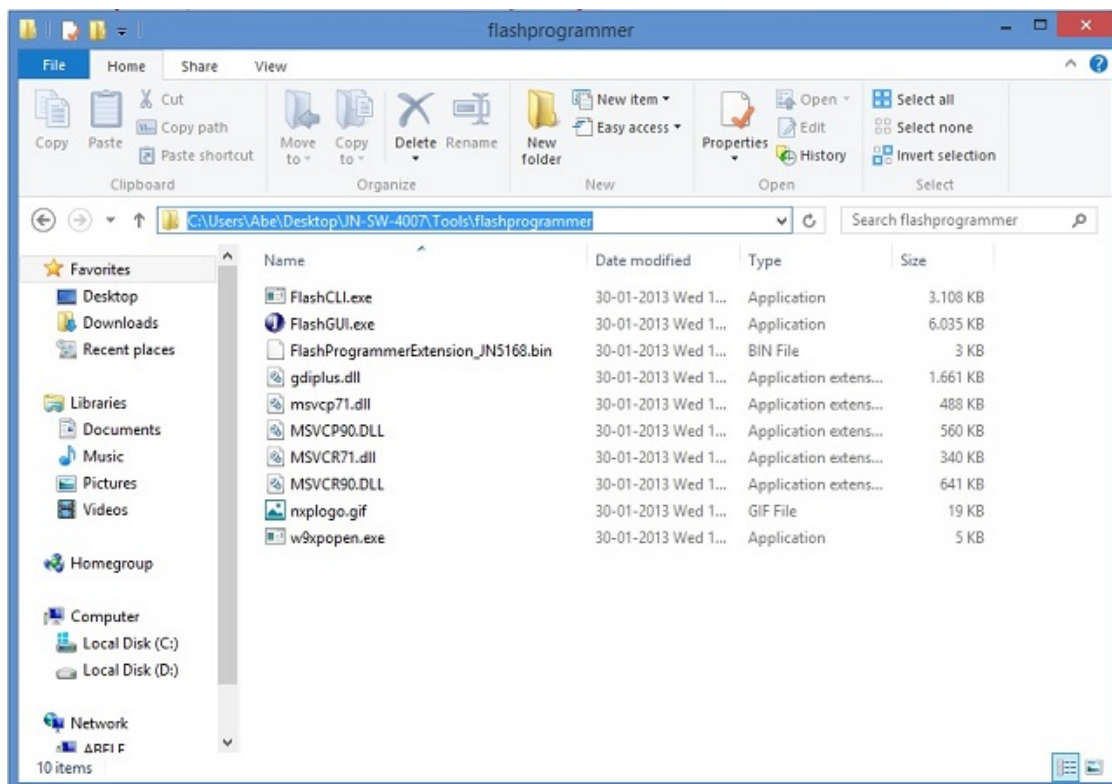


## Upgrading Flash GUI tool (for JN5168)

We must upgrade Flash GUI tool to release 1.8.9, because the tool just installed does'n work with JN5168.

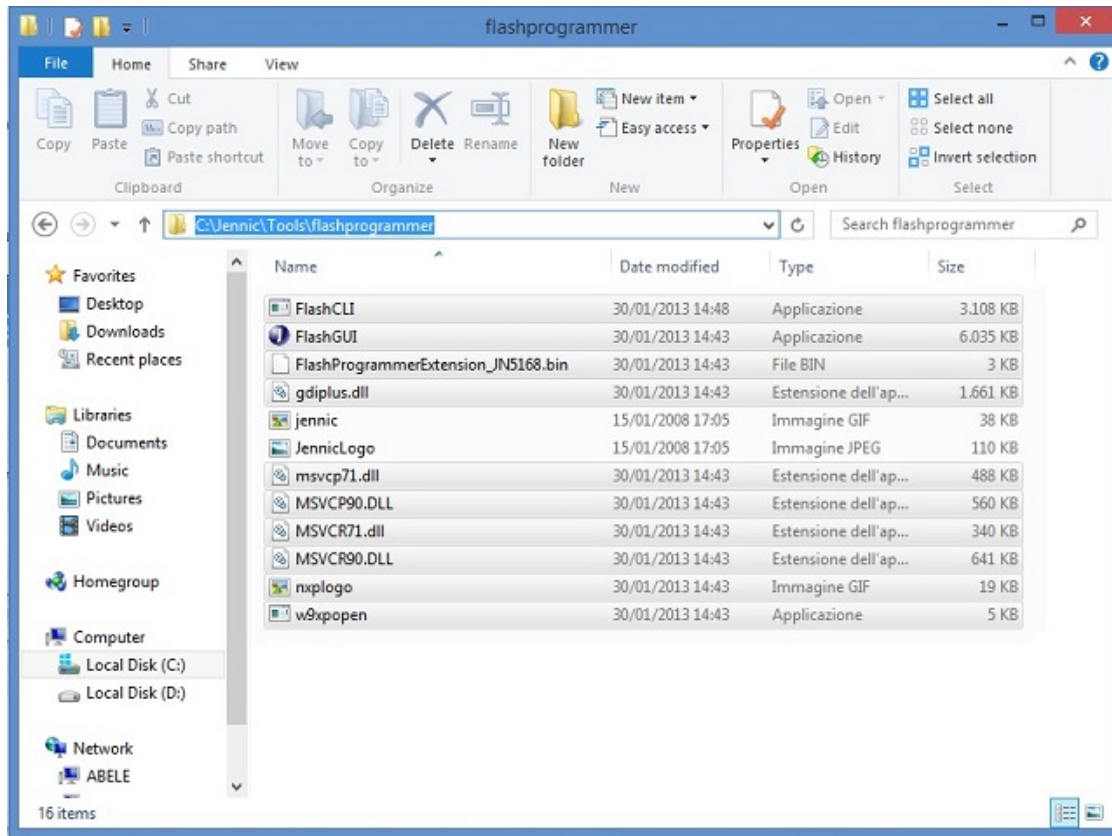
To make this, unpack the zip file JN-SW-4007.zip into new folder and navigate to **Tools\FlashProgrammer** located inside this new folder.

*you can find the JN-SW-4007 file inside Install.zip as show in **First step** chapter on this document*



Select all files, right click an copy.

Open folder **C:\Jennic\Tools\flashprogrammer** and **paste** all file inside this folder.



Now Jennic Eclipse setup is complete and the suite is ready for work with Rialto Project!

## WINXP TROUBLESHOOTING

### INSTALLING DRIVER FOR USB DONGLE

With some release of Windows Xp, you must manually install the USB drivers of the Rialto board.



In this case, go to [FTDI driver](#) page download and install Windows driver.

## CYGWIN1.DLL TROUBLES WITH WINXP

Jennic SDK Toolchain install CygWin API inside the folder C:\Jennic\cygwin.

Some XP installation have the file “cygwin1.dll” inside C:\windows\system32 folder.

This is a cause for fatal error during project building.

If it occurs, build fails and on the “Console” tab of the Jennic Eclipse you can see one or more messages as follow:

```
290 [main] ? (680) c:\Jennic\Tools\ba-elf-ba2\bin\ba-elf-gcc.exe: *** fatal error -
cygheap base mismatch detected - 0x611668E0/0x611688E0.
This problem is probably due to using incompatible versions of the cygwin DLL.
Search for cygwin1.dll using the Windows Start->Find/Search facility
and delete all but the most recent version. The most recent version *should*
reside in x:\cygwin\bin, where 'x' is the drive on which you have
installed the cygwin distribution. Rebooting is also suggested if you
are unable to find another cygwin DLL.
```

**To solve this trouble with Jennic Eclipse Builder, we suggest to try to remove file “cygwin1.dll” located inside folder C:\Windows\system32**

*We suggest to rename the file in your system folder and check if there are no other troubles*

Be careful if you have other applications that use cygwin!! Removing file from OS system folder must be checked before.

For more detail and further information regarding cygwin, you can visit the [cygwin](#) site.

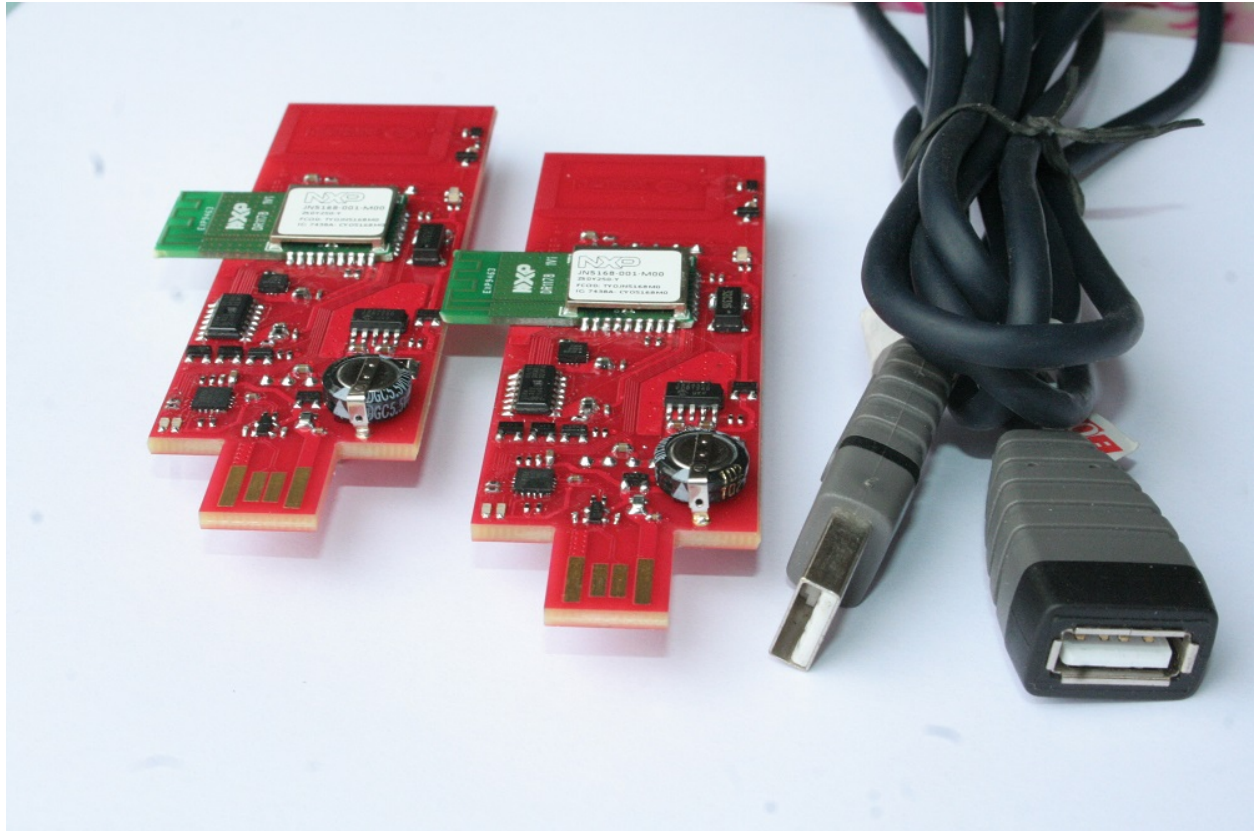
For further information about this trouble, click [here](#).



## Quick start guide

### Hardware requirements

- Silica Rialto Board kit
- One or more USB type-A extension cable
- PC with terminal software (such as HyperTerminal)



### Software requirements

- Jennic SDK toolchain (JN-SW-4041), JN5168 stack library (JN-SW-4065) and Flash GUI tool (JN-SW-4007)
- Rialto\_Jennic firmware
- PC terminal emulator (such as HyperTerminal)

---

#### Note:

**If you have not yet installed Eclipse Jennic, before proceeding, go to:**

*Installing Jennic Development Suite*

---

## Hardware setup

Plug Rialto Board into USB port (also using USB type-A extension cable)

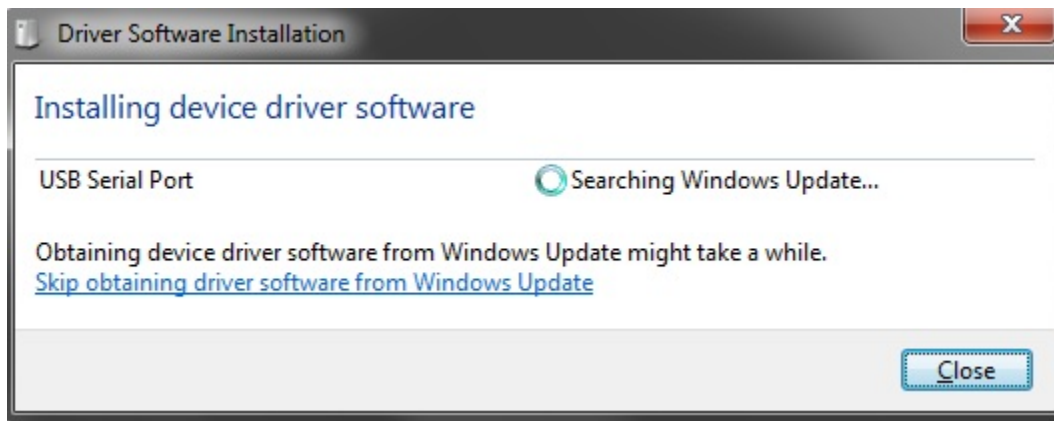


with USB type-A extension cable

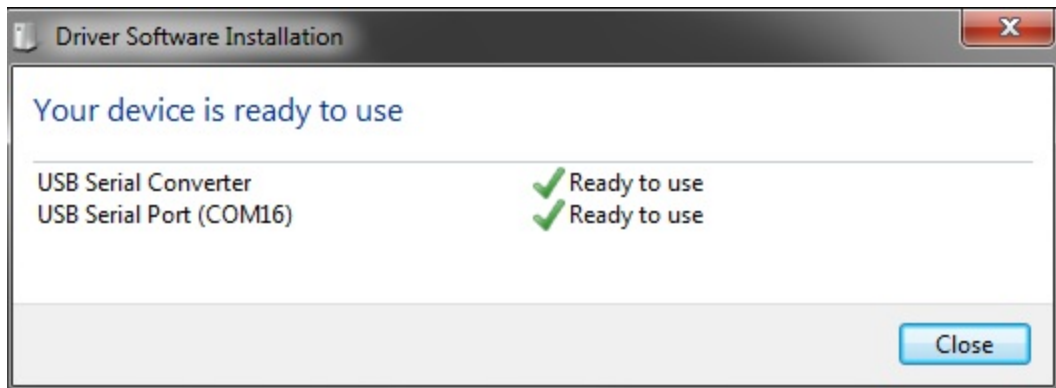




Wait for properly USB dongle driver installation

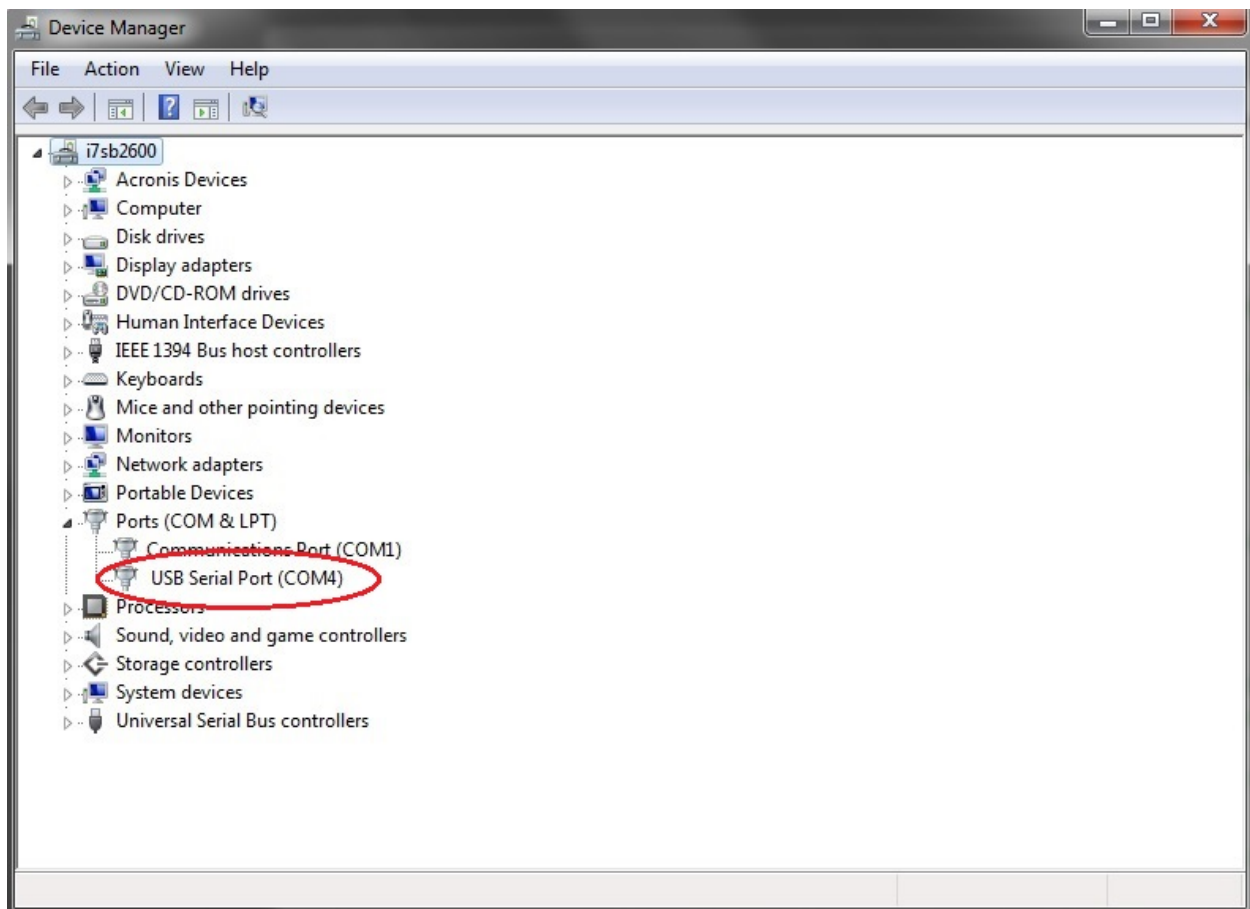


When drivers are ready you can connect one or more Rialto Board (at least two for firmware evaluation)



**Tip:**

See Control Panel of your PC and note COM Port number configured for each Rialto Board (figure above). For any troubles during USB driver installation in Windows XP see at [WINXP TROUBLESHOOTING](#)



*Connecting two Rialto Board (minimun number for full firmware evaluation), take care at COM number installed on your PC*



### SerizII plugin

Rialto board has in bottom layer two strip (6 and 4 way) and can be used as a plugin for SerizII.



**Important:** Before using Rialto boards as SerizII plugin, you must download and install the proper firmware revision of the SerizII board (**SerizRialto.zip**). For all instruction and information about, see at [Silica SerizII](#)

---

**Take care plugging Rialto on SerizII: be shure that both boards (Rialto and SerizII) are power off before!!**

## HyperTerminal settings

Set your HyperTerminal COMx parameter:

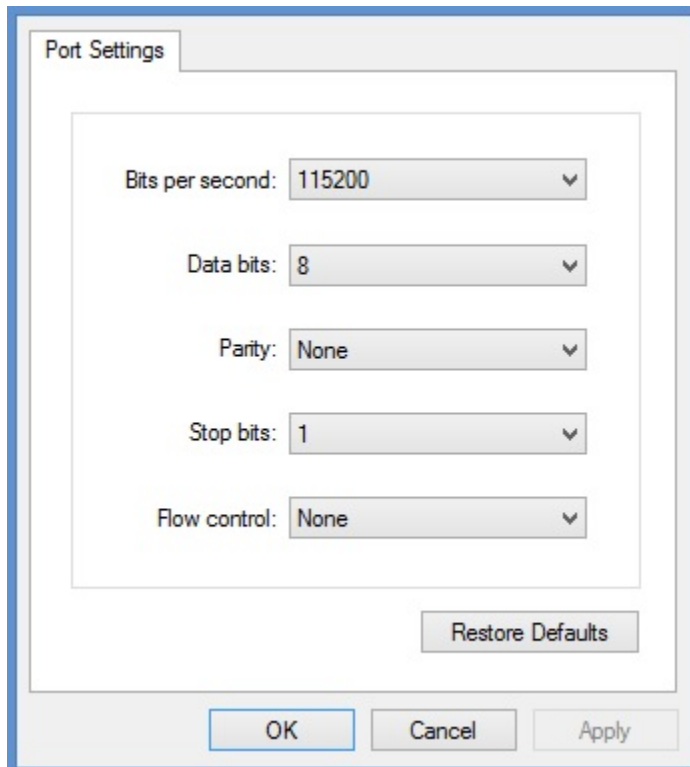
speed = 115200 baud

data with = 8

parity = none

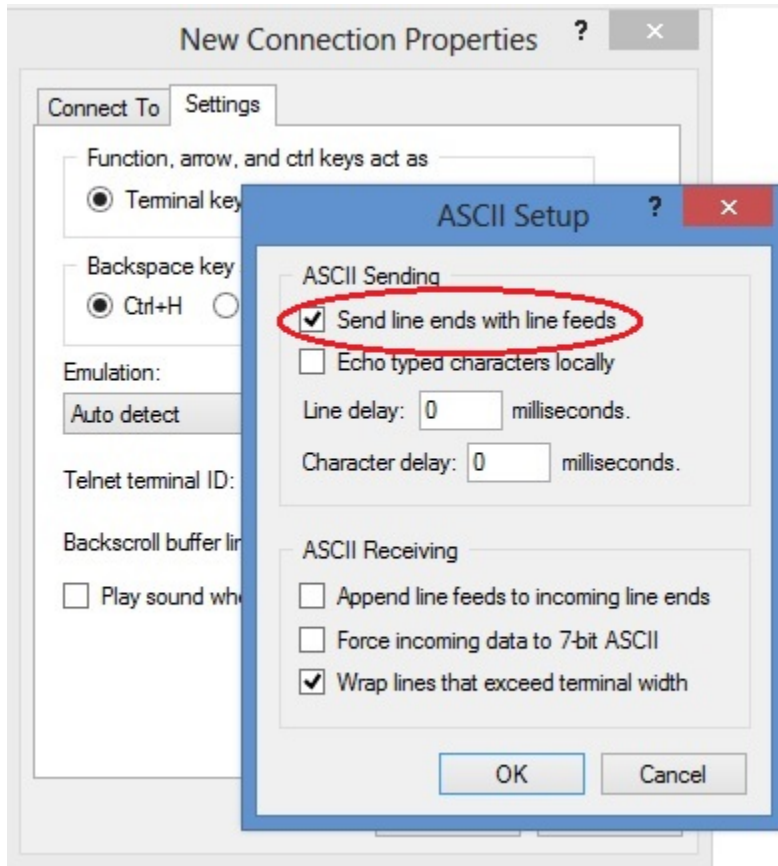
stop bit = 1

flow control = none



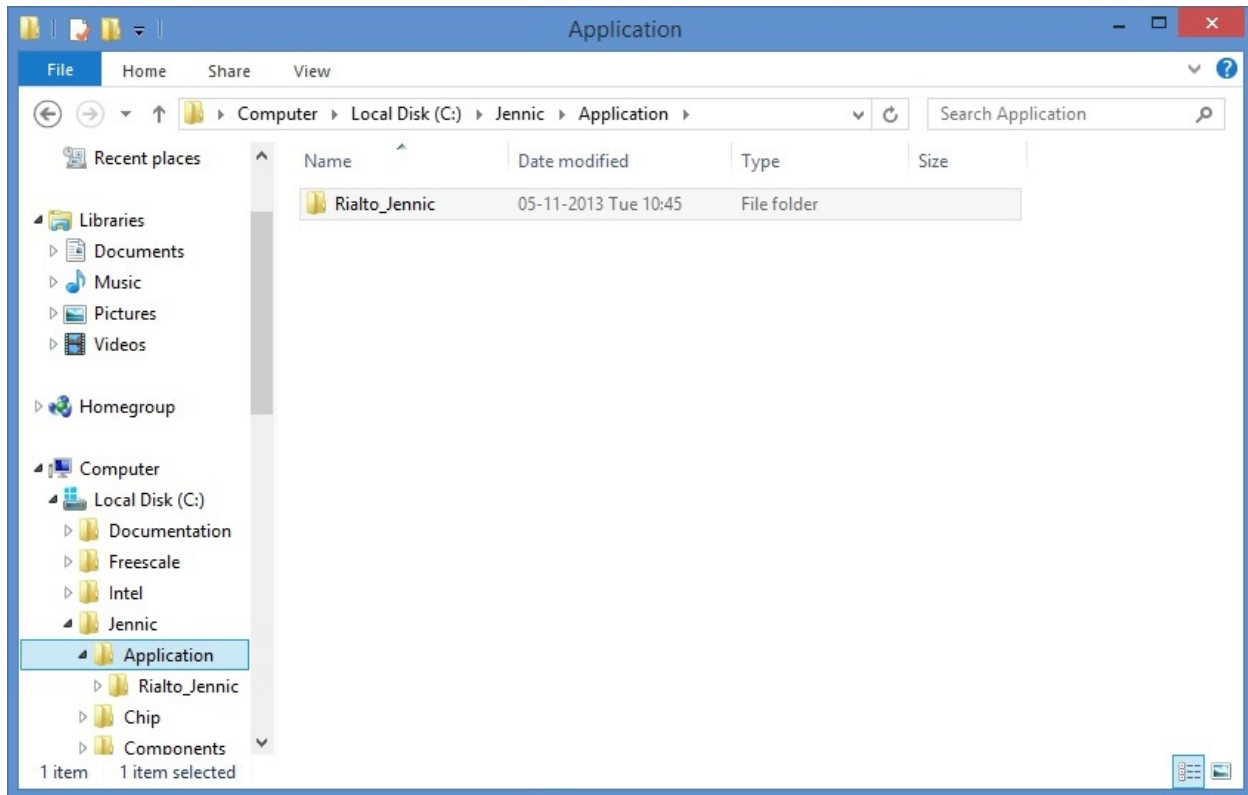
In Ascii Setup windows, check “Send line ends with line feeds”



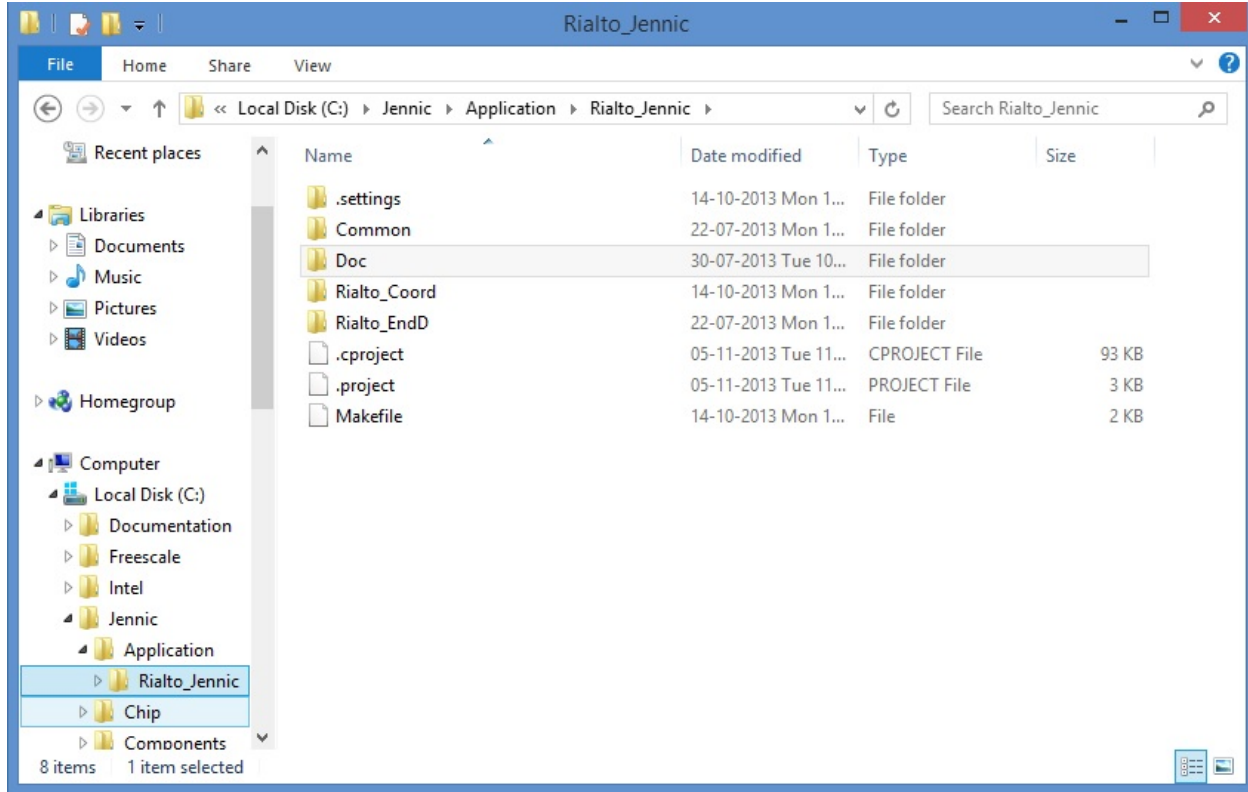


## Rialto Board FW installation & setup

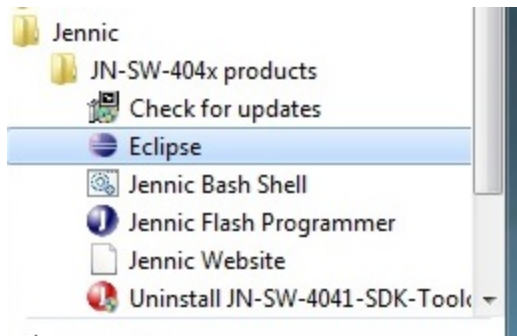
- In the folder C:\Jennic\Application create new folder named **Rialto\_Jennic**



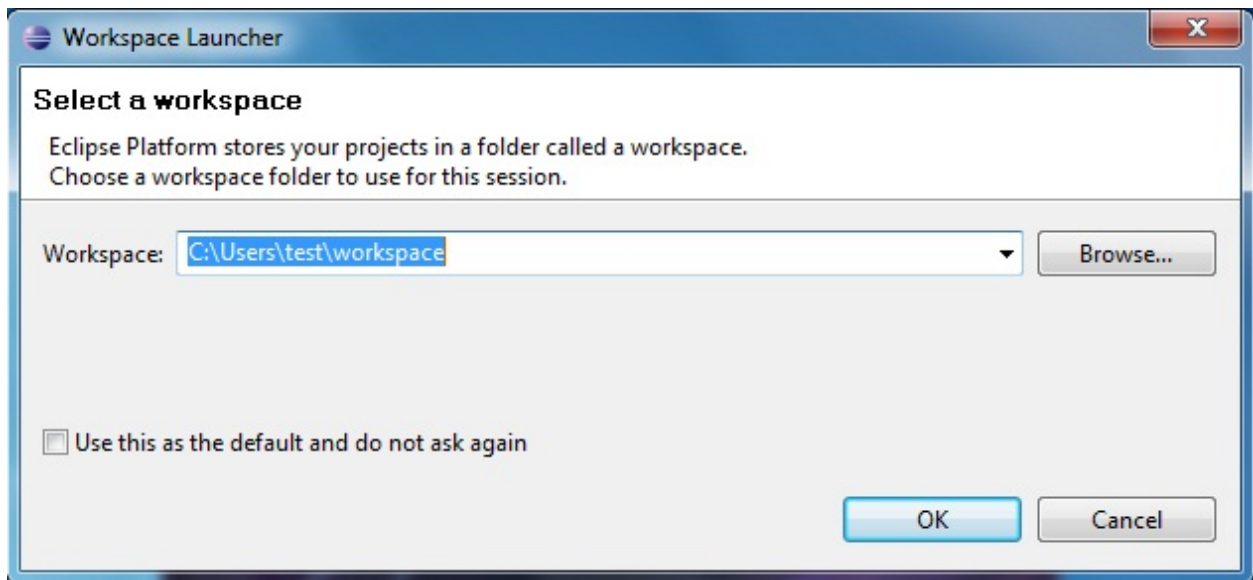
- Unzip all files from Rialto\_Jennic\_1\_0.zip into the folder **C:\Jennic\Application\Rialto\_Jennic** just created



- Go to Start -> Jennic -> JN-SW-404x products -> Eclipse and click on to start Jennic Eclipse

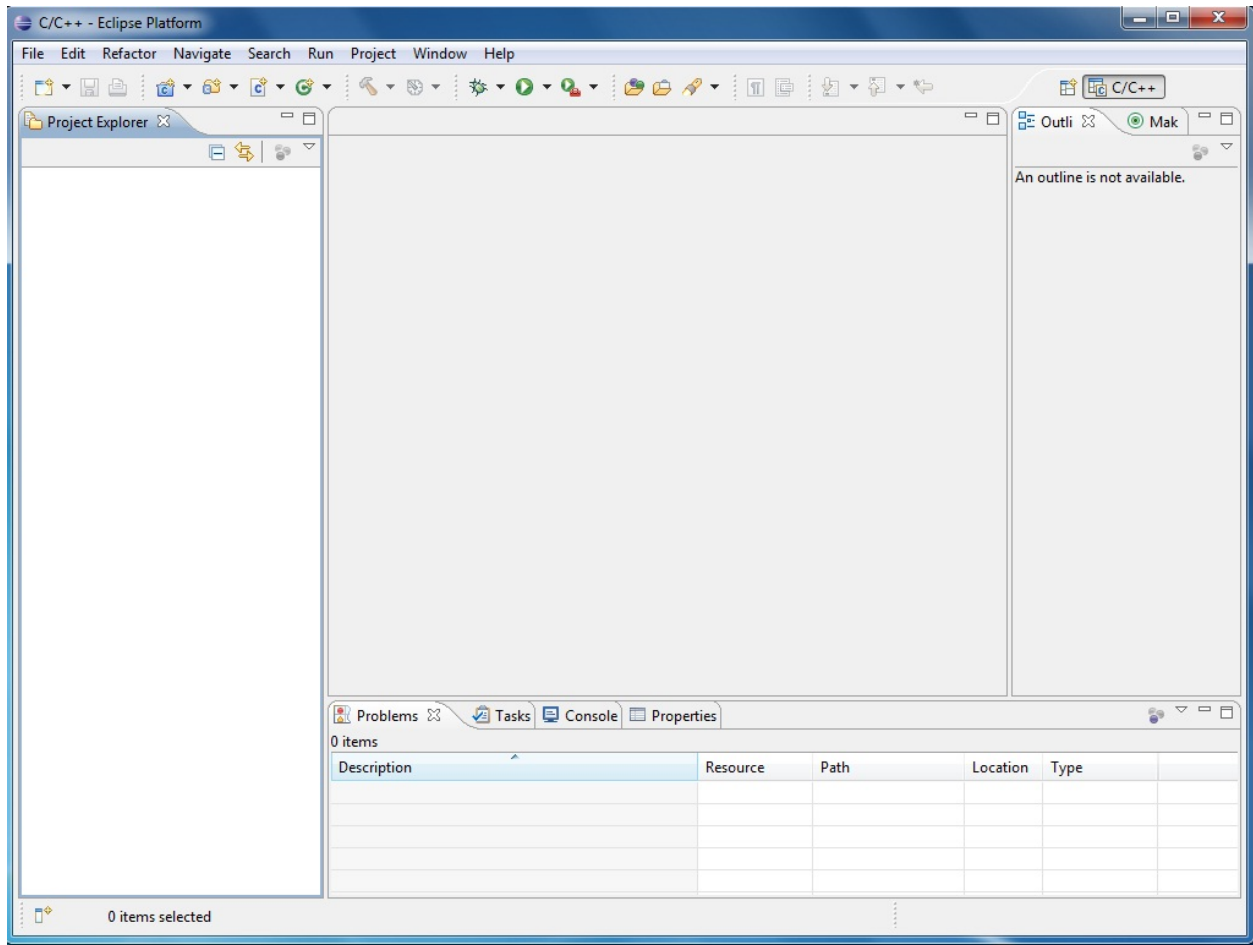


- Check if workspace setting is like figure below. Then click OK to proceed.

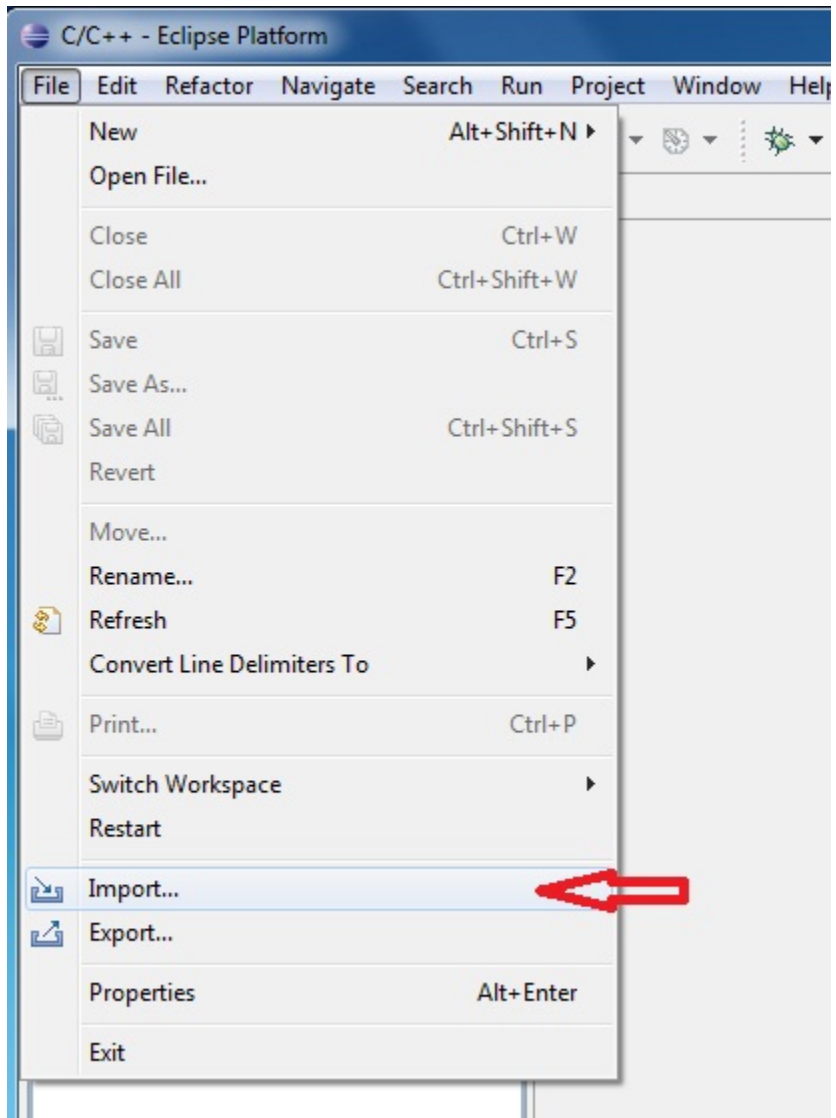


- Now you can see the Eclipse Main Window

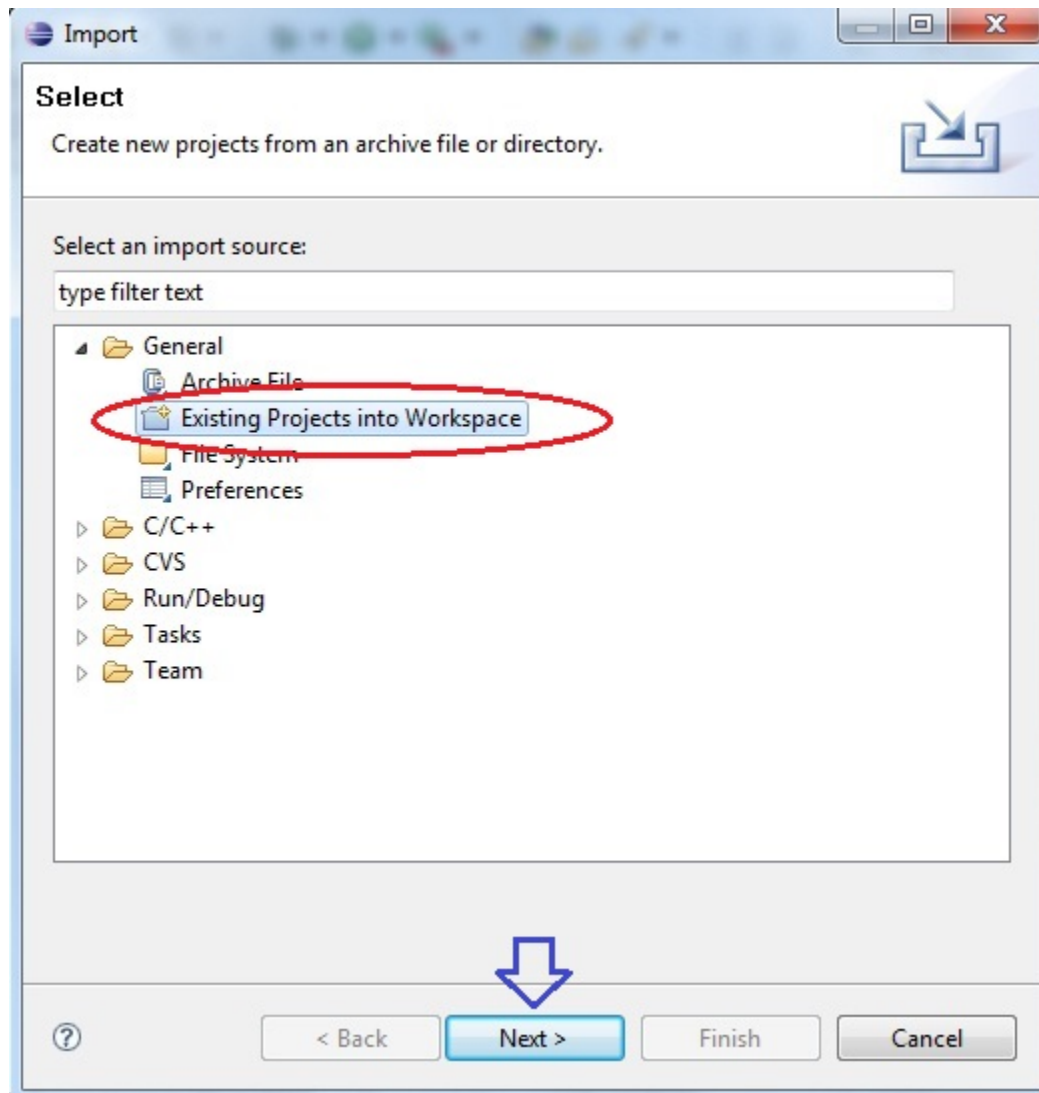




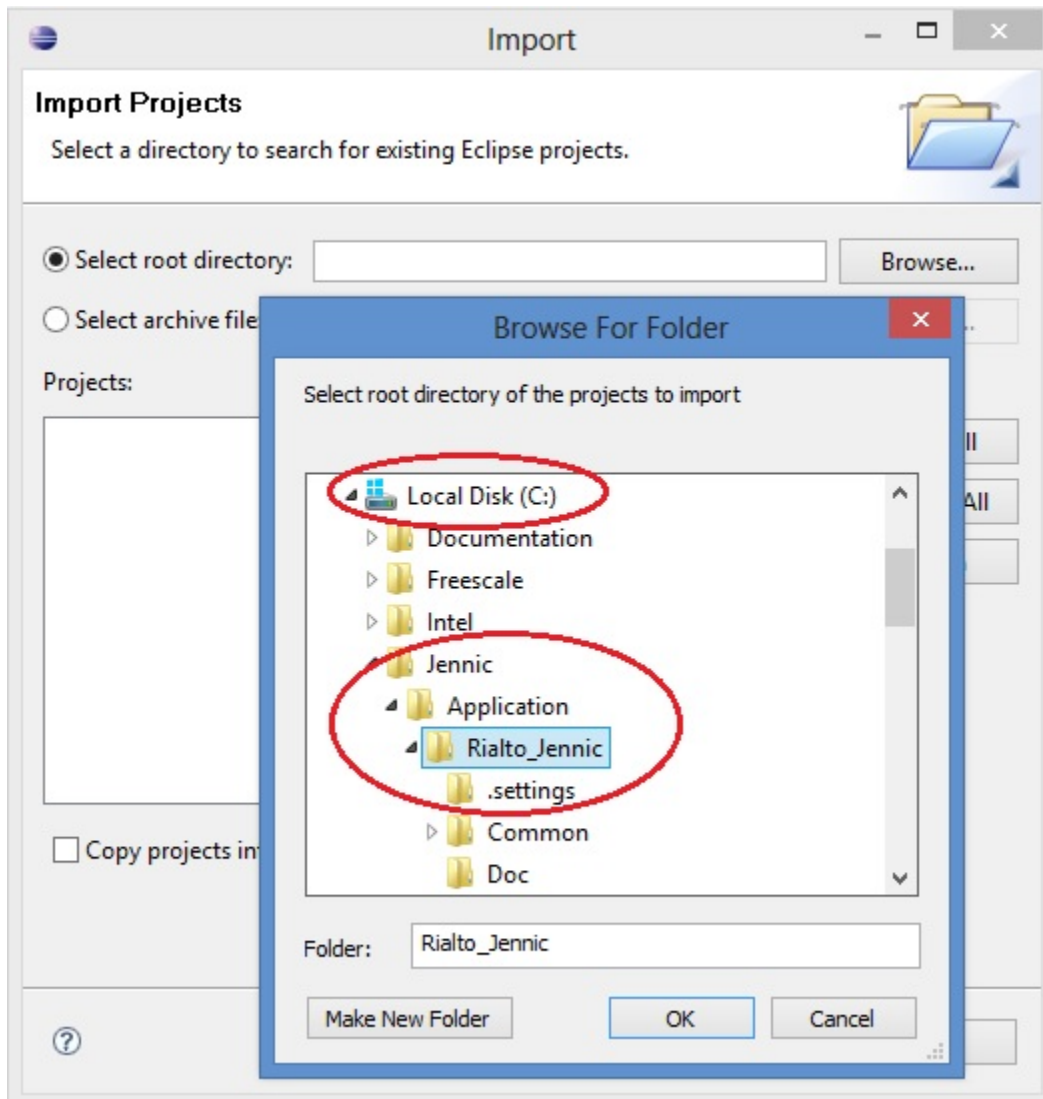
- Select menu File -> Import



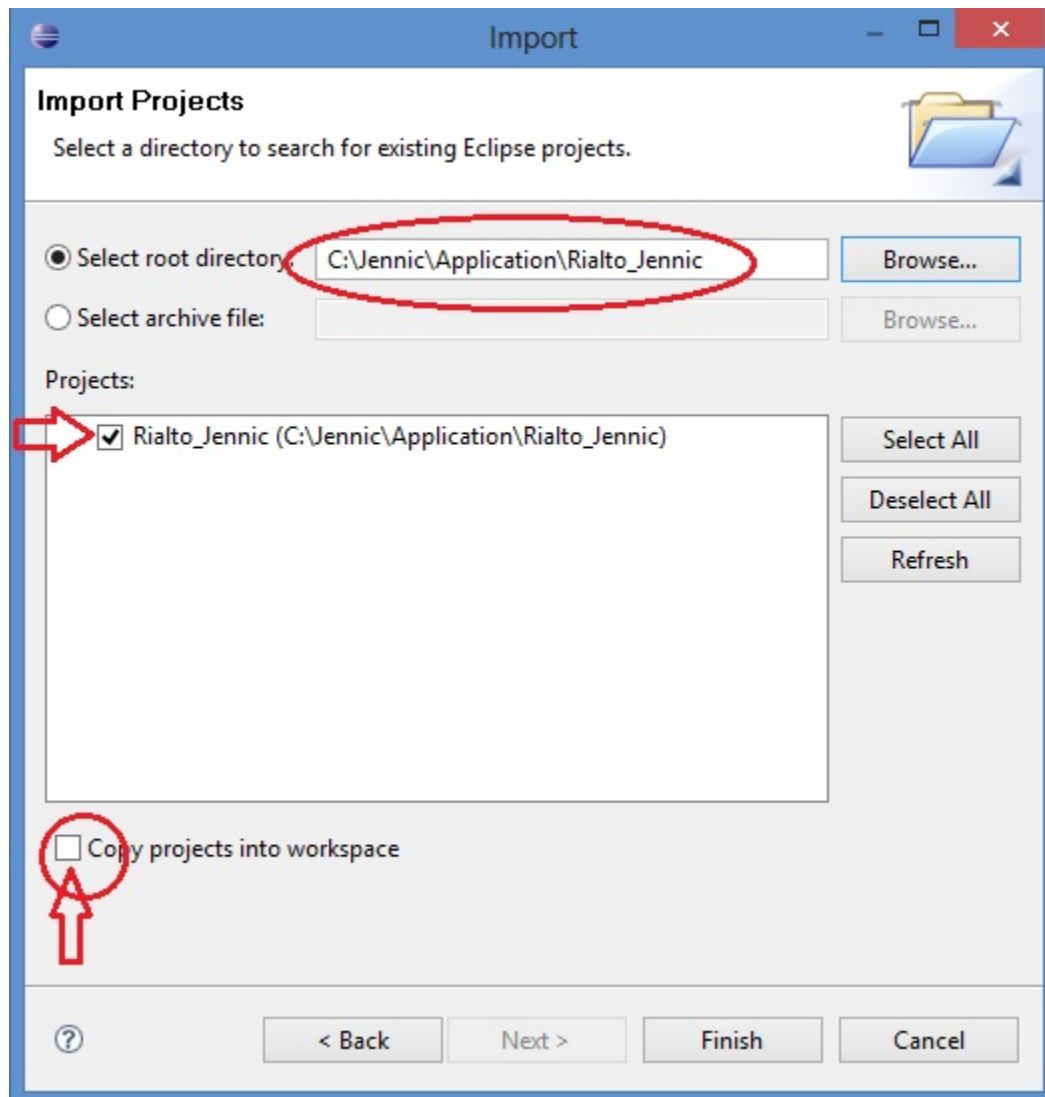
- In the dialog box that will open, click on **General**, select **Existing Projects Into Workspace** and after click “Next” button: new dialog will open.



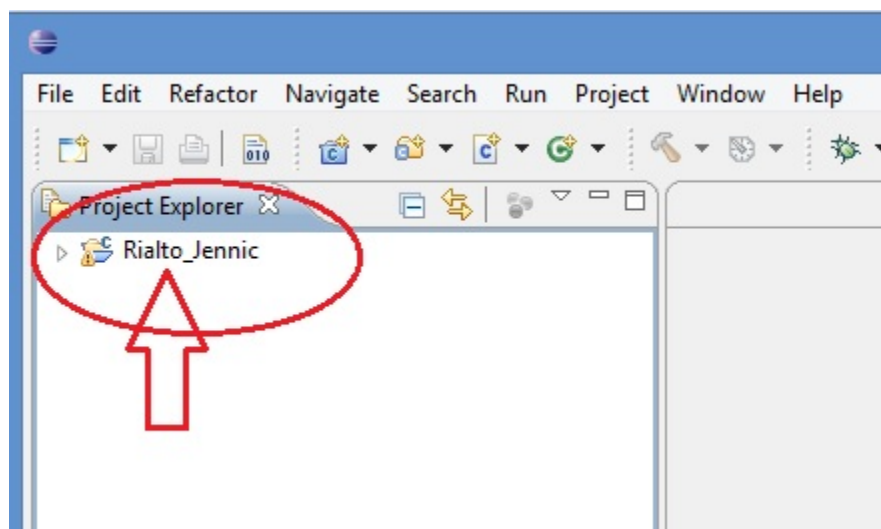
- Click on “Browse...” button an navigate to **C:\Jennic\Application\Rialto\_Jennic** folder. Click on “OK” button



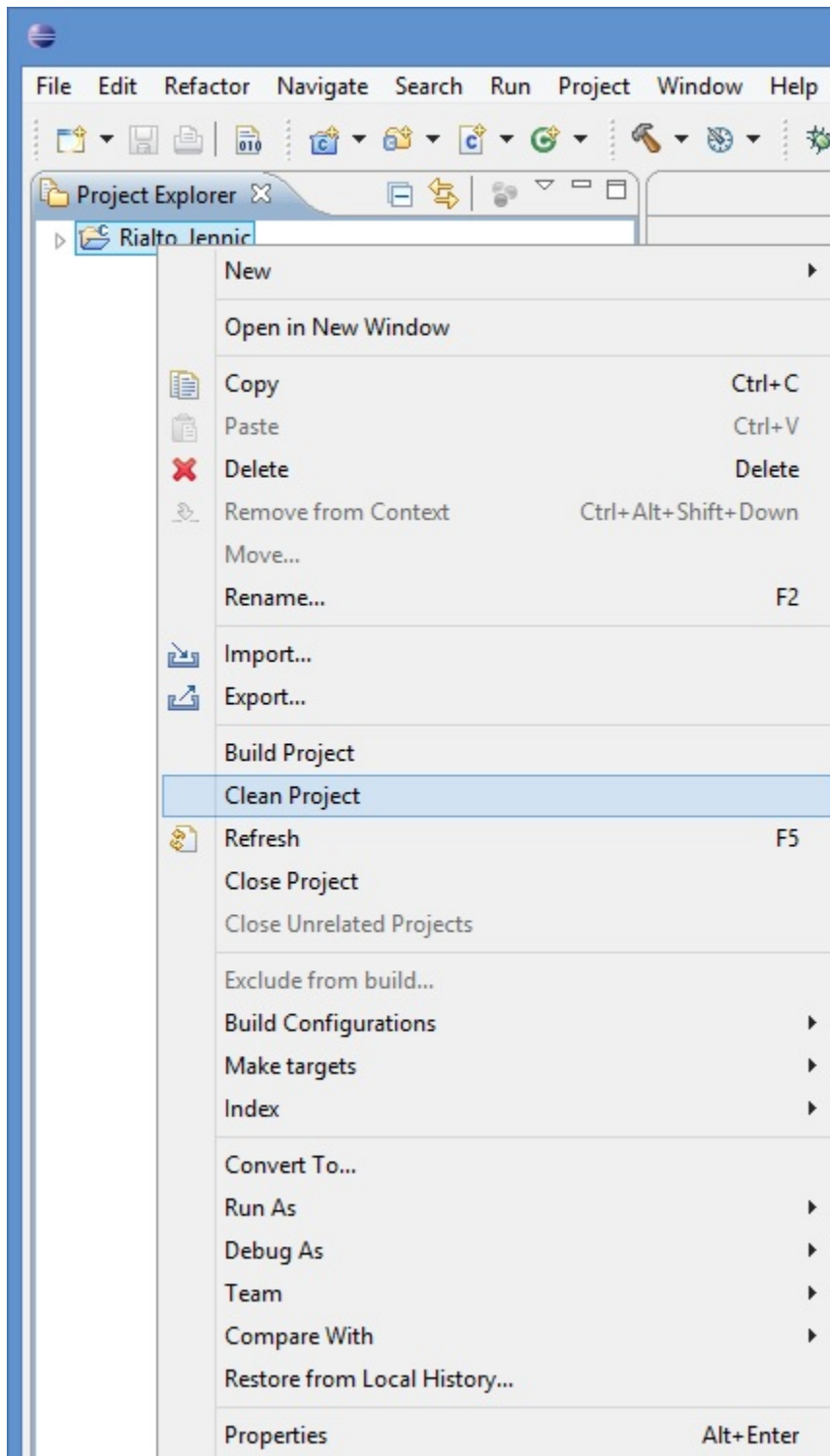
- Check options and setting as the image below, then click “Finish” button to import project.



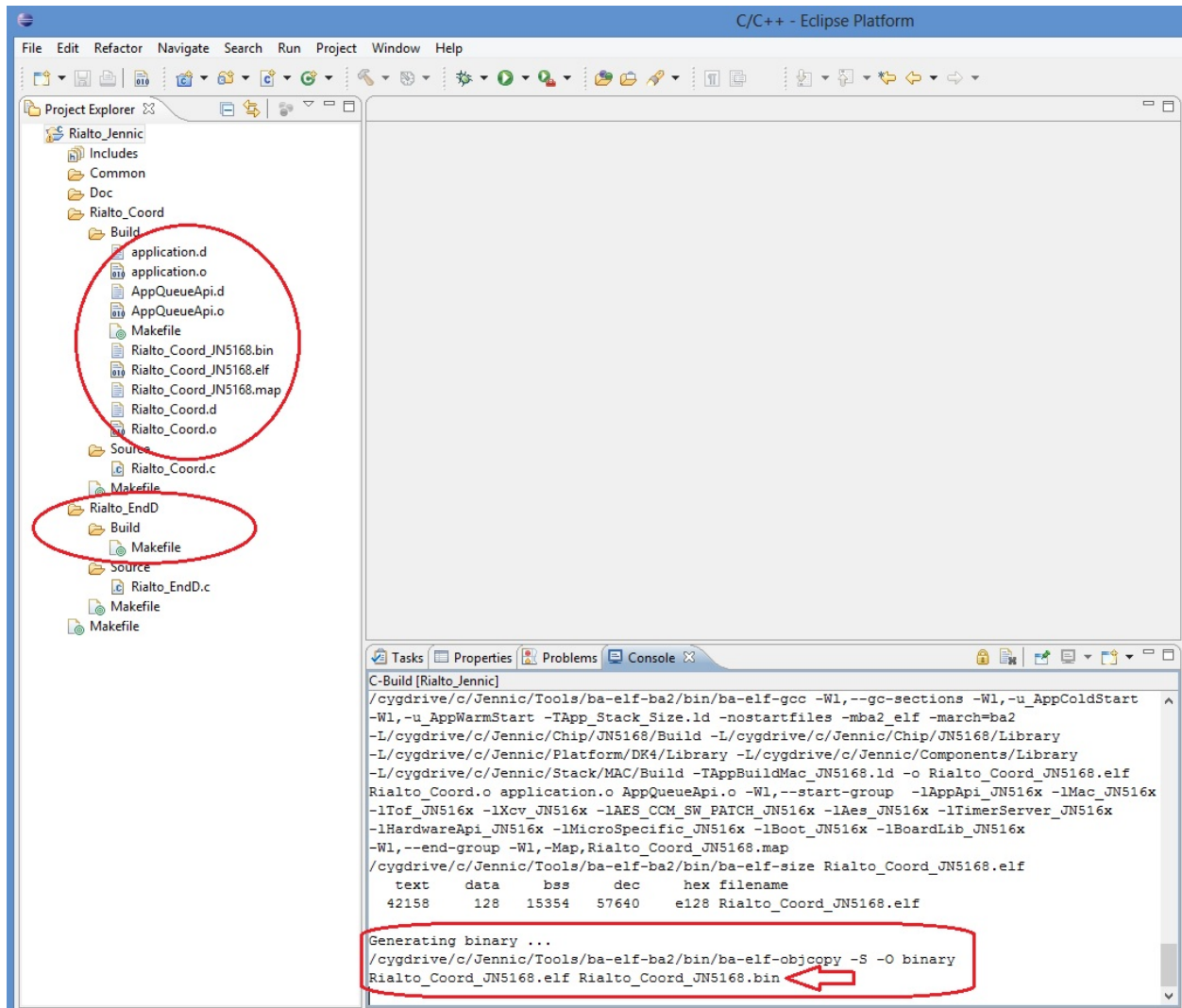
- Wait for project import, then you can see Rialto\_Jennic project in the Project Explorer windows of Eclipse Platform.



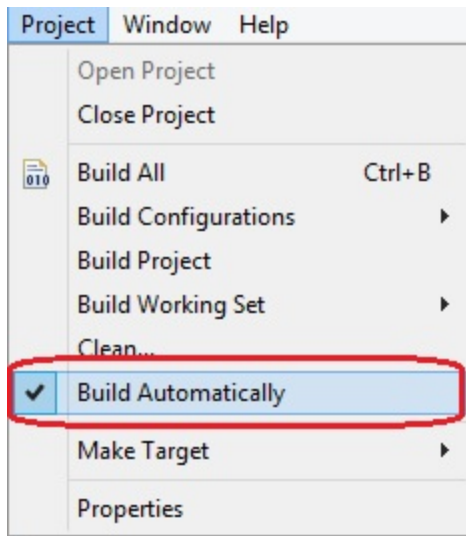
- **First of all**, right click over “Rialto\_Jennic” in the Project Explorer window, then select “Clean Project”. After cleaning, a first build will start automaticaly



- Take care at image above. Expand project, and see at “Console” tab: you can find a log that ends with “Generating binary .... Rialto\_Coord\_JN5168.bin”



**Tip:** If you can't see in the "Console" tab the message above, make shure that "Build Automatically" option (inside "Project" menu) is set

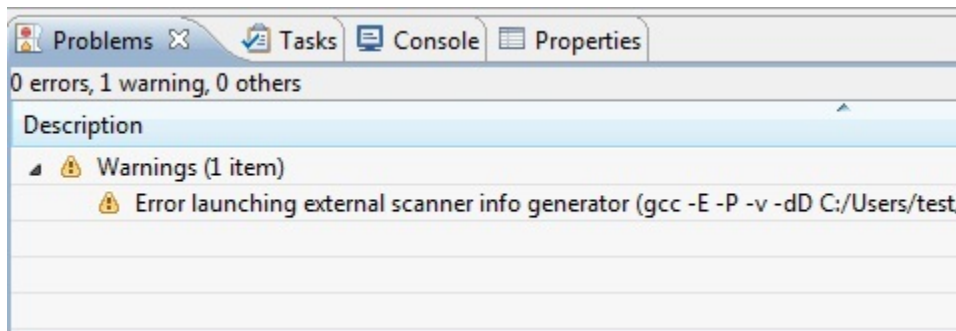


---

**Note:** The binary file for Coordinator has been generated and ready for Rialto Board programming

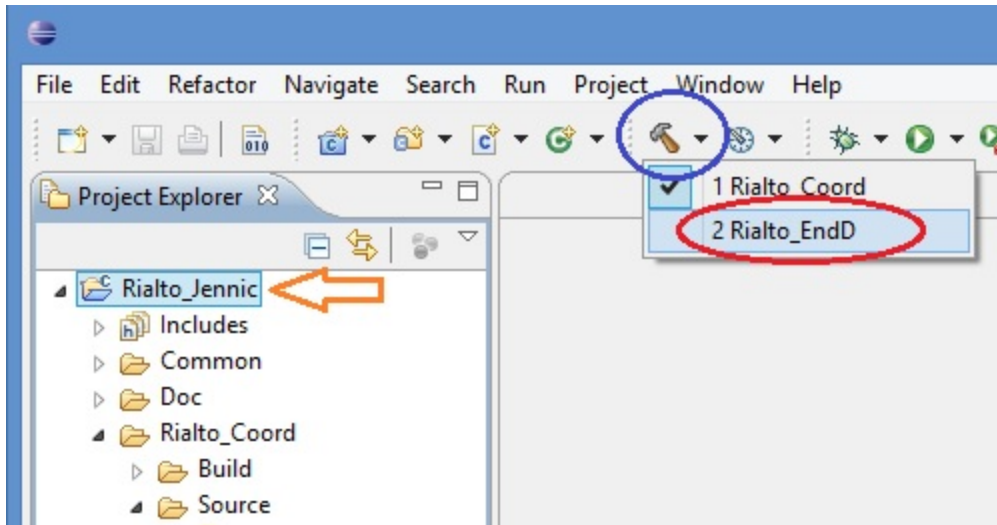
---

**Tip:** *don't care if you have this warning (see Problems tab).*

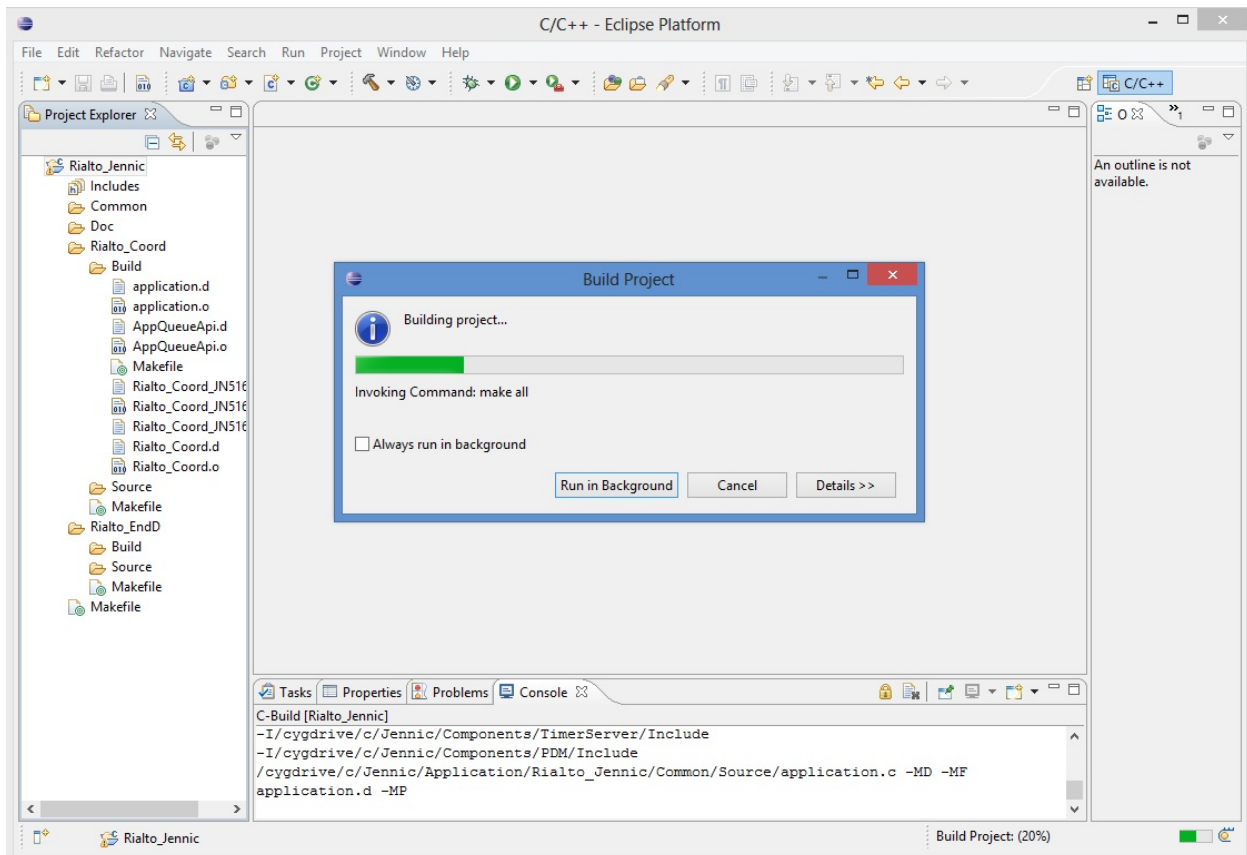


- Now you can build End\_Node project. Click on drop-down arrow next to Hammer Icon (blue circled in image below)

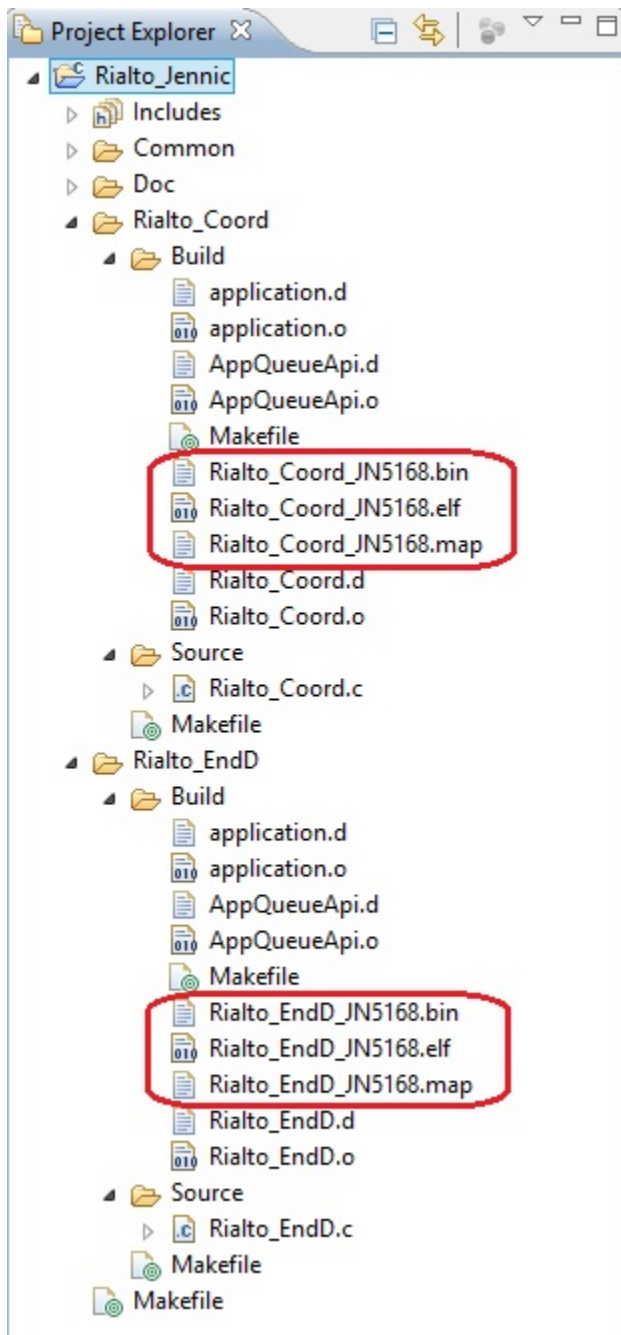




then click on “Rialto\_EndD”. Build will start.



- When build has finished, in the Project Explorer tab expand Rialto\_Coord and Rialto\_EndD Build folders. The result in image below.

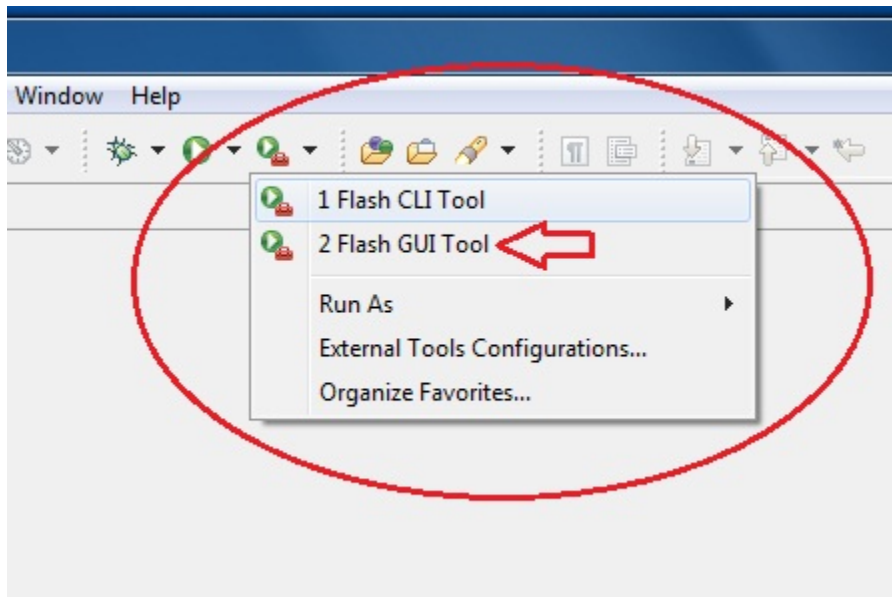


You have built the two binary files and you are ready for program Rialto Board

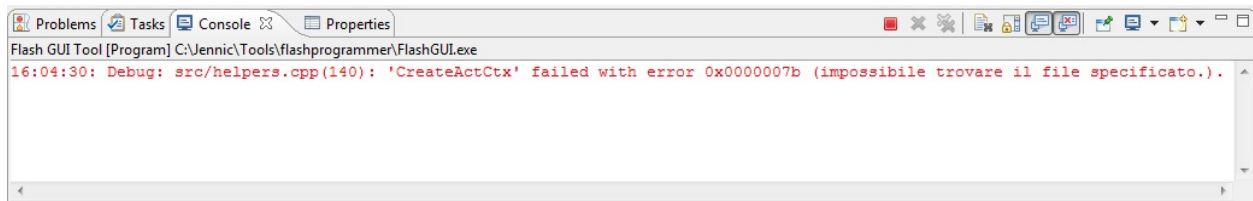
## Programming Rialto Board with Flash GUI Tool

Before starting Flash GUI programmer tool, you must connect Rialto Board whit built-in USB dongle. Please refer to Hardware Setup chapter on this document to check connection

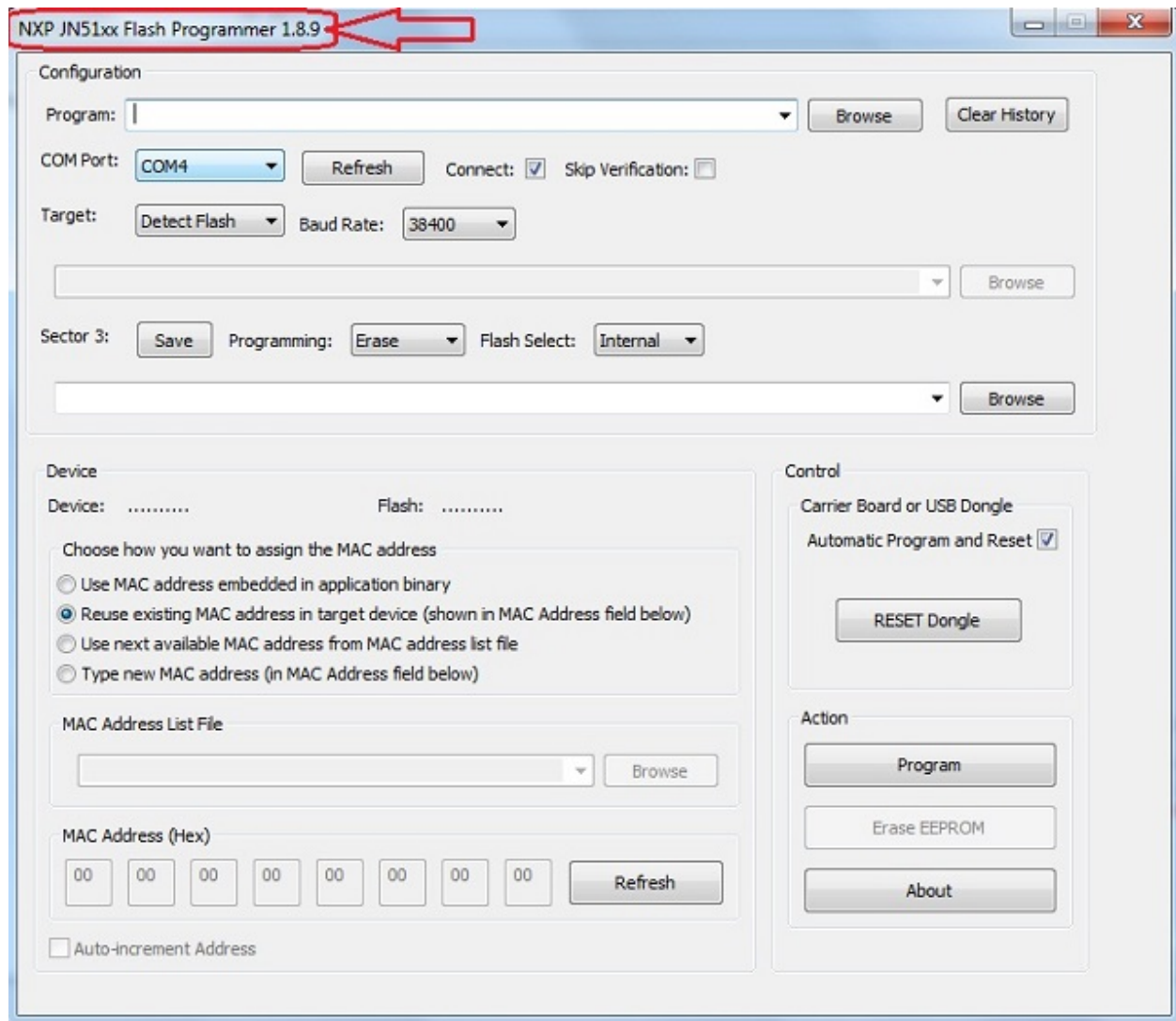
- Flash Gui Tool can be accessed by clicking on the drop-down arrow next to Tools Symbols on the toolbar, then selecting “Flash GUI Tool”



*Dont't care of error in Console tab: no troubles or malfunctioning will be affect board program!*



- See at Flash GUI windows: check if version is **1.8.9**. If not, refer *Upgrading Flash GUI tool (for JN5168)* on **Installing Jennic Development Suite** guide

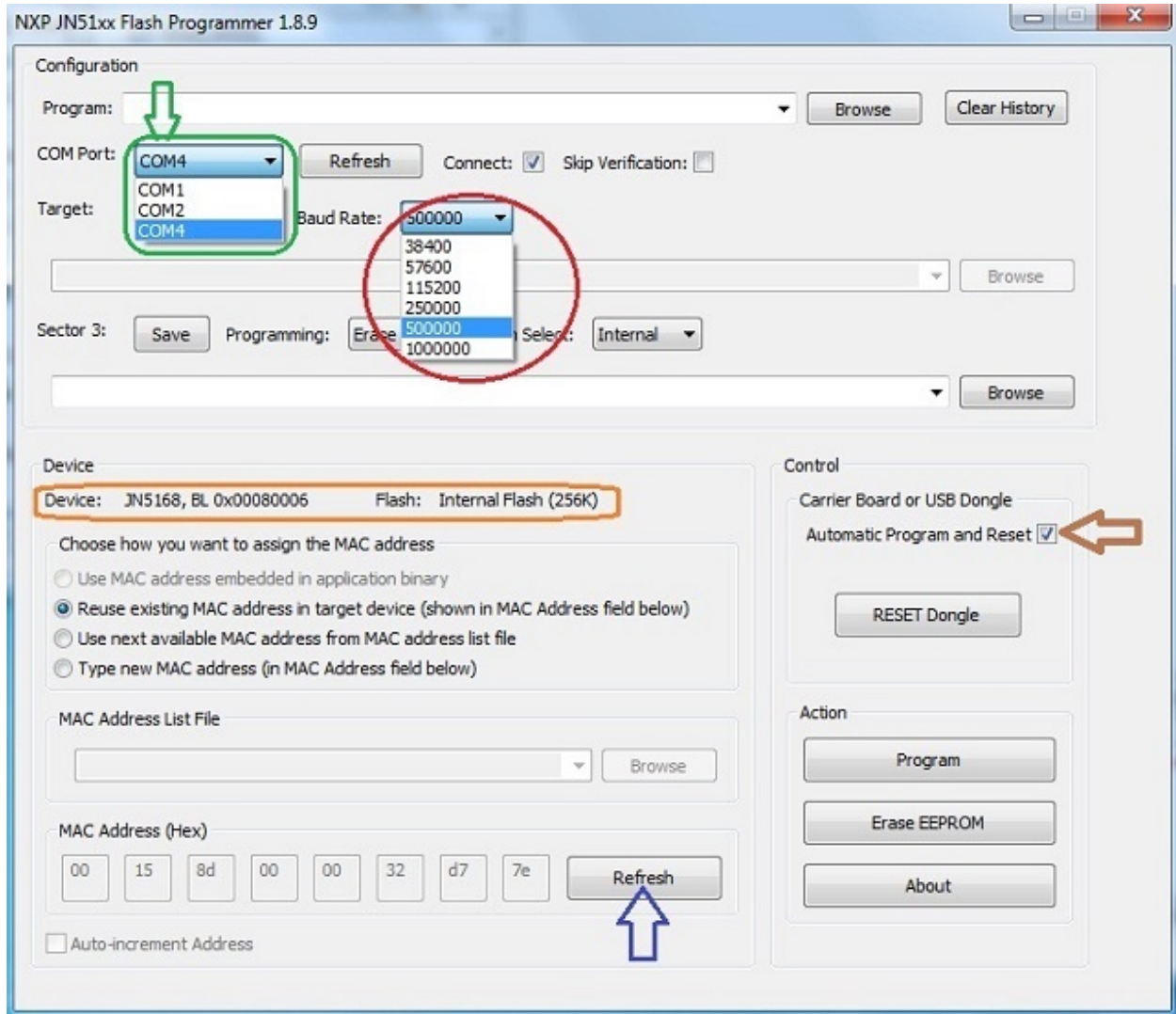


- **Step 1:** make these actions:

- 1) Check and/or select appropriate COM port (green circled).

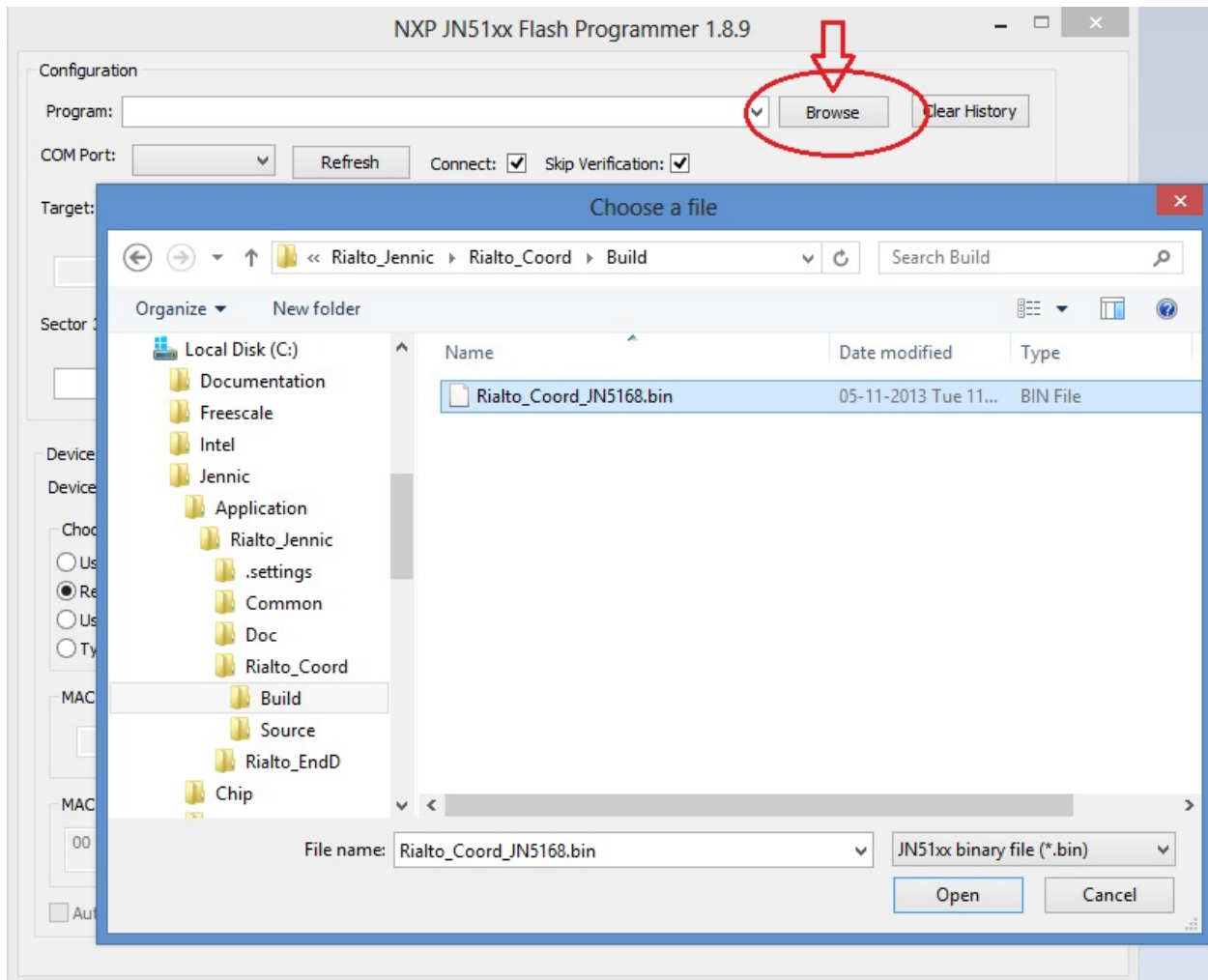
You can find out which serial communications port your PC has allocated to the Rialto Board by checking in the Control Panel→Hardware→device Manager→COM port

- 2) Click on “Refresh” button (blue arrow) and check the result (orange circled); if device not appear, check cable and USB connection (COM port installed)
- 3) Select in the Baud rate drop-down menu “500000” (for maximum programming time performance)
- 4) Check “Automatic Program and Reset” checkbox (brown arrow)



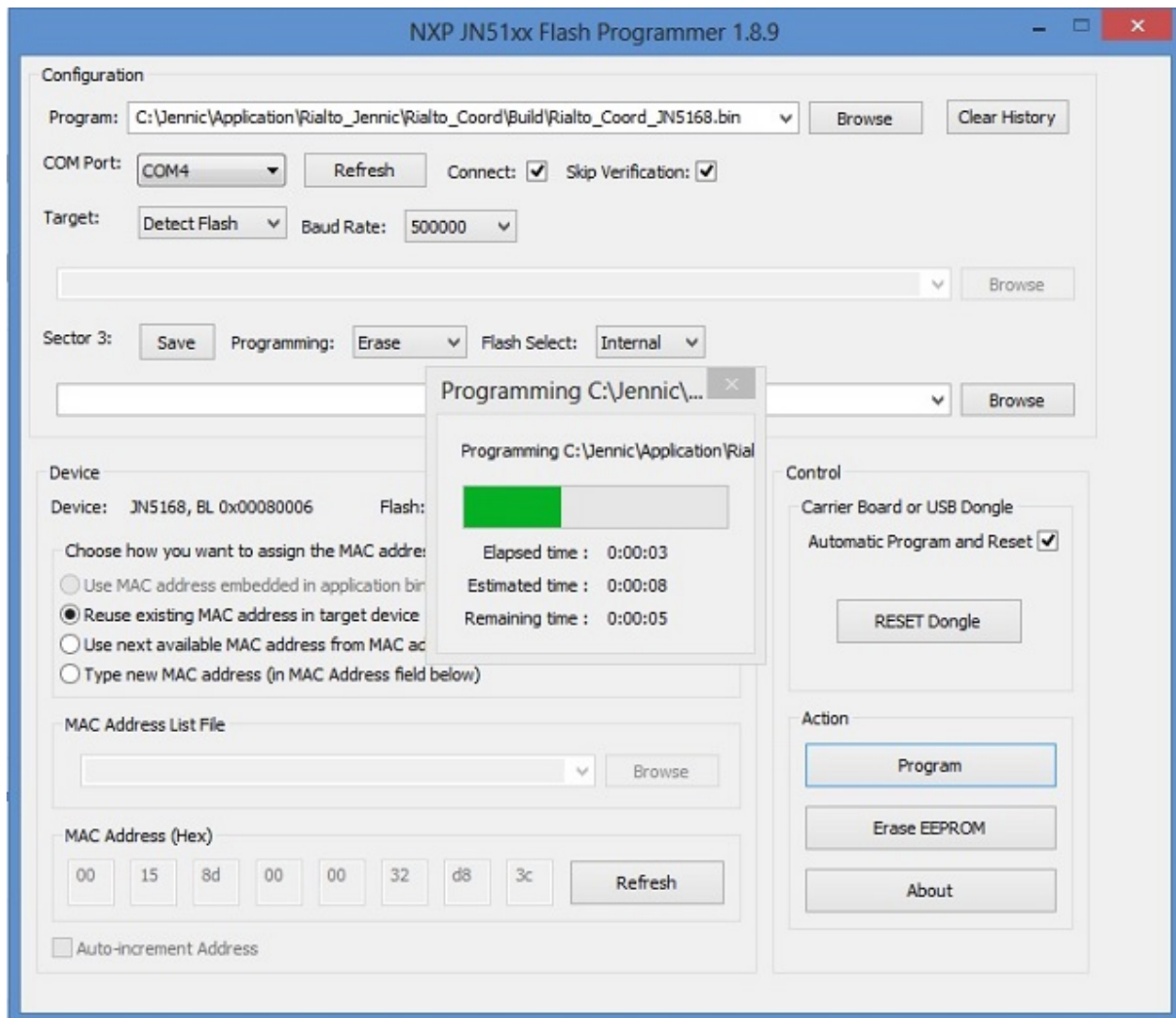
**Note:** Take care at COM port: if it's already busy with other applications (such as HyperTerminal), you cannot find it in COM port dropdown menu of Flash GUI Tool

- **Step 2:** click "Browse" button red circled, navigate to **C:\Jennic\Application\Rialto\_Jennic\Rialto\_Coord\Build** and select bin file **"Rialto\_Coord\_JN5168.bin"**. Then click "Open" button.



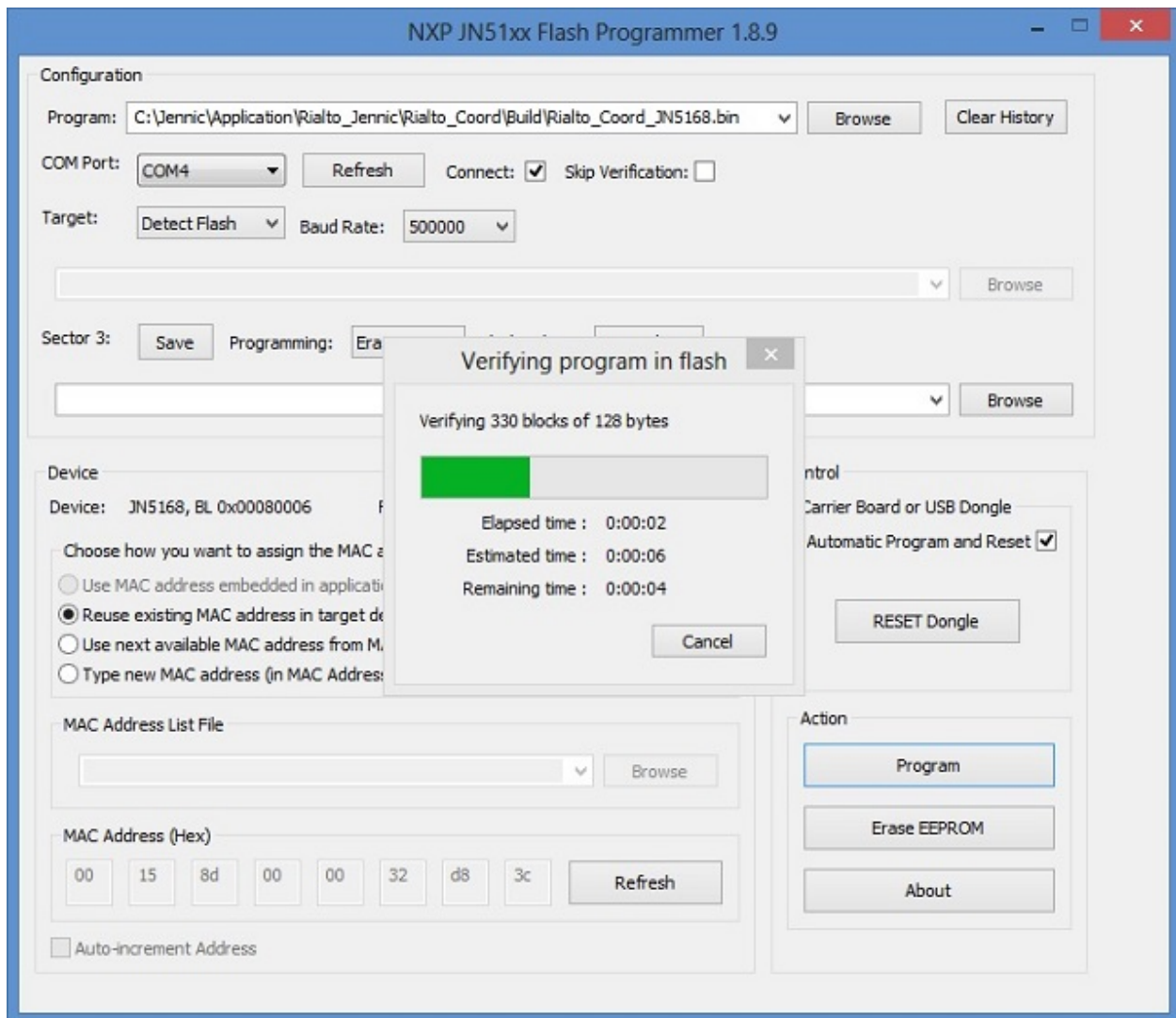
- **Step 3:** in the “Program” window will appear  
**C:\Jennic\Application\Rialto\_Jennic\Rialto\_Coord\Build\Rialto\_Coord\_JN5168.bin.**  
Click on “Program” button to start board programming.



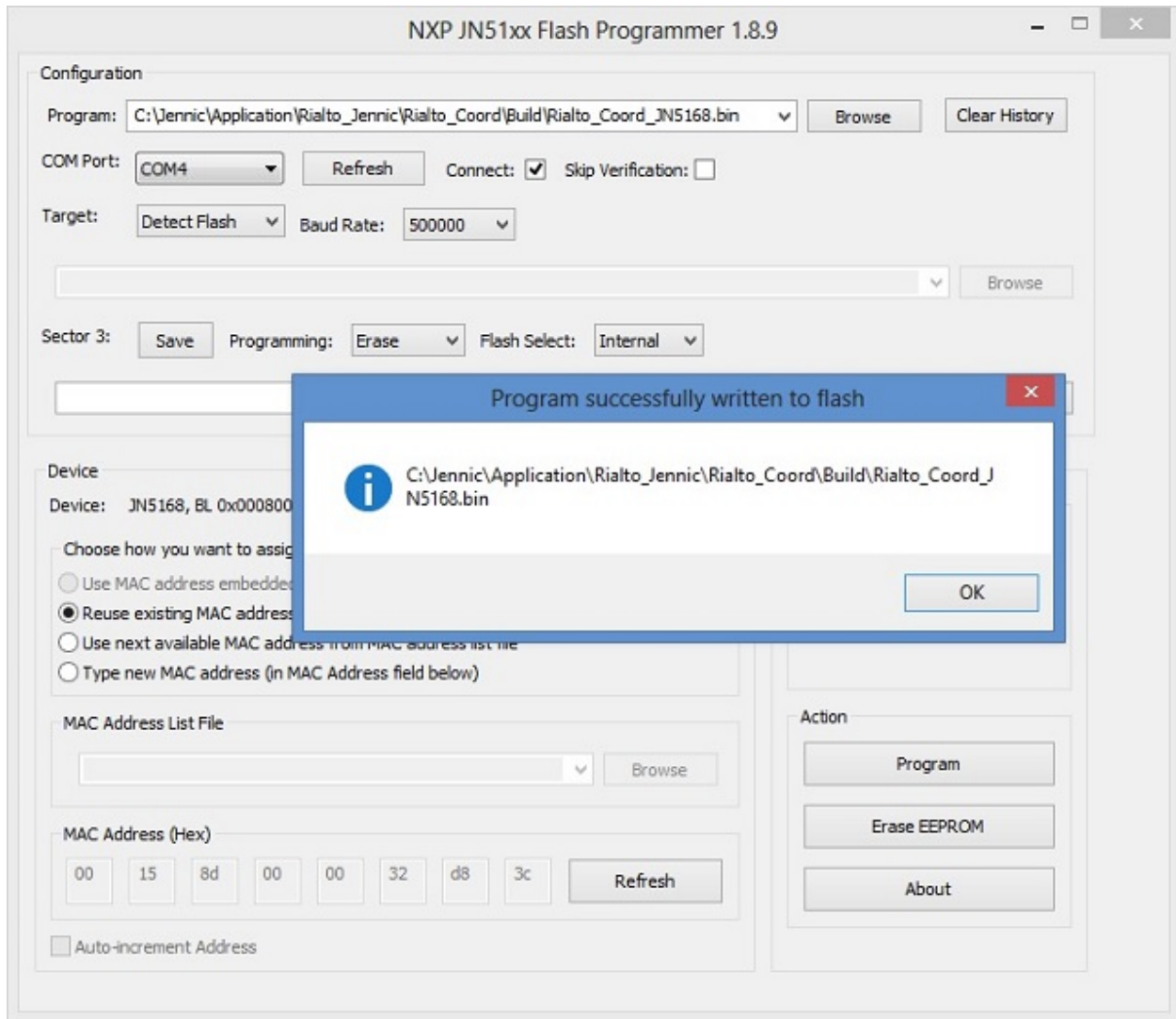


- **Step 4:** wait for verifying...

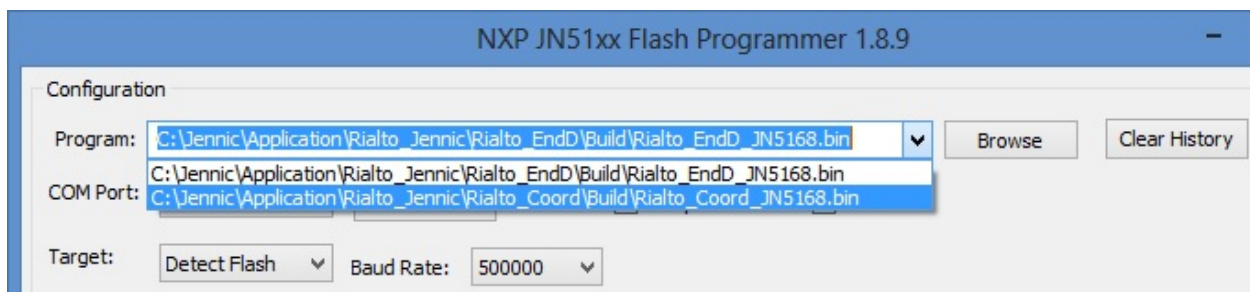




- **Step 5:** Click “OK” button and board will reset and start.



**Tip:** To program End-Node, repeat from step 2 but navigate to `C:\Jennic\Application\Rialto_Jennic\Rialto_EndD\Build` and select "Rialto\_EndD\_JN5168.bin". When you have programmed even Coordinator that End-node, you can select FW binary by clicking the drop-down arrow (red circled in image below) and select from drop-down menu.



Now you have finished all setup necessary to evaluate, debug and make changes inside Rialto Firmware.

You can plug the Rialto boards into USB ports of your PC, open HyperTerminal sessions and use Serial Monitor command for evaluate the main Firmware functions.

For further details you can read the guides:

*Serial Monitor guide*

*Tip & Tricks*

## “Ready to run” Rialto Kit

**Ready to run** is a pre-programmed kit that perform basic function for Jennic evaluation purpose. We can find two Rialto board, first one as Coordinator and second as End-Node.



---

**Important:** Before using Rialto Boards as a SerizII plugin, read note inside ‘important’ box on chapter [SerizII plugin](#)

---

## Working with Rialto board

Simply use of Rialto Board, is to plug Coordinator and End-Node on two USB port of your PC See at com port configured, and then open two HyperTerminal sessions to evaluate Rialto network

**Coordinator Serial Monitor main menu**

```

*****
* Rialto - JN5168 demo board *
*****

PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>_

```

Connected 00.00.15    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

### End-Node Serial Monitor main menu

```

*****
* Rialto - JN5168 demo board *
*****

PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data enabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>

```

Connected 00.00.12    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

By default, Coordinator has “verbose mode” enabled and End-Node has “auto send sensors data” enabled. “Verbose



mode” shows for every transaction (both TX and RX) more data detail, such as HEX format (enclosed in square bracket) When verbose mode is disabled, Serial Monitor gets out only the string or data. It is useful to receive VCARD data from SerizII and save these in VCF format.

“Auto send sensors data” performs read&send sensors data (ADXL362 and SENS300), every time that End-Node receive a poll from Coordinator. When disabled, auto send function will not work.

---

**Note:** For more details about Serial Monitor, see [Serial Monitor guide](#)

---

### First Lab

The Lab1 that you can perform, is by following the next steps:

- Connect Coordinator and End-Node Rialto boards on USB port of your PC
- Open terminals, connect appropriate serial port and check correct settings (see [HyperTerminal settings](#))
- Wait for network connection (blue-led will flash fast both in Coordinator and End-Node)
- See at red-led flash (poll, trasmit or receive, default is 8 seconds)
- See at terminals windows: you can see regullary (every poll-time), both coordinator and end-node, details of sensors data trasmitted and received
- Disable End-Node “Auto send sensors data” typing command ‘!’.
- Change End-Node PAN.ID (i.e. 1234) typing ‘C’ command. Board will restart and then wait for new Coordinator (blue-led flash slow)
- See blue-led of Coordinator board: when it flash slow, the Coordinator detect poll failure and automatically dissociate End-Node (it will take about 30-35 seconds). Serial monitor will inform when it occur.
- Now change Coordinator PAN.ID (‘C’ command) and set same as End-Node previously changed (i.e. 1234) Coordinator restarts itself.
- In a few seconds, End-Node will reassociate with new Coordinator and start to send sensors data.
- Enter End-Node sleep mode (typing ‘Z’ command). When leds off, plug out End-Node. It will send every 15 seconds sensors data to Coordinator. In sleep-mode the power drain by the end-node is very low. You can take out from USB port and the End-node will run on the small supercap on the board for few minutes.

---

**Important:** Sleep Wake Time can be set changing default definition in define.h file. For more information, read [Poll and Sleep settings](#)

---

### Lab 2 with SerizII

The purpose af Lab2 is to show how to add a new end-node to an exhisting network, by setting authomatically the end-node PAN-ID to mach the exhisting network PAN-ID. The processo is: - the network is running ( coordinator on PC ) - the end-node to be added, which is the SerizII, will read the PAN-ID stored on the NTAG on the ccordinator. - the end-node PAN-ID is set to the network PAN-ID - data can be now exchanged by end node to coordinator, the Vcard data.

Rialto board is designed as a SerizII plugin (see [SerizII plugin](#)). In Lab1, plug End-Node board on SerizII and Coordinator on USB port of your PC. If you want, you can connect Rialto End-Node board, plugged on SerizII, to PC using USB type-A extension cable. Serial Monitor will run, without affecting SerizII communications. Then, follow next steps:

- Open terminals, connect appropriate serial port and check correct settings (see [HyperTerminal settings](#))
- See SerizII display: it will inform that JN5168 End-Node is installed.
- Wait for End-Node restart. See blue-leds flash, that will be slow. No network connection and no device associated. (You can also verify by typing ‘N’ command on Coordinator serial monitor)
- If the PC serial monitor is connect to the End-Node, on main menu you can see that ‘Auto send sensors data’ is disabled, because of this is a default setting by SerizII connection.
- Take Coordinator near SerizII RFID antenna: the Seriz will read the Coordinator PAN.ID from the onboard Ntag. SerizII will update the End-Node PAN.ID.
- End-Node will restart, and in a few seconds will be associated to the Coordinator.
- Blue-leds will flash fast to indicate that Network is connected, and also the SerizII display will show this.
- Disable verbose mode on the Coordinator (typing ‘!’ command)
- Place the business card in near SerizII RFID antenna; see at Serial Monitor windows of Coordinator and you can see all data readed from business card.
- You can copy and paste business data (using NotePad) and save this file with extension **.vcf** and directly import to your contact database.

---

**Important:** When End-Node is connected to SerizII, Sleep mode entering is not allowed. Even ‘Auto send sensors data’ is disabled, but you can enable it using ‘!’ Serial Monitor command.

---

### Lab 3 with SerizII

The purpose af Lab3 is to show how to add a new end-node to an exhisting network, by setting authomatically the end-node PAN-ID to mach the nexhisting network PAN-ID. The processo is: - the network is running ( ccordinator on SerizII ) - the default address of the end-node to be added is read by Seriz from the NTAG on the end-node - the coordinator is set to the end-node read PAN-ID, to be able to comunicate with the end-nod to be added - the end-node PAN-Id is set to the network PAN-ID - the coordinator is put back on the original networn PAN-ID, to which the end-node is now added

In Lab3, plug Coordinator on SerizII and End-Node on USB port of your PC. Simillary to Lab2, you can connect Rialto Coordinator board, plugged on SerizII, to PC. After this, see here steps for Lab2.

- Open terminals, connect appropriate serial port and check correct settings (see [HyperTerminal settings](#))
- See SerizII display: it will inform that JN5168 Coordinator is installed.
- Wait for Coordinator restart: blue-led will flash slow, no network is connected.
- Take End-Node near SerizII RFID antenna: the Seriz will read the End-Node PAN.ID from the onboard Ntag. SerizII will update the Coordinator PAN.ID, wait for End-Node association and will perform update of End-Node PAN.ID to the previos network PAN.ID.
- Wait for End-Node restart, and in a few seconds it will associate to the Coordinator.
- End-Node, every time that receive poll command, send sensors data to Coordinator. You can see these data both Coordinator and End-Node Serial Monitor windows
- SerizII display show sensors data received and network state for each End-Node associated: short Network address and state (Sleep or not). You can verify also using Serial Monitor command ‘N’.
- Now place End-Node in ‘Sleep mode’: SerizII will show the End-Node new state (sleep) and receive (every Sleep Wake Time) End-Node sensors data.

---

**Important:** Sleep Wake Time can be set changing default definition in define.h file. For more information, read [Poll and Sleep settings](#)

---

After this 3 Labs, use [Serial Monitor guide](#) and enjoy with Rialto Boards!!

## Firmware specification

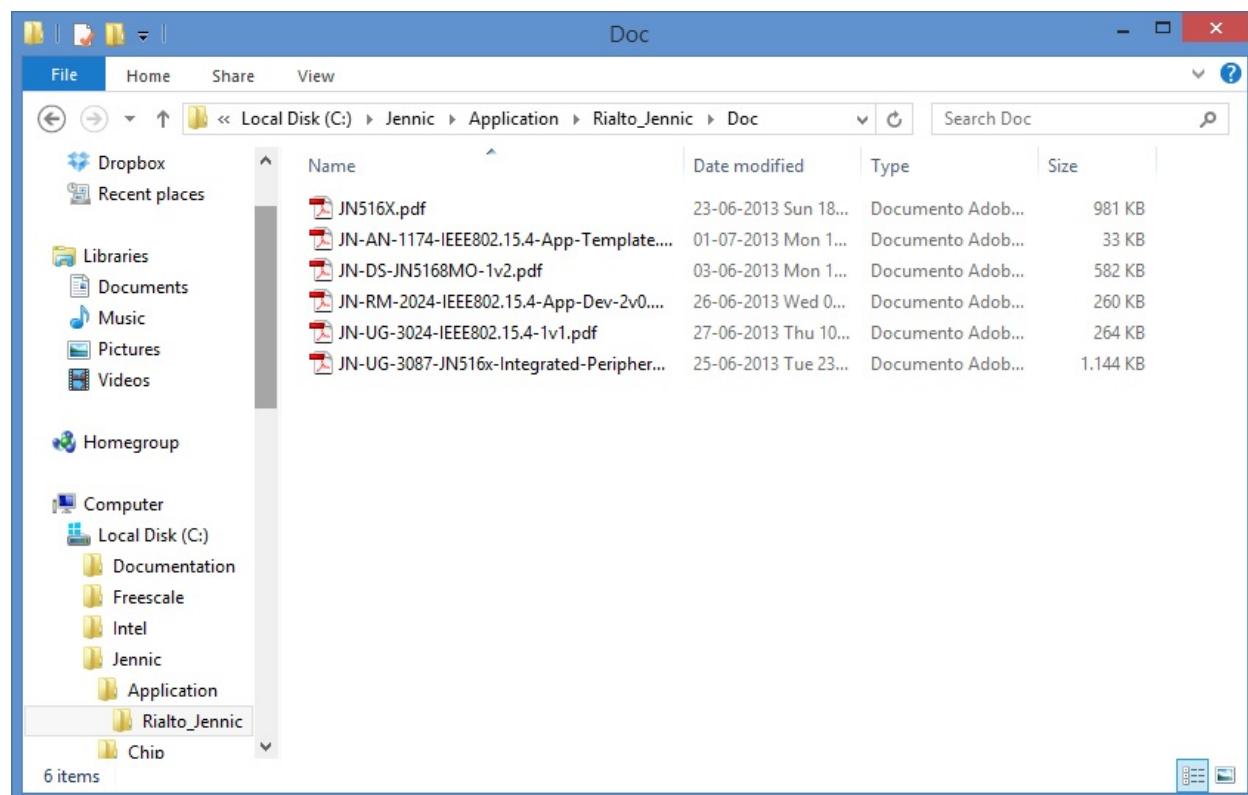
### Firmware overview

The Rialto\_Jennic firmware has been developed using **JN-AN-1174-JN516x-802-15-4-Application-Template** skeleton code.

This template include basic application of IEEE802-15-4 stack and some callback function that can be personalized and help end-user application development.

More detail regarding the structure of this template can be found on document

“JN-AN-1174-IEEE802.15.4-App-Template.pdf” inside “Doc” folder included in Rialto\_Jennic zip file



JN-AN-1174-IEEE802.15.4-App-Template.pdf reference manual refers to other documents in this folder, especially for:

**IEEE 802.15.4 Wireless Networks User Guide (JN-UG-3024)**

**IEEE 802.15.4 Application Development Reference Manual (JN-RM-2024)**



This is the complete set of guides to know the structure and operation of the skeleton code contained in the template

Other documents:

**JN-UG-3087-JN516x-Integrated-Peripherals-API.pdf** JN516x integrate peripherals API user guide (hardware library functions details)

**JN-DS-JN5168MO-1v2.pdf** JN5168 module hardware manual

**JN516X.pdf** JN5168 wireless microcontroller data sheet.

## Firmware restrictions

The application skeleton assumes the following restrictions (directly derived from JN-AN-1174 Application Template):

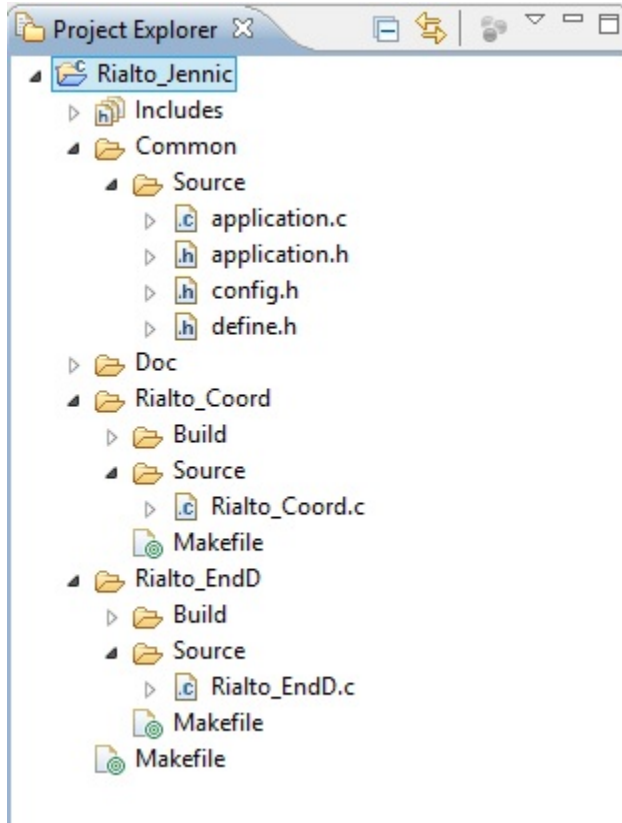
- You have one device which will act as the PAN Co-ordinator.
- You have at least one other device which will act as an End Device.
- You will use pre-determined values for the PAN ID and the short addresses (for the PAN Co-ordinator and for the End Device(s)).
- The network topology will be a Star network.
- The network will be non-beacon enabled (meaning that the PAN Co-ordinator will not transmit regular beacons).
- Short addressing will be used.
- Data transfers will be direct transmissions with acknowledgements.
- There will be no security implemented.

## Main firmware features

Rialto\_jennic firmware is based on IEEE802.15.4 network stack. The application also features a serial monitor which, with simple control commands, allows you to evaluate the status of the network and perform basic functions of data transmission / reception. The network topology is a “star”, with one Coordinator and more End-Node. The coordinator accepts up to a maximum of 8 End-Node, assigns them a short address and operates a regular polling of all devices associated to check your status. When a End-Node poll fails for 4 next times, the node is automatically dissociated from the Coordinator. Similary, when End-Node doesn’t receive poll and the polling timeout is expired, dissociates itself automatically and restarts the scan sequence to search for a coordinator. This is needed because that the network stack IEEE802.15.4, included in the JN-SW-4065 libraries, doesn’t seem manage orphan devices. In other words, this mode of operation (polling) allows for constantly monitoring the status of the network and update the presence or absence of networked devices.

Rialto Board End-Node firmware implements a demonstration of Low Power functionality. Using Serial Monitor command ‘Z’, you can place the board in “sleep mode”. As a result, Serial Monitor will be disabled, and any Serial Monitor commands will not performed when board is in sleep mode. Please note that restart is the only way to exit from sleep mode.

## Project structure



The application's file structure includes the following folders:

- **Rialto\_Coord** – contains source files and makefiles for the PAN Co-ordinator
  - **Rialto\_Coord.c** specific application file for Coordinator functions (see chapter 3.2.1 of “JN-RM-2024” resource manual for main details)
- **Rialto\_EndD** – contains source files and makefiles for an End Device
  - **Rialto\_EndD.c** specific application file for End-Node functions (see chapter 3.2.2 of “JN-RM-2024” resource manual for main details)
- **Common** – contains:
  - **application.c** common functions used both by coordinator that end-node
  - **application.h** common functions and variables global definition
  - **define.h** common costant defines
  - **config.h** network parameter and macro definition

## Firmware & documents download

Firmware project file Rialto\_Jennic.zip, installation file Install.zip and all other documentation for the Rialto project can be found at Silica ArchiTech page. Registration is needed to access at download section. Click [here](#) to go to ArchiTech main page.

## Serial Monitor guide

### Using serial monitor

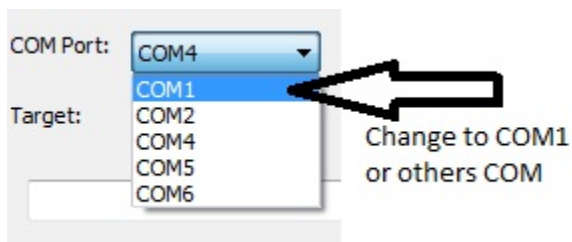
You can use USB built-in dongle on Rialto Board for programming or Serial Monitor purpose.

The only attention is that you must “deselect” COM port in the Flash GUI Tool (or close program) before connect with Terminal (i.e. HyperTerminal)

This is because the same COM port is used by Flash Programmer and (when application firmware starts) by Serial Monitor.

*Example: if your Rialto Board is connected to COM4, if possible, change COM Port settings in Flash GUI Programmer BEFORE attempting to connect HyperTerminal*

Best way is to close NXP Flash GUI Tool before connect Terminal to device COM port.



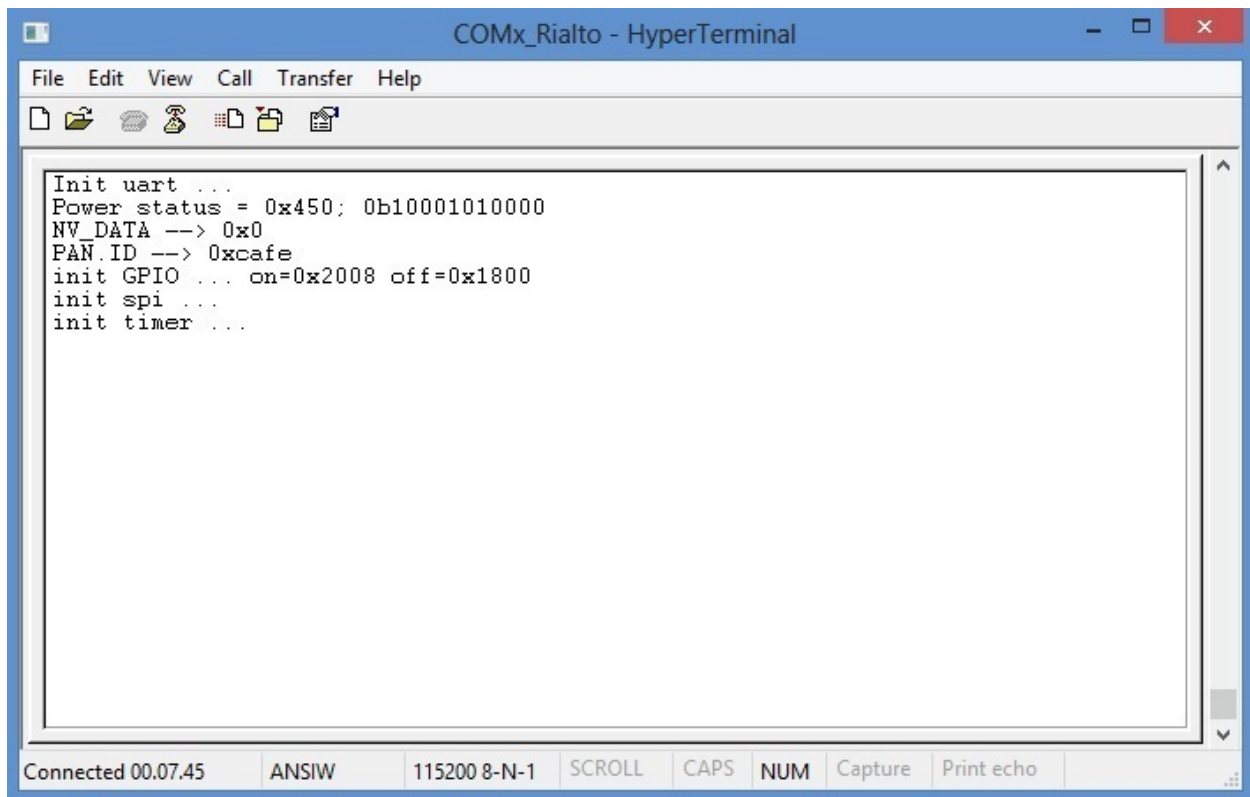
---

**Note:** Startup messages are same in End-Node and Coordinator

---

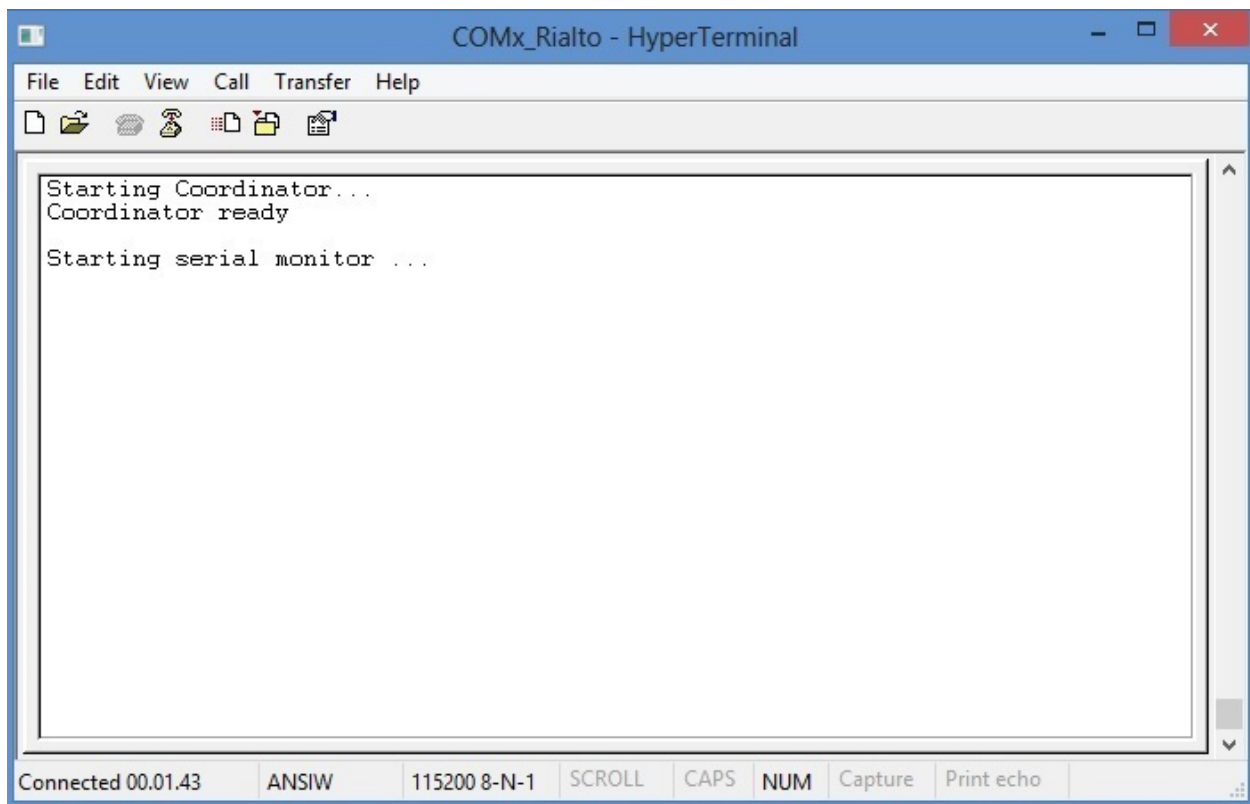
### Coordinator Serial Monitor

When Coordinator is power up, it sends (in sequence):



```
Init uart ...
Power status = 0x450; 0b10001010000
NV_DATA --> 0x0
PAN.ID --> 0xcafe
init GPIO ... on=0x2008 off=0x1800
init spi ...
init timer ...
```

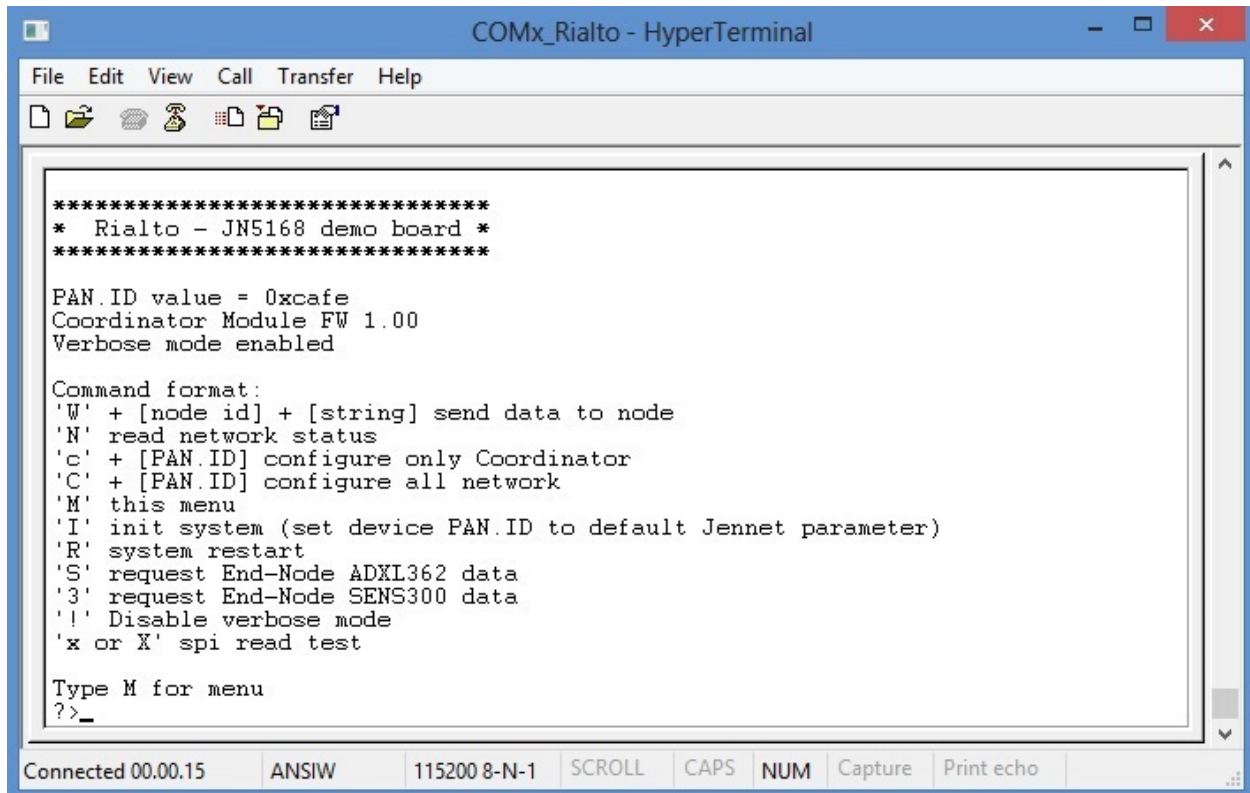
Connected 00.07.45   ANSIW   115200 8-N-1   SCROLL   CAPS   NUM   Capture   Print echo



```
Starting Coordinator...
Coordinator ready
Starting serial monitor ...
```

Connected 00.01.43   ANSIW   115200 8-N-1   SCROLL   CAPS   NUM   Capture   Print echo

and after starts main menu as follow:



```

COMx_Rialto - HyperTerminal

File Edit View Call Transfer Help

*****
* Rialto - JN5168 demo board *
*****

PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

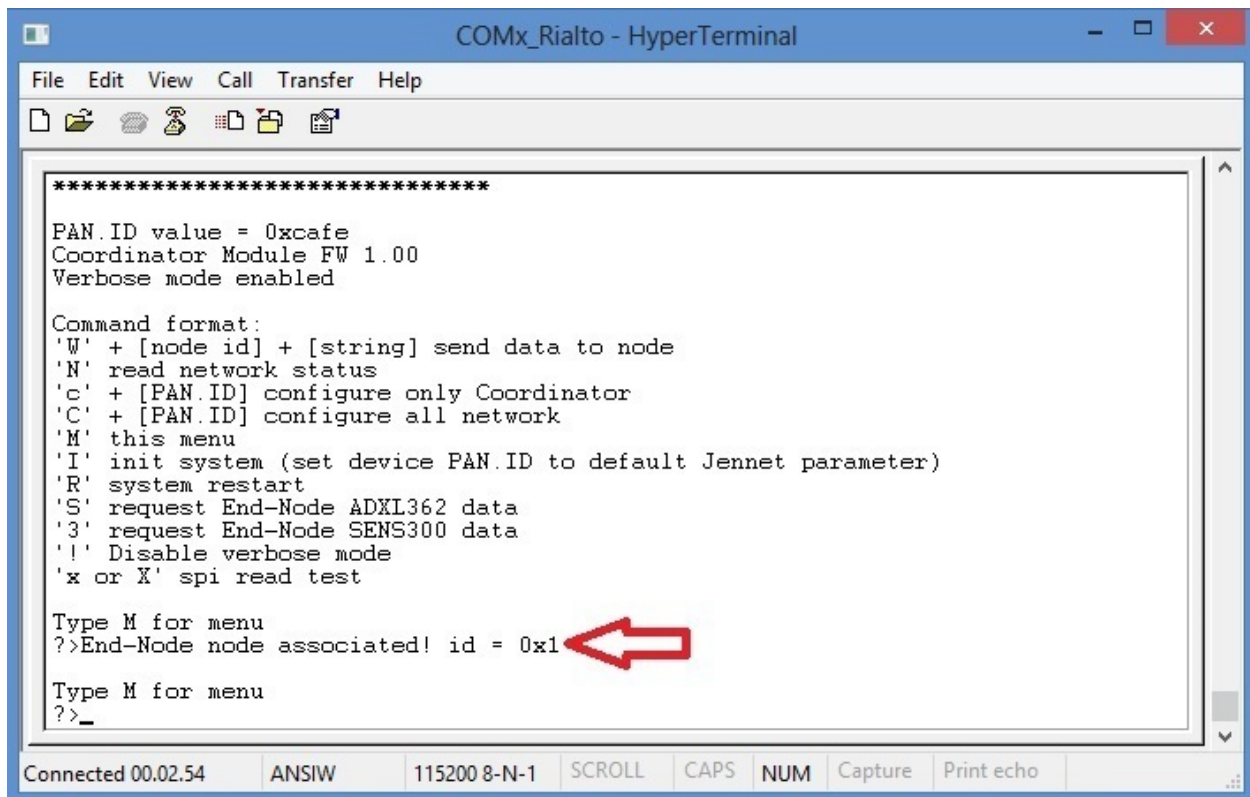
Type M for menu
?>_

Connected 00.00.15  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

### Serial Monitor main messages and commands

- End-Node detected and associated



```

COMx_Rialto - HyperTerminal

File Edit View Call Transfer Help

*****

PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>End-Node node associated! id = 0x1
Type M for menu
?>_

Connected 00.02.54  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

- End-Node poll fail and dissociated (details orange circled)

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>End-Node node associated! id = 0x1

Type M for menu
?>poll fail! End_node dissociated id->0x1 short->0x1 numfail-->0xffffffff

Type M for menu
?>

Connected 00.05.41  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

- Send data to End-Node (“W” command)

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>
Type the string in the following format:
'end-node address(decimal), string(ascii)'
send with 'enter' key - abort with 'end' key
W.DATA-->1.test
Take care at command format: [end-node address] + comma + [string to send]

Connected 00.08.05  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

- Sendig data to End-Node result

Red circled: detail of data message

Blue circled: sending End\_Node address

Green circled: data string hex format

Yellow circled: data string terminator (same for all data sending)

Orange circled: Data sending result

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

'x or X' spi read test

Type M for menu  
?>End-Node node associated! id = 0x1

Type M for menu  
?>

Type the string in the following format:  
'end-node address(decimal), string(ascii)'  
send with 'enter' key - abort with 'end' key

W.DATA-->1.test  
data\_string = 1 test  
End-Node address 0x1  
hex data to send:  
[0x74][0x65][0x73][0x74][0xd][0xa][0x0]  
Data to send len = 0x7; add = 0x1; string = test

Data transmitted

Type M for menu  
?>frame tx ok!!

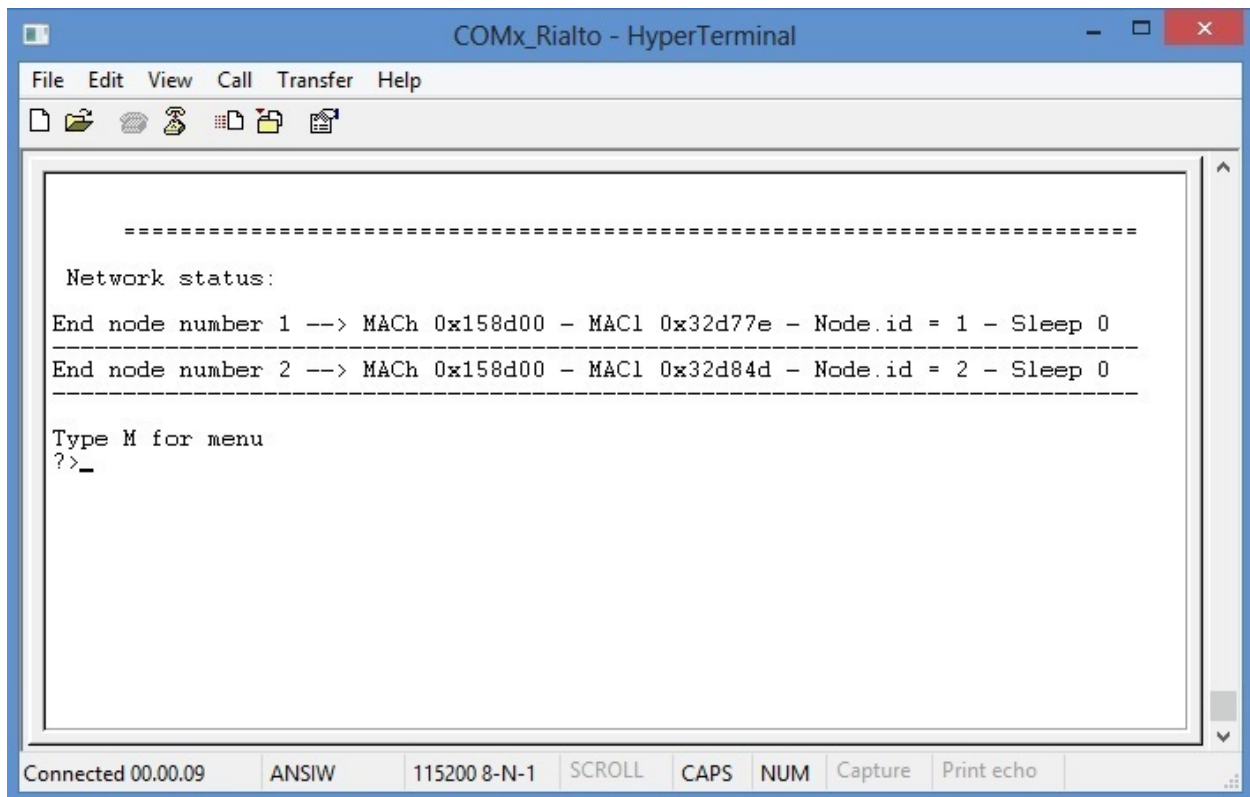
Type M for menu  
?>

Connected 00.03.08    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

**Important:** In the Coordinator Monitor **W** command will send a “string” to a selected End\_Node. You can see the received message inside End\_Node Serial Monitor addressed by Coordinator

- Network status (“N” command)





COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

-----

Network status:

End node number 1 --> MACH 0x158d00 - MAC1 0x32d77e - Node.id = 1 - Sleep 0

-----

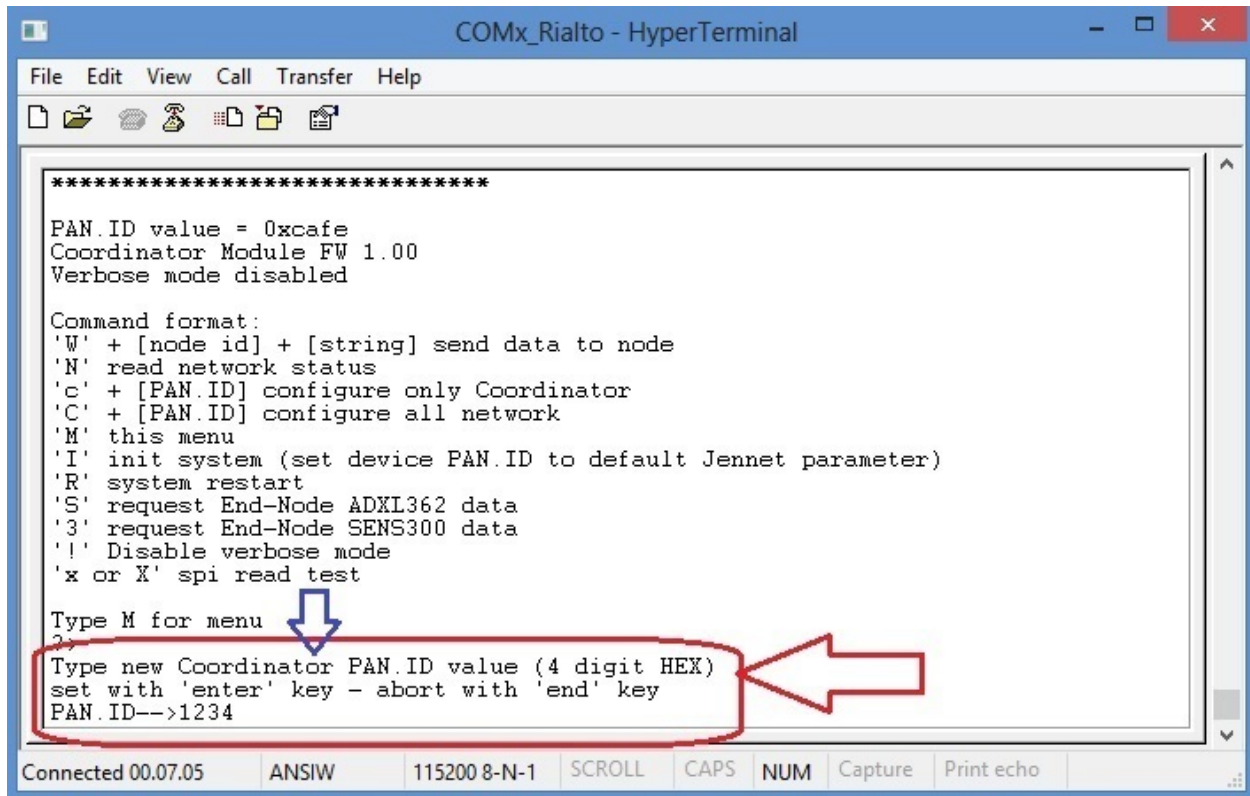
End node number 2 --> MACH 0x158d00 - MAC1 0x32d84d - Node.id = 2 - Sleep 0

-----

Type M for menu  
?>\_

Connected 00.00.09    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Set Coordinator PAN.ID ('c' command)



COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

\*\*\*\*\*

PAN.ID value = 0xcafe  
Coordinator Module FW 1.00  
Verbose mode disabled

Command format:

'W' + [node id] + [string] send data to node  
'N' read network status  
'c' + [PAN.ID] configure only Coordinator  
'C' + [PAN.ID] configure all network  
'M' this menu  
'I' init system (set device PAN.ID to default Jennet parameter)  
'R' system restart  
'S' request End-Node ADXL362 data  
'3' request End-Node SENS300 data  
'!' Disable verbose mode  
'x or X' spi read test

Type M for menu  
?>

Type new Coordinator PAN.ID value (4 digit HEX)  
set with 'enter' key - abort with 'end' key  
PAN.ID-->1234

Connected 00.07.05    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

Please, note that 'c' command (lowercase) set only coordinator PAN.ID. To configure all network PAN.ID, use 'C'

command (uppercase)

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

```

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>
Type new Coordinator PAN.ID value (4 digit HEX)
set with 'enter' key - abort with 'end' key
PAN.ID-->1234
New Coordinator PAN.ID = 0x1234
pan=0x1234; p=0x0
pan=0x1234; p=0xaa551234
Restart system to set new pan.id configuration ... wait
=

```

Connected 00.09.46    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Set network PAN.ID (“C” command)

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

```

*****
PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>
Type new NETWORK PAN.ID value (4 digit HEX)
set with 'enter' key - abort with 'end' key
PAN.ID-->1234

```

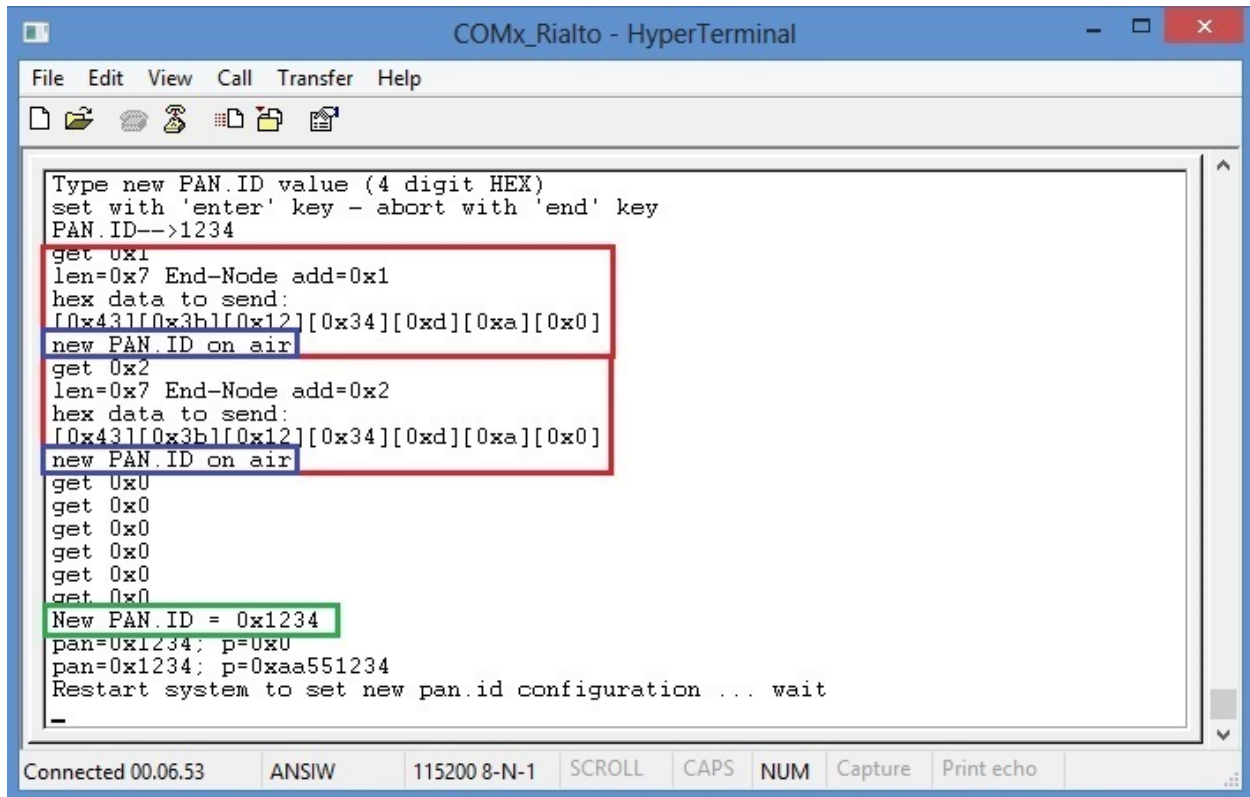
Connected 00.13.45    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- See image below:

Red circled: detail of PAN.ID configuration message

Blue circled: PAN.ID message status

Green circled: PAN.ID network global settings



```
COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
Type new PAN.ID value (4 digit HEX)
set with 'enter' key - abort with 'end' key
PAN.ID-->1234
get 0x1
len=0x7 End-Node add=0x1
hex data to send:
[0x43][0x3b][0x12][0x34][0xd][0xa][0x0]
new PAN.ID on air
get 0x2
len=0x7 End-Node add=0x2
hex data to send:
[0x43][0x3b][0x12][0x34][0xd][0xa][0x0]
new PAN.ID on air
get 0x0
get 0x0
get 0x0
get 0x0
get 0x0
get 0x0
New PAN.ID = 0x1234
pan=0x1234; p=0x0
pan=0x1234; p=0xaa551234
Restart system to set new pan.id configuration ... wait
-
Connected 00.06.53  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

*after system restart*

```

*****
* Rialto - JN5168 demo board *
*****
PAN.ID value = 0x1234
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>_

```

Connected 00.00.17    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Default PAN.ID (“I” command)

```

*****
PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>pan=0xcafe; p=0x0
pan=0xcafe; p=0xaa55cafe
Restart system to set new pan.id configuration ... wait
_

```

Connected 00.17.34    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Request End-Node ADXL362 data (“S” command)

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help

PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'l' Disable verbose mode
'x or X' spi read test

Type M for menu
?>
Type End-Node address to request ADXL362 data read:
'end-node address(decimal)'
send with 'enter' key - abort with 'end' key
S to End-Node-->1

```

Connected 00.01.36    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Received End-Node ADXL362 data

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help

send with 'enter' key - abort with 'end' key
S to End-Node-->1
data string = 1,S;
End-Node address 0x1
hex data to send:
[0x53][0x3b][0xd][0xa][0x0]
Data to send len = 0x5; add = 0x1; string = S;

Data transmitted

Type M for menu
?>frame tx ok!!

Type M for menu
?>
frame received len = 0xf
ADXL362 data from End-Node 0x1:
Xaxis=0xffc7; Yaxis=0xffff8; Zaxis=0x47f; TempSensor=0x1cf
data string in hex format:
[0x53][0x3b][0xff][0xc7][0xff][0xf8][0x4][0x7f][0x1][0xcf][0x0][0x1][0xd][0xa][0x0]

Type M for menu
?>

```

Connected 00.20.22    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Request End-Node SENS300 data (“3” command)



COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

PAN.ID value = 0xcafe  
Coordinator Module FW 1.00  
Verbose mode enabled

Command format:  
'W' + [node id] + [string] send data to node  
'N' read network status  
'c' + [PAN.ID] configure only Coordinator  
'C' + [PAN.ID] configure all network  
'M' this menu  
'I' init system (set device PAN.ID to default Jennet parameter)  
'R' system restart  
'S' request End-Node ADXL362 data  
'3' request End-Node SENS300 data  
'!' Disable verbose mode  
'x or X' spi read test

Type M for menu  
?>  
Type End-Node address to request SENS300 data read:  
'end-node address(decimal)'  
send with 'enter' key - abort with 'end' key

S to End-Node-->1

Connected 00.03.35    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Received End-Node SENS300 data

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

send with 'enter' key - abort with 'end' key  
S to End-Node-->1  
data string = 1,3;  
End-Node address 0x1  
hex data to send:  
[0x33][0x3b][0xd][0xa][0x0]  
Data to send len = 0x5; add = 0x1; string = 3;

Data transmitted

Type M for menu  
?>frame tx ok!!

Type M for menu  
?>

frame received len = 0xf  
SENS300 data from End-Node 0x1:  
Temperature=0x4a9b; Humidity=0x64fd; Ligth=0x8; Dummy=0x0  
data string in hex format:  
[0x33][0x3b][0x4a][0x9b][0x64][0xfd][0x0][0x8][0x0][0x0][0x0][0x1][0xd][0xa][0x0]  
]

Type M for menu  
?>

Connected 00.26.22    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

---

**Important:** All sensors data are transmitted in 32 bit HEX format, without any conversion in ASCII format. Values are position dependent, and must be read scanning the data string received

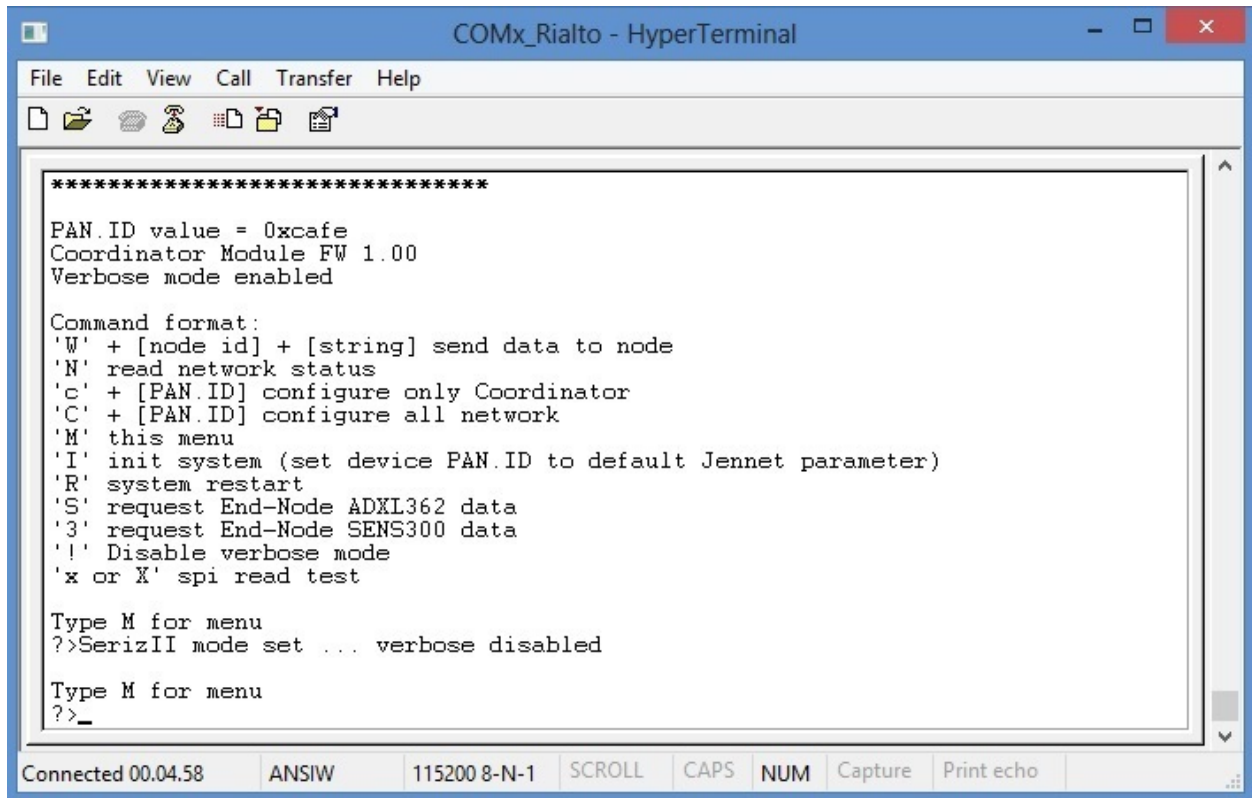
---

---

**Note:** In the Coordinator Monitor **S** and **3** command will answers a selected End\_Node to read and send sensor data. You can see sensor data both in End\_Node monitor and Coordinator monitor

---

- Disable verbose mode ("!" command)



```
*****
PAN.ID value = 0xcafe
Coordinator Module FW 1.00
Verbose mode enabled

Command format:
'W' + [node id] + [string] send data to node
'N' read network status
'c' + [PAN.ID] configure only Coordinator
'C' + [PAN.ID] configure all network
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' request End-Node ADXL362 data
'3' request End-Node SENS300 data
'!' Disable verbose mode
'x or X' spi read test

Type M for menu
?>SerizII mode set ... verbose disabled

Type M for menu
?>_

Connected 00.04.58  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

---

**Note:** In the Coordinator Serial Monitor, verbose mode is also set by SerizII. With verbose mode disabled, Serial Monitor messages are limited to data string and few other information. Check by yourself this operation mode.

---

## End-Node Serial Monitor

- End-Node main menu



```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
*****
* Rialto - JN5168 demo board *
*****

PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data enabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>

```

Connected 00.00.12    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Coordinator found, End-Node association

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data enabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

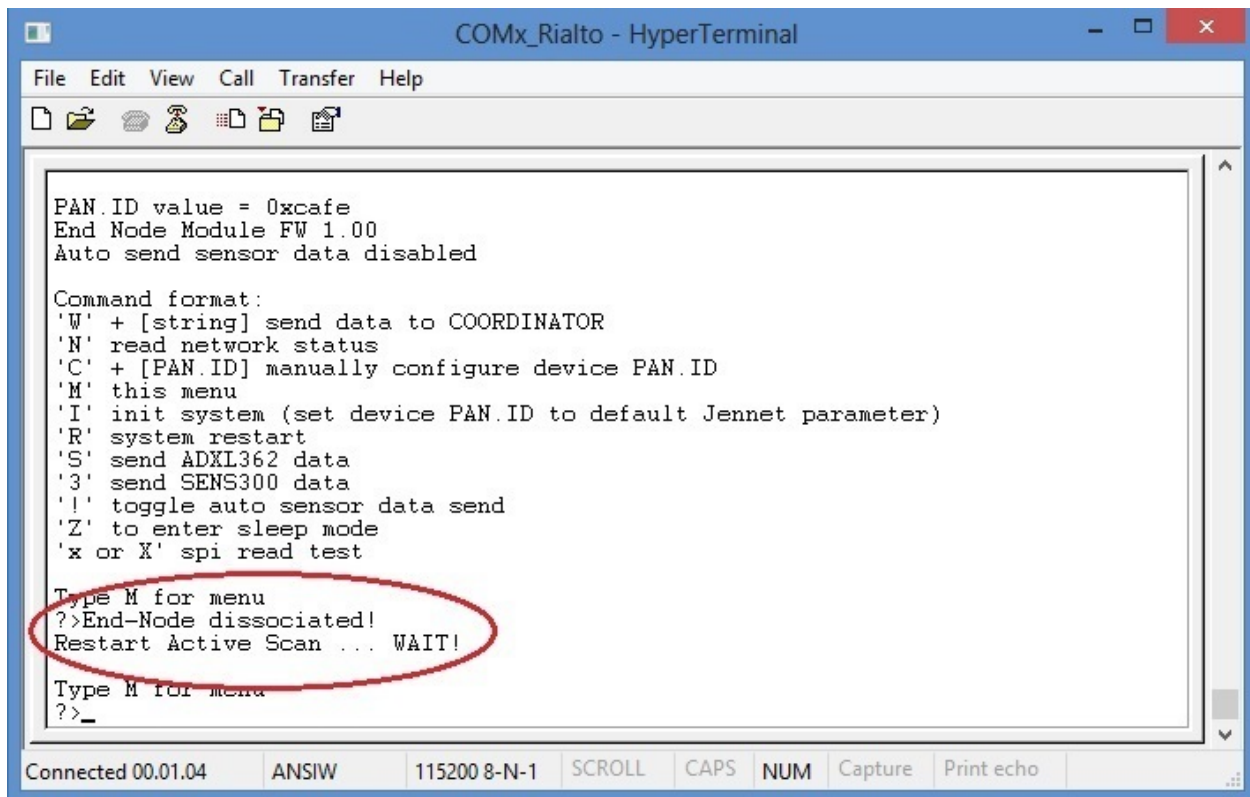
Type M for menu
?>start association
Node associated id=0x1

Type M for menu
?>

```

Connected 00.22.29    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Coordinator lost, End-Node dissociation



```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data disabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

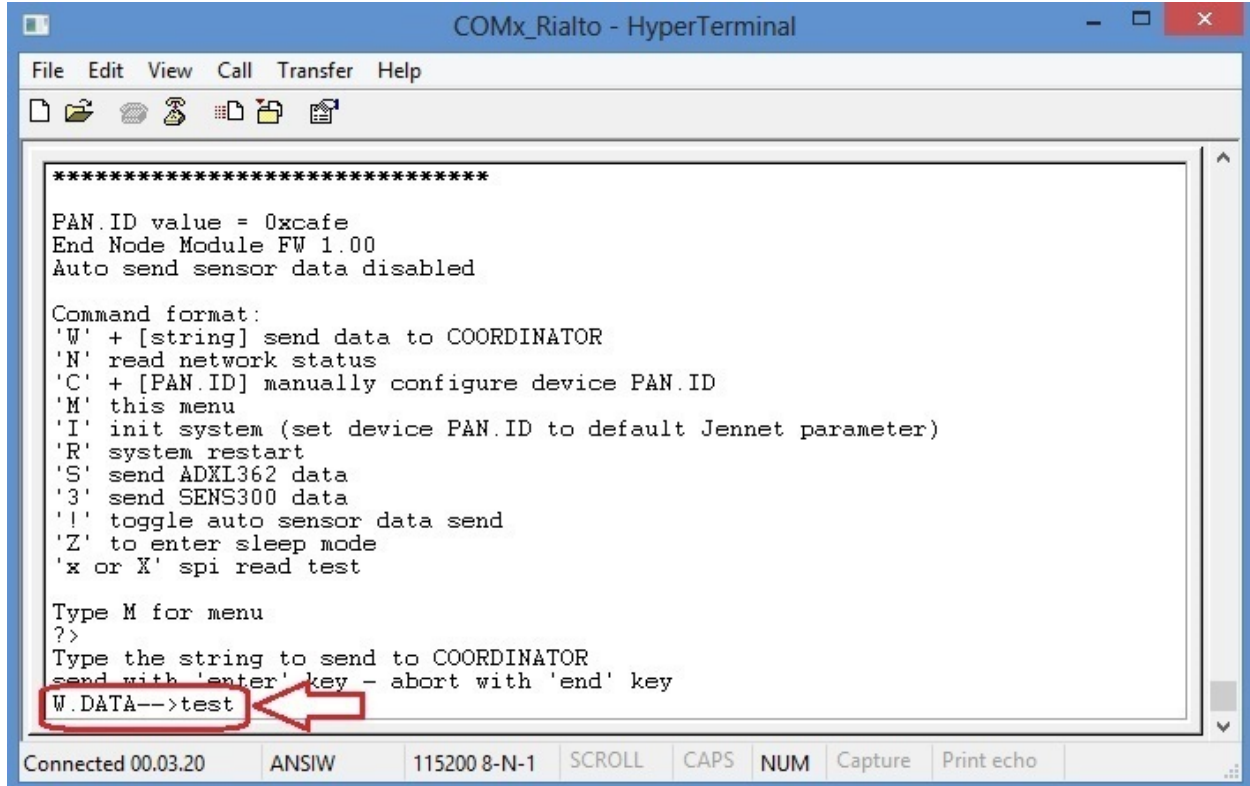
Type M for menu
?>End-Node dissociated!
Restart Active Scan ... WAIT!

Type M for menu
?>_

```

Connected 00.01.04    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Set data to send to Coordinator (“W” command)



```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
*****
PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data disabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>
Type the string to send to COORDINATOR
send with 'enter' key - abort with 'end' key
W.DATA-->test

```

Connected 00.03.20    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- Data sending

Blue arrow: string to send

Orange circled: hex string format

Red circled: End-Node address that send the message

Blue circled: Data sending result

```

COMx_Rialto - HyperTerminal
File Edit View Call Transfer Help
[Icons]
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>
Type the string to send to COORDINATOR
send with 'enter' key - abort with 'end' key
W.DATA-->test
data string = test
hex data to send:
[0x74][0x65][0x73][0x74][0x0][0x1] 0xd][0xa][0x0]
Data to send len = 0x9, add = 0x0, string = test
END-NODE sending ...
Data transmitted

Type M for menu
?>frame tx ok!!

Type M for menu
?>_

```

Connected 00.08.25    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- End-Node network status (“N” command)

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

-----

Network status:

Node.id = 1 - PAN.ID = 0xcafe - Coordinator Address = 0x0

-----

Type M for menu  
?>

Connected 01.22.32    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- End-Node ADXL362 data sending (“S” command)

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

'I' init system (set device PAN.ID to default Jennet parameter)  
'R' system restart  
'S' send ADXL362 data  
'3' send SENS300 data  
'!' toggle auto sensor data send  
'Z' to enter sleep mode  
'x or X' spi read test

Type M for menu  
?>Sensor read

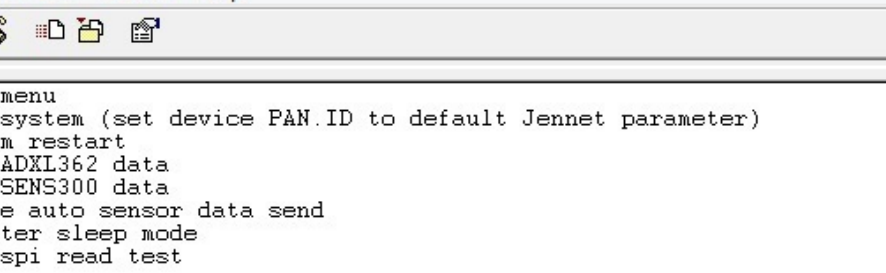
ADXL362 Xaxis=0xffc7; Yaxis=0xffff7; Zaxis=0x464; TempSensor=0x1b8  
len = 0xi  
ADXL362 hex data to send:  
[0x53][0x3b][0xff][0xc7][0xff][0xf7][0x4][0x64][0x1][0xb8][0x0][0x1][0xd][0xa][0x0][0x0]  
END-NODE sending  
ADXL362 data on air

Type M for menu  
?>frame tx ok!!

Type M for menu  
?>\_

Connected 00.14.06    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- End-Node SENS300 data sending (“3” command)



```
File Edit View Call Transfer Help

'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>Reading S300 ...
s300 temperature = 0x4a20; hum = 0x83d5; light = 0x9
len = 0x1
SENS300 hex data to send:
[0x33][0x3b][0x4a][0x20][0x83][0xd5][0x0][0x9][0x0][0x0][0x0][0x1][0xd][0xa][0x0][0x0]
END NODE sending ...
SENS300 data on air

Type M for menu
?>frame tx ok!!

Type M for menu
?>_

Connected 00.00.23  ANSIW  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

- End-Node Auto Send Sensors Data toggle ("!" command)

disable auto send

COMx\_Rialto - HyperTerminal

File Edit View Call Transfer Help

\*\*\*\*\*

PAN.ID value = 0xcafe  
End Node Module FW 1.00  
Auto send sensor data enabled

Command format:  
'W' + [string] send data to COORDINATOR  
'N' read network status  
'C' + [PAN.ID] manually configure device PAN.ID  
'M' this menu  
'I' init system (set device PAN.ID to default Jennet parameter)  
'R' system restart  
'S' send ADXL362 data  
'3' send SENS300 data  
'!' toggle auto sensor data send  
'Z' to enter sleep mode  
'x or X' spi read test

Type M for menu  
?>sensor data auto send = FALSE

Type M for menu  
?>

Connected 00.12.39    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo



enable auto send

```

*****
PAN.ID value = 0xcafe
End Node Module FW 1.00
Auto send sensor data disabled

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>sensor data auto send = TRUE

Type M for menu
?>_

```

Connected 00.11.05    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

- End-Node Sleep Mode (“Z” command)

```

Command format:
'W' + [string] send data to COORDINATOR
'N' read network status
'C' + [PAN.ID] manually configure device PAN.ID
'M' this menu
'I' init system (set device PAN.ID to default Jennet parameter)
'R' system restart
'S' send ADXL362 data
'3' send SENS300 data
'!' toggle auto sensor data send
'Z' to enter sleep mode
'x or X' spi read test

Type M for menu
?>sensor data auto send = FALSE

Type M for menu
?>entering sleep mode ... wait
Send sleep information to Coordinator
END-NODE sending ...
frame tx ok!!
Disable serial monitor ...
enter sleep mode!

```

Connected 00.14.26    ANSIW    115200 8-N-1    SCROLL    CAPS    NUM    Capture    Print echo

---

**Important:** In End-Node module, configuration command “C” perform only local device PAN.ID settings. If you use this command, End-Node will dissociate himself until it find a Coordinator with same new PAN.ID. The setting sequence is identical to that of the Coordinator command “c”(lowercase)

---

## Tip & Tricks

### Stand-alone use of Rialto Boards

“Ready to run” Rialto boards are designed to quick and fast evaluation with PC and SerizII board.

You can use the board both connected with PC for stand-alone evaluation or personal purpose.

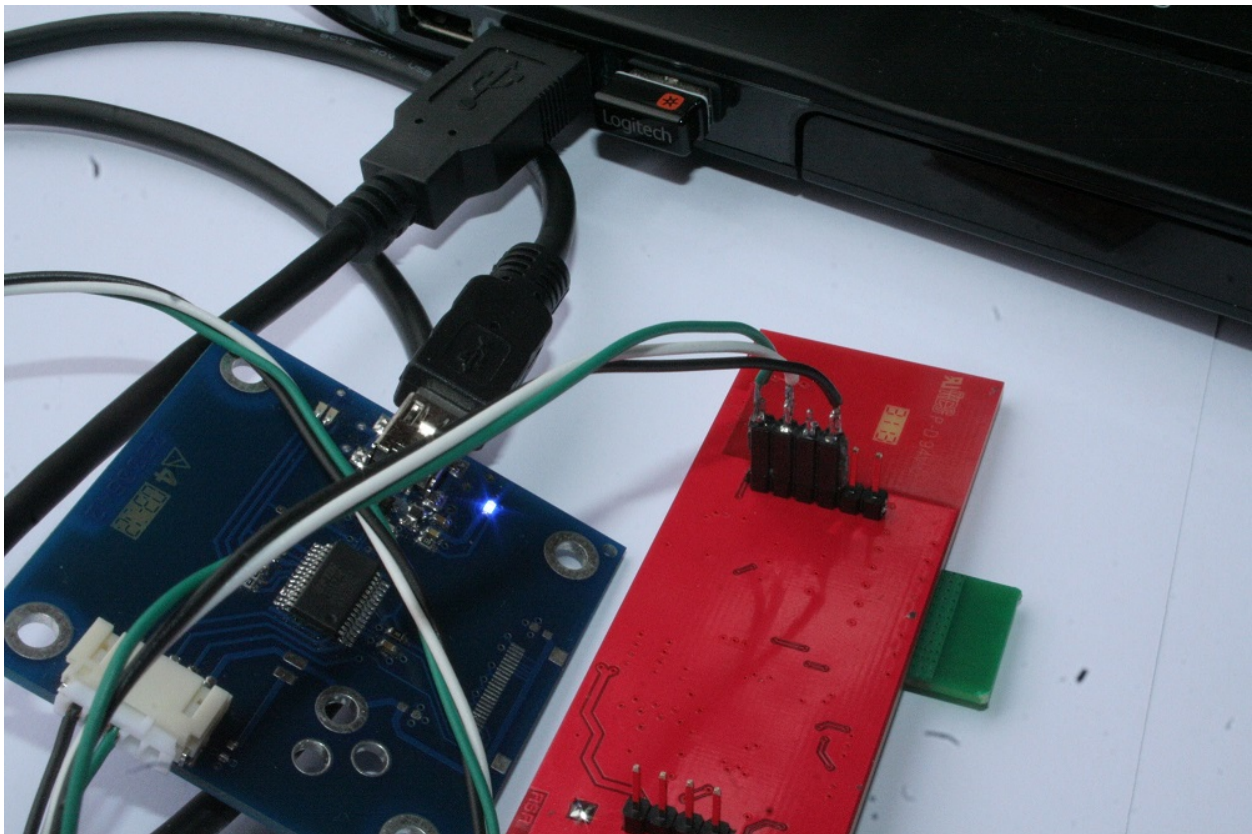
In this case, take care at USB com port installed for each board.

For full informations about, see *“Ready to run” Rialto Kit*

### Serial channel details

For best developement purpose, you can change serial monitor out:

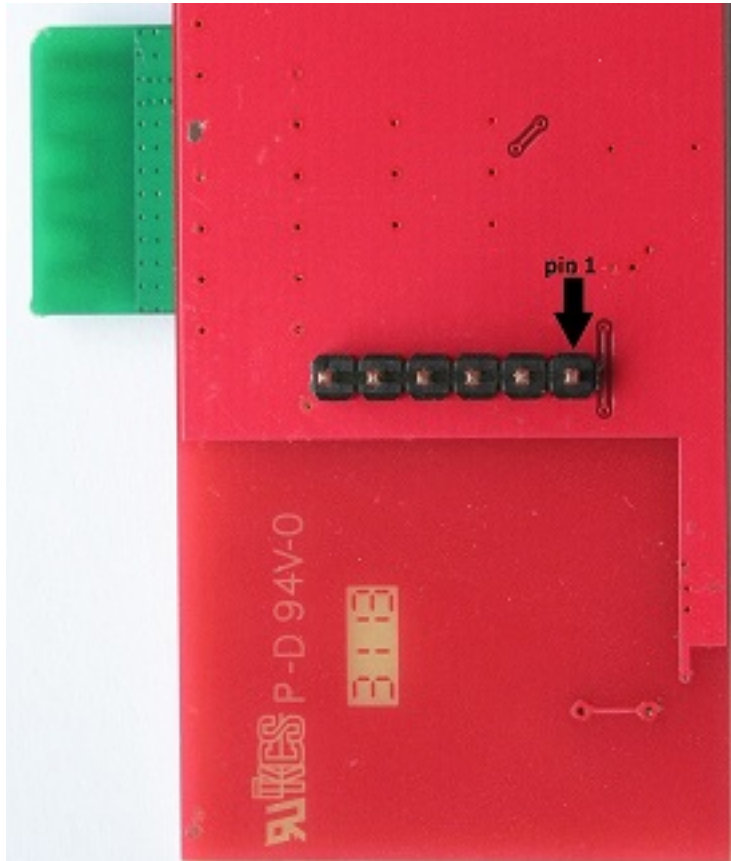
- USB to serial adapter (TTL out)
- Custom cable to connect 6 pin strip from Rialto Board to USB serial adapter





Rialto 6 pin strip schematich

```
pin 1 --> RxD TTL level
pin 2 --> TxD TTL level
pin 3 --> NC
pin 4 --> GND
pin 5 --> NC
pin 6 --> RESET
```



## How to change Serial Monitor out

At the top of **Rialto\_Coord.c** and **Rialto\_EndD** file you can find a variable statement

```
uint8 vprintf_uart = E_AHI_UART_0;
```

```
/*
*****
Serial Monitor define
*****
*/

uint8 vprintf_uart = E_AHI_UART_0;
```

The Coordinator and End-Node, by default, use USB dongle serial channel (E\_AHI\_UART\_0) for both programming and serial monitor. You can change it to E\_AHI\_UART\_1, and serial monitor will communicate using serial channel connected to 6 pin strip.

---

**Important:** Serial channel 1 (E\_AHI\_UART\_1) is default communication channel to SerizII board. When you switch Rialto Serial Monitor out to this channel, all SerizII communication functions will be disabled. The result of this change is the loss of functionality with SerizII

---

## Poll and Sleep settings

In the file **define.h** you can find the Poll and Sleep settings. By default, polling is executed every 8 seconds, the number of maximum poll failure is 4. This gives a maximum non-response time equal to 32 seconds: if End-Node will not be found for this timing, Coordinator will disassociate it from Network. End-Node use instead a polling timeout, that is equal  $((\text{POLL\_TIME} * \text{MAX\_FAIL}) + 1)$  seconds, and by default gives a value of 33 seconds. Then, the time necessary for disassociation will be just over 30 seconds.

The Sleep time is by default 15 seconds: this means that every End-Node in sleep mode will awakes every 15 seconds, sends sensors data and will return in sleep mode. With this setting, the sleep mode has a duration of about 5 minutes. After this time, the device dies. You can change the Sleep Time definition and make it more great to obtain a more long device life.

When in Sleep-mode, if End-Node will not find a valid Coordinator for SLEEP\_RETRY times, stops itself with Blue-Led on. By default, this gives End\_Node timeout equal to 45 seconds. After this time, End-Node lights up Blue-Led and let it die. You can change number of SLEEP\_RETRY in order to have more retry (only during sleep mode)

```

/*****
* SLEEP and POLL SETTINGS
*****/

// POLL definition --> leave before MAX_SLEEP_FAIL definition
// Coordinator
#define POLL_TIME ONE_SEC*8
#define MAX_FAIL 4
// End-Node
#define POLL_TIMEOUT (POLL_TIME * MAX_FAIL)+ONE_SEC

//End-Node SLEEP MODE time definition --> seconds - leave before MAX_SLEEP_FAIL_
↳definition
#define SLEEP_TIME 15

//Max End-Node sending retry --> only valid in sleep mode
#define SLEEP_RETRY 3

//Coordinator SLEEP MODE fail definition
#if ((SLEEP_TIME*MAX_FAIL) >= ((POLL_TIME/ONE_SEC)*MAX_FAIL))
#define MAX_SLEEP_FAIL ((SLEEP_TIME*MAX_FAIL)/(POLL_TIME/ONE_SEC)+1)
#else
#define MAX_SLEEP_FAIL = MAX_FAIL
#endif

```

## Led functions

Rialto board has 2 LEDs, one blue and one red, which indicate the following features

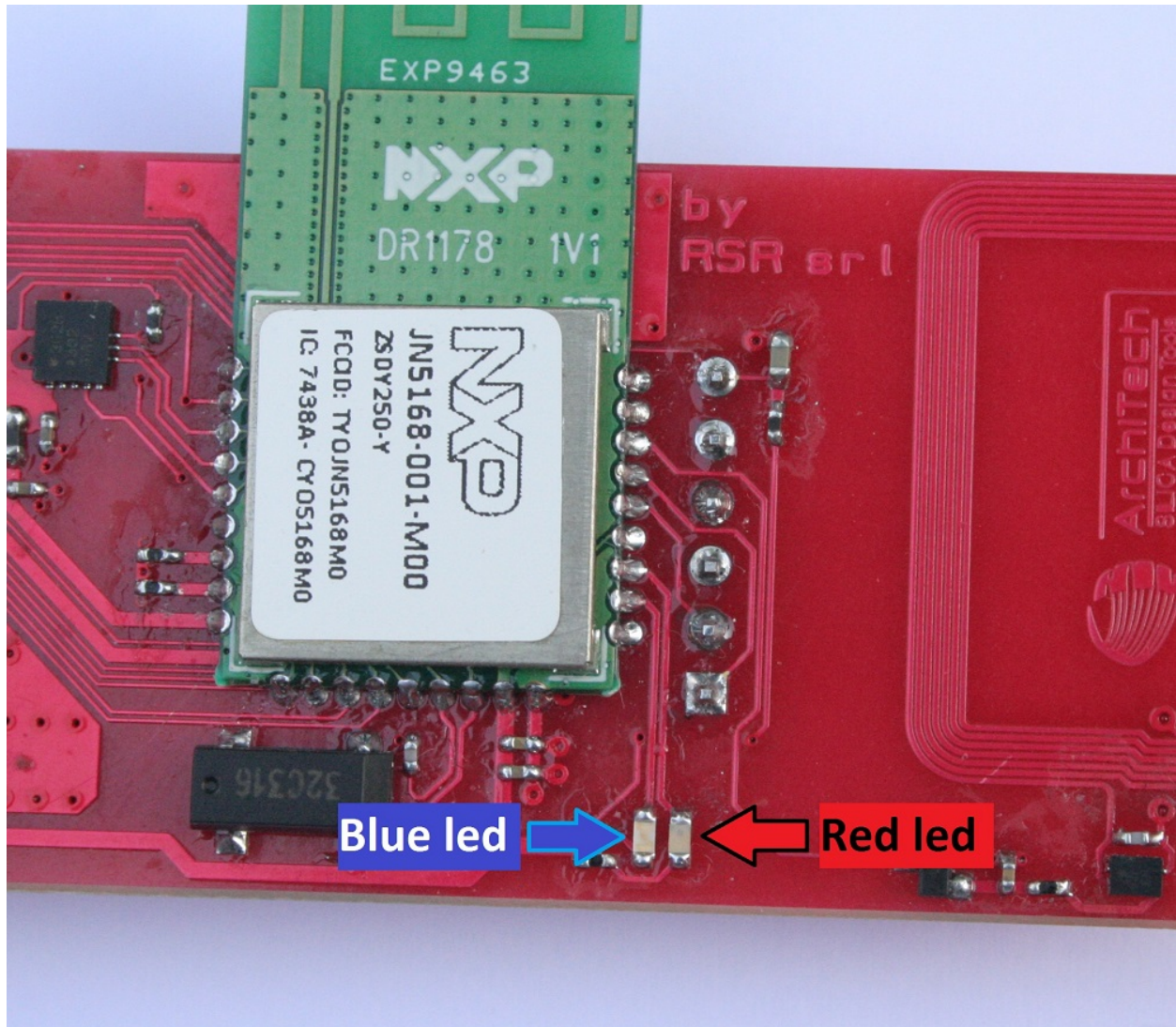
### Coordinator

- Blue led, slow lamp → network scanning active, no devices associated

- Blue led, fast lamp -> network ready, almost one device associated
- Red led -> flashing to indicate radio Rx/Tx

#### End-Node

- Blue led, slow lamp -> ready to associate, no network or coordinator found
- Blue led, fast lamp -> network ready, device associated (coordinator found)
- Blue led, on -> sleep fail, no coordinator found
- Red led -> flashing to indicate radio Rx/Tx



#### Firmware debugging

JN5168 modules does't support Jtag emulation. If you need to develop new application, you can use "vPrintf" function.

We also suggest you to see documentation [JN516x Wireless Microcontrollers](#) for any specific further detail.

- search

## C

CwInst, 4

## I

index, 1

## P

PrjFiles, 50