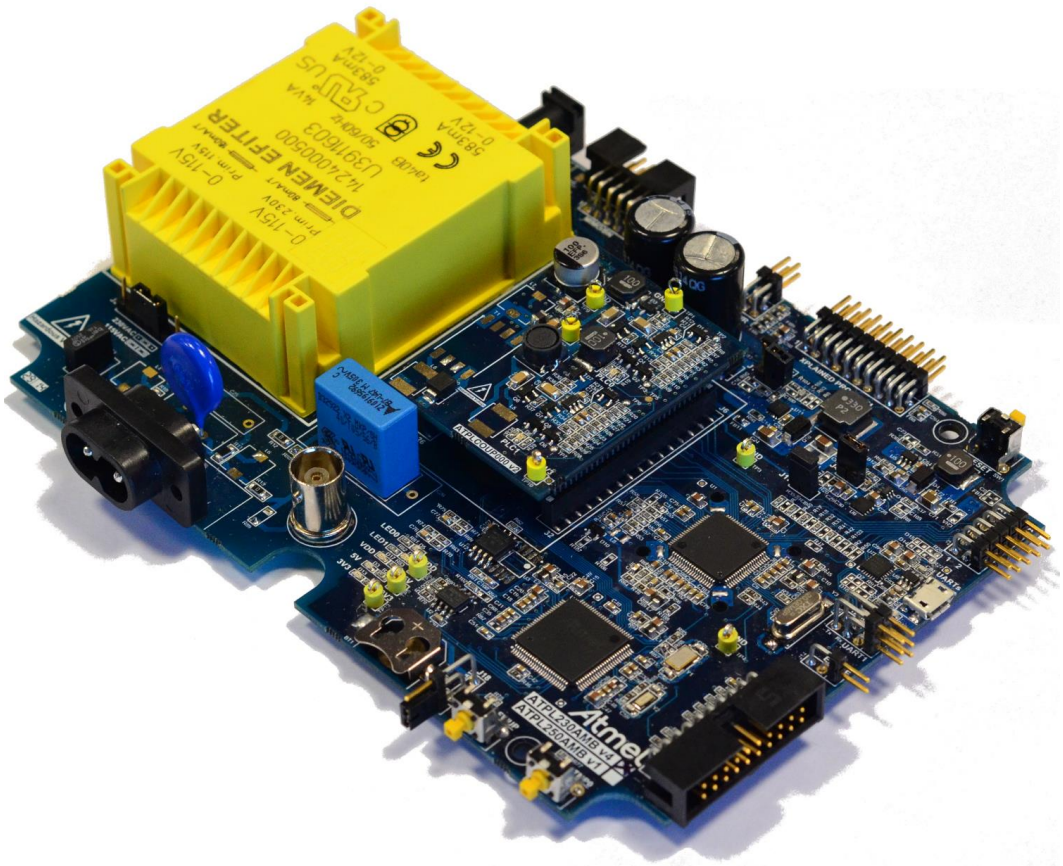
ATPL230A

ATPL230A-EK Kit User Manual

USER GUIDE



Introduction

ATPL230A-EK is an evaluation kit for the ATPL230A modem for Power Line Communication from Atmel® Corporation. ATPL230A is a power line communications base band modem, compliant with the PHY layer of PRIME (Power Line Intelligent Metering Evolution) specification.

ATPL230AMB is PLC multi-purpose modem board based on the ATPL230A transceiver and on Atmel®|Smart™ SAM4C ARM® Cortex™ M4 microcontroller. This development board provides a full featured platform to develop a complete communications system over Power Line Communication technology.

This guide describes how to use the kit and get start with it.

Contents

- Welcome letter that presents you the evaluation kit and the contents.
- Boards:
  - Two ATPL230AMBv4 modem boards.
  - Two ATPLCOUP001v1 coupling boards.
  - Two ATPLCOUP006v1 coupling boards.
- Cables:
  - Two micro A/B-type USB cables.
  - Two power cord cables IEC320-C8.
- Jumpers:
  - Two voltage jumper with pitch 5.08mm.
  - Two erase jumper with pitch 2.54mm.

Features

- ATPL230A is a compact and high-efficient device for a wide range of Smart Grid applications such as Smart Metering (Smart Meters and Data Concentrators), Lighting, Industrial/Home Automation, Home and Building Energy Management Systems, Solar Energy and Plug-in Hybrid Electric Vehicle (PHEV) Charging Stations.
- ATPL230A PRIME device includes enhanced features such as additional robust modes and frequency band extension. ATPL230A is able to operate in independently selectable transmission bands up to 472 kHz, achieving baud rates ranging from 5.4 kbps up to 128.6 kbps.
- ATPL230A has been conceived to be bundled with an external Atmel MCU. ATPL230AMB modem board mounts the ATPL230A transceiver and on SAM4C ARM Cortex M4 microcontroller. This development board provides a full featured platform to develop a complete communications system over Power Line Communication technology.
- Evaluation platform performance for the Atmel ATPL230A to develop a complete communications system based on PLC technology:
  - Channel characterization.
  - Noise level measurement.
  - Sensitivity level measurement.
  - Maximum reachable distance.
  - Power consumption.
  - Possibility to verify the different standard frequency bands complying with the existing regulations (CENELEC, FCC, ARIB) setting the different PLC couplings boards.
  - A transformer lets you supplied the board with universal 115-230 V_{AC} 50-60 Hz power input.

  - Boards have a JTAG interface for MCU debugging and programming purposes and two debugging UARTs. And also it provides Battery Backup and slow crystal oscillator to support SAM4C embedded Real time Clock (RTC) and low power modes. Several wake up conditions are available, such as mains crossover detection and voltage rails recovery condition.
- Software application examples available based on PRIME Stack:
  - Atmel provides an Atmel PRIME PHY layer library which is used by the external MCU to take control of ATPL230A PHY layer device. Three example projects about the PRIME PHY layer are provided with the kit.
  - And also the Atmel PRIME Stack for Service Node and some user applications.

Table of Contents

1. Evaluation Kit Specifications

1.1 Safety recommendations

These development boards must be only used by expert technicians. ATPL230AMB is directly powered from mains grid, so hazardous voltage (100/230V_{AC}) is present on the board. To avoid user access to dangerous parts, ATPL230AMB must always be used within its enclosure. All required connectors and configuration jumpers are easily accessible without electrical shock risk.

A normal use of ATPL230AMB does not require removing the enclosure cover. If this action is necessary, it must be performed by qualified staff after being sure that mains connection has been previously removed. Be careful it is only for indoor use.

This development board does not have any switch on mains connection to switch on or off it. It must always be connected to an easy accessible mains socket.

Do not connect any probe to high voltage sections if the board is not isolated from the mains supply to avoid damaging of measurement instruments.

This board can be used with coin lithium batteries, which are highly contaminated products. Used batteries must always be recycled.

The boards' kits are shipped in a protective anti-static package. The board system must not be subjected to high electrostatic discharge.

We strongly recommend using a grounding strap or similar ESD protective device when handling the board in hostile ESD environments (offices with synthetic carpet, for example) without enclosure. Avoid touching the component pins or any other metallic element on the board.

ATPL230AMB is a CE mark product which passes EN60950-1 safety standard, EN50065-1, EN50065-2-3, EN600065-7 EMC and FCC (as current carrier system) standards. It also satisfies Pb-Free and ROHS directive.

ATMEL does not assume responsibility for the consequences arising from any improper use of this board.

Boards' kits are intended for further engineering, development, demonstration, or evaluation purposes only. It is not a finished product except as may be otherwise noted on the board/kit.

1.2 Electrical characteristics

This section shows the electrical characteristics of the kit's boards. See the following tables:

Table 1-1. **Power Supply Requirements.**

Parameter	Condition	Min.	Typ.	Max.	Unit
AC mains Voltage Range		100		230	V_{AC}
Mains Frequency			50/60		Hz
Maximum Input Current				200(1)	mA
Isolation Voltage	ACDC power supply and PLC coupling transformer			3000	V_{AC}

Note that the ATPL230AMB can be supplied either with $100V_{AC}$ or $230V_{AC}$ by setting the proper jumpers (pitch = 5.08mm) in the voltage selector, J2, as depicted in the Figure 6-20. By default, voltage jumper is set for $230V_{AC}$. For more information about power supply, see section 3.5.1.

Note: This current is measured when board is supplied with $100V_{AC}$ and board is in worst consumption conditions. That is when it emits against very low impedance in higher channel and it is supplying an extra board through the DC jack J15.

Table 1-2. **Power Consumption.**

Parameter	Condition	Min.	Typ.	Max.	Unit
Max Power Consumption	ACDC maximum output power			13	W
TX Power Consumption	FW PHY Tester Tool App Low Impedance Load (PRIME LISN). Measured on V_{DD} (16V) DCDC output.			8320(1)	mW
	FW PHY Tester Tool App High Impedance Load (CISPR LISN). Measured on V_{DD} (16V) DCDC output.			1840(1)	mW
RX Power Consumption	Measured on 3.3V LDO output		325(1)		mW
Low Power Mode Current Consumption			< 1(2)		uA

Notes: 1. These measurements were taken with a non-optimized FW (the PHY Tester Tool project included in the kit with a default configuration in Tx mode and in Rx mode) from a power consumption point of view and they highly depend on the architecture of the power supplies. These measurements correspond to the whole PCBA design and not only to ATPL230A device. All PCB peripherals are supplied, i.e. SAM4C16C and ATPLCOUP001 coupling board is emitting in channel 1. Refer to Atmel ATPL230A datasheet for an optimized power consumption measurement result.

2. Output current of a 3Volts CR1225 battery.

2. Evaluation Kit Overview

ATPL230A is a PRIME modem for Power Line Communication that implements PRIME CENELEC, FCC and ARIB profiles. It has been conceived to be bundled with an external Atmel MCU.

ATPL230A is oriented in a wide range of Smart Grid applications such as Smart Metering, Lighting, Industrial/Home Automation, Home and Building Energy Management Systems, Solar Energy and Plug-in Hybrid Electric Vehicle Charging Stations.

ATPL230AMB is PLC multi-purpose modem board based on the ATPL230A transceiver and on SAM4C ARM Cortex M4 microcontroller. This development board provides a full featured platform to develop a complete communications system over Power Line Communication technology.

This document describes how to starting to work with the Atmel ATPL230-EK by explaining the PC tools, software examples and hardware provided and giving you the necessary documents to create your PLC application by means of small and easy examples.

2.1 Design support

To make it faster and easier for you to evaluate, prototype, develop and program with Atmel® products, we offer a variety of design resources, including development tools, software, boards, kits and documentation.

For any technical support request, please refer to our Design Support webpage: http://www.atmel.com/design-support/.

There any user can search the Atmel knowledge base to find tips, help topics, and answers to common questions. In case that the obtained information is not helpful any user can *Open a Support Case* indicating a description of the case, product information, etc.

2.2 ATPL230A-EK contents

Additional information of this user guide as hardware documentation, software projects and PC tools to get started can be found in our Atmel website, http://www.atmel.com/tools/ATPL230A-EK.aspx. To download this information you need a *myAtmel* account, www.atmel.com/myAtmel. After that, please contact with plc@atmel.com to get the password access kit contents site. Once you have access to the *ATPL230A Evaluation Kit* Project you can find the available releases for the ATPL230A-EK. You can get these items, navigating through the different folders of the packing kit. **Please do not hesitate to visit our web site to get the last kit updates**.

Packing kit contents are:

1. A welcome letter, *ATPL230A-EK_WL*, which presents you the evaluation kit and the contents.
2. ATPL230A-EK Kit User Manual, *doc43075*.
3. *Hardware* folder contents:
    a. ATPL230A datasheet, *doc43053*.
    b. Some application notes about hardware issues: different Atmel PLC coupling boards, crystal selection guidelines, layout recommendations, critical design guidelines, etc.
    c. Schemes, PCBs layout, Gerbers and BOM files of ATPL230AMB, ATPLCOUP001 and ATPLCOUP006 boards.
4. *Software* folder contents:
    a. PRIME_vaa.bb.cc.dd folder, contains several projects for IAR and Atmel Studio:
        - Three PHY example projects in an unique workspace, see *phy.atpl230amb.zip* file:
            – Apps_Phy_Tester_Tool. This application configures PRIME PHY layer and its serial interface to communicate with Atmel PLC PHY Tester Tool to send and

receive PLC messages from/to the PLC line and check the PLC transmission/reception processes between ATPL230AMB boards. Atmel PLC PHY Tester tool can be downloaded from the *PCTools* folder.

  - Apps_Phy_Tx_Test_Console. This application lets the user to configure a proper set up to perform both EMC emissions and immunity tests on ATPL230AMB board. These tests are based on the use of PRIME PHY layer with a terminal console firmware that eases the configuration of several transmission parameters such as modulation, frame data length and time interval between frames.
  - Apps_Phy_Sniffer_Tool. This application configures PRIME PHY layer to monitor the PLC data traffic on ATPL230AMB boards and sends via serial communication this traffic to the ATPL Multiprotocol Sniffer tool. This tool can be downloaded from the *PCTools* folder. Every coupling board is intended to be used in their corresponding channel(s) only. By default, sniffer project is compiled for ATPLCOUP001 board. This means that only PRIME channel 1 is supported.

- A PRIME Service Node project, which is composed of several projects for both IDE tools:
  - A PRIME FW stack project, *prime_service_bin.zip* file.
  - A PRIME user application project (DLMS application). It is an application example that shows how the PRIME API should be used. This application configures the ATPL230AMB board as a Service Node with DLMS capabilities and simulates the data exchange between the Base Node and the Service Node. The Service Node responds dummy DLMS messages after receiving data requests from the Base Node. For this example, a PRIME Concentrator is required. Depending on the operation mode (as a Real Operating System or as Microcontroller) there are two projects: *prime_service_dlmsemu_fi.atpl230amb.zip* file for OSS based on FreeRTOS and *prime_service_dlmsemu_ui.atpl230amb.zip* file running as microcontroller.
  - A PRIME user application project (modem application). This application configures the ATPL230AMB board as a Service Node. It is an application example that shows how to serialize the PRIME API when the user application is running in an external device. See *prime_service_modem.atpl230amb.zip* file.
- Atmel provides a USI Host drivers example *usi-host.zip*. These files allow the user to integrate them in his own application and start the PRIME operation via serial communication with the PLC module. It is composed by a set of functions identical to the ones described in the PRIME specification. See doc43085 for more information.
- Scripts folder. It contains the .bin files and scripts to download easily the Service node (PRIME stack and DLMS or modem application) commented previously, in the boards' kit. Also it contains a PRIME Base Lite Node script. Atmel provides a binary file of a PRIME Base Lite Node. This Atmel PRIME Base Lite Node version is limited to manage up to 10 Service Node connections. This Base Node Lite lets us to communicate with ATMEL PLC tools and uses it to evaluate the Atmel PRIME Service nodes.

b. Common software documentation folder. It contains some application notes as the description of the Atmel PRIME firmware stack, doc43085. It describes in detail all layers from the Atmel PRIME implementation as well as configuration options provided, target

platforms, default architecture, all vendor-specific PIBs and the provided solutions by Atmel. Also each layer is specified with its corresponding primitives and access points.

c. Evaluation License Agreement document.

5. *PCTools* folder contents:
   a. Atmel PLC PHY Tester tool, for checking the point to point PLC transmissions between ATPL230AMB boards.
   b. ATPL Multiprotocol Sniffer tool to monitor data traffic on PRIME networks and gather information of a PRIME network.
   c. Atmel PRIME Manager tool, that displays information about the devices connected to the network and manages the PRIME network. As well as, it lets you to the firmware upgrade of the service nodes and monitors the data traffic –sniffer function-. There are two versions, one for Base Lite node (1.x.x) and another one for Service node (2.x.x).
   d. SAM-ICE™ Drivers. Users may need to install this driver the first time the SAM-ICE is connected to the PC.
   e. USB Drivers (Silicon usb drivers). Users may need to install these drivers the first time the ATPL230AMB board is connected to the host PC by means of a serial USB connection.

We recommend installing the evaluation kit contents in the root C:\ to avoid problems with very long paths.

Unpack and inspect the kit carefully. Contact your local Atmel distributor, should you have any issues concerning the contents of the kit.

The two ATPL230AMB boards with the ATPLCOUP001 are encapsulated with enclosures and shipped in protective anti-static foam. The two coupling boards, ATPLCOUP006, are shipped in shielded bags.

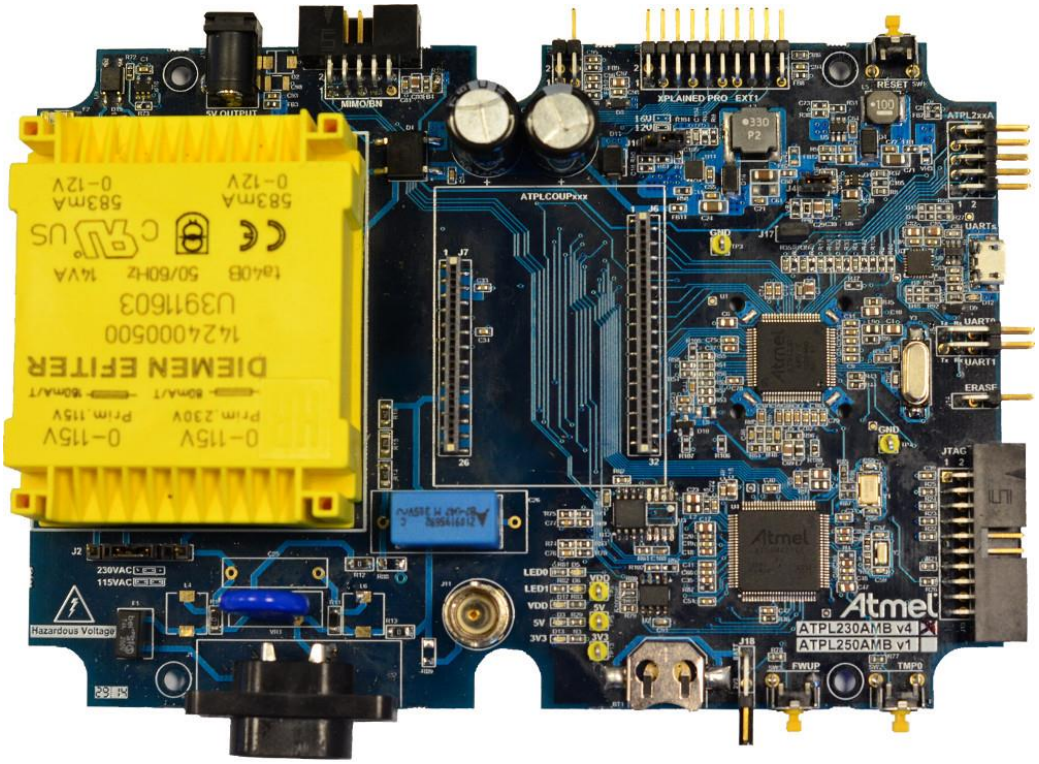
The boards must not be subject to high electrostatic discharge. We recommend using a grounding strap or similar ESD protective device when handling the board in hostile ESD environments. Avoid touching the components pins or any other metallic elements on the board.

Note that kit does not provide any battery. The coin battery is provided for user convenience in case the user would like to exercise the date and time backup function of the SAM4C device when the board is switched off.

Figure 2-1. Packed Atmel ATPL230A-EK.

Figure 2-2. Unpacked Atmel ATPL230A Evaluation Kit.

Figure 2-3. ATPL230AMB Modem board.



Both ATPL230AMB boards are provided with an example application preprogrammed, the PHY Tester embedded software for SAM4C16C. After installing the Atmel PLC PHY Tester Tool in your PC, users can interface with the device and start exploring its capabilities, for example, checking the point to point PLC transmissions between the two ATPL230AMB boards.

Take into account that the ATPL230A-EK provides two coupling boards for CENELEC-A band, Figure 2-4, set over the ATPL230AMB board. In addition to the ATPLCOUP001 boards, evaluation kit adds two coupling boards for FCC bands, PRIME channels 3, 4, 5, 6, 7 and 8, Figure 2-5. So the Atmel PLC PHY Tester Tool lets you send and receive PLC messages with both coupling boards according to the board selected in the PC tool. And depending on the board selected you will select the PHY parameters and the PLC channel.

So that, with ATPLCOUP001 board only lets you send and receive PLC messages in CENELEC-A band. And with ATPLCOUP006 board in FCC bands.

Please refer to chapter 6.2 for further information.

Figure 2-4. ATPLCOUP001 Coupling board.

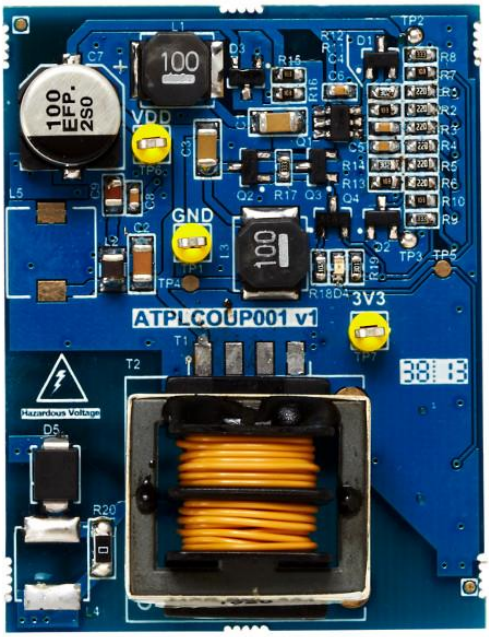
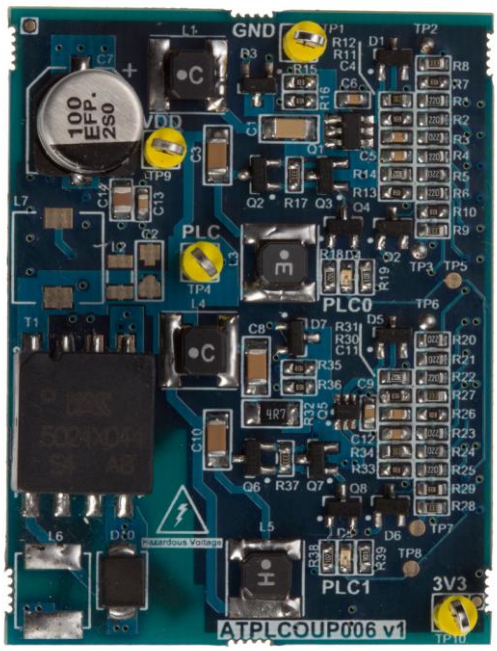


Figure 2-5. ATPLCOUP006 Coupling board.



3. ATPL230AMB Hardware

3.1 Overview

This section summarizes the Atmel ATPL230AMB board design. It introduces system-level concepts, such as power supply, MCU, PLC coupling, memories, peripherals and interface board.

ATPL230AMB is a PRIME multi-purpose development board based on the ATPL230A PRIME PLC transceiver and on the SAM4C16C ARM Cortex-M4 microcontroller. ATPL230AMB modem board provides a platform to develop a complete communications system over PRIME PLC technology.

Figure 3-1. ATPL230AMBv4 board.

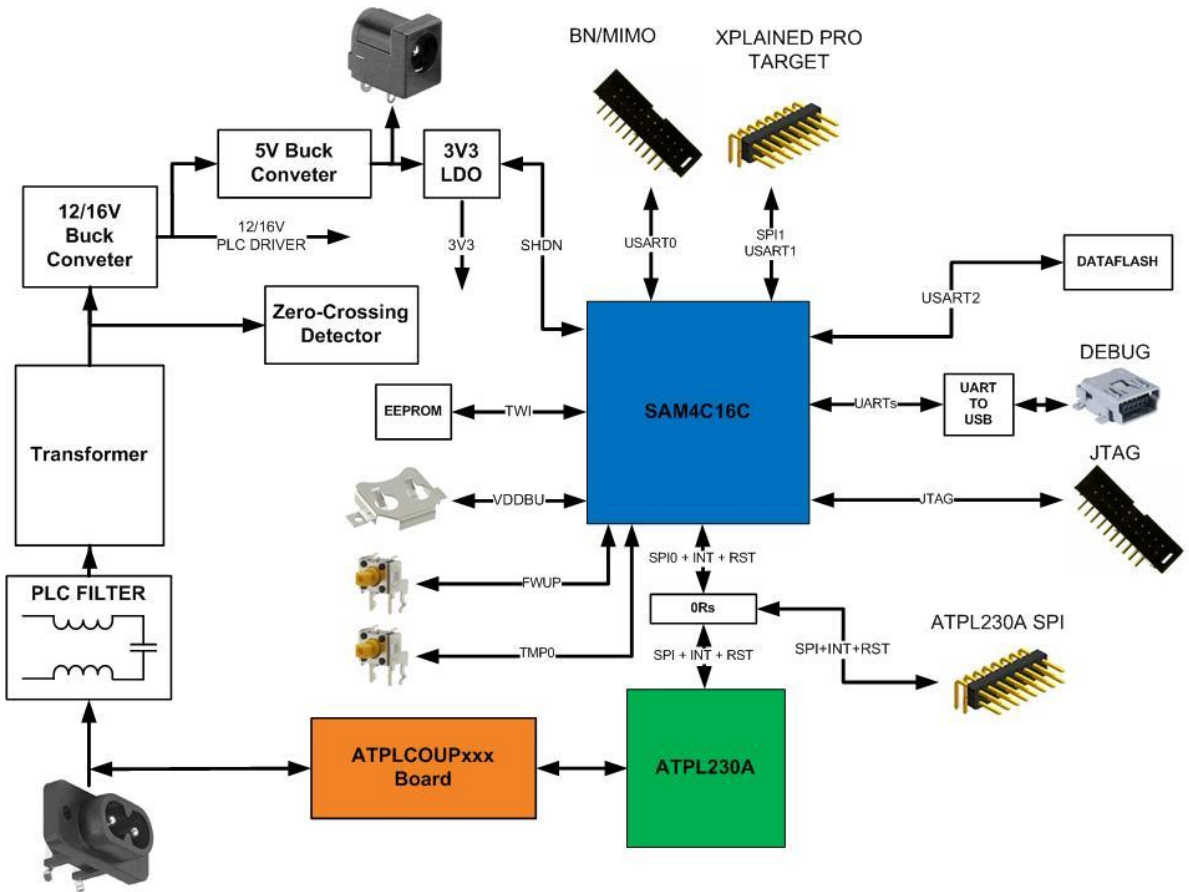
3.2 Features

The ATPL230AMBv4 board includes the following features:

- Power supply:
  - Non switched ACDC isolated power supply: 100-230V_{AC}, 50-60Hz.
  - 5 volts rail is accessible by means of a DC Jack connector (J15).
- ATPL230A PRIME Transceiver:
  - Power Line Carrier modem for 50 and 60 Hz mains.
  - 97 carriers OFDM PRIME compliant.
  - DBPSK, DQPSK, D8PSK modulation scheme available.
  - Additional enhanced modes available: Robust modes.
  - Configurable single transmission channel from 42 kHz to 472 kHz (CENELEC-A, FCC/ARIB).
  - Baudrate selectable from 5400 to 128600 bps.
  - Automatic Gain Control (AGC) and signal amplitude tracking.
  - Viterbi soft decoding and PRIME CRC calculation.
  - 128-bit AES encryption.

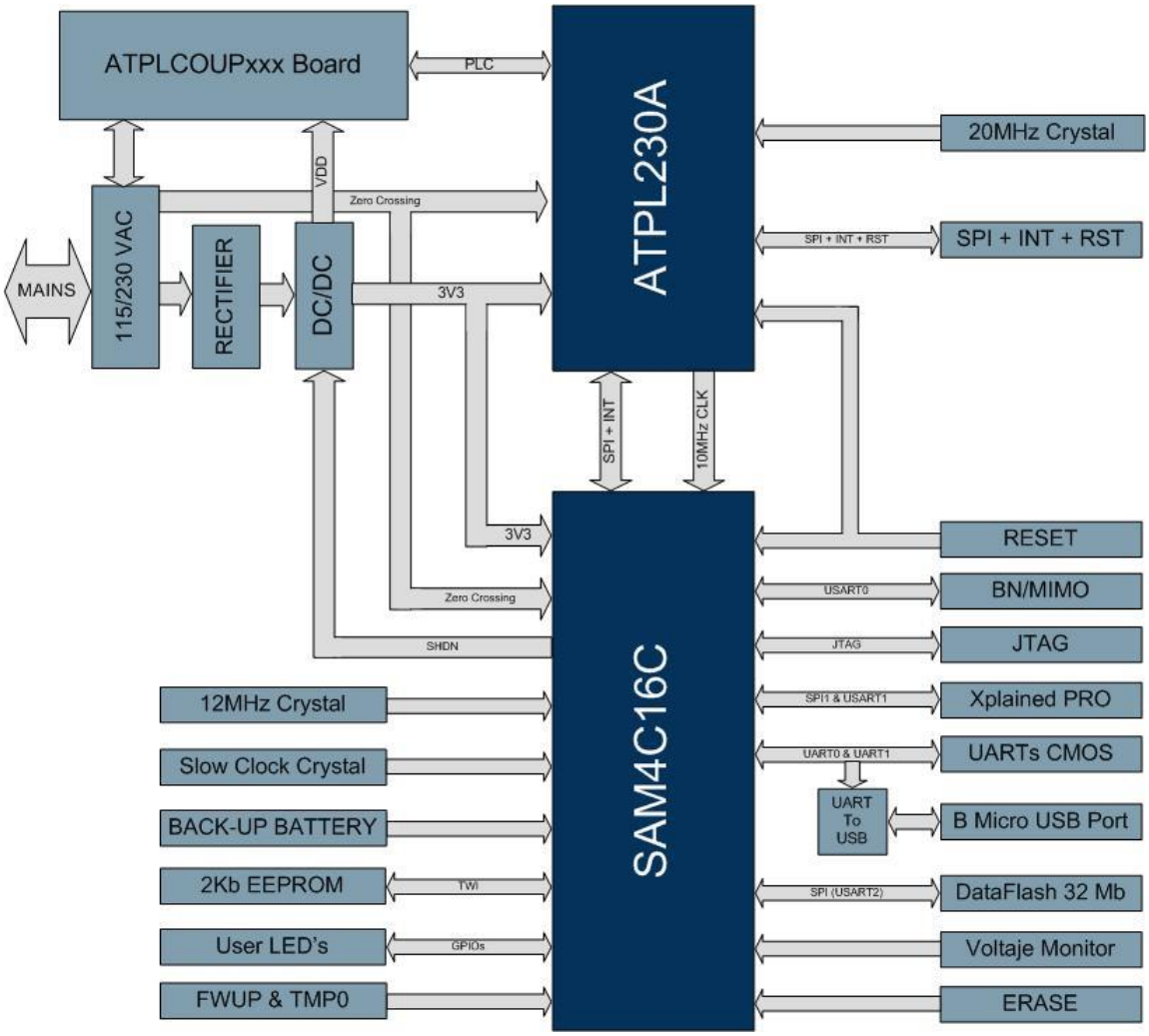
  - Channel sensing and collision pre-detection.
- Support to PRIME PLC coupling boards ATPLCOUP*XXX*
- Mains zero-crossing detector circuit
- SAM4C16C MCU ARM Cortex-M4
- External Memories:
  - Serial EEPROM (do not populate).
  - DataFlash memory.
- Peripherals:
  - Voltage monitor.
  - Back-up battery holder.
  - User's LEDs.
  - Force Wake Up switch button.
  - Tamper switch button.
  - Reset button.
- Interface:
  - JTAG debugging port.
  - Xplained PRO Master/Slave Interface.
  - UARTs over USB and CMOS levels.
  - ATPL230A SPI.
  - Control of 3V3 power supply.

Figure 3-2. ATPL230AMBv4 multi-purpose modem board.



3.3 Block diagram

Figure 3-3. ATPL230AMBv4 Block diagram.



3.4 Mechanical and user considerations

This development board is directly powered from mains grid, so hazardous voltage is present on the board. To avoid user access to dangerous parts, ATPL230AMB must always be used in its enclosure. All required connectors and configuration jumpers are easily accessible without removing the enclosure cover.

A normal use of the ATPL230AMB does not require removing the enclosure cover. If this action is necessary, it must be performed by qualified staff being sure that mains connection has been previously removed.

ATPL230AMB is a CE mark product which passes EN60950-1 safety standard and EN50065-1, EN50065-2-3, EN600065-7 EMC and FCC (as current carrier system) standards. It also satisfies Pb-Free and ROHS directive.

ATPL230AMB supply voltage is taken from mains grid (100/230V_{AC}, 50-60Hz), J1 connector.

ATPL230AMB dimensions are 165mm x 114mm x 30mm (LxWxH) and the enclosure dimensions are 174.4mm x 123.9mm x 38.5mm (LxWxH).

The operating temperature range is about -10 to 85ºC.

3.5 Hardware description

In this section the modules of the ATPL230AMBv4 board are described. Take into account that the board's BOM; is not a final design, so they include devices that could be no necessary in the customer designs once the design has been optimized.

Hardware files are contained in the Hardware folder: ".\Hardware\HW_SCH&PCB\ATPL230AMBv4".

3.5.1 Power supply

ATPL230AMB board can be powered either with $100V_{AC}$ or $230V_{AC}$ by setting the proper jumpers in the voltage selector (J2, Figure A-2). J1 IEC-320-C8 connector allows cable connection to mains grid. This design uses an encapsulated transformer (T1, Figure A-2) plus a full bridge rectifier (D1, Figure A-2) to obtain a DC voltage without increasing noise in PLC frequency bands (42 to 472 kHz), as may occur with switched ACDC power supplies. F1 and VR1 are used as protective devices in the equipment input and F2 protects the transformer output against over current situations.

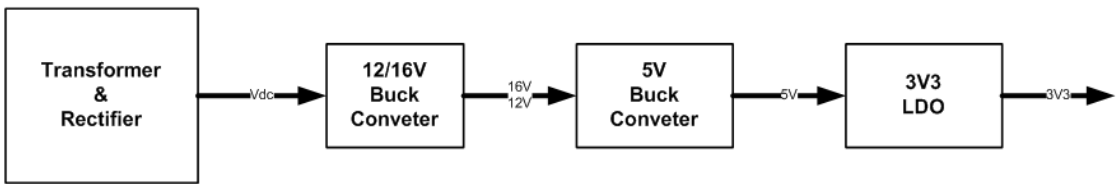
By default, the voltage jumpers' configuration is for $230V_{AC}$. See Figure 6-20.

The maximum transformer output power of 14VA is oversized compared to the maximum current consumption of ATPL230AMB when it is used as a PLC service node. However, this design is intended to power up other development kits which may have considerable power consumption if they include components such TFT displays.

The unregulated DC voltage is used as input of the DCDC buck converter -high frequency step-down switching regulator (U11, Figure A-2) which generates the configurable V_{DD} voltage. V_{DD} is mainly used as power supply of the PLC class D amplifier and also as input of the 5V DCDC buck converter -high frequency step-down switching regulator (U12, Figure A-2).

5V voltage rail is only used to provide an external power supply by means of DC jack connector (J15, Figure A-8). 3V3 is linearly regulated (U13, Figure A-2) and is used to power up ATPL230A and all other digital devices. To measure the current consumption of the 3volts power supply, connect an ammeter instead of the jumper J17.

Figure 3-4. Power supply diagram.



Switching frequency of DCDC buck converters used in this evaluation kit has been chosen to be higher than maximum PLC frequency band supported by ATPL230A device.

IMPORTANT

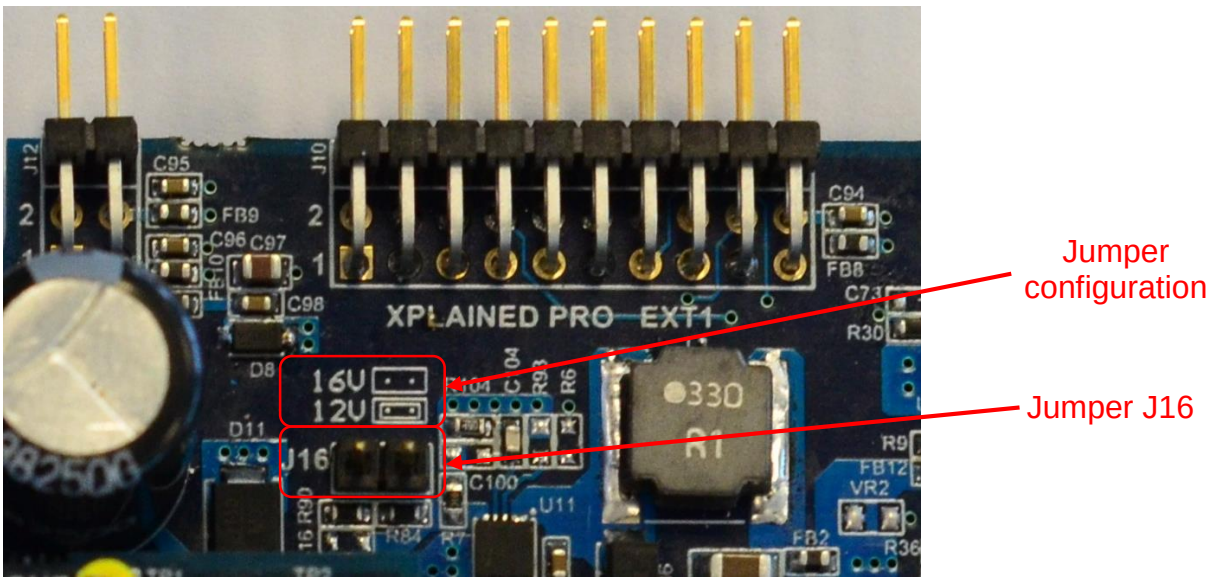
We recommend characterizing the potential impact of the selected SMPS for customer designs on the PRIME transmission channel.

V_{DD} could be two different voltages, 16 volts or 12 volts, depending on the jumper position. If the jumper is not placed, the voltage V_{DD} is 16 volts. If the jumper is placed in J16, V_{DD} is 12 volts. By default, the board has not a jumper, so board provides 16 volts. These different voltages are used to supply the PLC coupling driver board.

DANGER

Be careful with this issue, because the PLC coupling driver board ATPLCOUP*XXX* could be damaged. See the features of these boards to know the working voltage.

Figure 3-5. V_{DD} selection in ATPL230AMB board.



ATPL230AMB can also be powered from USB connector (J9, Figure A-8) or Xplained PRO interface (J12) without requiring connection to mains. Note that in these cases V_{DD} is not available so the PLC amplifier cannot be used.

The following test points and LEDs allow checking that these power supplies are running properly (see Figure A-2):

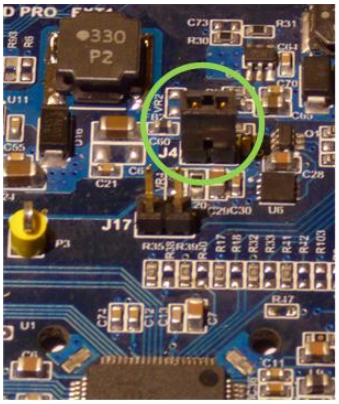
- V_{DD}: TP6 and green LED D17.
- 5V: TP5 and green LED D3.
- 3V3: TP13 and green LED D13.
- GND: TP3 & TP4.

3.5.1.1 Control of 3.3 volts power supply

The ATPL230AMB provides activate or deactivate the 3.3 volts regulator by SHUTDOWN pin, SHDN. User can be deactivate the 3.3 volts regulator before enter in a low mode power consumption of SAM4C16C, that can be powered by the battery. This allows decrease the consumption of the board.

J4 lets us enable the 3.3 volts regulator always, independently of SHDN pin. By default, this option is deactivated, 3v3 is always on, independently of the value of SHDN, to activate this option remove the jumper in J4 (Figure A-6).

Figure 3-6. J4 enabling 3.3 volts jumper.



By default, jumper J4 is set.

3.5.1.2 Zero crossing detector

A simple isolated circuitry (U10, Figure A-2) is used to detect mains zero crossing events. This VNR signal is used directly in the ATPL230A (VZ Cross) as well as in SAM4C16C microcontroller through an input (PB11) port as another wake up condition.

Note: By default, the zero crossing detector circuit is not populated.

3.5.2 ATPL230A PLC Transceiver

3.5.2.1 ATPL230A Overview

The ATPL230AMB includes an ATPL230A (U1, Figure A-3) PLC transceiver that is a PRIME compliant ASIC specifically designed for PLC Base and Service nodes PHY layer implementation.

ATPL230A has been conceived to be bundled with an ATMEL MCU running the Physical Layer API and being controlled by means of a serial synchronous communication interface (SPI).

Please refer to ATPL230A datasheet on the Atmel website or in doc43053 for a detailed description.

3.5.2.2 ATPL230A Clocking

ATPL230A requires a 20MHz crystal oscillator (Y3, Figure A-3). And SAM4C16C requires a 12 MHz crystal oscillator (Y1, Figure A-6).

The 20MHz clock signal could be used as internal reference time of the PLC modem, ATPL230A, and also to generate a 12MHz. So, it could be connected the output clock signal (CLKOUT) of ATPL230A like an input clock (CLKIN) of SAM4C16C when ATPL230A is configured in bypass mode. In this way, only one high frequency crystal oscillator is required. For this option that is mounted by default in the board, R85 is soldered but R67 and R68 are not populated, and remember that ATPL230A must be configured properly.

Clocking item is widely detailed in the datasheet, doc43053.

3.5.3 SAM4C16C Flash Microcontroller

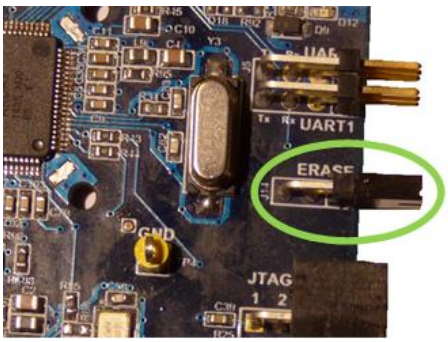
3.5.3.1 SAM4C16C Overview

The Atmel SAM4C16C microcontroller (U4, Figure A-6) is a system-on-chip solution for smart energy applications, built around two high-performance 32-bit ARM Cortex-M4 RISC processors. It operates at a maximum speed of 120 MHz and feature up to 1MB of embedded Flash, 152 Kbytes of SRAM and on-chip cache for each core.

The peripheral set includes advanced cryptographic engine, anti-tamper, floating point unit (FPU), five USARTs, two UARTs, two TWIs, up to seven SPIs, as well as a PWM timer, two 3-channel general-purpose 16-bit timers, temperature compensable low power RTC running on backup area down to 0.5 µA, and a 50 x 6 segmented LCD controller.

The ERASE pin can be used to reinitialize the Flash content, so setting a jumper in J14 connector (Figure A-6), the flash content is erased. This pin integrates a pull-down resistor of about 100kΩ, so that, it can be left unconnected for normal operations. When the ERASE pin is tied high during less than 100 ms, it is not taken into account. The pin must be tied high during more than 220 ms to perform a Flash erase operation. Please refer to SAM4C datasheet (doc11102) for a further description on Atmel's website.

Figure 3-7. J14 jumper, ERASE.



3.5.3.2 SAM4C16C Clocking

A 12 MHz Crystal oscillator is used as SAM4C16 clock input (Y1, Figure A-6). But board uses the clock output of ATPL230A.

A slow clock crystal oscillator of 32.768 kHz (Y2, Figure A-6) is used as SAM4C16 clock base in low power mode and for the embedded Real Time Clock (RTC).

3.5.4 PLC Coupling

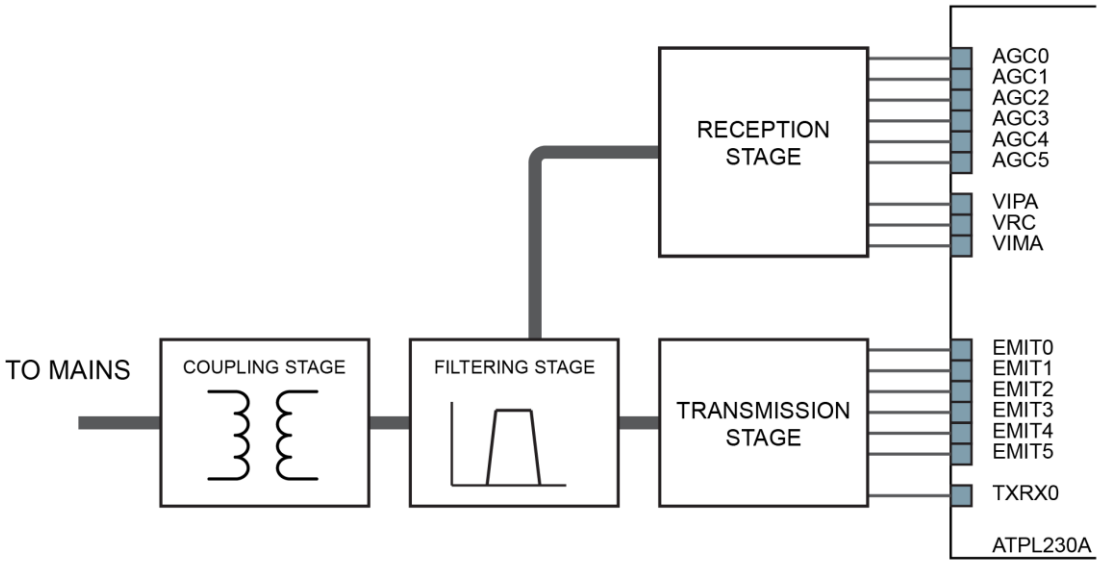
Atmel PLC technology is purely digital and does not require external DAC/ADC, thus simplifying the external required circuitry. Generally Atmel PLC coupling reference designs make use of few passive components plus a Class D amplification stage for transmission.

Figure A-4 and Figure A-5 show external components required by ATPL230A for PLC reception and transmission respectively.

PLC coupling reference design is composed by the same sub-circuits:

- Coupling Stage.
- Reception Stage.
- Transmission Stage.
- Filtering Stage

Figure 3-8. PLC Coupling example.



3.5.4.1 Coupling stage

The coupling stage blocks the DC component of the line to/from which the signal is injected / received (i.e.: 50/60 Hz of the mains). This is carried out by a high voltage capacitor (C26, Figure A-4). Coupling stage could also voltage isolate the coupling circuitry from the external world by means of a 1:1 PLC transformer. Capacitor is laying out in ATPL230AMB. The optional PLC transformer is included in ATPLCOUP001 board (voltage isolated), see section 4.

Footprint of BNC connector (J11, Figure A-4) is included in the board, but is not mounted by default. Removing the R12 and R13 and soldering R88 and R89 resistors, the PLC coupling signal can be isolated from the mains grid and that connector allows performing measurements of transmitted and received PLC signal without side effects (noise) coming from the grid.

3.5.4.2 Reception stage

The reception stage adapts the received analog signal to be properly captured by the internal reception chain. Reception circuit is independent of the PLC channel which is being used. It basically consists of:

- Anti-aliasing filter (RC Filter), R49 & C43, Figure A-5.
- Automatic Gain Control (AGC) circuit. The AGC circuit avoids distortion on the received signal that may arise when the input signal is high enough to polarize the protective diodes in direct region (D10, Figure A-5).
- Driver of the internal ADC. The driver to the internal ADC comprises a couple of resistors and a couple of capacitors. This driver provides a DC component and adapts the received signal to be properly converted by the internal reception chain.

3.5.4.3 Transmission stage

The transmission stage adapts the EMIT signals and amplifies them if required (Figure A-4). It can be composed by:

- Driver: A group of resistors which adapt the EMIT signals to either control the Class-D amplifier or to be filtered by the next stage.
- Amplifier: If required, a Class-D amplifier which generates a square waveform from 0 to V_{DD} is included.
- Bias and protection: A couple of resistors and a couple of Schottky barrier diodes provide a DC component and provide protection from received disturbances.

Transmission stage shall be always followed by a filtering stage.

3.5.4.4 Filtering stage

The filtering stage is composed by band-pass filters which have been designed to achieve high performance in field deployments complying at the same time with the proper normative and standards.

The in-band flat response filtering stage does not distort the injected signal, reduces spurious emission to the limits set by the corresponding regulation and blocks potential interferences from other transmission channels.

The filtering stage has three aims:

- Band-pass filtering of high frequency components of the square waveform generated by the transmission stage.
- Adapt Input/Output impedances for optimal reception/transmissions. This is controlled by TXRX signals.
- And, in some cases, Band-pass filtering for received signals.

When the system is intended to be connected to a physical channel with high voltage or which is not electrically referenced to the same point then the filtering stage must be always followed by a coupling stage.

These components are not lying out on ATPL230AMB board because are dependent on the application parameters such frequency band transmission. A set of boards known as ATPLCOUP*XXX* have been design by Atmel to implement any possible transmission scheme supported by ATPL230A. ATPL230A-EK includes ATPLCOUP001 and ATPLCOUP006 boards which are described in chapters 4 and 5 respectively. Other coupling boards have been designed. The Application Note doc43052 provides a description of the PLC Coupling Reference Designs available and all the features and characteristics.

3.5.4.5 ATPLCOUP boards

Table 3-1 summarizes the main characteristics of currently available PLC coupling reference designs. Please refer to Atmel doc43052 for a complete description of ATPLCOUP boards.

Table 3-1. **ATPLCOUP boards.**

Board Name	Frequency Band	Branch	Electrical Isolation	PRIME Channel	CENELEC Band	ARIB	FCC
ATPLCOUP000	41-89 kHz	Double	Yes	1	A	-	-
ATPLCOUP001	41-89 kHz	Single	Yes	1	A	-	-
ATPLCOUP002	206–417 kHz	Double	Yes	4, 5, 6, 7	-	X	-
ATPLCOUP003	41-89 kHz	Double	No	1	A	-	-
ATPLCOUP004	41-89 kHz	Single	No	1	A	-	-
ATPLCOUP006	151-472 kHz	Double	Yes	3, 4, 5, 6, 7, 8	-	-	X

Although different ATPLCOUPXXX can be used on the same ATPL230AMB board, they may require different voltage for the class D amplifier (V_{DD}). As is commented in 3.5.1, V_{DD} can be regulated to 16 or 12 volts depending on J16 jumper position. It is important to note that ATPLCOUP001 must be used with 16V.

The PRIME PHY 1.3 specification uses the frequency band from 41.992 kHz to 88.867 kHz (47 kHz bandwidth). This is achieved by using OFDM modulation with signal loaded on 97 (96 data and one

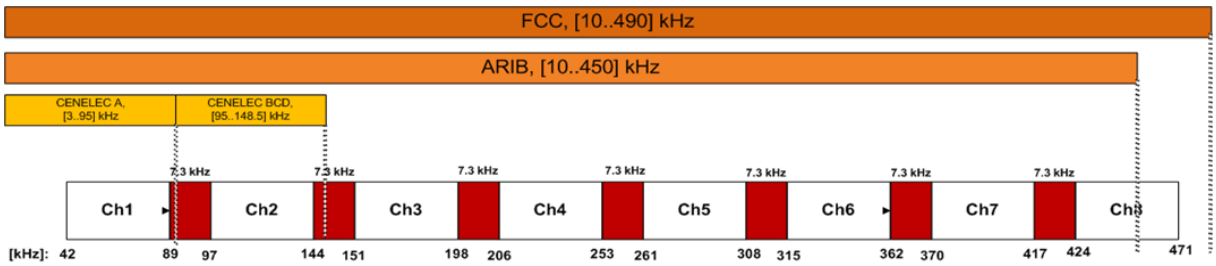
pilot) equally spaced subcarriers. Differential modulation is used, with one of three possible constellations: DBPSK, DQPSK or D8PSK.

The ATPL230A architecture provides enhanced performances over the PRIME specification with the new robust modes and the ARIB/FCC frequency band extension. PRIME has two additional robust modes:

- Robust DQPSK.
- Robust DBPSK.

The current PRIME standard is adapted to European regulations. The evolution of PRIME has, as one of its key features, a frequency band extension that allows choosing up to 8 different channels (PRIME 1.4). This performance makes PRIME becoming into a more flexible platform.

Figure 3-9. FCC & ARIB bands.



This technology only allows one channel active at a time. The limits of each channel are shown in the next table and can be compared with the figure above.

Table 3-2. **Frequency Band limits for each channel.**

Channel	Start freq. (kHz)	End freq. (kHz)	CENELEC	ARIB	FCC
1	41,992	88,867	X	X	X
2	96,68	143,555	X	X	X
3	151,367	198,242	-	X	X
4	206,055	252,93	-	X	X
5	260,742	307,617	-	X	X
6	315,43	362,305	-	X	X
7	370,117	416,992	-	X	X
8	424,805	471,68	-	-	X

3.5.5 Peripherals

These peripherals are not necessary to implement a PRIME device, they are included to show some features of the ATPL230A for a customer designs.

3.5.5.1 External Memories

The ATPL230AMB Modem Board includes a Flash Memory connected mean a SPI interface (U3/U12, Figure A-7) with the SAM4C16C.

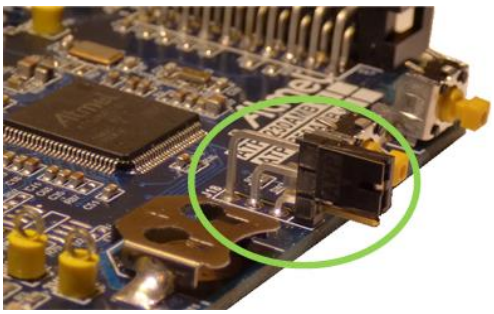
The ATPL230AMB Multi-purpose board includes the possibility to mount a serial EEPROM memory connected by Two Wires Interface (U2, Figure A-7) with the SAM4C16C. Please refer to AT24Cxx datasheet for a further description on Atmel's website.

3.5.5.2 SAM4C MCU Real Time Clock and back-up battery

SAM4C16 MCU embedded Real Time Clock (RTC) can be used as calendar and time base counter. A back-up battery (Figure A-6), slow clock crystal and low power modes are required to keep the RTC running during power down or mains unplugged conditions.

The ATPL230AMB includes a Battery (BT1, Figure A-6) for maintain active the RTC when the power supply of 3v3 shutdown and SAM4C16C enter in a low-power mode. J18 jumper lets us supply the board with the battery setting the jumper between VDDBU and BATT position.

Figure 3-10. J18 jumper in battery position.



By default, jumper J18 sets VDDBU to 3V3 supply.

3.5.5.3 Voltage Monitor

The ATPL230AMB monitors V_{DD} and 5V voltage rails to detect backup mode entering conditions and also wake up events by means of its dedicated hardware.

5V falling condition is the most recommended trigger event to enter backup mode on ATPL230AMB design:

- Configure PB23 as positive input of analog comparator and compare it with AREF.
- Once 5V rail falls below 4.5V (depending on R70, R74 values) (i.e., mains grid connection has been removed) the analog comparator interrupt is triggered.
- Before going to backup mode, configure PB23 as wake up port to return to active mode once power supply is available again.

The wake-up events allow the microcontroller to exit the backup mode. When a wake-up event is detected, the Supply Controller performs a sequence which automatically enables the core and the SRAM power supply and clocks. See Figure A-7 for details.

Lack of activity on VZ CROSS signal (PB11) can also be used to enter in backup mode.

3.5.5.4 Tamper and Wake-Up

The purpose of backup mode is to achieve the lowest possible power consumption in a system that executes periodic wake-ups to perform tasks but which does not require fast start-up time.

Wake-up events allow the device to exit backup mode. Force Wake-Up pin, FWUP, can be used as a wake-up source. In ATPL230AMB board, FWUP has been connected to switch button SW3.

Anti-tamper pins (TMP0-TMP3) detect intrusion, for example, into a smart meter case. Upon detection through a tamper switch, automatic, asynchronous and immediate clear of registers in the backup area, and time stamping in the RTC will be performed. Anti-tamper pins can be used in all modes. Date and number of tampering events are stored automatically. Tampering input 0, TMP0, is connected to switch button SW2. Wake up pins multiplexed with anti-tampering functions are possible sources of wake up as well in case an anti-tampering event is detected.

3.5.5.5 User leds

The board incorporates two user LEDs (LED0 & LED1), green and red (D5 & D6, Figure A-7), connected to PB14 and PB15 respectively of the SAM4C16C.

3.5.6 Interface Ports

3.5.6.1 Reset circuitry

ATPL230AMB can be manually reset by using a push button (SW1, Figure A-8) or by means of an external reset signal available on the Base Node/MIMO interface connector. This reset restarts the SAM4C16C and the ATPL230A include his PLL.

Also ATPL230A shall be reset from SAM4C16C with an asynchronous reset by PC6 and with a synchronous reset by PC7, see Figure A-8.

3.5.6.2 ATPL230A SPI

ATPL230AMB provides the option to connect the SPI, the Reset and the interruption signal of ATPL230A device with an external microcontroller. This option is available in a 5-pin dual row male header (J3, Figure A-8) allowing others Atmel development boards make use of the ATPL230A PLC transceiver. For enable this option is necessary do not placed R17, R32, R35, R39 and R41 and solder R2, R18, R33, R38, R40, R42 and R50.

It is recommended to avoid unintentional reset, do not placed R43, R44 and R45.

3.5.6.3 SAM4C JTAG Debug Port

The SAM4C16C JTAG interface is available in a standard 20-pin male header J13 (see Figure A-8) for debugging and programming purposes. The JTAG/ICE connector is implemented on the ATPL230AMB board for the connection of a compatible ARM JTAG emulator interface, such as the SAM-ICE from Segger.

Notes:
1. The NRST signal is connected to SW1 system button and also to an external reset signal available on the Base Node/MIMO interface connector.
2. The 0 ohm resistor R26 may be removed in order to isolate the JTAG port from this system reset signal.
3. The TDO pin is in input mode with the pull-up resistor disabled when the Cortex M4 is not in debug mode. To avoid current consumption on VDDIO and/or VDDCORE due to floating input, the internal pull-up resistor corresponding to this PIO line must be enabled.

Please refer to the SAM4C16C datasheet for a further description of JTAG debug port.

3.5.6.4 Debugging UARTs

ATPL230AMB uarts, UART0 and UART1, are user accessible by means of micro USB type B connector (J9, Figure A-8). A single chip bridge is used to convert UARTs TTL CMOS to USB levels (U8, Figure A-8). Note that this bridge is powered from USB 5V power supply, so it is only available when USB cable is attached to any other USB host port. That single chip drive, CP2105-F01-GM of Silicom Labs, has two ports. The enhanced port is connected to UART0 and the standard port is connected to UART1.

It is possible to power ATPL230AMB directly from USB connector. However, due to power limitations, this option does not allow PLC transmissions. Nevertheless, this option is very useful for several applications such as FW downloading or debugging.

Furthermore, UARTs CMOS signals are also available in a triple row male connector (J5, Figure A-8).

3.5.6.5 Xplained PRO Master

Xplained Pro is an Atmel's proprietary interface port intended to connect different development boards, such as metering and PLC communication boards. This point-to-point interface offers SPI and USART

communication capabilities and requires one target board (master) and an extension module (slave). ATPL230AMB is an Xplained Pro target device with power supply extension connector.

ATPL230AMB Xplained Pro provides the following features:

- SPI (from the SPI1).
- UART (from the USART1).
- I2C (from the TWI0)
- 2 ADC inputs (from PA4 and PB13).
- 1 IRQ input (from PA17).
- 5 GPIO's (from PA18, PB19, PB20, PB21 and PC8).

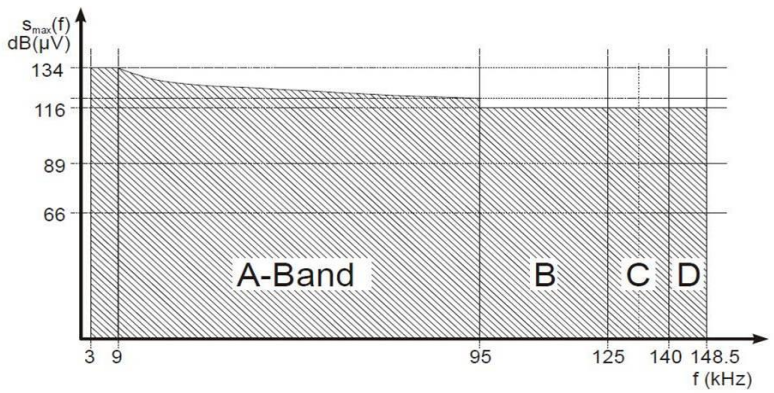
ATPL230AMB is an Xplained Pro Master device with power supply extension connector (J12, Figure A-8).

4. ATPLCOUP001 Hardware

4.1 Overview

ATPLCOUP001 is a PLC coupling board designed to communicate in CENELEC-A band, especially in PRIME band from 41 to 89 kHz (PRIME channel 1). ATPLCOUP001 mounts a single branch with voltage isolation from mains to the PLC coupling driver board. The goal of this design is provided to the customers with a cost optimized performance transmission board in CENELEC-A band for PRIME channel 1. This board is set by default in the ATPL230AMB board of the ATPL230A-EK.

Figure 4-1. CENELEC bands.

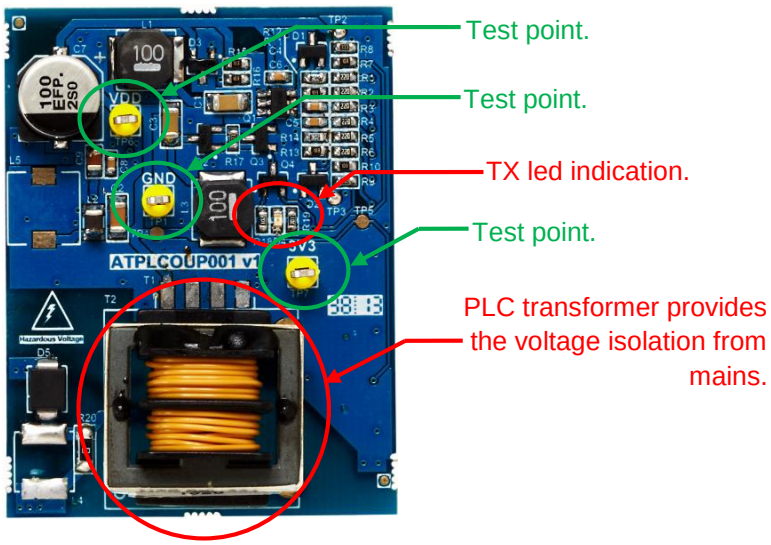


4.2 Features

The ATPLCOUP001v1 board includes the following features:

- Specially designed to communicate in CENELEC-A frequency band (41.992 – 88.867 kHz).
- Voltage Isolation from mains with a transformer, *MSR EXL-324*, soldered in top layer board.
- Single branch:
  - Low impedance optimized.

Figure 4-2. ATPLCOUP001v1 PLC Coupling board (top view).



4.3 Mechanical and user considerations

ATPLCOUP001 is delivered with the ATPL230A-EK. Board-to-board SMD connectors, J1 and J2, are used to connect the ATPLCOUP001 into connectors J6 and J7 of ATPL230AMB board (Figure A-4). These J1 and J2 connectors are in bottom layer of ATPLCOUP001 and they have the following part numbers:

- J1: SAMTEC FTR-130-54-L-S.
- J2: SAMTEC FTR-124-54-L-S.

The ATPLCOUP001 board is directly powered from mains grid, so hazardous voltage is present on the board. To avoid user access to dangerous parts, ATPLCOUP001 must always be used in its enclosure.

ATPLCOUP001 is a CE mark product that passes EN 50065-1, EN 50065-2-3 and EN60065-7 EMC standards. It also satisfies Pb-Free and ROHS directive.

ATPLCOUP001 dimensions are 51.5mm x 39.5mm x 18mm (LxWxH).

The operating temperature range is about -40 to 85ºC.

4.4 Hardware description

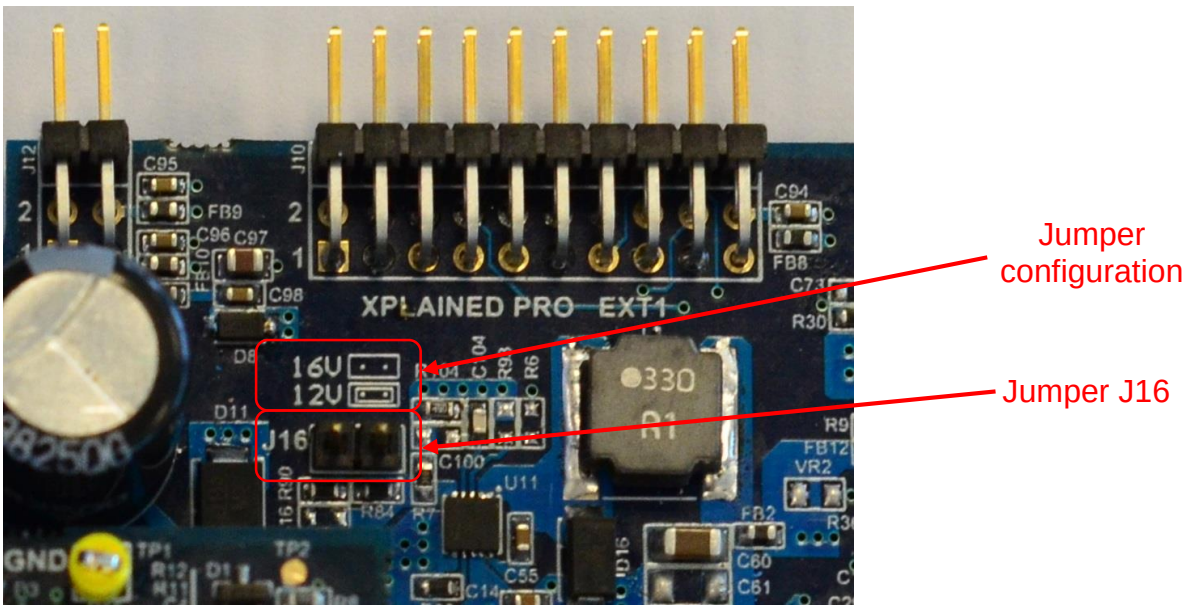
Hardware files are contained in the Hardware folder: *".\Hardware\HW_SCH&PCB\ATPLCOUP001v1"*.

ATPLCOUP001 is a galvanic isolated reference design which provides a cost optimized PLC coupling reference design between 41 kHz and 89 kHz within the CENELEC-A band. It is based on a single branch design which filtering stage has a flat band pass response with typical field impedances. ATPLCOUP001 has a good performance in terms of transmitted channel power with low impedances and complies with EN5065-1, EN5065-2-3 and EN5065-7 normative.

For more information, see PLC coupling reference designs document, doc43052.

Take into account that, when ATPLCOUP001 is connected to ATPL230AMB, V_{DD} voltage must be **16 volts** to avoid damaging the coupling board, so jumper in J16 must **not** be **set** (see section 3.5.1 and Figure A-2). By default, the jumper is not placed.

Figure 4-3. V_{DD} selection in ATPL230AMB board.

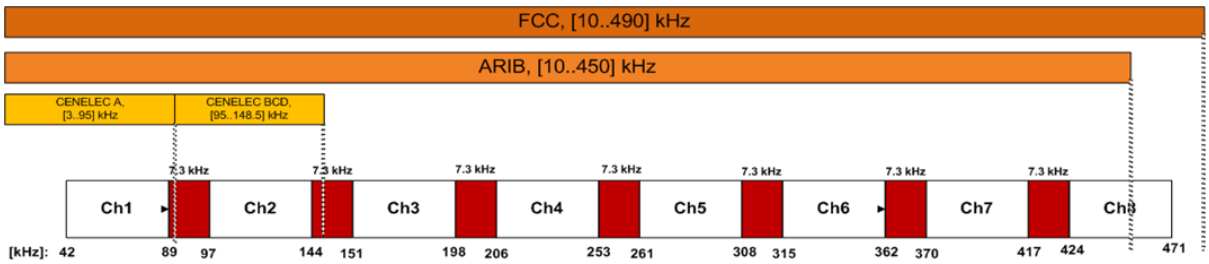


5. ATPLCOUP006 Hardware

5.1 Overview

ATPLCOUP006 is a PLC coupling board designed to communicate in ARIB and FCC bands, especially in PRIME band from 151 to 472 kHz (PRIME channels 3, 4, 5, 6, 7 and 8). ATPLCOUP006 mounts a double branch with voltage isolation from mains to the PLC coupling driver board. The goal of this design is provided to the customers with a full performance transmission board in FCC band. This board is not set by default in the ATPL230AMB board of the ATPL230A-EK.

Figure 5-1. FCC and ARIB bands.

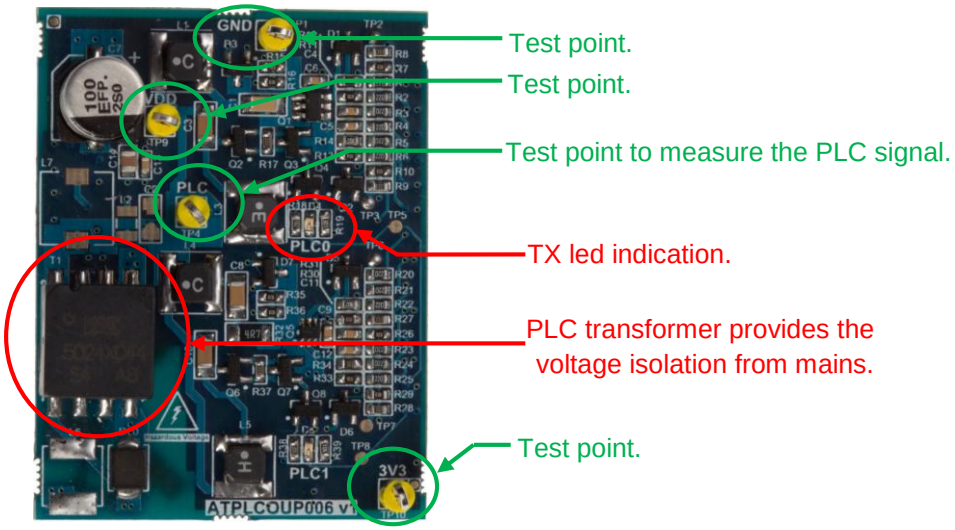


5.2 Features

The ATPLCOUP006v1 board includes the following features:

- Specially designed to communicate in ARIB and FCC frequency bands (151,367 – 471,68 kHz).
- Voltage Isolation from mains with a transformer, *VAC T60403K5024X044*, soldered in top layer board.
- Double branch, each one for a range of impedances:
  - Low impedance optimized.
  - High impedance optimized.

Figure 5-2. ATPLCOUP006v1 PLC coupling board.



5.3 Mechanical and user considerations

ATPLCOUP006 is delivered with the ATPL230A-EK. Board-to-board SMD connectors, J1 and J2, are used to connect the ATPLCOUP006 into connectors J6 and J7 of ATPL230AMB board (Figure A-4).

These J1 and J2 connectors are in bottom layer of ATPLCOUP006 and they have the following part numbers:

- J1: SAMTEC FTR-130-54-L-S.
- J2: SAMTEC FTR-124-54-L-S.

The ATPLCOUP006 board is directly powered from mains grid, so hazardous voltage is present on the board. To avoid user access to dangerous parts, ATPLCOUP006 must always be used in its enclosure.

ATPLCOUP006 is a CE mark product that passes EN 50065-1, EN 50065-2-3, EN60065-7 EMC and FCC (as current carrier system) standards. It also satisfies Pb-Free and ROHS directive.

ATPLCOUP006 dimensions are 51.5mm x 39.5mm x 18mm (LxWxH).

The operating temperature range is about -40 to 85ºC.

5.4 Hardware description

Hardware files are contained in the Hardware folder: *".\Hardware\HW_SCH&PCB\ATPLCOUP006v1"*.

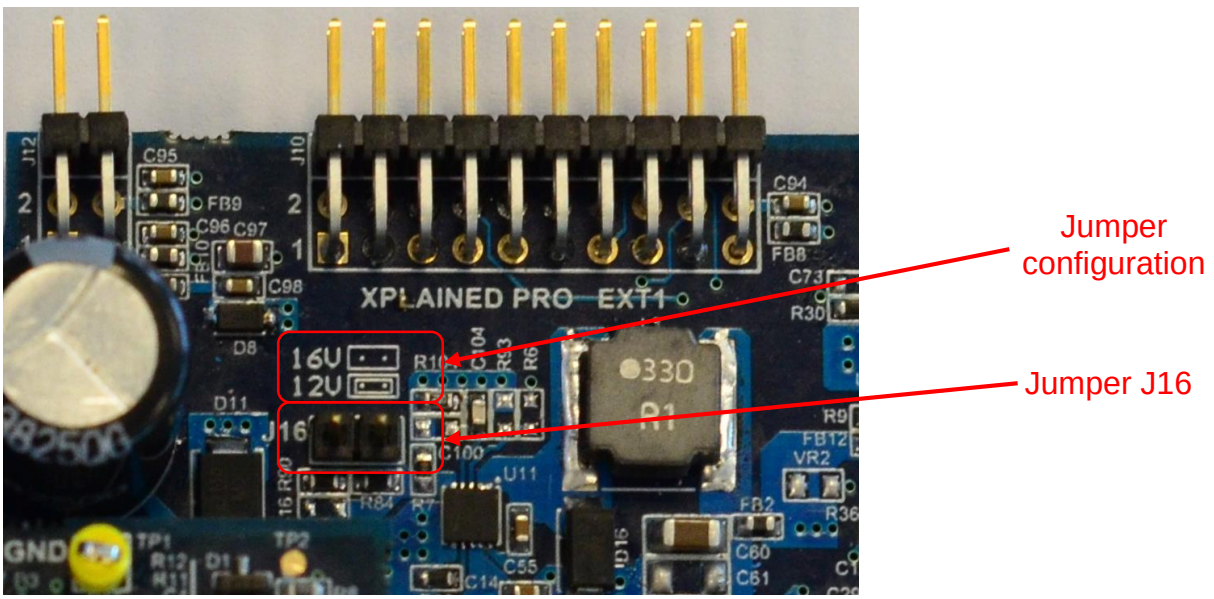
ATPLCOUP006v1 board is a PLC coupling driver board with double branch design and galvanic voltage isolation. ATPLCOUP006 has been designed to transmit in ARIB and FCC band, especially in PRIME band from 151 to 472 kHz (PRIME channels 3, 4, 5, 6, 7 and 8).

It has a good performance in terms of transmitted channel power over a range of load impedance values complying with FCC standard as current carrier system, see FCC normative. ATPLCOUP006 is composed of two transmission branches which only differ on the filtering stage. A 12V power supply voltage for the class-D amplifier is recommended to be used with ATPLCOUP006.

For more information, see PLC coupling reference designs document, doc43052.

Take into account that, when ATPLCOUP006 is connected to ATPL230AMB, V_{DD} voltage must be **12 volts** to avoid damaging the coupling board, so jumper in J16 must be **set** (see section 3.5.1 and Figure A-2). By default, the jumper is not placed.

Figure 5-3. V_{DD} selection in ATPL230AMB board.



6. ATPL230A Evaluation Kit: Getting started

The purpose of this chapter is to introduce you the Atmel ATPL230A device and its functionalities.

First of all, the software is presented to create, build, program and debug your application using the appropriate IDE tools (section 6.1).

Chapter 6.2 describes a simple PLC application that lets you check the device communication in a point-to-point connection (PHY layer example).

Chapter 6.3 describes the PHY TX Test Console application, which lets you configure a proper setup to perform both EMC emissions and immunity tests on ATPL230AMB board.

Chapter 6.4 describes the PRIME PHY Sniffer project, which, is able to monitor data traffic on the PRIME network.

Chapter 6.5 describes the PRIME Stack and we present you the structure of a PRIME project and how to create a final application.

Chapter 6.6 explains the setup and operations required to create a smart PLC network using the included PRIME Service Node example and Base Node Lite binary. This network communicates by means of PRIME (PoweR line Intelligent Metering Evolution).

Finally, chapter 6.7 introduces you the Atmel PRIME Manager tool. This tool lets you establish a serial communication with the boards by means of Atmel Universal Serial Interface.

Note: The software described in this manual is under the Atmel`s *Evaluation License Agreement.pdf* document. You can find it in the Software folder.

6.1 Introduction to the embedded system

The purpose of this section is to guide new users through the initial settings of IAR Embedded Workbench or Atmel Studio (AS), and compile a PRIME PHY project. The section shows setup of a PRIME project to generate a debug target that can be loaded into the microcontroller.

Kit projects are supported by IAR 7.10.1 version or AS 6.2 version or above. From this point on, it is assumed that a working copy of this IDE is installed in your computer. The IAR's homepage, http://www.iar.com, is a suitable source to download (i.e.: 30-day time-limited evaluation license). And the Atmel's homepage, http://www.atmel.com, is suitable for downloading the Atmel Studio 6 (free download).

6.1.1 IAR Embedded Workbench

IAR Embedded Workbench is a high-performance C/C++ compiler assembler, linker, librarian, text editor, project manager, and C-SPY® Debugger in an integrated development environment for applications based on 8-, 16-, and 32-bit microcontrollers. IAR Embedded Workbench is compatible with other ARM EABI compliant compilers and supports the SAM4C core family (example projects are developed only for 7.10.1 versions or above).

6.1.2 Atmel Studio 6

Atmel Studio 6 is the integrated development platform (IDP) for developing and debugging Atmel ARM Cortex-M and Atmel AVR® microcontroller (MCU) based applications. The Atmel Studio 6 IDP gives you a seamless and easy-to-use environment to write, build and debug your applications written in C/C++ or assembly code.

Atmel Studio 6 is free of charge and is integrated with the Atmel Software Framework (ASF) — a large library of free source code with 1,600 ARM and AVR project examples. ASF strengthens the IDP by providing, in the same environment, access to ready-to-use code that minimizes much of the low-level

design required for projects. Use the IDP for our wide variety of AVR and ARM Cortex-M processor-based MCUs, including our broadened portfolio of Atmel SAM3 ARM Cortex-M3 and M4 Flash devices.

Figure 6-1. Atmel Studio 6.

Download the latest version from the following link: http://www.atmel.com/microsite/atmel_studio6/

6.1.3 Atmel SAM-ICE JTAG Probe

Atmel SAM-ICE (a dedicated Atmel J-Link version) is a USB-powered JTAG emulator supporting Atmel ARM-based microcontrollers.

Atmel SAM-ICE is a JTAG emulator designed for Atmel SAMA5, SAM3, SAM4, SAM7 and SAM9 ARM core-based microcontrollers, including the Thumb® mode. It supports download speeds up to 720 Kbytes per second and maximum JTAG speeds up to 12 MHz. It also supports Serial Wire Debug (SWD) and Serial Wire Viewer (SWV) from SAM-ICE hardware V6.

SAM-ICE support is integrated in most professional integrated development environments (IDEs) such as IAR, Keil and many others.

More details are available here: http://www.atmel.com/tools/ATMELSAM-ICE.aspx.

Figure 6-2. Atmel SAM-ICE JTAG.

Note: Evaluation kit does not provide an Atmel SAM-ICE.

To use Segger tools with Atmel Studio 6.2, download Atmel's latest USB driver driver-atmel-bundle-7.0.712.exe from the following link: https://gallery.atmel.com/Products/Details/07bf16c1-444f-4ac8-8f40-9d4005575dca or take it from the PCTools folder: ".*PCTools\SAM-ICE_Drivers*". And install the file.

6.1.4 J-Link / SAM-ICE JTAG Probe Software & Documentation Pack

The J-Link / SAM-ICE JTAG software and documentation pack includes:

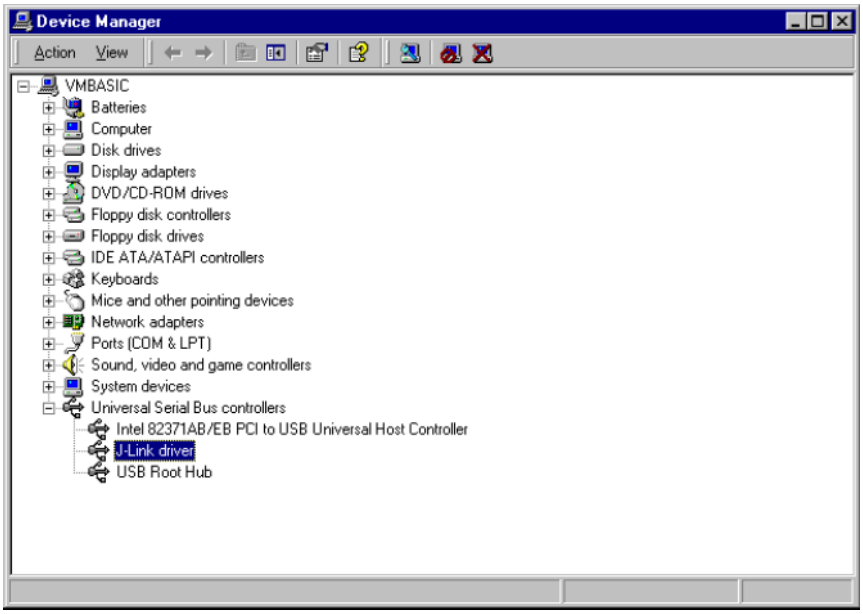
- GDB Server - Support for GDB and other debuggers using the same protocol. GUI & command line version.
- J-Link Configurator - Free utility to manage a various number of J-Links connected to the PC via USB or Ethernet.
- J-Link Commander - Simple command line utility, primarily for diagnostics and trouble shooting.
- J-Link Remote Server - Free utility which provides the possibility to use J-Link / J-Trace remotely via TCP/IP.
- SWO Viewer - Free tool which shows terminal output of the target performed via SWO pin.
- J-Mem - Memory viewer.
- J-Link DLL Updater - Allows updating 3rd party applications which use the J-Link DLL.
- Free flash programming utilities - Simple command line utilities which allow programming a bin file into the internal/external flash memory of popular evaluation boards.
- USB driver (Includes driver for J-Links with CDC functionality).
- Manuals: UM08001 (J-Link User Guide), UM08003 (J-Flash User Guide), UM08004 (RDI User Guide), UM08005 (GDB Server User Guide), UM08007 (Flasher ARM User Guide).
- Release notes for J-Link DLL, J-Flash and J-Link RDI DLL.
- J-Flash, including sample projects for most popular evaluation boards.
- J-Link RDI – Support for ARM RDI standard. Makes J-Link compatible with RDI compliant debuggers.

Installing the software will automatically install the J-Link USB drivers. It also allows the update of applications that use the J-Link DLL.

The last version of the driver for the SAM-ICE JTAG Probe can be downloaded from the http://www.segger.com website using the following link: http://www.segger.com/jlink-software.html. The package for Windows, *Setup_JLinkARM_V496b.zip*, is located in the following folder: *".\PCTools\SAM-ICE Driver"*.

Once drivers have been installed you may verify the driver installation by consulting the Windows® device manager. If the driver is installed and your SAM-ICE is connected to your computer, the device manager should list the J-Link driver as a node below "Universal Serial Bus controllers" as shown in the following screenshot.

Figure 6-3. Device manager.



6.1.5 Downloading a file using command script files

J-Link Commander can also be used in script mode that allows the user to use this application for batch processing and without user interaction. When using J-Link commander in script mode, the path of a script file is passed to it. The syntax in the script file is the same as when using regular commands in J-Link commander (one line per command).

To do easier to load the bin files of the Service and Base Lite nodes software examples, Atmel provides you a script, *program_bin.bat*, that lets you download the proper bin files in the right memory address position. You can find them in the following directory: *"\Software\PRIME_vaa.bb.cc.dd\Scripts\SN"* or *"\Software_vaa.bb.cc.dd\Scripts\BN"*.

Example: Scripts to load SN (Service Node with DLMS capabilities running as microcontroller mode) show an error when programming process falls. A typical error could be when the J-Link tool is a different version of the written in the script or in a different path. To solve it, edit the path according to your installation folder in the .bat file.

program_bin.bat file code:

```
"c:\Program Files (x86)\SEGGER\JLink_V496b\JLink.exe"
program_bin.jlink
```

Note: Edit the path to JLink.exe according to your installation folder and J-Link version.

program_bin.jlink file code:

```
exec device = ATSAM4C16C //device
speed 0
r //reset
h //halt
//load bin files
loadbin apps_prime_service_dlmsemu_ui_230.bin, 0x01000000
//loadbin apps_prime_service_modem_flash.bin, 0x01000000
loadbin prime_service_stack_0.bin, 0x010F0000
loadbin prime_service_stack_0.bin, 0x010E0000
```

```
r //reset
g //go
qc //quit and close
```

Note: Take into account that the previous memories allocation are to load binaries compiled with IAR

6.1.6 Atmel Software Framework (ASF)

The Atmel Software Framework (ASF) is a collection of embedded software for the Atmel Flash MCUs: megaAVR, AVR XMEGA, AVR UC3 and SAM devices.

It simplifies the use of our microcontrollers by providing an abstraction to the hardware and high-value middleware. ASF is designed to be used for evaluation, prototyping, design and production phases. The intention of ASF is to provide a rich set of proven drivers and code modules developed by Atmel experts to reduce customer design-time. It simplifies the usage of microcontrollers, providing an abstraction to the hardware and high-value middleware.

ASF is integrated in the Atmel Studio IDE with a graphical user interface or available as standalone for GCC, IAR compilers. ASF can be downloaded for free. ASF is an open-source code library designed to be used for evaluation, prototyping, design and production phases.

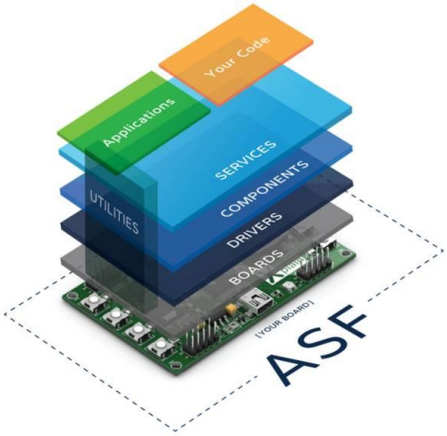
The Atmel Software Framework is split in six main parts, the *avr32/* directory, the *xmega/* directory, the *mega/* directory, the *common/* directory, the *sam/ directory* and the *thirdparty/* directory. These six directories represent the Atmel AVR UC3 architecture, the Atmel megaAVR, the Atmel AVR XMEGA architecture and the Atmel SAM architecture, what are common between all architectures and finally third party libraries.

Each architecture (and the common directory) is split into several subdirectories, these directories contain the various modules; boards, drivers, components, services and utilities.

See the list below and Figure 6-4 for an overview of how the various modules are wired together:

- *Boards* contain mapping of all digital and analog peripheral to each I/O pin of Atmel's development kits.
- *Drivers* is composed of a *driver.c* and *driver.h* file that provides low level register interface functions to access a peripheral or device specific feature. The services and components will interface the drivers.
- *Components* is a module type which provides software drivers to access external hardware components such as memory (e.g. Atmel DataFlash®, SDRAM, SRAM, and NAND flash), displays, sensors, wireless, etc.
- *Services* is a module type which provides more application oriented software such as a USB classes, FAT file system, architecture optimized DSP library, graphical library, etc.
- *Utilities* provide several linker script files, common files for the build system and C/C++ files with general usage define, macros and functions.
- *Applications* provide application examples that are based on services, components and drivers modules. These applications are more high level and might have multiple dependencies into several modules.

Figure 6-4. ASF modules structures.



Download link for more information: http://www.atmel.com/tools/AVRSOFTWAREFRAMEWORK.aspx. Please do not hesitate to visit our web site to get the last library updates.

6.1.7 First steps with IAR

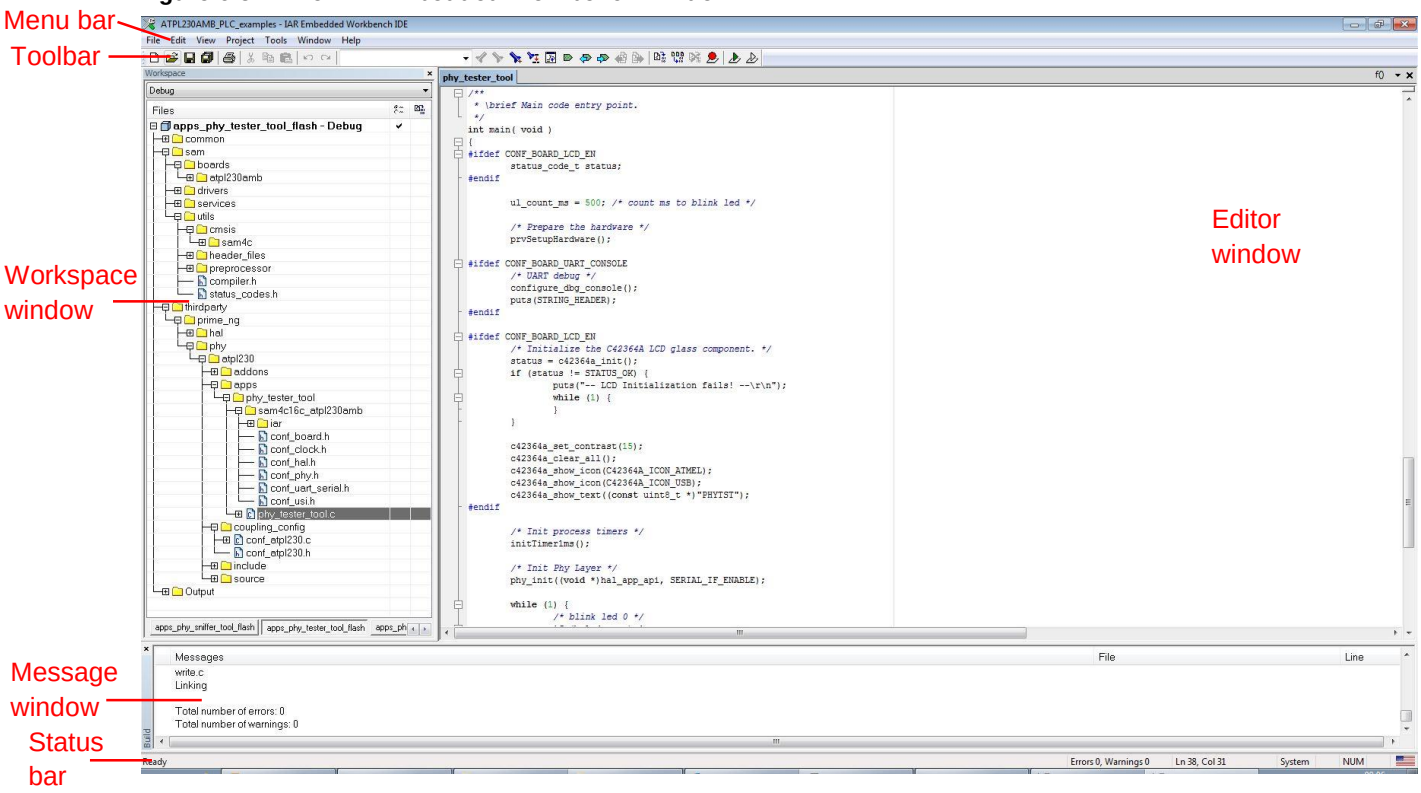
When working with programming in general it is important to have some structure in your coming projects and code. IAR Embedded Workbench is made to support such demands. The upper abstraction of a task is called Workspace, within each workspace you can add projects. The projects added in a workspace could be supporting the same device or have something in common. Each project contains code and settings for each target. So what we need to do is first make a workspace, then add a new project to this workspace. When this is done you should be able to include an application code to your project and make all the settings for the target SAM4C16C on your ATPL230AMB board.

IAR Embedded Workbench supports the SAM4C core family (preferred 7.10.1 versions or above).

Once the IAR Embedded Workbench has been installed. Open IAR Embedded Workbench. Click the Start button on the Windows taskbar and choose All Programs>IAR Systems>IAR Embedded Workbench for ARM>IAR Embedded Workbench. The file *IarIdePm.exe* is located in the *common\bin* directory under your IAR Systems installation, in case you want to start the program from the command line or from within Windows Explorer. The workspace file has the filename extension *eww*. If you double-click a workspace filename, the IDE starts. If you have several versions of IAR Embedded Workbench installed, the workspace file is opened by the most recently used version of your IAR Embedded Workbench that uses that file type, regardless of which version the project file was created in.

The following figure shows the main window and its default layout.

Figure 6-5. The IAR Embedded Workbench window.



Let's have a closer look to the environment now. Basically, the environment is split into five different areas:

- Editor window: allows you to edit the source files.
- Workspace window: shows the project structure.
- Message window: displays messages from the compiler.
- The Menu bar lets us the menu commands.
- The IDE toolbar—available from the View menu—provides buttons for the most useful commands on the IDE menus, and a text box for typing a string to do a quick search.
- The Status bar at the bottom of the IAR Embedded Workbench IDE main window —available from the View menu— contains useful help about how to arrange windows that they can be enabled from the View menu.

Open the PRIME PHY workspace for SAM4C16C platform *ATPL230AMB_PLC_examples.eww*. For that, on the start page, click on File>Open>Project/Solution. And select the project in the folder: *".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\apps\wrkspcs\iarew_workspace"*.

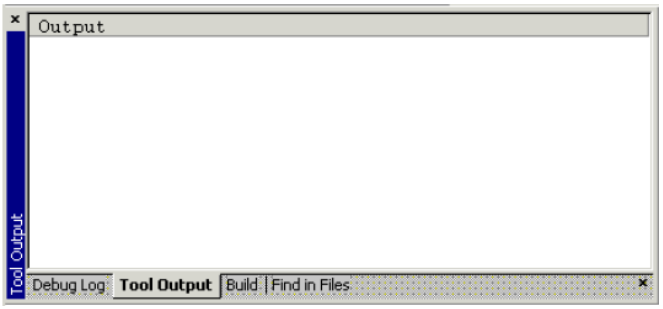
Once you have loaded the workspace, select the apps_phy_tester_tool project. Now, you can see the PRIME PHY project structure (expand the tree structure) in the workspace window. That structure is showed in the Figure 6-5.

6.1.7.1 Building, programming and debugging a project with IAR

Now you can create, build, program and debug the Atmel PRIME Examples using the IAR. But before to do this, you can configure and customize your project as you want; i.e., adding the Output window, show the line numbers, change the language options, etc.
Tool Output window is available by choosing View>Messages>Tool Output. The Tool Output window displays any messages output by user-defined tools in the Tools menu, provided that you have selected

the option *Redirect to Output* Window in the *Configure Tools* dialog box. When opened, by default, this window is grouped together with the other message windows.

Figure 6-6. Tool Output window.



The Language options are available by choosing Tools>Options. Use this page to specify the language to be used in windows, menus, dialog boxes, etc.

For example, it is very useful to enable line number display feature. For that, show the editor window and tick the *Show line numbers* options. Editor options window is available in Tools>Options. In addition to this, you can use this window to configure the editor.

In order to build the project, choose a build configuration in the combo box of the workspace window. By default, the IDE creates two build configurations when a project is created—Debug and Release. Every build configuration has its own project settings, which are independent of the other configurations. For example, a configuration that is used for debugging would not be highly optimized, and would produce output that suits the debugging. Conversely, a configuration for building the final application would be highly optimized.

You can build your project either as an application project or a library project. You have access to the build commands both from the *Project* menu and from the context menu that appears if you right-click an item in the Workspace window. To build your project as an application project, choose one of the three build commands *Make* (F7), *Compile* (Ctrl+F7), and *Rebuild All*. They will run in the background, so you can continue editing or working with the IDE while your project is being built.

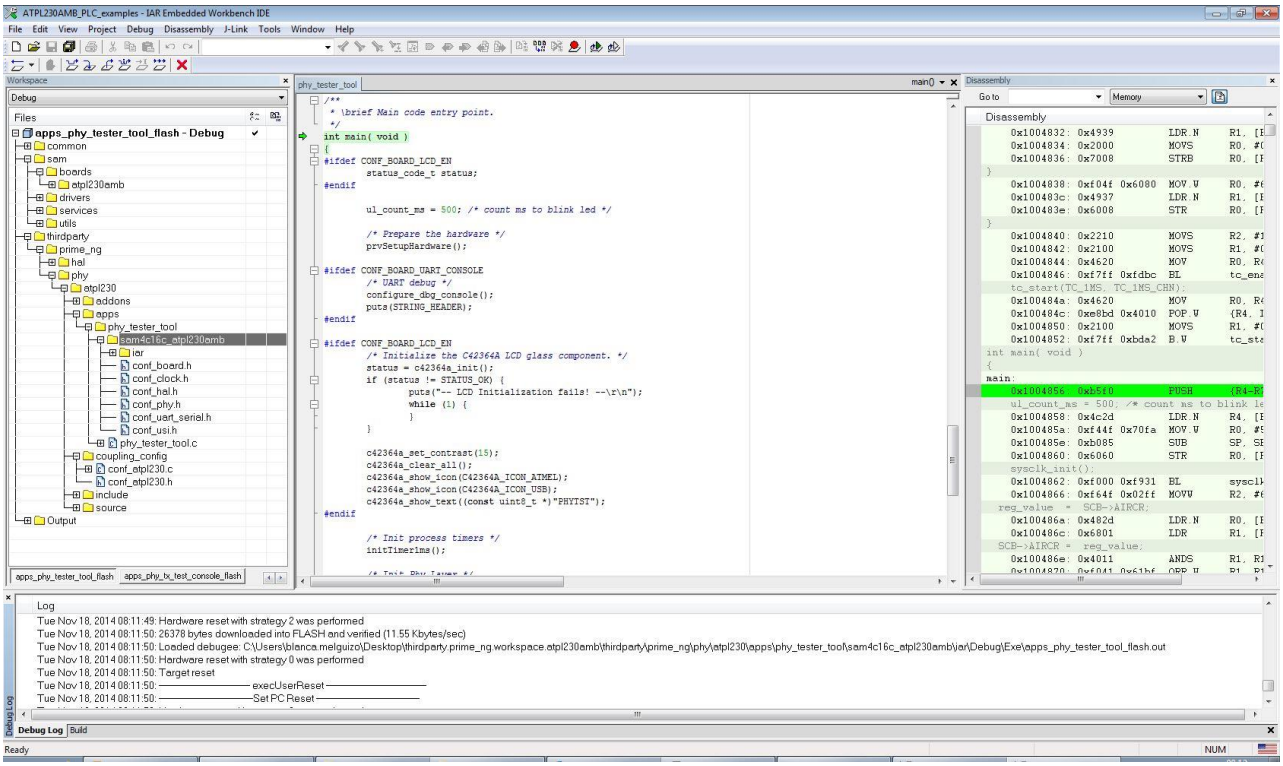
Error messages are displayed in the *Build window*. If your source code contains errors, you can jump directly to the correct position in the appropriate source file by double-clicking the error message in the error listing in the *Build window*, or selecting the error and pressing Enter. After you have resolved any problems reported during the build process, you can directly start debugging the resulting code at the source level.

Process to build, compile, load and debug the project over the board could be:

1. Choose Project>Make or click the Make button  on the toolbar. The part should compile with no errors.
2. Connect the SAM-ICE JTAG probe.
3. Supply on the board.
4. Choose Project>Download and Debug or click the *Download and Debug* button  on the toolbar to download your program to the board.
5. The file *phy_tester_tool.c* is now open in the editor window and the program is stopped at the start. Choose Debug>Go or click the Go button  on the toolbar to start the application. Your IAR IDE window should now look like Figure 6-7.
6. Once the board is powered, the green led D5, LED0, is blinking.
7. To stop C-SPY, click the Break button  on the debug bar.

8. To exit C-SPY, click the Stop Debugging button  on the toolbar.
9. To exit the IAR Embedded Workbench IDE, choose File>Exit. You will be asked whether you want to save any changes to editor windows, the projects, and the workspace before closing them.

Figure 6-7. The IAR Editor window.



For examples of building application and library projects, see the tutorials in the Information Center. For more information about building library projects, see the IAR C/C++Development Guide for ARM.

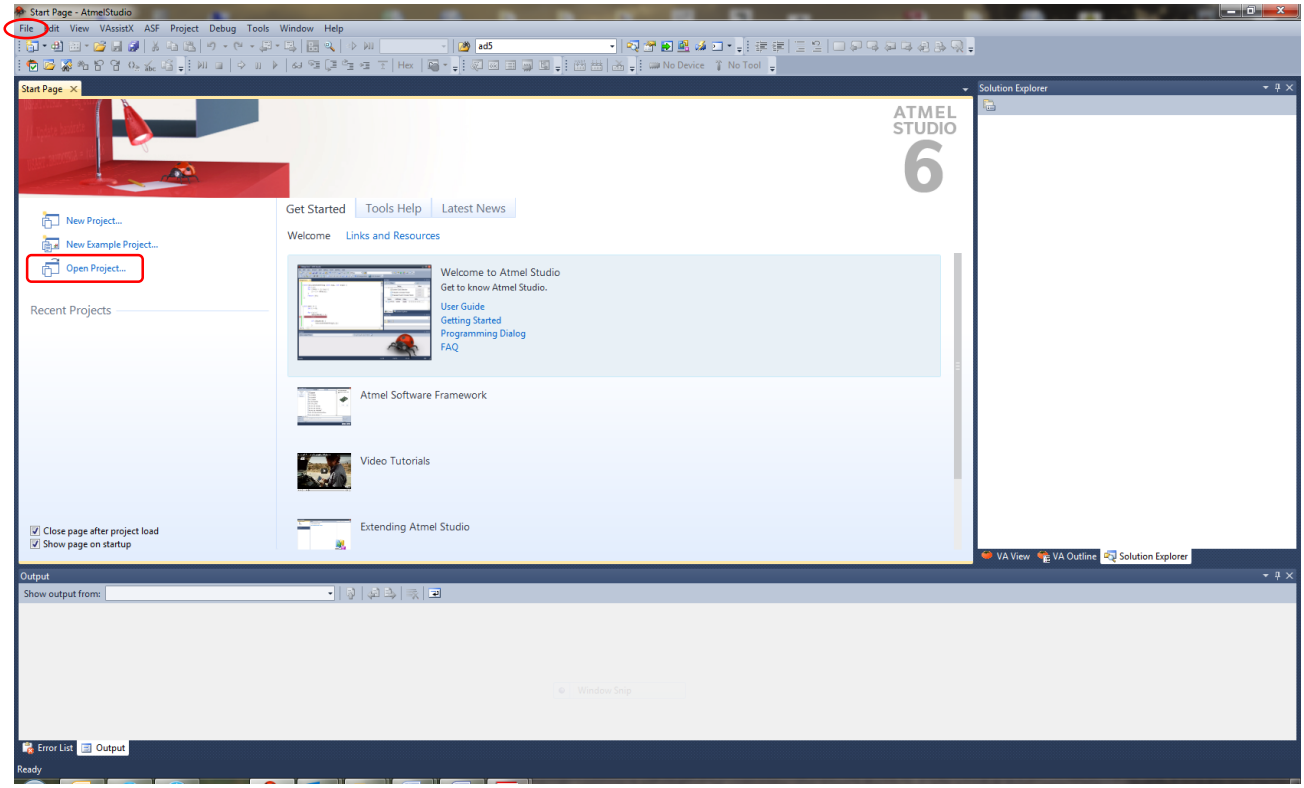
6.1.8 First steps with Atmel Studio 6.2

Atmel Studio 6.2 supports the SAM4C core. Once Atmel Studio 6.2 is installed in your computer, launch Atmel Studio 6.2. Click the Start button on the Windows taskbar and choose All Programs>Atmel>Atmel Studio 6.2. The workspace file has the filename extension *atsln*. If you double-click a workspace filename, the IDE starts.

Note: Opening Atmel Studio 6.2 takes some time.

The following figure shows the main window and its default layout.

Figure 6-8. Start page of Atmel Studio 6.2.

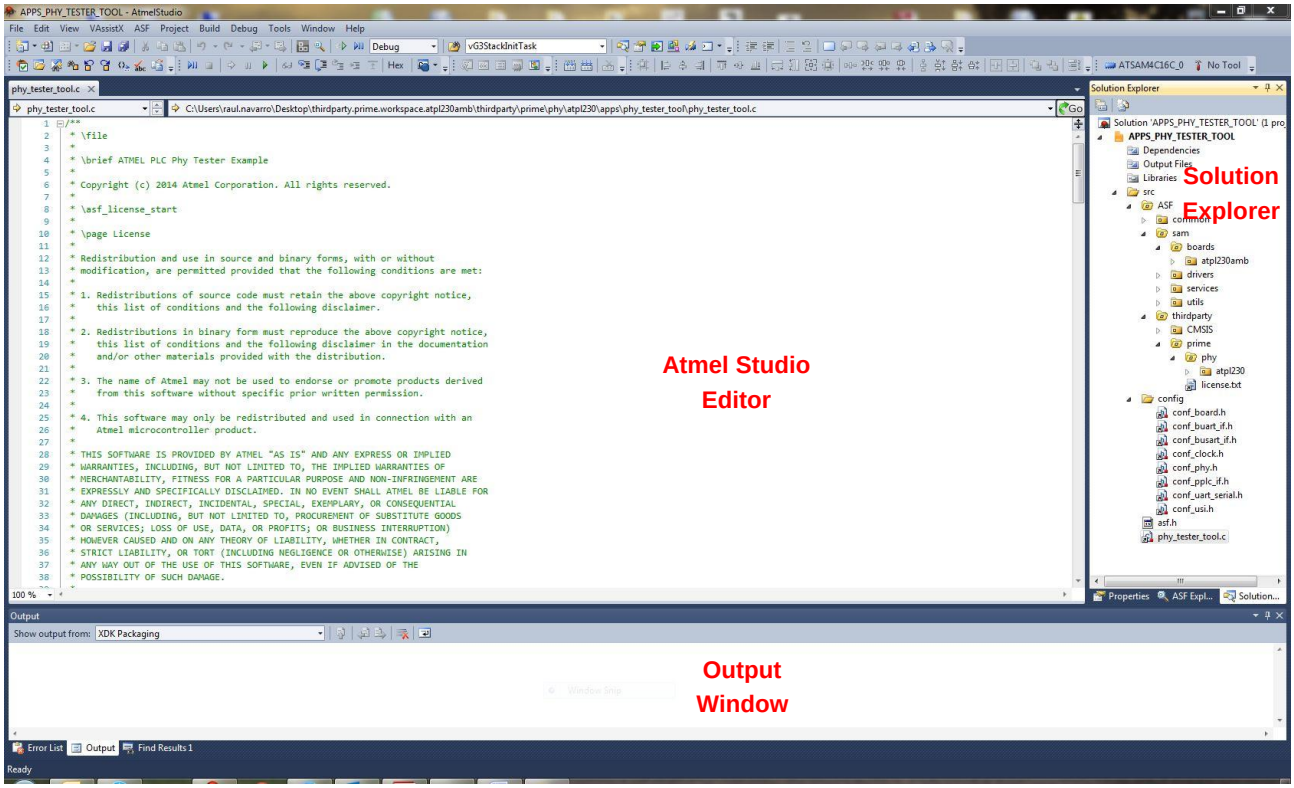


To avoid problems depending on the length of the path with Atmel Studio, we recommend install the Software folder contents of the evaluation kit in the root C:\.

And now, you can open the PRIME PHY workspace for SAM4C16C platform, *ATPL230AMB_PLC_examples.atsln*. For that, you have to click on Open Project or on File>Open>Project/Solution on the Start page and select the project in the folder: *".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\apps\wrkspcs\as_solution"*.

Once you have loaded the workspace, select the apps_phy_tester_tool project. The Solution should appear in the integrated development environment as in the figure below.

Figure 6-9. Atmel Studio 6.2 interface.



Let's have a closer look at the environment now. Basically, the environment is split into three different areas:

- Atmel Studio Editor: allows you to edit the source files.
- Solution Explorer: shows the project structure.
- Output Window: displays messages from the GCC compiler.

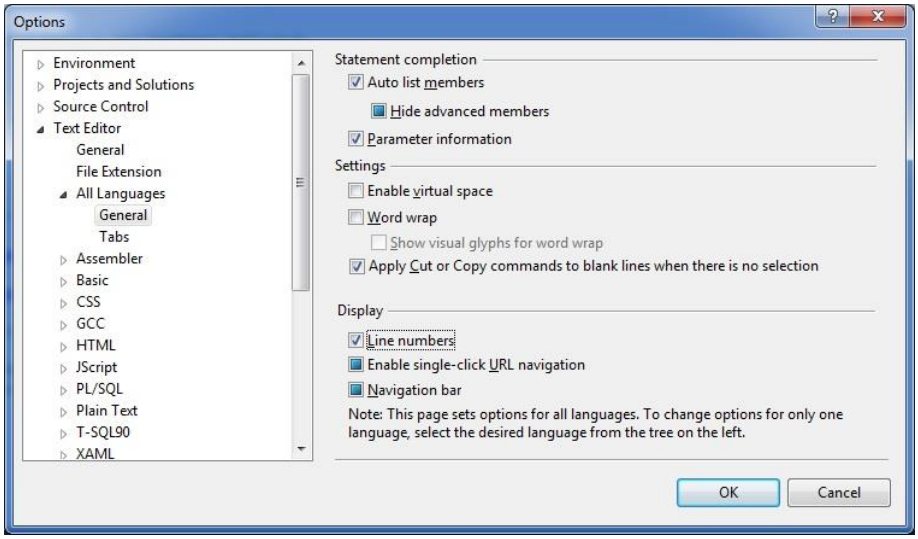
Once you have opened the *apps_phy_tester_tool.atsln* project you can see the PRIME PHY project structure (expand the tree structure) in the solution explorer area. That structure is showed in the Figure 6-9.

6.1.8.1 Building, programming and debugging a project with AS

Now you can create, build, program and debug the Atmel PRIME PHY Examples using the AS. But before to do this, you can configure and customize your project. For example it is very important to enable line number display feature in Atmel Studio 6.2 editor. For that:

- Access to Editor Function by clicking on Tools>Options and access to *All Languages* window in the *Text Editor* tab.
- Enable the Display Line numbers function.

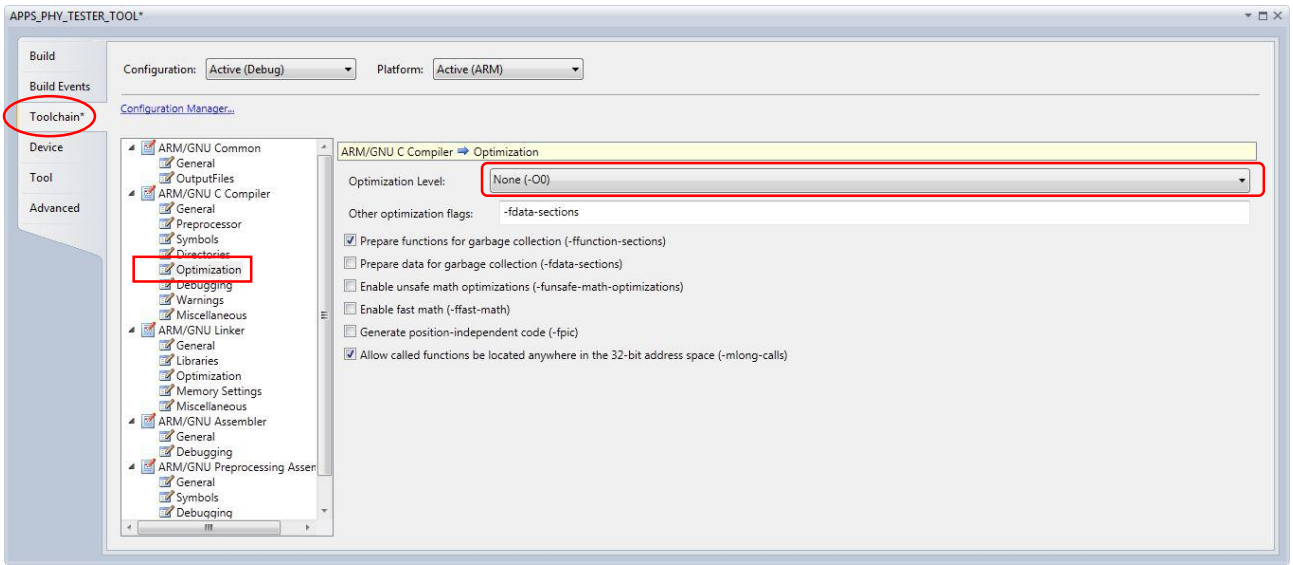
Figure 6-10. Line numbers enabling.



Another important feature is to disable the optimization in Atmel Studio 6.2 editor when you are in *Debug* mode to avoid jumping into the lines of code without order –due to the optimization-. For that:

- Access to Project Properties by clicking on Project>Properties and access to *Toolchain* window in the *Project Properties* tab.
- Select *Optimization* option in *ARM/GNU C Compiler* main tree.
- Select *None* option in the display *Optimization Level* function.

Figure 6-11. Optimization option window

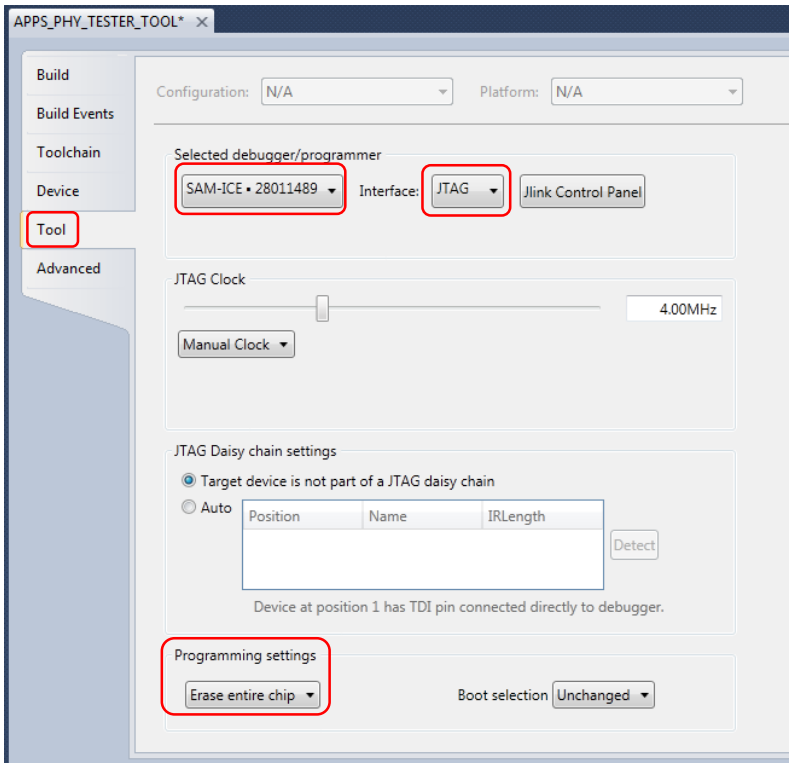


In order to build the project, click on the *Build* Solution button  or on Build>Build Solution. Make sure the SAM/ICE cable is connected from your board to your PC through the J13 connector. Power the ATPL230AMB board.

Then, download the program in the internal flash of the SAM4C16C by clicking on the *Start Debugging and break* button  .

The first time Atmel Studio will ask to select the Debug Tool. Select the on-board SAM-ICE (the serial number in parenthesis differs from one board to another) and check the programming settings combo box, see Figure 6-12.

Figure 6-12. Select tool instance.



Once programmed, start the code execution by clicking on the green arrow  .

When the debug session is running, the *Stop* button  allows you to stop the program execution and exit the debug session. If you just want to stop the program and keep the debug session active, simply click on the *Pause* button  .

If you modify any of the files of the project, you need to do a *Rebuild* and not only a *Build*. Do a right-click on the project name in the Solution Explorer and then click on the *Rebuild* button.

In case you only want to download the program on the SAM4C16C without debugging, clicking on the *Start Without Debugging* button  .

Close the project on the toolbar File>CloseSolution.

For further information, please refer to the tool's embedded help (in the menu bar) or visit the webpage: http://www.atmel.com/microsite/atmel_studio6/default.aspx.

6.2 PLC application example 1 – PHY Tester

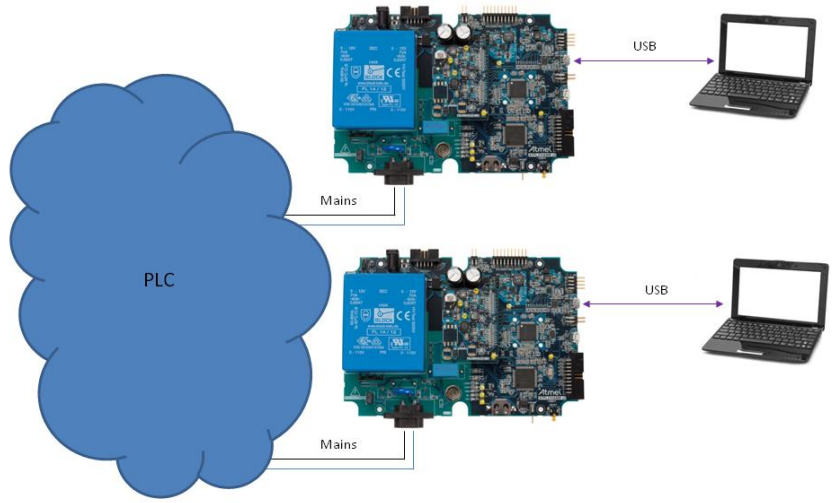
The boards of the kit, by default, are programmed with the embedded PLC PHY Tester tool firmware for SAM4C16C device, *apps_phy_tester_tool.bin*. It is an application example that shows the capabilities of the ATPL230A in a point-to-point connection (physical layer). This application requires a pair of boards and a PC tool, Atmel PLC PHY Tester tool, which has to be installed in the user's host PC to interface with the boards.

In any case, if you want to load this file again, you have to build the project *apps_phy_tester_tool* to generate the output file to program. See section 6.2.4 to know more about programming the ATPL230AMB boards.

Atmel recommends to load the binary generated with the last PHY Tester Tool project released in the kit to evaluate the board with last improvements.

After installing the Atmel PLC PHY Tester tool in your PC(s), connect the two boards to the grid and to the host(s) PC(s) as shown in the following figure.

Figure 6-13. ATPL230AMB Boards connection scheme.

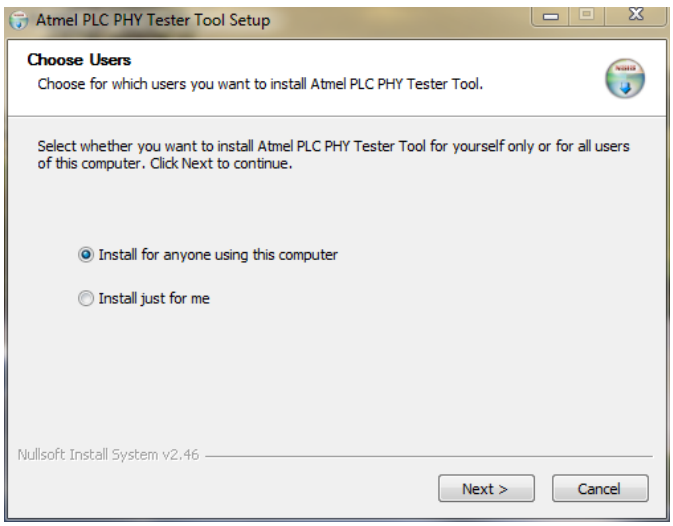


Following chapters explain to you how to install the PC tool, supply the boards, and select the UART0 to communicate with the SAM4C16C. Load the firmware and run the application.

6.2.1 Atmel PLC PHY Tester tool Installation

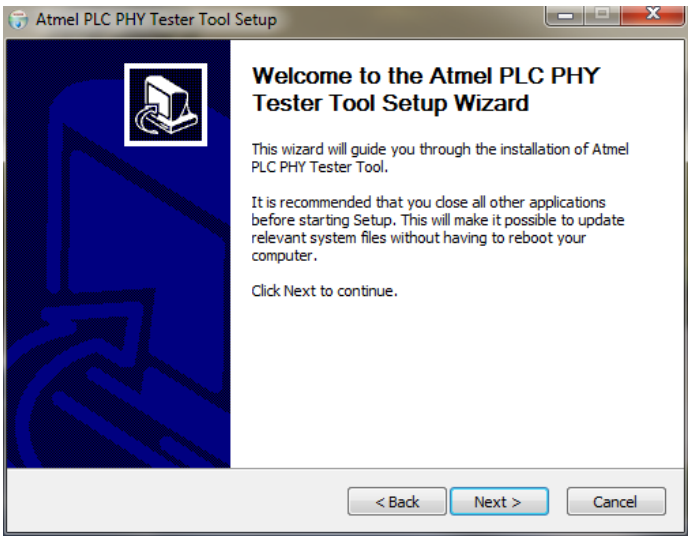
To install Atmel PLC PHY Tester tool in a Windows Operating System, execute the provided installer in the Tools folder: "*.\PCTools\ATMEL_PLC_PHY_Tester\ATMEL_PLC_PHY_Tester_Tool_vX.Y.Z.exe*", and follow the installation wizard. The installer wizard should open. To follow with the installation, click Next.

Figure 6-14. Installation process, slide 1.



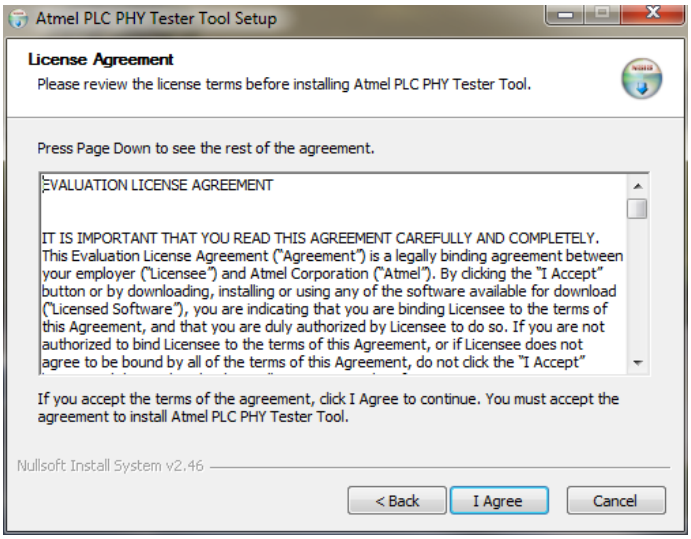
Select the users' permissions and click Next.

Figure 6-15. Installation process, slide 2.



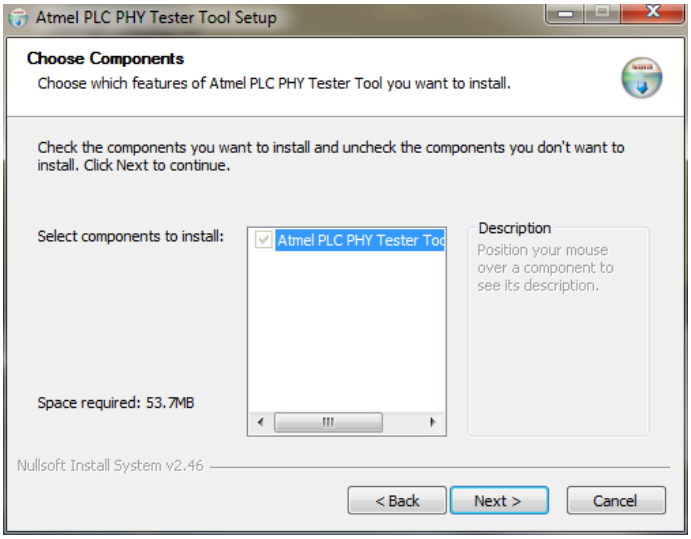
Click Next to continue.

Figure 6-16. Installation process, slide 3.



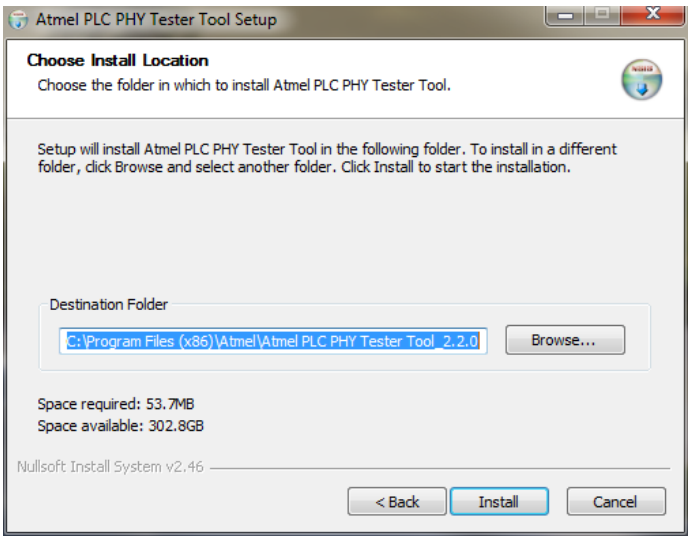
Click I Agree to continue.

Figure 6-17. Installation process, slide 4.



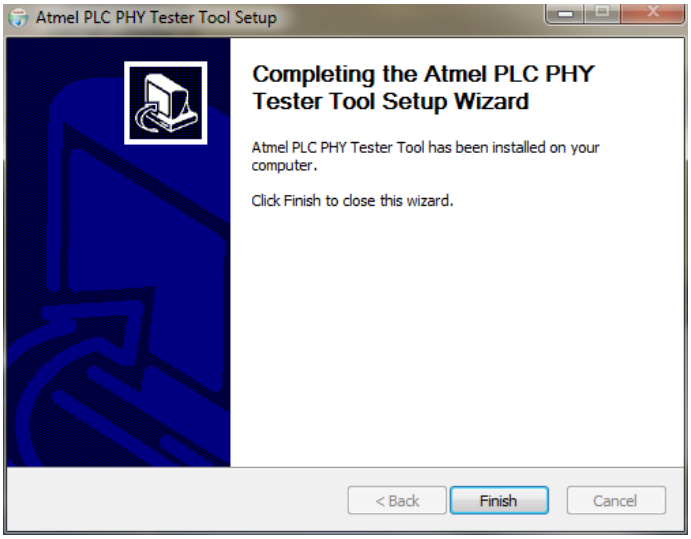
Click Next.

Figure 6-18. Installation process, slide 5.



Setup will install the program in the *Destination Folder*. To install in a different folder, click Browse and select your destination folder. Click Install.

Figure 6-19. Installation process, slide 6.



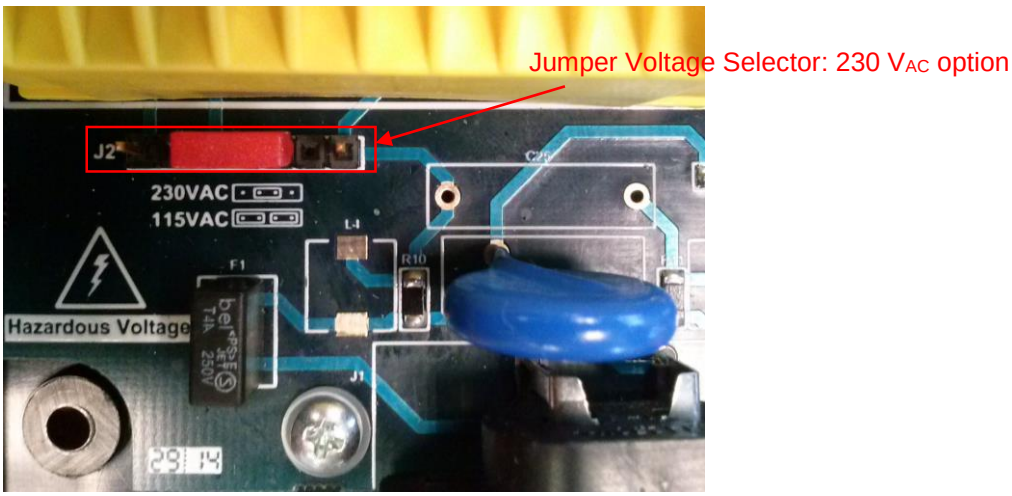
Click Finish.

Now the program is installed in your computer and a shortcut should have been created in your desktop.

6.2.2 Supplying the boards

Kits are provided with power cord cables in order to connect the boards to the mains. Mains connector is shown below in Figure 6-20. Please connect the provided power cord cable with the kit to the *Power Cord Connector, J1,* in order to supply the board.

Figure 6-20. ATPL230AMB mains and voltage jumper selector.



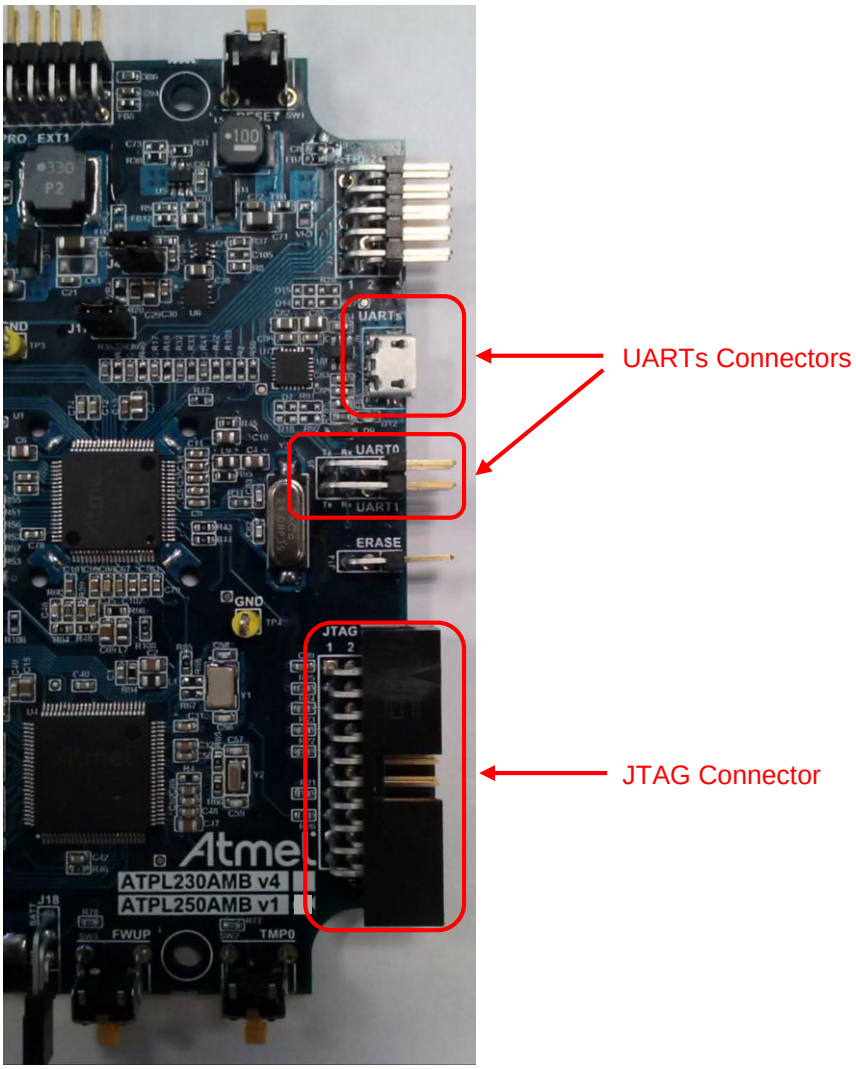
Note that the ATPL230AMB board can be supplied either with $100V_{AC}$ or $230V_{AC}$ by setting the proper jumpers in the voltage selector, J2, as depicted in the Figure 6-21. By default, voltage jumper is set for $230V_{AC}$. For more information about power supply section, see section 3.5.1.

Figure 6-21. Jumper configuration for $100V_{AC}$ or $230V_{AC}$.

6.2.3 USB connection

By default, the programmed firmware for Atmel PHY Tester tool establishes serial communication with **UART0**. Boards have such UART0 available either by micro-B USB connector, J9, or the triple pin row CMOS connector, J16. See the figure below and sections 3.5.6.4 for more information about the USB device. Kits are provided with two micro USB cables in order to connect the user's host(s) PC(s) with the two boards.

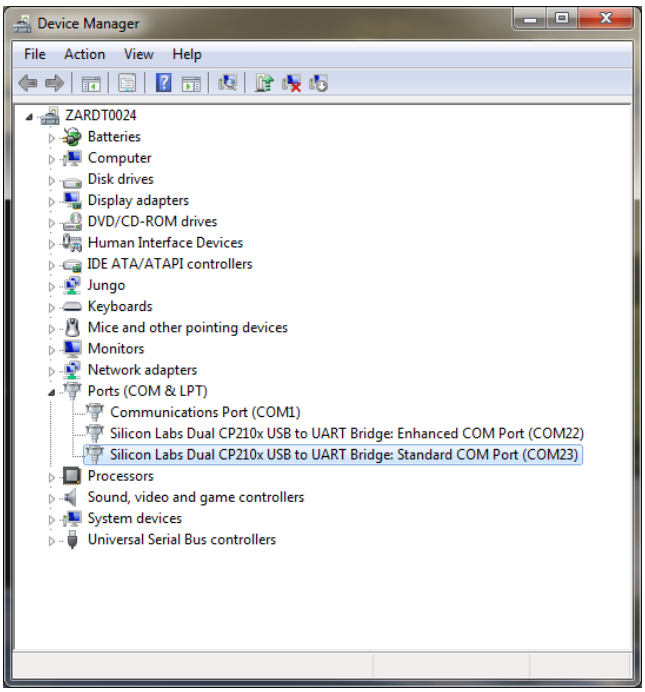
Figure 6-22. UART & JTAG connectors.



Connect the USB cable to the micro-B USB connector and the host PC. If the PC does not recognize the USB, download the USB driver from the manufacturer webpage or take it from the *PCTools* folder: "*.\PCTools\USB_Drivers*". Once the driver is downloaded, unpack the driver archive to a folder on the host PC's hard-disk. Connect the USB cable to the board. The new hardware installation will recognize the new board and will guide you through the USB driver installation. When the wizard asks for the driver to install, navigate to the directory where the driver archive has been unpacked to.

Identify the new hardware in the *Windows Device Manager*. The assigned COM port number is needed when configuring the PHY Tester tool application later. See the following figure for an example of a COM port assignment.

Figure 6-23. Windows device manager.



As you can see in the figure above, the CP210x USB to UART Bridge Virtual COM Port (VCP) appears as two COM ports (Enhanced and Standard COM ports) in the Device Manager. They are assigned the lowest available COM ports for operation. In the ATPL230AMB design, the Enhanced COM port corresponds to UART0 and the Standard COM port to UART1, so select the Enhanced COM Port when you use the Atmel PHY Tester PC tool.

6.2.4 Programming the embedded file

The boards of the kit are programmed with the embedded PLC PHY Tester tool firmware for SAM4C16C device, *apps_phy_tester_tool.bin*. In this chapter we explain how to load an embedded file. The process and tools to load the embedded file in the ATPL230AMB boards are always the same. Remember that all these tools and performance are described in chapter 6.1.

To be able to develop applications, build binaries and program the firmware on the SAM4C16C device, you can use the IAR Workbench or the Atmel Studio IDE.

In order to program the firmware on the board, the JTAG connector is used (see section 3.5.6.3 about JTAG programming mode) and JTAG probe is required. See previous Figure 6-22, which shows the JTAG connector, J13, of the board. **Note that kits do not provide a J-Link ARM or SAM-ICE JTAG probe in order to connect to the user's host PC and the boards to download and debug the projects**.

The process to load the file should be as is explained below; in that process we use a programming tool, J-Link Tool. Remember that the J-Link USB drivers must have been downloaded previously from the Segger webpage (see section 6.1.4) and they depend on your operating system:

A. Place the JTAG connector of the J-Link or SAM-ICE in the J13, JTAG connector of the board. Check pin number 1 of J13 connector to place the cable in the right position. See the Figure 6-22.
B. Switch on the power supply of the board.
C. There are two ways to program the board:
    a. Launch the IAR or AS IDE and select the PHY Tester tool project. Build the project *apps_phy_tester.eww* or *apps_phy_tester.atsln* to generate the output file. Now you can

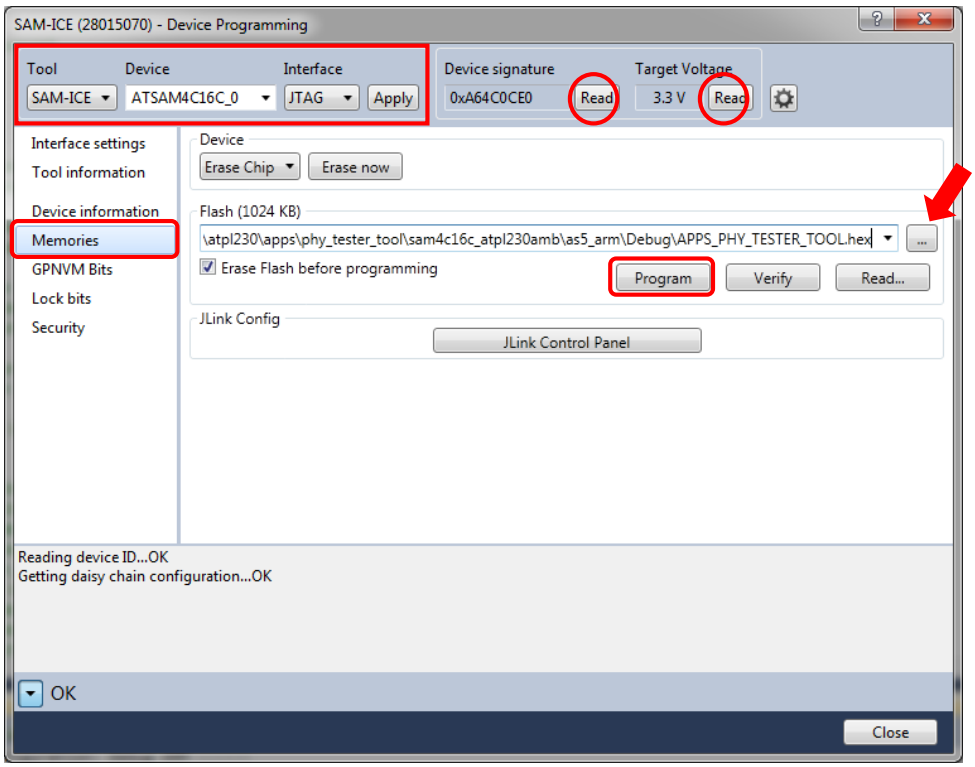
download the file to the board. Process to load the output file is commented in sections 6.1.7.1 or 6.1.8.1.

b. Another way could be using the *Device Programming* Instance of the Atmel Studio IDE. Once the output file has been created, you can load the program in the flash memory. In the menu bar, go to Tools>Device Programming. Select the tool, device and interface and press *Apply* button. Then, press *Read* button of *Device signature*. Go to *Memories* window, select the output file (.hex or .elf) and press *Program* button. When finished, power cycle the board to run the program.

Remember that, every PHY Tester tool example project is contained in the following folder: *".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\phy\atpl230\apps\phy_tester_tool\sam4c16c_atpl230amb".*

And also in the PHY common workspace, ATPL230A_PLC_examples, which is contained in following Software folder: *".Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\apps\wrkspcs\".*

Figure 6-24. Device Programming instance.



PHY Tester Tool project has been created for the default PLC coupling board, ATPLCOUP001v1. So, if you are going to use another coupling board, you must build the PHY Tester Tool project with the correct configuration. For that, open the IDE tool used and open the project application, *apps_phy_tester_tool.atsln* or *apps_phy_tester_tool.eww*. After that, select the file *conf_phy.h*, that it is located in the PHY project configuration directory:

"./Software/PRIME_vaa.bb.cc.dd/phy.atpl230amb\thirdparty\prime_ng\phy\atpl230\apps\phy_tester_tool\sam4c16c_atpl230amb", find the define function to select the coupling board configuration (see Figure 6-25). Change the board name to desire board and build to generate the output file.

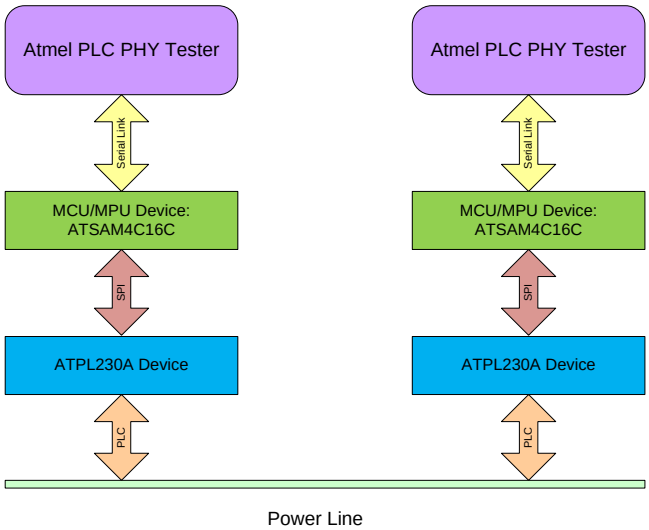
Figure 6-25. Coupling board configuration definition.

```
44 #ifndef CONF_PHY_H_INCLUDE
45 #define CONF_PHY_H_INCLUDE
46
47 /* Select Coupling Board Configuration (see values in atpl230.h) */
48 #define BOARD_COUPLING    ATPLCOUP001_v1
49
50 #endif  /* CONF_PHY_H_INCLUDE */
51
```

6.2.5 Running the PLC application example 1

The Atmel PLC PHY Tester tool is used to control the application running on the SAM4C16C+ATPL230A. As you can see in Figure 6-26, the two boards are plugged into the same power line. Users have to execute two instances of the PHY Tester tool – which has been previously installed in the host(s) PC(s) – in order to enable communication between both boards. Please note that these two instances may or may not run on the same computer.

Figure 6-26. Atmel PLC PHY Tester concept.



In order to know if the boards were programmed successfully you can check if the green led LED0, D5, is blinking. This indicates that the PHY Tester Tool application is running on SAM4C16C device.

You must select the same coupling boards to plug in both ATPL230AMB boards. Check the coupling identifier that you can find in the coupling board.

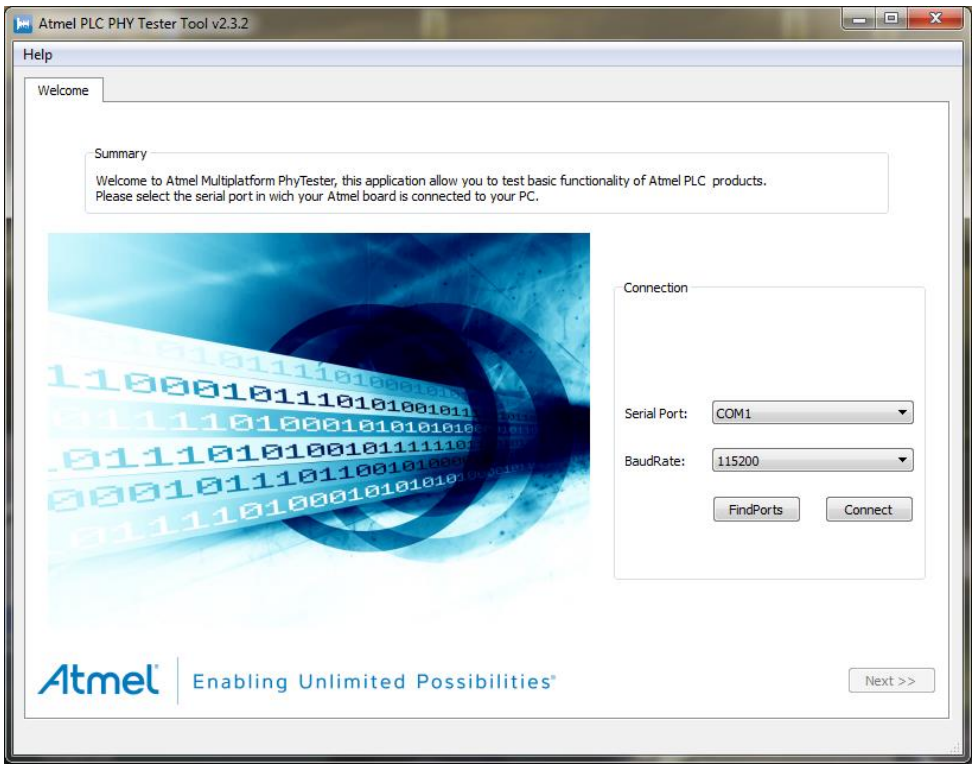
These coupling boards must be the proper one for the frequency band you want to send/receive, otherwise please remove them and connect the proper ones.

By default, ATPL230AMB board sets an ATPLCOUP001 coupling board, so V_{DD} voltage of ATPL230AMB must be 16 volts. V_{DD} can be regulated to 16 or 12 volts depending on the J16 jumper position. In this situation, jumper J16 must not set. See section 3.5.1 and Figure A-2 for more information.

Other coupling boards may require different voltage for the class D amplifier (V_{DD}).

Once the application is launched, *Starting Window* will appear (see Figure 6-27).

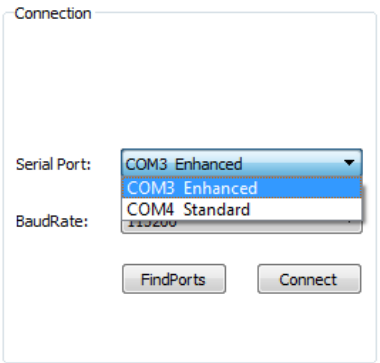
Figure 6-27. Welcome instance.



The first to do is configure the corresponding COM port for each board. In this window we select the serial connection configuration:

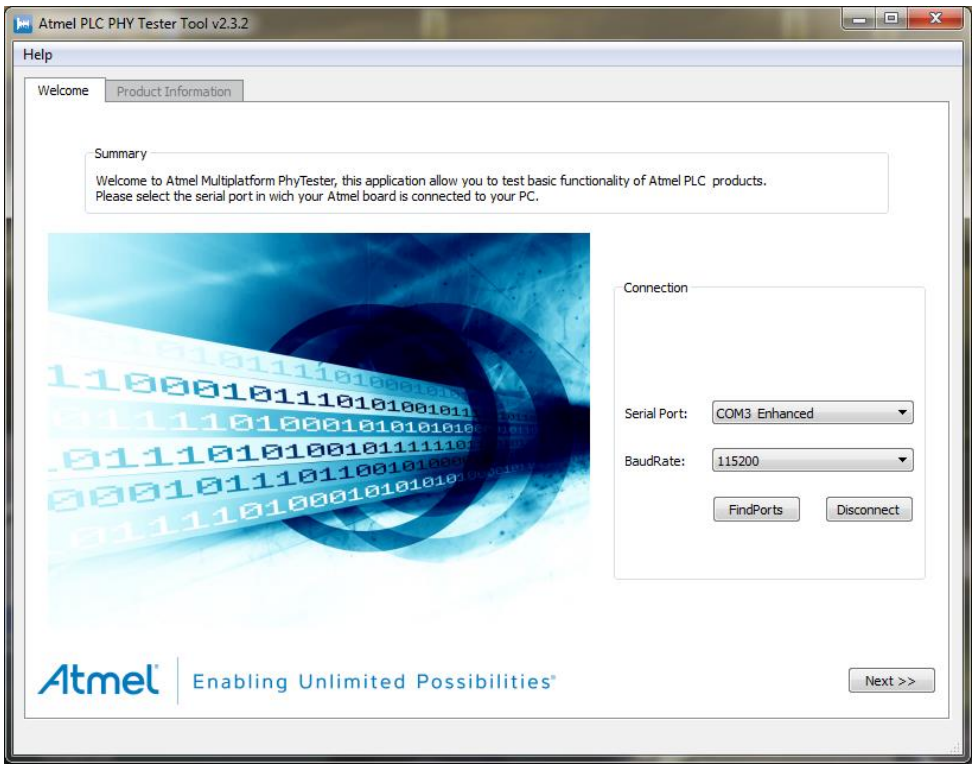
- Select in the Serial Port combo box the proper port to connect (see Figure 6-28). As it is explained in section 6.2.3, communication is by the Enhanced COM (UART0). If your COM port does not appear (see section 6.2.3), press Find Ports button.
- Select the BaudRate combo box of 115200 bauds.

Figure 6-28. Serial port selection.



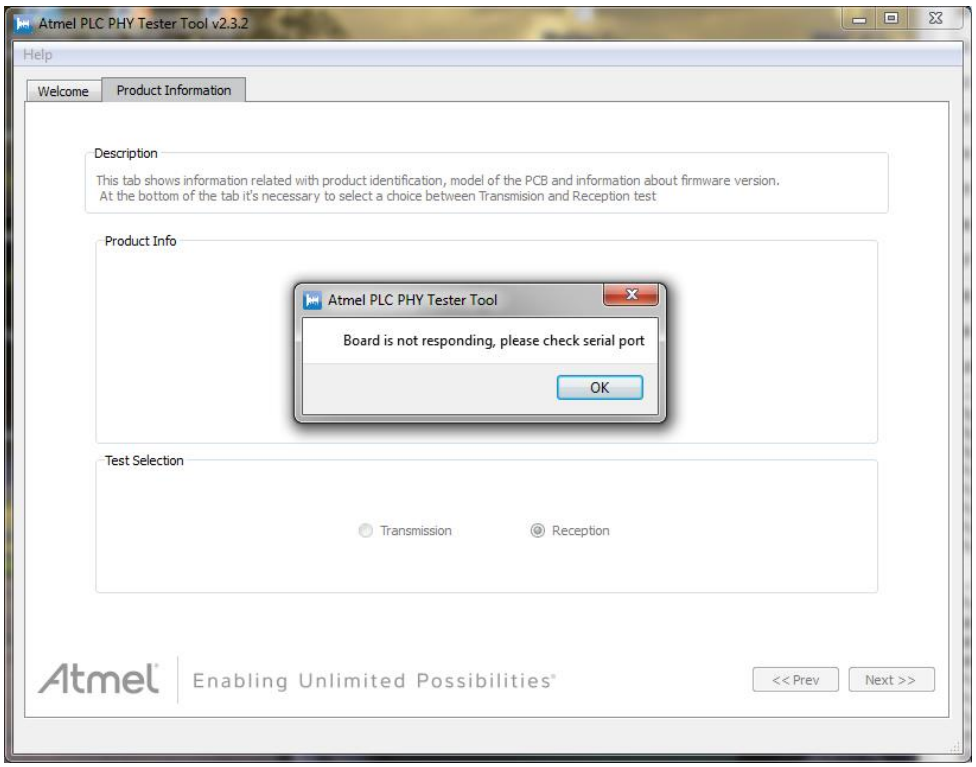
Once COM port is selected, click the Connect button. As soon as the button is clicked, the button text will change to *Connecting*. Then, the application and the board start a process of identification and, after few seconds, the button text will change to *Disconnect*. This means that the identification process has finished. A new Tab (*Product Information*) is appended to the wizard and Next button is enabled allowing the user to go to the following step of the configuration. See Figure 6-29.

Figure 6-29. Communication enabled.



In case the tool cannot establish a communication with the board, the tool shows the following error message.

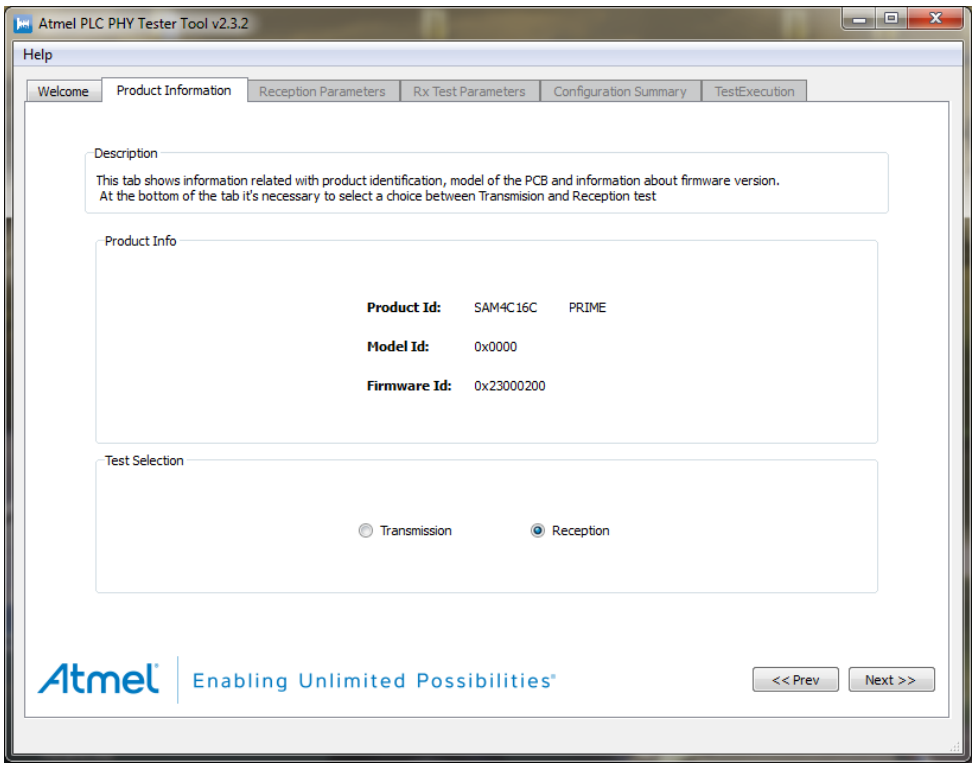
Figure 6-30. Communication error.



Click the OK button and press Prev button to get back to *Welcome* tab. Now press Disconnect button and check your connections. Either you have not selected the right Enhanced COM port or the board is not supplied or the downloaded firmware is not the right. After these operations, you can retry to establish the communication again between the board and the computer.

Once the communication is right, *Product Information* tab of the PHY Tester tool is shown below in Figure 6-31.

Figure 6-31. Product Information tab of the Atmel PLC PHY Tester tool.



The *Product Information* Tab shows basic information of the type of board connected to, and also asks the user to select the kind of test to be performed.

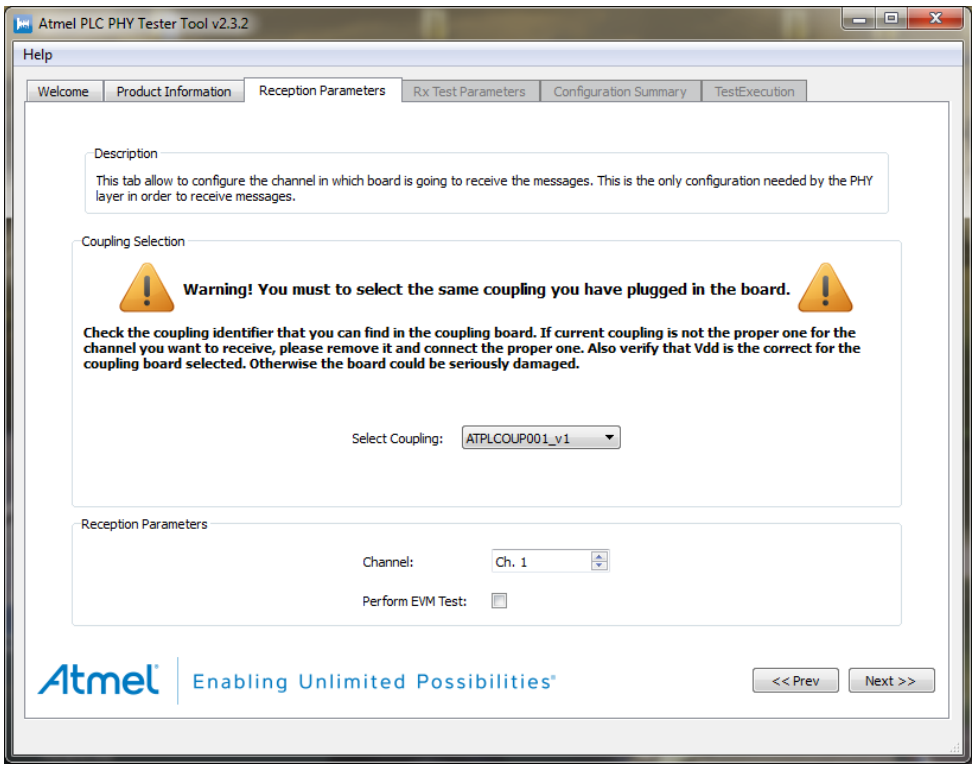
The showed information is related to the physical layer implemented in the firmware of the board:

- **Product ID:** it shows a text string that identifies the Atmel PLC product (platform).
- **Model ID:** It is a 16-bit unsigned integer that identifies the model of the board.
- **Firmware ID:** It is a 32-bit unsigned integer that identifies the physical layer firmware running in the board.

Now the user has to do a selection depending on whether the user selects transmission or reception test, different tabs are added. For reception tests, "*Reception Parameters*" and "*Rx Test Parameters*" tabs are added. For transmission tests, "*Transmission Parameters*" and "*Tx Test Parameters*" tabs are added. Finally, independently of the kind of selected test, two more tabs are added: "*Configuration Summary*" and "*Test Execution*".

First, we will describe the process to configure a board as receptor and after that we will describe how to configure the other board as emitter. Selecting the *Reception* option and clicking the Next button, a tab appears as the following image (Figure 6-32).

Figure 6-32. Reception Parameters tab.



This tab allows you to configure the channel reception of the PLC coupling board and the PLC coupling board plugged. Note the warning message displayed before to continue, check the proper voltage to use (V_{DD} selection). In this tab you can select one of the eight channels (1 to 8) and one of the five PLC coupling boards designed by Atmel.

In this example, the firmware load is for ATPLCOUP001 board, so that, it is only designed for PRIME channel 1.

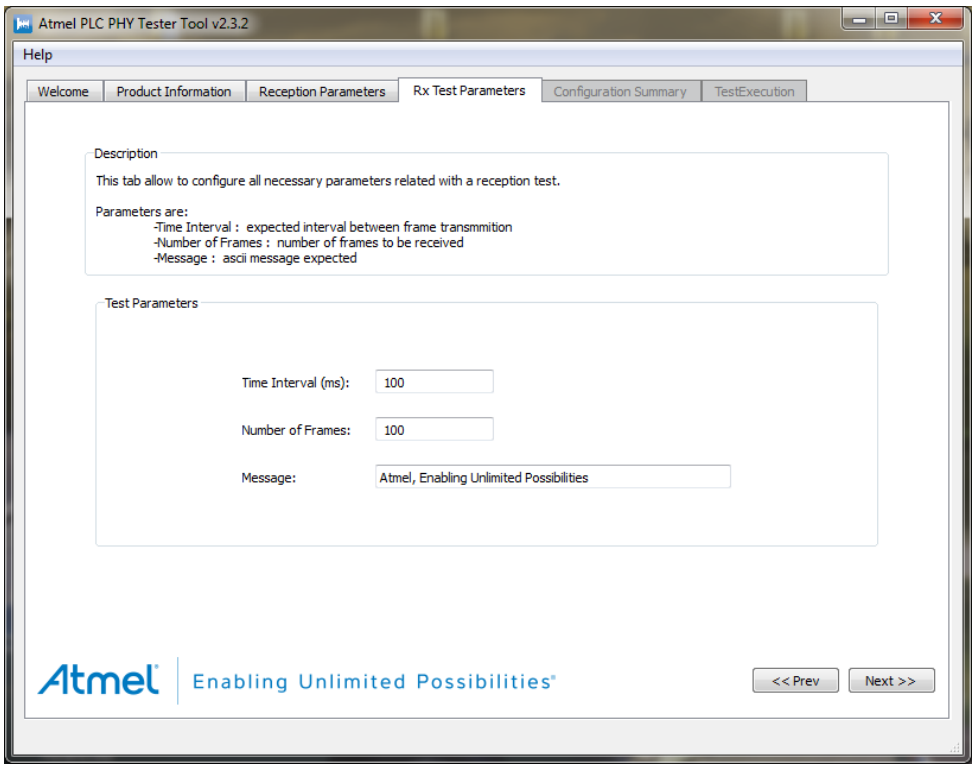
Selecting the option *Perform EVM Test* you can change the message and interval of transmission in order to make a test that evaluates the PHY layer performance. For more information check application note, doc43072**.** The Perform EVM Test box is disabled by default.

Click the Next button to continue.

The next tab shows the *RX Test Parameters*, see Figure 6-33. This tab is where the following reception test parameters are configured:

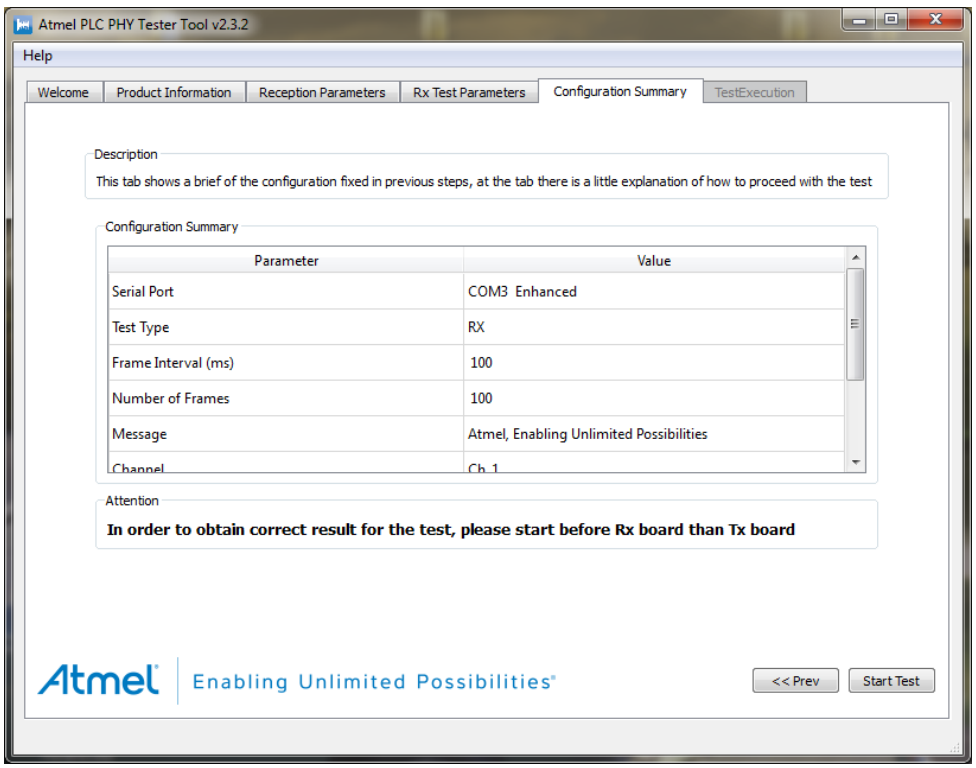
- Time Interval (milliseconds): expected interval between frame transmissions.
- Number of Frames: number of frames to be received.
- Message: ASCII message expected.

Figure 6-33. RX test parameters.



Default parameters are selected. Click the Next button to continue.

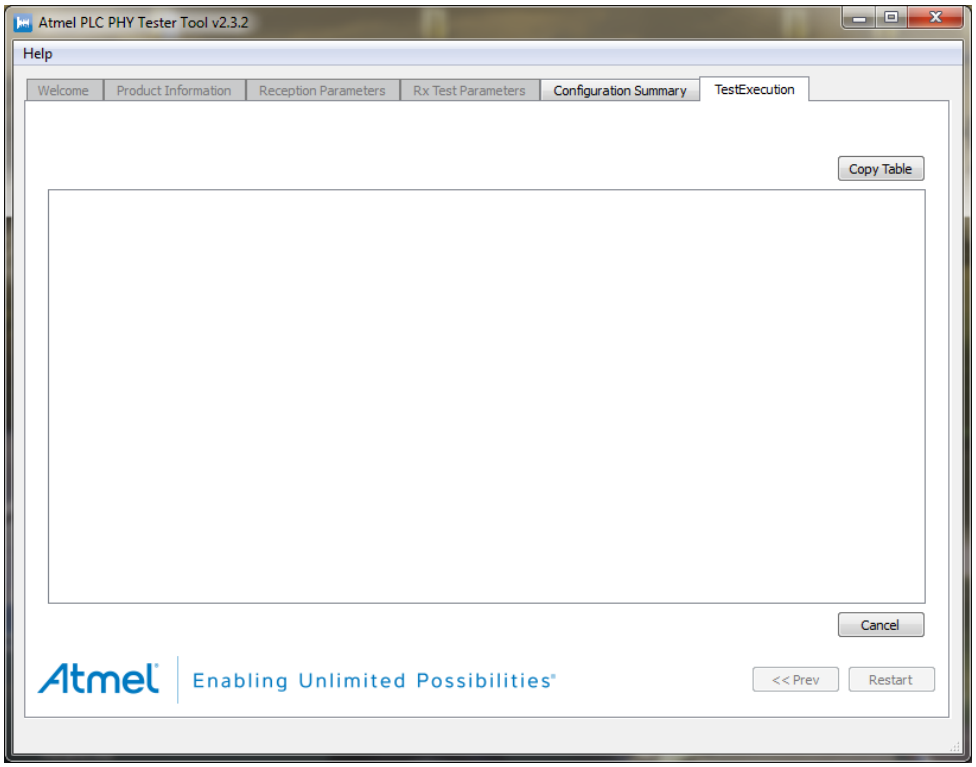
Figure 6-34. Configuration Summary tab.



The previous figure, *Configuration Summary* tab, shows a table where all the configuration parameters and their selected values are listed. It is recommended to check that all values correspond to the desired configuration before to continue.

To start the process, click the Start Test button. A new tab is enabled, at first the table is empty because any frame has been received. Note that there is a timeout to wait the frame reception.

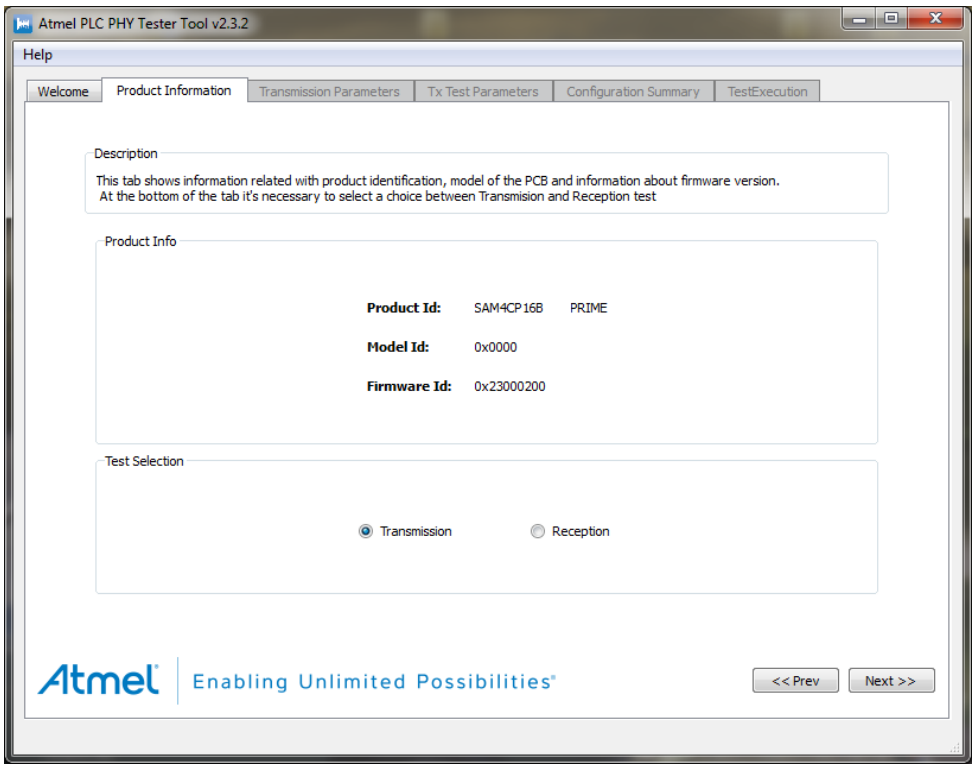
Figure 6-35. Test Execution tab.



Once the receiver board has been configured, the emitter board must be configured. Launch another Atmel PLC PHY Tester tool and once the transmission board is supplied and USB cable connected, configure the corresponding COM port for the board in the window *Starting Window*.

Once COM port is selected, click the Connect button. As soon as the button is clicked, the button text will change to *Connecting*. Then, the application and the board start a process of identification and, after few seconds, the button text will change to *Disconnect*. This means that the identification process has finished. A new Tab (*Product Information*) is appended to the wizard and Next button. This time we select in *Product Information* tab, the Transmission process (Figure 6-36).

Figure 6-36. Transmission option selection.



Once the transmission option is selected, click the Next button.

The *Transmission Parameters* tab appears (Figure 6-37) that allows you to configure the PLC coupling board plugged and the transmission parameters:

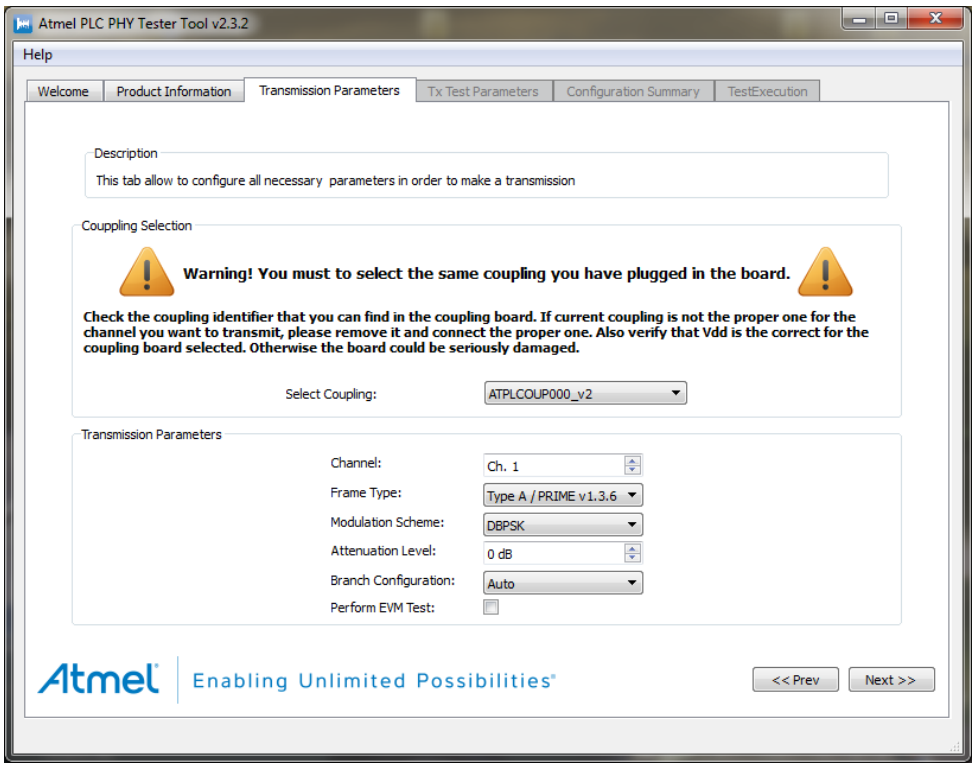
- PLC coupling board. Different boards are defined for different band plans and isolation modes.
- Channel. It allows selecting the channel in which the frames are going to be transmitted, depending on the coupling plugged to the board different channels can be available.
- Frame Type. Configure the board to transmit frame type A, B or C.
- Modulation Scheme. Configure the modulation scheme for frames.
- Attenuation Level. It allows attenuating the output certain amount of dBs (0-21 dB).
- Branch Configuration. It configures the output stage depending on the kind of impedance presented to the board.
- Perform EVM Test. Selecting the option *Perform EVM Test* you can change the message and interval of transmission in order to make a test that evaluates the PHY layer performance. For more information check application note: 43072_PHY_Performance_Verification**.**

Note the displayed warning message before to select the values. In this example, we select (Figure 6-37) the following values:

- PLC coupling board. We select *ATPLCOUP001_v1*. **This parameter must match with the reception parameter (Figure 6-32) for the test to be successful.**
- Channel. We select *Ch. 1*. **This parameter must match with the reception parameter (Figure 6-32) for the test to be successful.**
- Frame Type. We select *PRIME 1.3.6*.
- Modulation Scheme. We select *TypeA/PRIME v1.3.6*.
- Attenuation level. We select *0 dB.*
- Branch configuration. We select *Auto*.

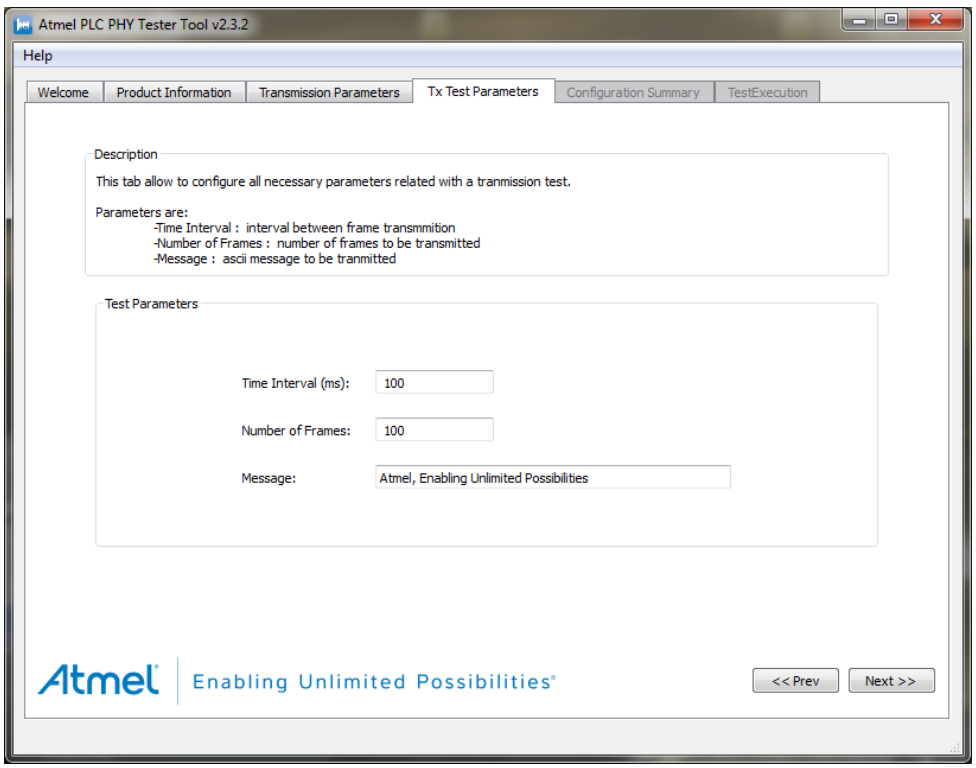
- Perform EVM Test. Disabled.

Figure 6-37. Transmission Parameters tab.



Click the Next button to continue.

Figure 6-38. TX Test Parameters tab.



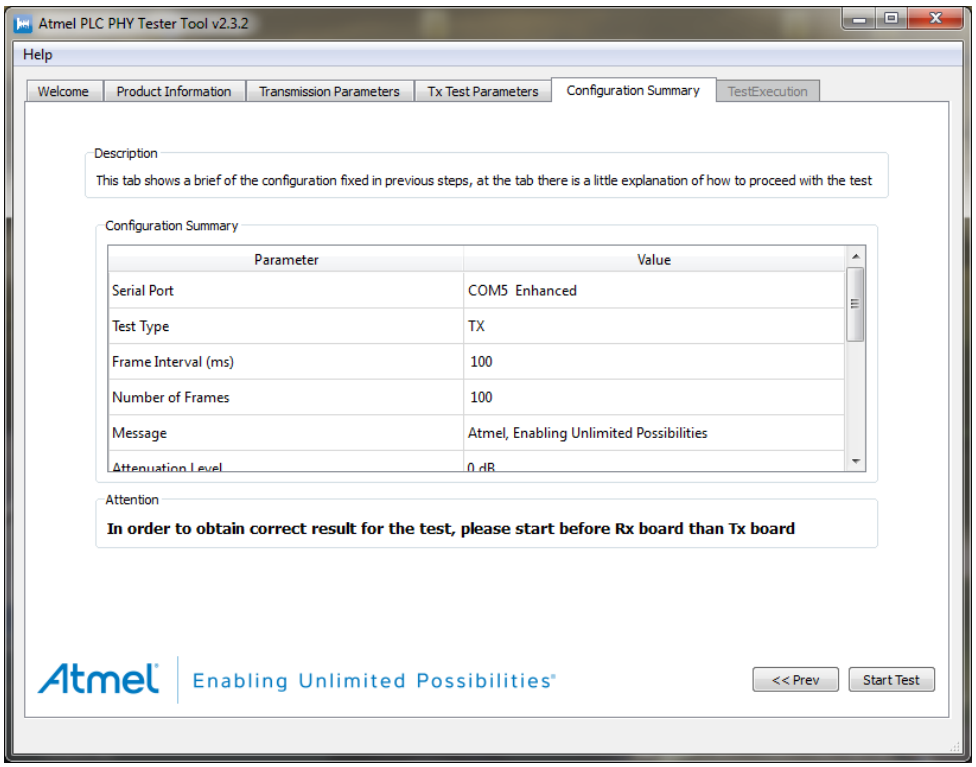
Previous figure shows the *TX Test Parameters* tab (Figure 6-38). This tab is where transmission test parameters are configured:

- Time Interval (milliseconds): desired interval between frame transmissions.
- Number of Frames: number of frames to be transmitted.
- Message: ASCII message to be transmitted.

These parameters **must match** the reception test parameters (Figure 6-33) for the test to be successful.

Default parameters are selected. Click the Next button to continue.

Figure 6-39. Configuration Summary tab.



The previous tab shows a table where all the configuration parameters and their selected values are listed. It is recommended to check that all values correspond to the desired configuration before continue.

To start the process, click the Start Test button. A new tab, *Test Executions* (reports of TX process), will appear with the frame sent and the TX result of the transmission process.

Now you can observe the transmission and reception process in both *Test Executions* windows. If messages are different, the receiver will not recognize them as a valid. If the configured interval and number of frames are different, the statistics computed at the end of the test may be inaccurate. In both board's displays the transmitted/received messages are showed.

During the transmission process the TX led of the coupling board is toggled. You can use it to check if the PLC messages are sent.

When all frames are sent, both *Test Executions* windows show some statistics. See the following figures.

Figure 6-40. Transmission test result.

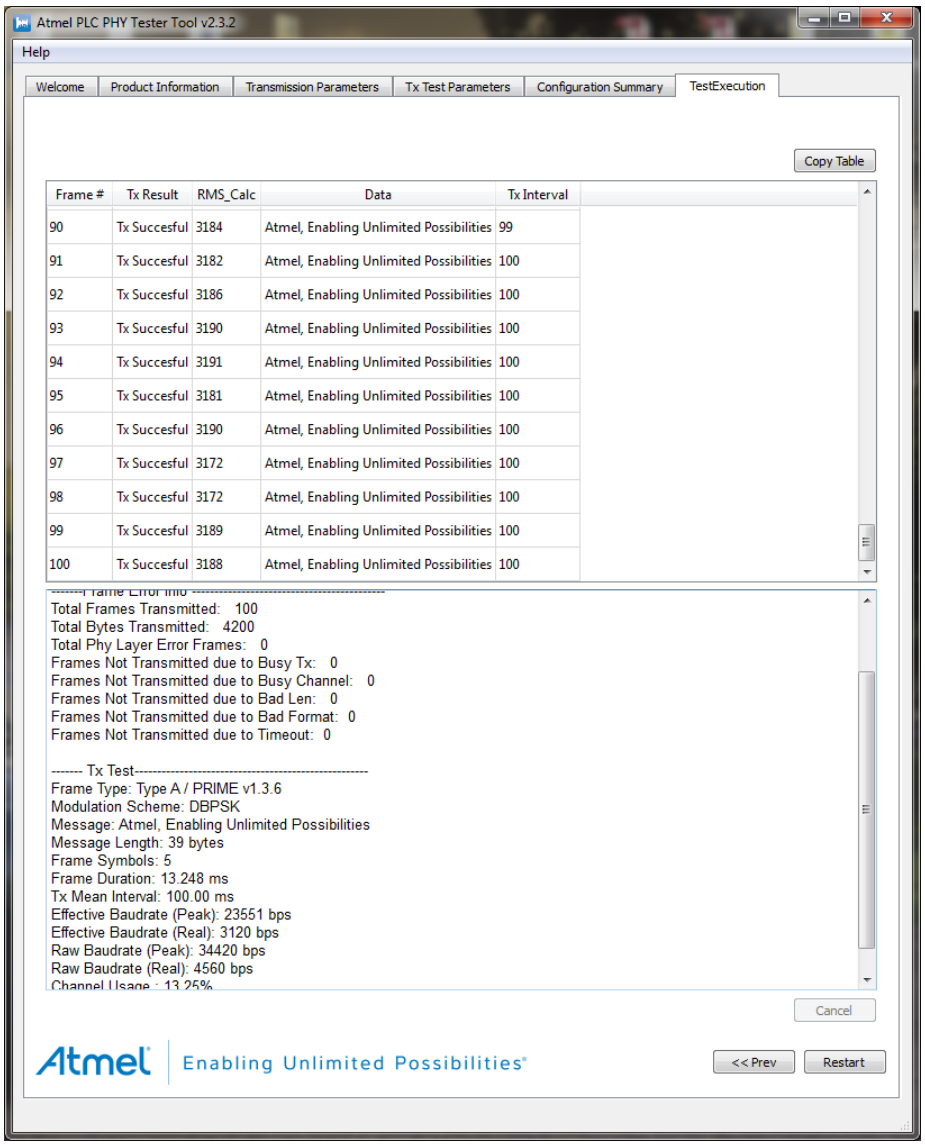
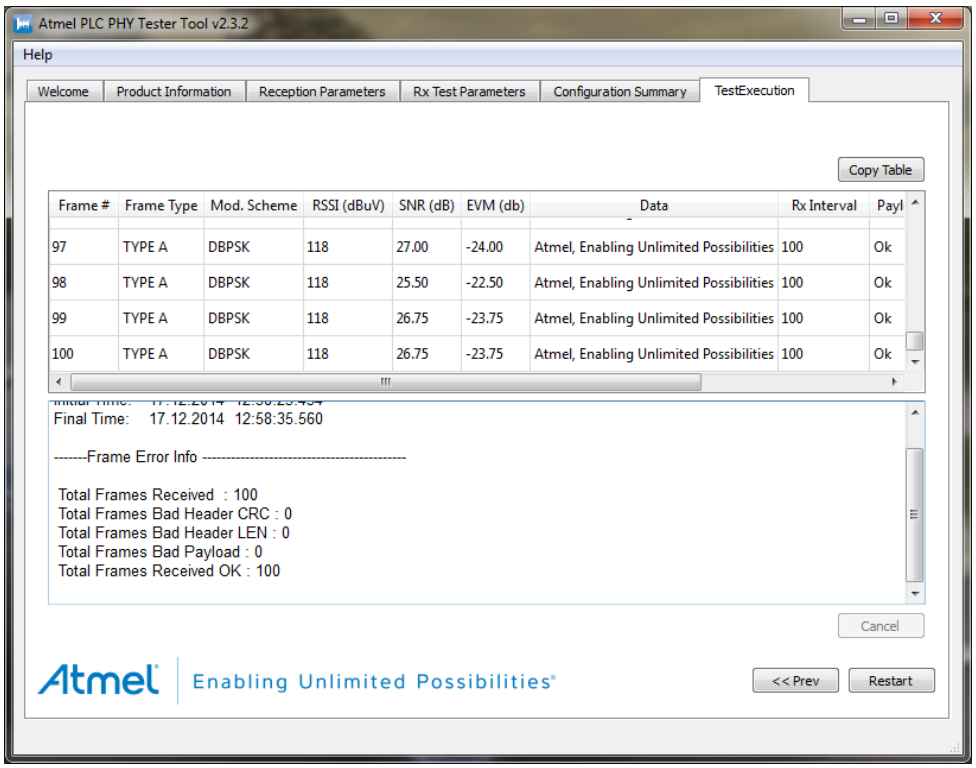


Figure 6-41. Reception test result.



While tests are executing, a row is added to the top table with information about the frame currently transmitted/received. The columns that contain these tables are the following.

Table 6-1. **Transmission/Reception parameters showed in columns.**

Transmission parameters showed		Reception parameters showed	
Parameter	**Description**	**Parameter**	**Description**
Frame #	It indicates the number of frame transmitted. It is useful to track the test progress.	Frame #	It indicates the number of frame received. It is useful to track the test progress.
Tx Result	It indicates the result of transmis-sion. If an error occurs, a descriptive text will appear.	Frame type	It indicates the frame type of the frame received.
Data	It shows the message transmitted in ASCII format.	Modulation scheme	It indicates the modulation scheme of the frame.
Tx Interval	It is the interval of time between the transmission of the current frame and the previous one.	RSSI	It indicates the strength of the signal received in dBuV.
		SNR	Signal Noise Ratio is a parameter calculated as is defined in the PRIME spec (dB).
		EVM	Error Vector Magnitude is a parameter calculated as is defined in the PRIME spec (dB).
		Data	It is the received info in ASCII format.

Transmission parameters showed		Reception parameters showed	
		Rx Interval	It is the interval of time between the reception of the current frame and the previous one.
		Payload Integrity	It shows if the content of the frame is correct or not.

After all frames have been transmitted/received, or the test has been cancelled, at the bottom of the tab it will appear a text box with information about the test. First of all, it will appear information about starting and ending time, this information is measured by the PC application.

After that, there is a section of information called *Frame Error* information that shows information about transmitted/received frames and possible errors. Finally another section shows a resume of the transmission/reception tests, this information contains much information as modulation scheme, message length, total frame received,… that is pretty straight forward but other fields must be explained. For that, please refer to the tool's embedded help (in the menu bar).

Once the values have been received, you can copy all values to check and analyze them by your own, clicking Copy Table button on the instances, the reception and transmission.

Click the Restart button to start the test again. It does first in the reception instance to avoid "lose" some frames.

The same TX/RX processes could be done using another ATPL coupling board. For that, after power down the ATPL230AMB, remove the ATPLCOUP001v1 board and set the new coupling board. Take into account that the new coupling board could require to set the jumper in J16 connector of the ATPL230MB board. Check the characteristics of the available ATPLCOUP boards. And even, you have to download over ATPL230AMB boards a new firmware build for the new coupling board in which the coupling board configuration has been changed (see Figure 6-25).

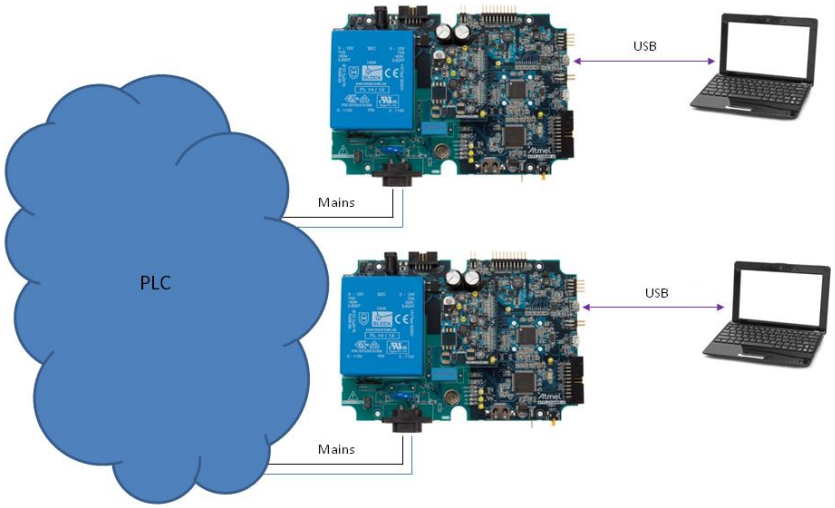
For further information about the tool, please refer to the tool's embedded help (in the menu bar).

6.3 PLC application example 2 – PHY TX Test Console

This example explains how to use the project application called *APPS_PHY_TX_TEST_CONSOLE*. This application lets the user to configure a proper setup to perform both EMC emissions and immunity tests for ATPL230AMB board. These tests are based on the use of PRIME PHY layer with a terminal console firmware (*apps_phy_tx_test_console.bin*) that eases the configuration of several transmission parameters such as modulation, frame data length and time interval between frames.

Following chapters explain to you how to supply the board, select the UART1 to communicate with the ATSAM4C16C, load the firmware and run the application. The setup is shown in the following figure.

Figure 6-42. Boards connection scheme.



6.3.1 Supplying the boards

Please refer to 6.2.2 in order to know how to supply the ATPL230AMB boards.

6.3.2 USB connection

Please refer to 6.2.3 in order to know how to connect the micro USB cable with the ATPL230AMB board. Remember to select the Standard COM Port, UART1. As is commented in section 3.5.6.4, UART 1 is available by USB connector J9.

UART1 CMOS signals are also available in a triple row male connector J5, see Figure 6-22.

6.3.3 Programming the embedded file

We have commented in section 6.2.4 the way to program a board.

Open the IDE tool used, Atmel Studio or IAR Embedded Workbench. Select the project *apps_phy_tx_test_console.atsln* or *apps_phy_tx_test_console.eww* and build it to generate the output file. Now you can download the file to the board.

Note that kits do not provide a J-Link ARM or SAM-ICE JTAG probe in order to connect to the user's host PC and the boards to download and debug the projects.

Remember that the J-Link USB drivers must have been downloaded previously from the Segger webpage (see section 6.1.4) and they depend on your operating system.

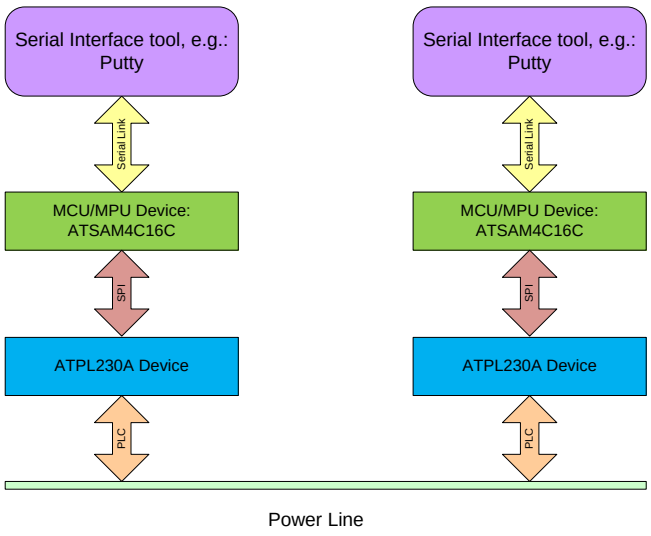
Remember that, every PHY TX Test console example project is contained in the following folder:
".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\phy\atpl230\apps\phy_tx_test_console\sam4c16c_atpl230amb".

And also in the PHY common workspace, ATPL230A_PLC_examples, which is contained in following Software folder:
".Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\apps\wrkspcs\".

6.3.4 Running the PLC application example 2

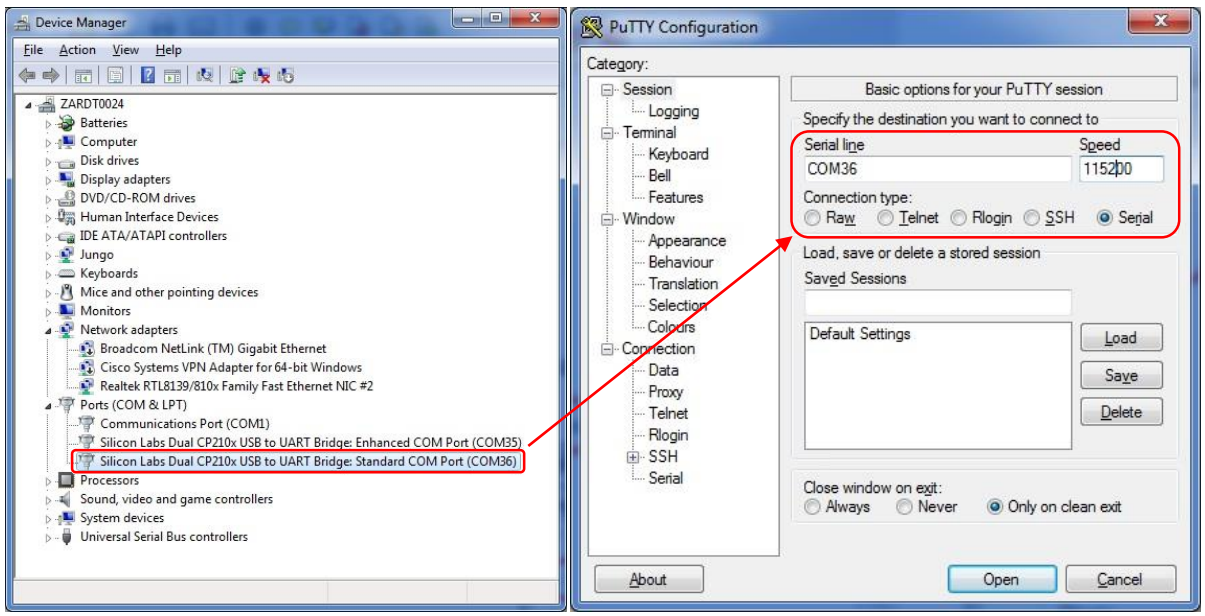
As the PLC application example 1, boards are plugged to the mains, see Figure 6-42. Users have to execute an instance of the serial interface tool – which has been previously installed to the host PC – in order to enable communication between both boards. Please note that these two instances may or may not run on the same computer.

Figure 6-43. PHY TX Test Console concept.



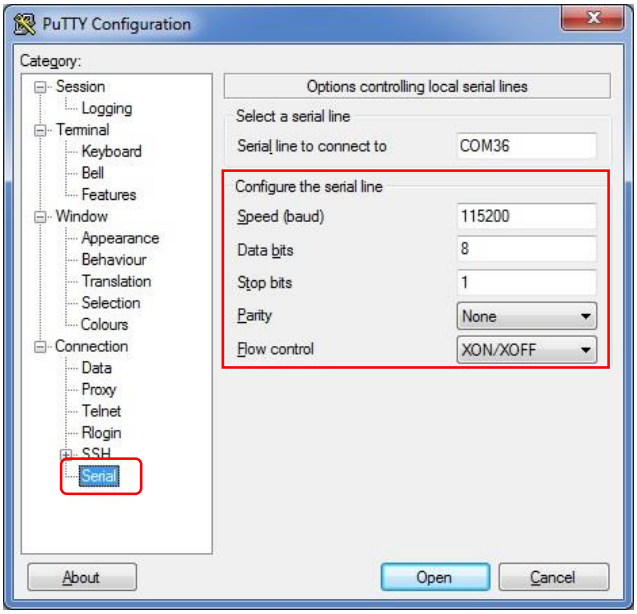
For this example a serial interface tool is required. HyperTerminal is not installed on Windows 7. You can use a PuTTY terminal instead. Once you have the serial terminal in your computer, open *putty.exe* and connect to the COM port number assigned to the micro-B USB cable (see Figure 6-22). As is commented in section 3.5.6.4, UART 1 is available by USB connector J9. UART1 CMOS signals are also available in a triple row male connector J5, see Figure 6-22. Remember to select the **Standard** COM Port, UART1.

Figure 6-44. COM Port selection.



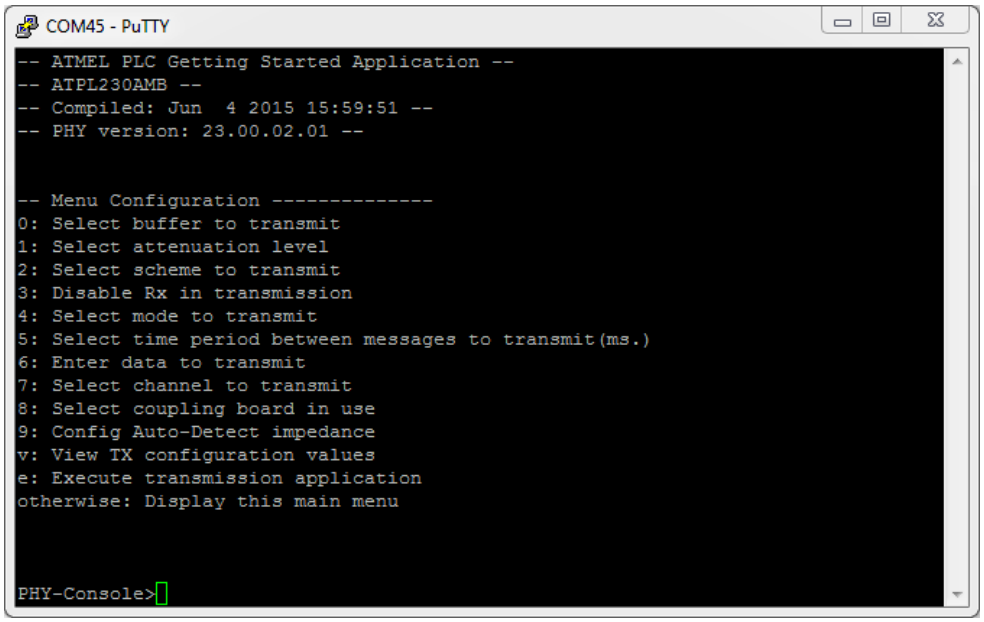
Set 115200 in the *Speed* field. In the *Serial* Category, change the *Flow Control* to None. The other fields should already be correctly configured. Finally, click Open.

Figure 6-45. PuTTY Configuration instance.



Once board is supplied, leds LED0 and LED1 blinks several times. After that, main menu is displayed (press Reset button in case board has been supplied previously to connect USB cable) in the Terminal window.

Figure 6-46. Main menu.



The description of each field is the following:

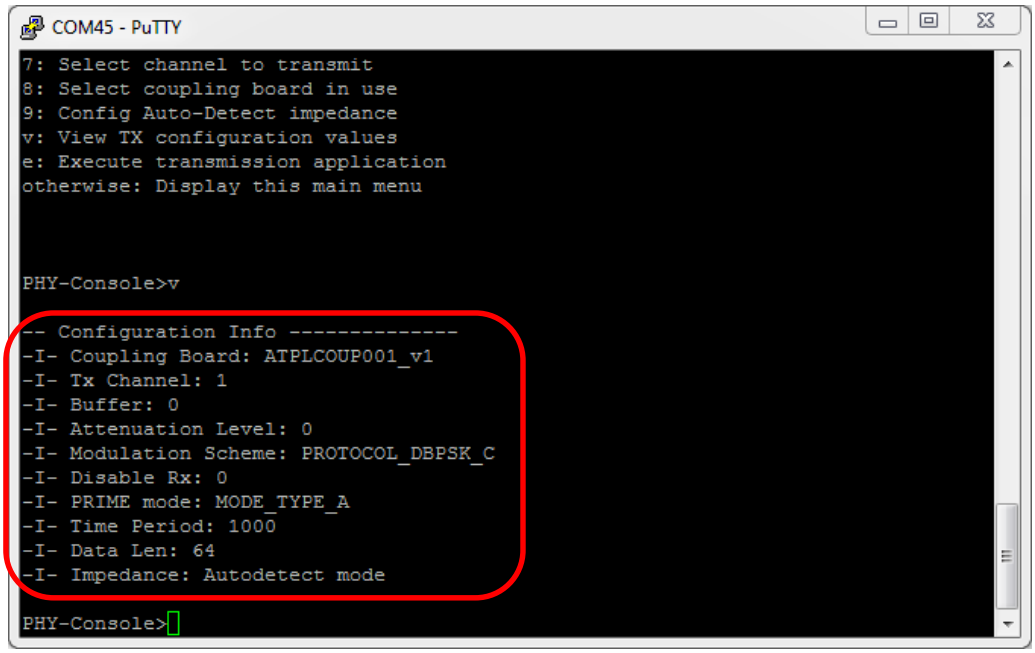
- **0: Select buffer to transmit.**
- **1: Select the attenuation level.** In this example is 0dB of attenuation and every step increments the attenuation in 3dB. In the current firmware the maximum attenuation value is 10 (30dB).
- **2: Select scheme to transmit.** In this example we choose 4 that is DBPSK+VTB**.**
- **3: Disable RX in transmission.** In this example is 0.
- **4: Select mode to transmit.** In this example is PRIME 1.3.6.
- **5: Select time period between messages to transmit (ms).** 1000ms in this example.

- **6: Enter data to transmit.** In this example data is fixed.
- **7: Select channel to transmit.** In this example is 1.
- **8: Select coupling board in use**. ATPLCOUP001v1 in this example.
- **9: Config Auto-Detect impedance.** Auto in this example.
- **v: View Tx configuration values**. Press *v* key of keyboard to check default configuration.
- **e: Execute transmission application**. Press *e* key in the keyboard to begin transmission and reception mode in both boards. And press *x* key of keyboard to stop the transmission process.

Default configuration is configured for ready for EMC tests:

- Coupling board: *ATPLCOUP001v1*. CENELEC-A band coupling board.
- TX channel: *0*. PRIME channel 1.
- Buffer: *0*. Buffer 0.
- Attenuation level: *0*. 0 dB.
- Modulation scheme: *PROTOCOL_DBPSK_VTB.* Differential BPSK with ViTerBi.
- Disable RX: *0*. RX enabled while board is emitting.
- PRIME mode: *MODE_TYPE_A*. PRIME 1.3.6.
- Time period between frames: *1000*. 1000ms.
- Data length: *64*. 64bytes.
- Impedance: *Fix high*. High branch emission fixed.

Figure 6-47. Default configuration menu.



In *phy_tx_test_console.C* file are all the possible values of the parameters from main menu fields.

So, for example, if you want to use another coupling board (Table 3-1), you have to change the coupling board default parameter. So the possible values are:

- 1: ATPLCOUP000_v2
- 2: ATPLCOUP001_v1
- 3: ATPLCOUP002_v1
- 4: ATPLCOUP002_v2

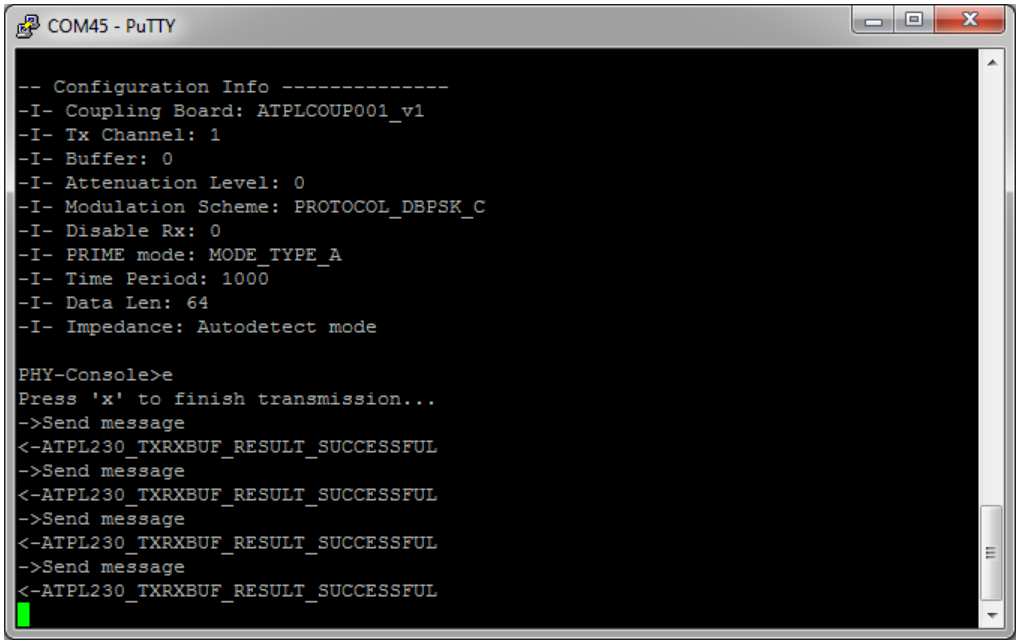
- 5: ATPLCOUP003_v1
- 6: ATPLCOUP004_v1
- 7: ATPLCOUP005_v1
- 8: ATPLCOUP006_v1

During the transmission, green led (LED0) is blinking indicating test is running. And the yellow led, PLC, on ATPLCOUP001 board blinks every time a PLC frame is sent. Terminal window shows the following messages (see Figure 6-48).

In the reception board, the red led (LED1) blinks in every PLC frame reception.

Respond to every action of main menu takes some time to the boards, is not executed immediately. Be patient.

Figure 6-48. Transmission messages.



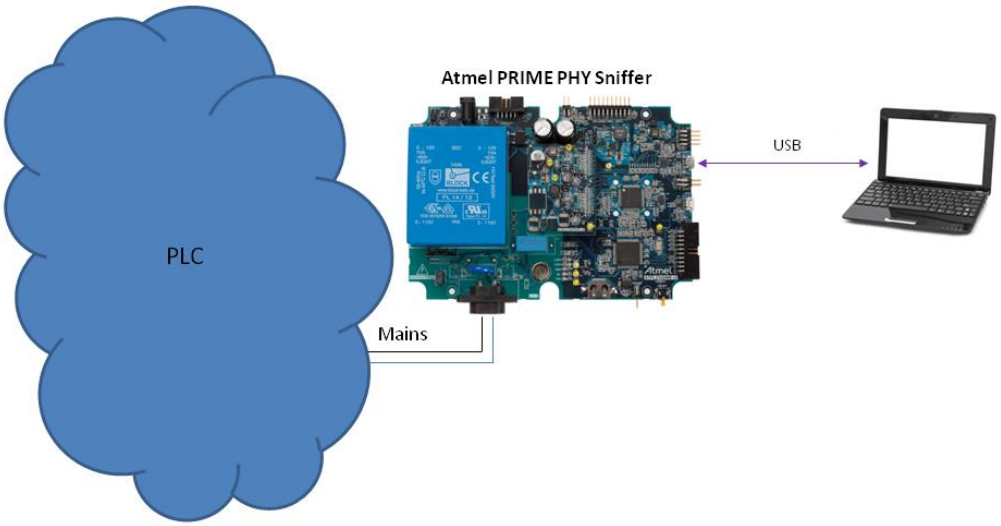
In case the configuration default has been changed, the board keeps the configuration unless power shutdown. If board is reset while keeping power supply on, it will restart the configuration mode after start-up.

6.4 PLC application example 3 – PHY Sniffer

In this example, we present you the PRIME PHY Sniffer project, *APPS_PHY_SNIFFER_TOOL*. PRIME PHY Sniffer project is able to monitor data traffic on the PRIME network by means of an ATPL230AMB board and the PC application, ATPL Multiprotocol Sniffer. For this example, only one ATPL230AMB board is required -and obviously a PRIME network to be monitorized-.

The circuitry in the coupling boards has an influence in the reception itself. As a consequence, each coupling board is intended to be used in their corresponding frequency channel(s) only. The application behaves properly when this correspondence is maintained.

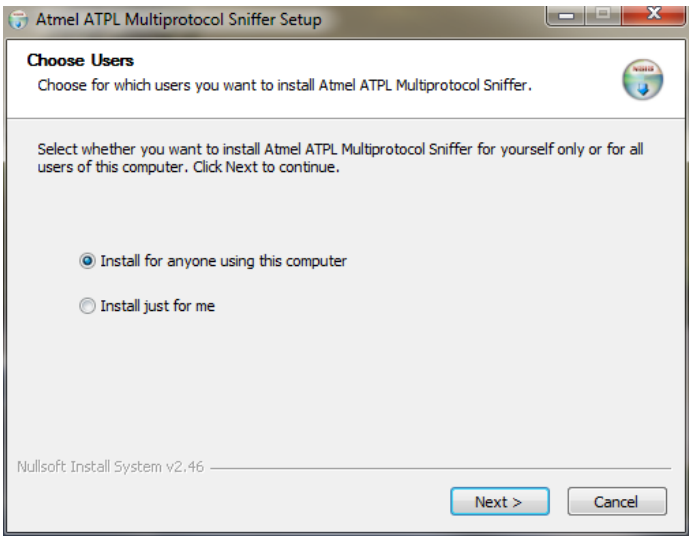
Figure 6-49. ATPL230AMB board connection scheme.



6.4.1 ATPL Multiprotocol Sniffer tool Installation

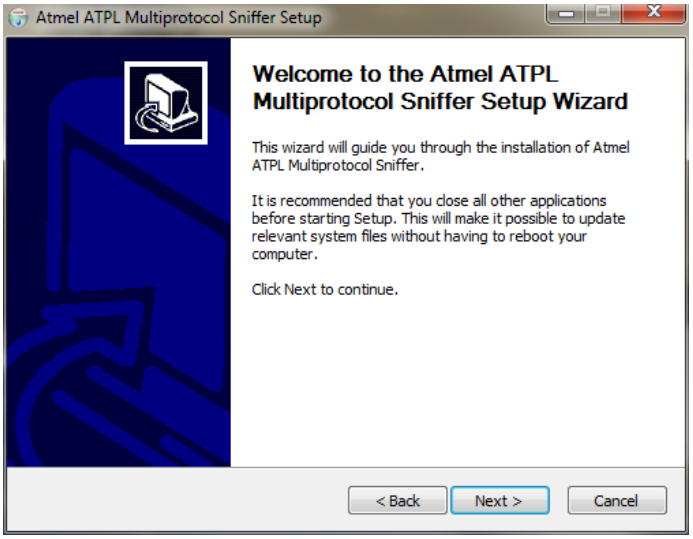
To install ATPL Multiprotocol Sniffer tool in a Windows Operating System, execute the provided installer in the PCTools folder *".\PCTools\ATPL_Multiprotocol_Sniffer\ATPL Multiprotocol Sniffer vX.Y.Z.exe"* and follow the installation wizard. The installer wizard should open. To follow the installation, click Next.

Figure 6-50. ATPL Multiprotocol Sniffer installation process, slide 1.



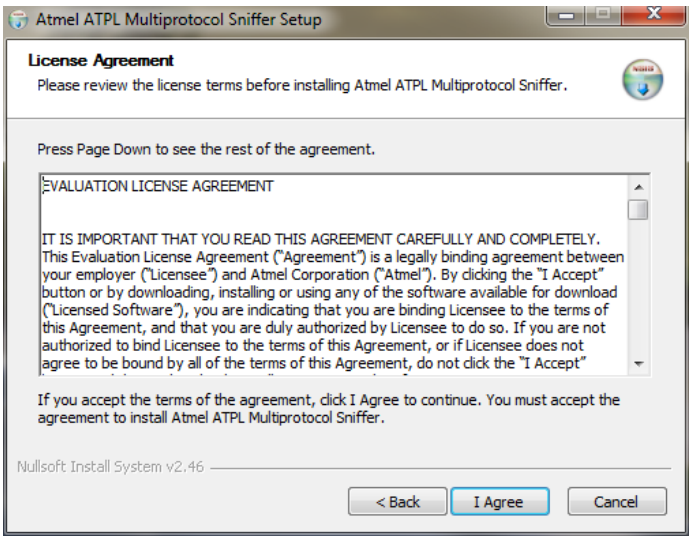
Select the users' permissions and click Next.

Figure 6-51. Installation process, slide 2.



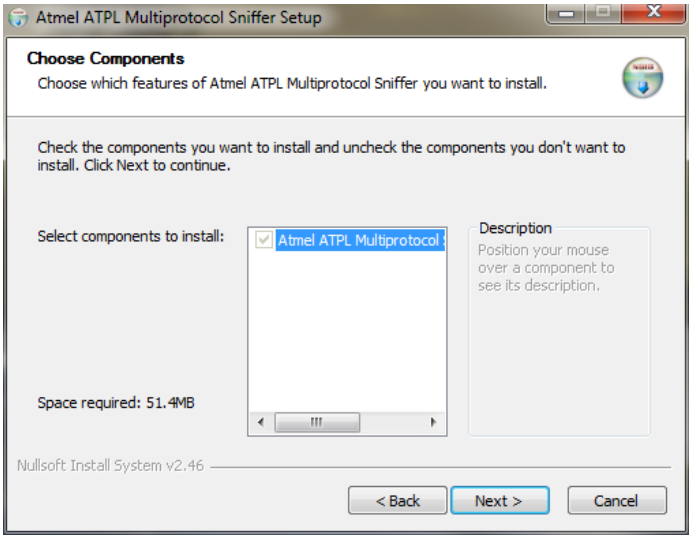
Click Next to continue.

Figure 6-52. Installation process, slide 3.



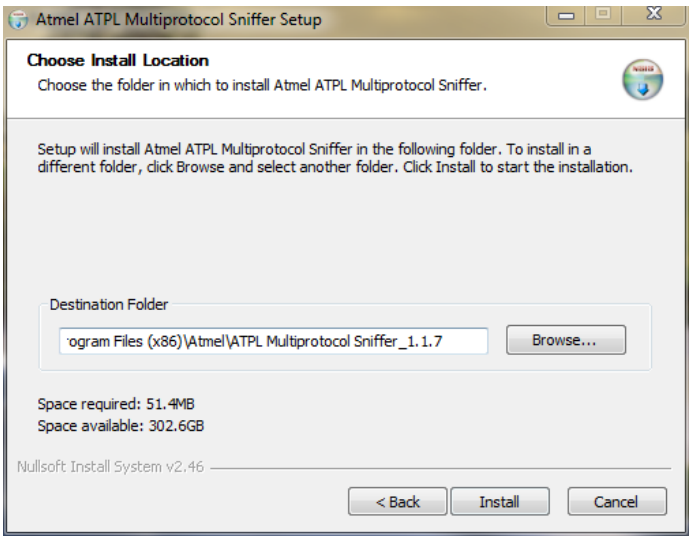
Read and accept term and conditions expressed in the End User License Agreement. Click I Agree to continue.

Figure 6-53. Installation process, slide 4.



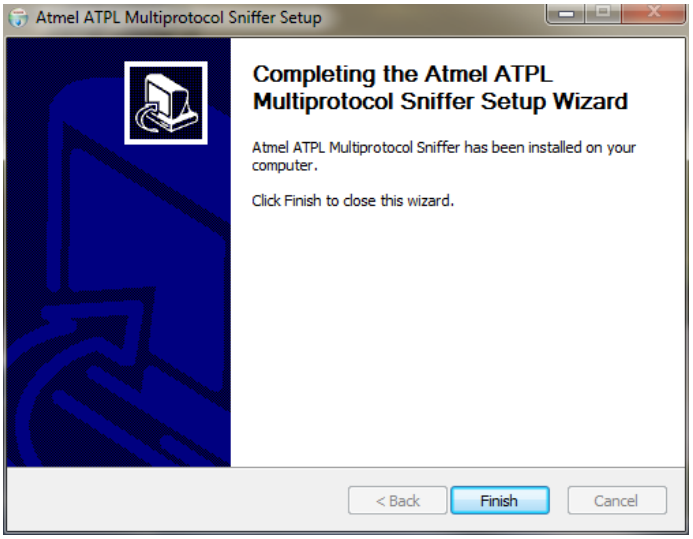
Click Next to install the component selected.

Figure 6-54. Installation process, slide 5.



Setup will install the program in the *Destination Folder*. To install in a different folder, click Browse and select your destination folder. Click Install to start the installation process.

Figure 6-55. Installation process, slide 6.



Click Finish.

Now the program is installed in your computer and a shortcut should have been created in your desktop.

6.4.2 Supplying the boards

Please refer to 6.2.2 in order to know how to supply the ATPL230AMB board.

6.4.3 USB connection

Please refer to 6.2.3 in order to know how to connect the micro USB cable with the ATPL230AMB board.

6.4.4 Programming the embedded files

We have commented in section 6.2.4 the way to program a board. To program the board as PLC sniffer, process will be the same: building the IDE project and downloading into the board.

Open the IDE tool used, Atmel Studio or IAR Embedded Workbench. Select the PHY sniffer tool project, *APPS_PHY_SNIFFER_TOOL.atsln* or *APPS_PHY_SNIFFER_TOOL.eww*, and now build it to generate the output file.

Note that kits do not provide a J-Link ARM or SAM-ICE JTAG probe in order to connect to the user's host PC and the boards to download and debug the projects.

> Remember that, every PHY example project is contained in the following folder: *".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\phy\atpl230\apps\phy_sniffer_tool\sam4c16c_atpl230amb".*
>
> And also in the PHY common workspace, ATPL230A_PLC_examples, which is contained in following Software folder: *".\Software\PRIME_vaa.bb.cc.dd\phy.atpl230amb\thirdparty\prime_ng\apps\wrkspcs\".*

Remember that the J-Link USB drivers must have been downloaded previously from the Segger webpage (see section 6.1.4) and they depend on your operating system.

As we commented in a previous section, every coupling board is intended to be used in their corresponding channel(s) only. By default, sniffer project is compiled for ATPLCOUP001 board. This means that only CENELEC-A frequency band, PRIME channel 1, is supported.

If you are going to use another coupling board, you must build the PHY sniffer project with the correct configuration. For that, open the IDE tool used and open the PHY sniffer project application, *APPS_PHY_SNIFFER_TOOL.atsln* or *APPS_PHY_SNIFFER_TOOL.eww*. After that, select the file, *conf_phy.h*, which it is in the PHY project configuration directory:

*"./Software/PRIME_vaa.bb.cc.dd/phy.atpl230amb\thirdparty\prime\phy\atpl230*apps\phy_sniffer_tool\sam4c16c_atpl230amb", find the define function to select the coupling board configuration (see Figure 6-56). Change the board name to desire board and build to generate the output file.

Figure 6-56. Coupling board configuration definition.

```
44 #ifndef CONF_PHY_H_INCLUDE
45 #define CONF_PHY_H_INCLUDE
46
47 /* Select Coupling Board Configuration (see values in atpl230.h) */
48 #define BOARD_COUPLING   ATPLCOUP001_v1
49
50 #endif  /* CONF_PHY_H_INCLUDE */
51
```

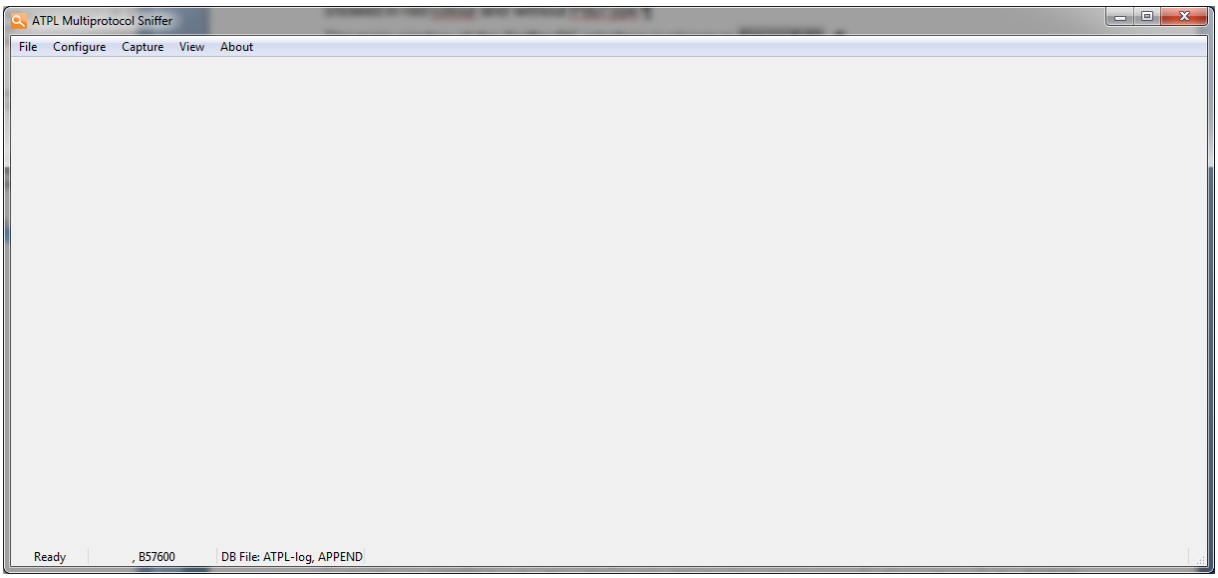
Check the Table 3-1 for the characteristics of the available ATPLCOUP boards.

6.4.5 Running the PLC application example 3

As you can see in Figure 6-49, the boards are plugged into the same power line. Users have to execute an instance of the ATPL Multiprotocol Sniffer tool – which has been previously installed in the host PC – in order to enable communication between the sniffer board and the PC. The ATPL Multiprotocol Sniffer tool is used to monitor data traffic on the network. You can also use the ATPL Multiprotocol Sniffer tool to monitor the PLC messages which they do not belong the PRIME standard then the messages will be showed in red color and without PduType.

The main window of the Sniffer PC interface is shown in Figure 6-57.

Figure 6-57. ATPL Multiprotocol Sniffer tool window.

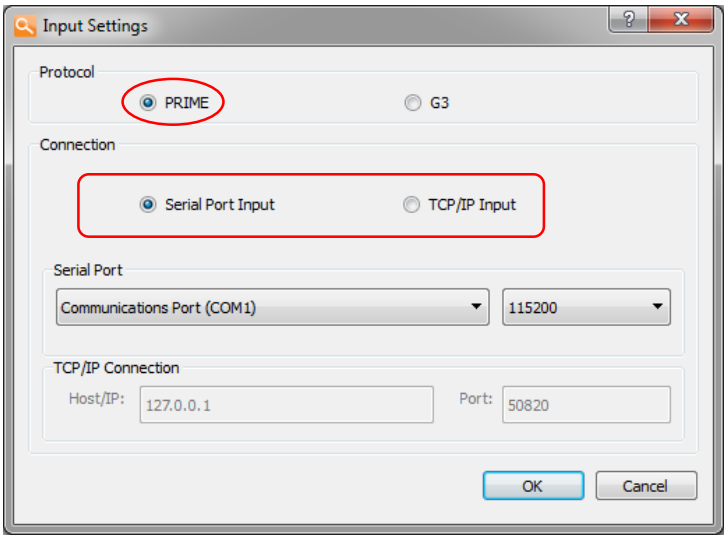


Once the application is launched, the COM port for the board needs to be configured. The COM port selection window is available by choosing Configure>Input (Ctrl+I). A new window *Input Settings* will appear as shown in Figure 6-58.

First of all, select the Power Line Communication protocol, in this case PRIME. After that select the COM port and set the speed. The default port is UART0 (enhanced COM port) and the speed for this

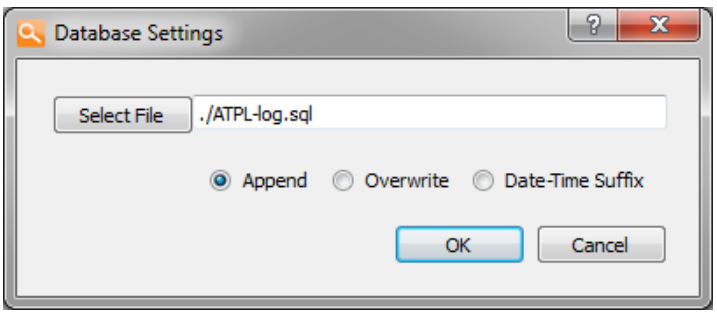
application is 256000 bauds. Also, this tool is able to connect to a remote device through the TCP/IP protocol.

Figure 6-58. Input Settings window.



The database file to store the traffic must be configured. If output logs are required and the location to store these choose Configure>Database… (Ctrl+D). A new window *Database Settings* will appear as shown in Figure 6-59, select the file name and click OK button. Database files can hold longer logs without having to split them in pieces. Also log stored files can be opened to review the file. The three options when you create a log database depends on if you want to keep the previous data or not. And it is possible to build your own scripts (for example, in Python) to analyze the data.

Figure 6-59. Database Settings window.

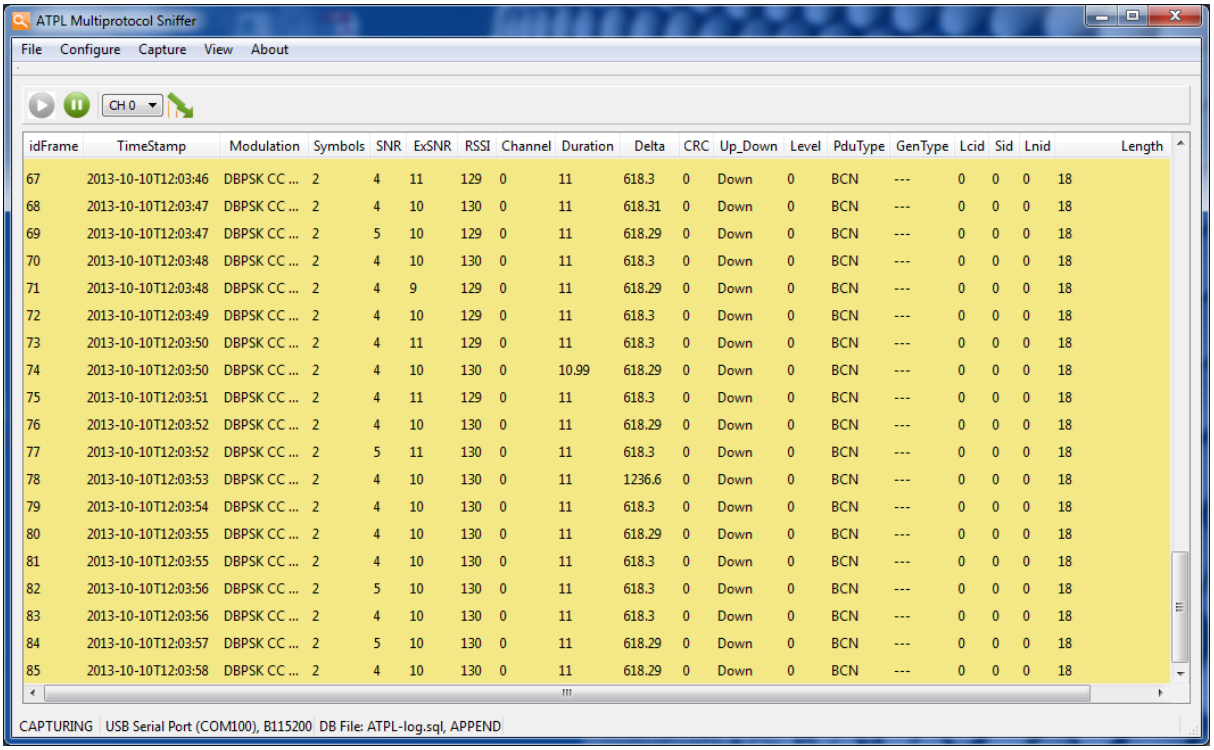


At this point, the tool is ready to start capturing data. If board is not powered, this is the point to supply it.

Click on the menu Capture>Start to begin logging data.

If tool establishes the communication with the COM port of the ATPL230AMB, the status bar at the bottom of the window will show the current setup and status of the tool. On a PRIME network the main window will look like as the Figure 6-60. Main window displays a table with the current log. It is updated in real time as frames are received from the hardware sniffer.

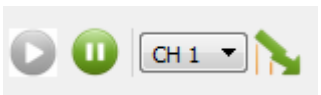
Figure 6-60. ATPL Multiprotocol Sniffer tool main window.



The capture window has a tool bar with four commands (see Figure 6-61):

- Pause command will stop the update of the scroll view, while the logging process will continue.
- To restart showing the live stream of PDUs, click Play button.
- Channel combo box allows selecting the PRIME channel to listen. Obviously the compatible PLC coupling board must be used.
- Thunder button will set the CRC configuration on the hardware device. If it is enabled, the hardware device will calculate the CRC on all the frames and discard frame errors. If it is enabled, all frames received will be sent to the PC software.

Figure 6-61. Tool bar.



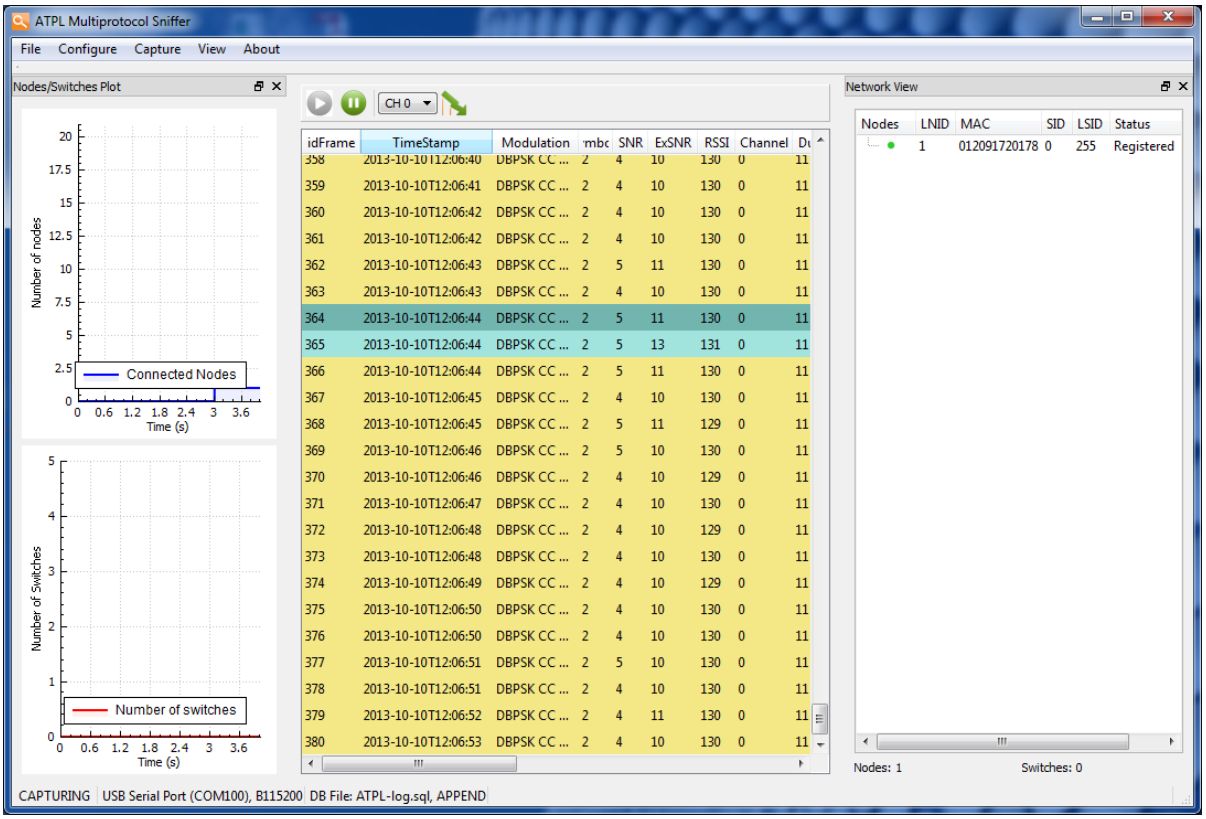
Main window displays a table with the current log. It is updated in real time as frames are received from the hardware sniffer. The data shown are: idFrame, Timestamp, Modulation, Symbols, SNR, ExSNR, RSSI, Channel, Duration, Delta, CRC, Up_Down, NAD, Level, PduType, GenType, Lcid, Sid, Lnid and Length.

While the PLC traffic is logged into a database, the software tries to infer the PLC network structure and status as seen by the Base node. This information is shown in several docking views. They are available on the menu *View*:

- Hexa view shows the hexadecimal display of the selected frame in the main view.
- Packet view shows the disassembled data of the selected frame in the main window. All the specified fields on the PRIME specification are shown.
- Network view shows the current status of the inferred network by the software. It is refreshed every time a change in the network is detected.

- Nodes/Switches plot shows the evolution of the network as seen by the sniffer. It plots the number of active nodes and switches on the network versus time. It is useful to detect problems of stability on the PLC network.
- Filter view allows selecting the frames shown in the main view table.

Figure 6-62. ATPL Multiprotocol Sniffer tool main window and several docking windows.



To uninstall the ATPL Multiprotocol Sniffer tool from your computer, go to *Start>All Programs>ATMEL>ATPL Multiprotocol Sniffer vX.Y.Z>Uninstall*.

For further information, please refer to the tool's embedded help (in the menu bar).

6.5 Introduction to PRIME Stack

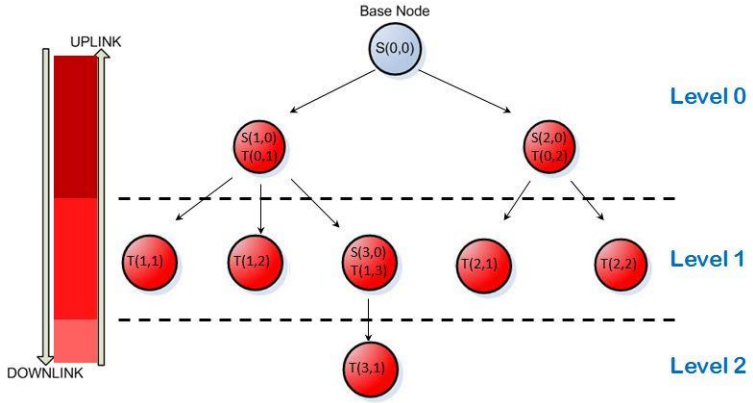
PLC is a medium with such special characteristics (asymmetry, noise variation in time, etc.) that makes it a hostile environment for successful communication when users are not familiar with these issues.

The PRIME (PoweR line Intelligent Metering Evolution) initiative is a solution for an entire Smart Grid environment which will contribute definitively to energy efficiency improvement and ultimately to addressing the pressing issue of climate change.

PRIME defines lower layers of a PLC narrowband data transmission system over the electric grid. All the system has been created to be low-cost and high-performance.

PRIME system is composed of sub-networks, each of them defined in the context of a transformer station. A sub-network is a tree with two types of nodes: the Base Node and the Service Node.

Figure 6-63. A typical PRIME network.

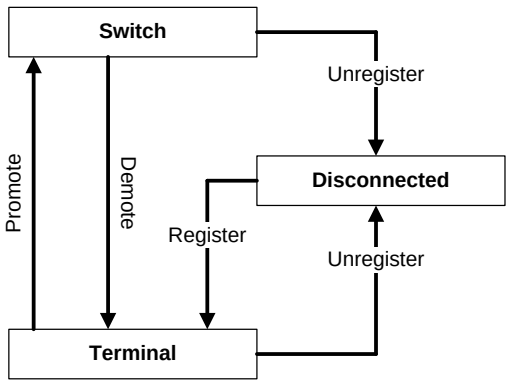


The Base Node is the root of the tree and acts as master node that provides connectivity to the sub-network. It manages the sub-network resources and connections. There is only one Base Node in a sub-network. This Base Node is initially the sub-network itself and other nodes should follow a process of registering in order to join this sub-network.

Any other node in the sub-network is a Service Node. Service Nodes are either leaves of the tree or branch points of the tree. These nodes start in a disconnected state and follow certain procedures to establish network connectivity. Each of these nodes is one point in the mesh of the sub-network. These nodes have two responsibilities: connecting themselves to the sub-network and switching the data of their neighbors in order to propagate connectivity.

Service Nodes change their behavior dynamically from “Terminal” functions to “Switch” functions and vice-versa. Changing of functional states occurs based on certain predefined events in the network.

Figure 6-64. Functional states of a Service Node.



As shown in the previous figure, the three functional states of a Service Node are:

- Disconnected: a Service Node starts in a disconnected state. In this state a node is not capable of communicating or switching the traffic of another node. The primary function of a Service Node in this state is to search for an operational network in its proximity and to try to register itself to it.
- Terminal: in this state a Service Node is capable of communicating by establishing connections. But it is not capable of switching the traffic of any other node.
- Switch: in this state a Service Node is capable of performing all Terminal functions. Additionally, it is capable of forwarding data to and from other devices in the sub-network. It is a branch point in the tree.

The current PRIME standard specifies a transmission system based on OFDM modulations scheme.

The OFDM PRIME PHY specification uses the frequency band from 41.992 kHz to 88.867 kHz (47 kHz bandwidth). This is achieved by using OFDM modulation with signal loaded on 97 (96 data and one pilot) equally spaced subcarriers. Differential modulation is used, with one of three possible constellations: DBPSK, DQPSK or D8PSK.

The ATPL230A architecture provides enhanced performances over the PRIME specification with the new robust modes and the ARIB/FCC frequency band extension. The PRIME 1.4 has two additional robust modes:

- Robust DQPSK
- Robust DBPSK

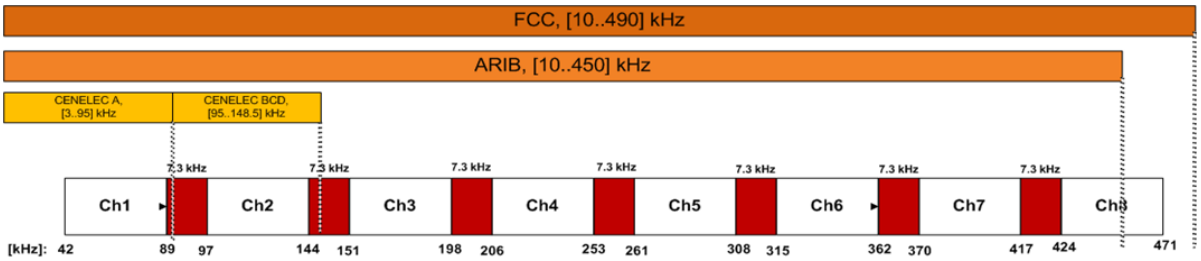
Thanks to this new performance, it is possible to achieve a highest robust mode gain compared to PRIME up to 14,5 dB more.

Table 6-2. **PRIME modulations.**

PRIME	Band Extension	Modulation
PRIME 1.3	Cenelec-A	DBPSK, DQPSK, D8PSK
PRIME 1.4	Cenelec-A FCC ARIB	DBPSK, DQPSK, D8PSK, Robust DBPSK, Robust DQPSK.

The current PRIME standard is adapted to European regulations. The evolution of PRIME has, as one of its key features, a frequency band extension that allows choosing up to 8 different channels. This performance makes PRIME becoming into a more flexible platform.

Figure 6-65. PRIME Frequency Band Extension, PRIME 1.4.



This technology only allows one channel active at a time. The limits of each channel are shown in the next table and can be compared with the figure above:

Table 6-3. **Frequency Band Limits of each Channel.**

Channel	Start freq. (kHz)	End freq. (kHz)	CENELEC	ARIB	FCC
1	41,992	88,867	X	X	X
2	96,68	143,555	X	X	X
3	151,367	198,242		X	X
4	206,055	252,93		X	X
5	260,742	307,617		X	X
6	315,43	362,305		X	X
7	370,117	416,992		X	X

8	424,805	471,68			X

In the following sections there are basic overviews of the libraries used and the description of the whole system integration (FreeRTOS, PRIME, PLC, ATPL230A and the SAM4C) in one project using the ASF structure.

6.5.1 FreeRTOS

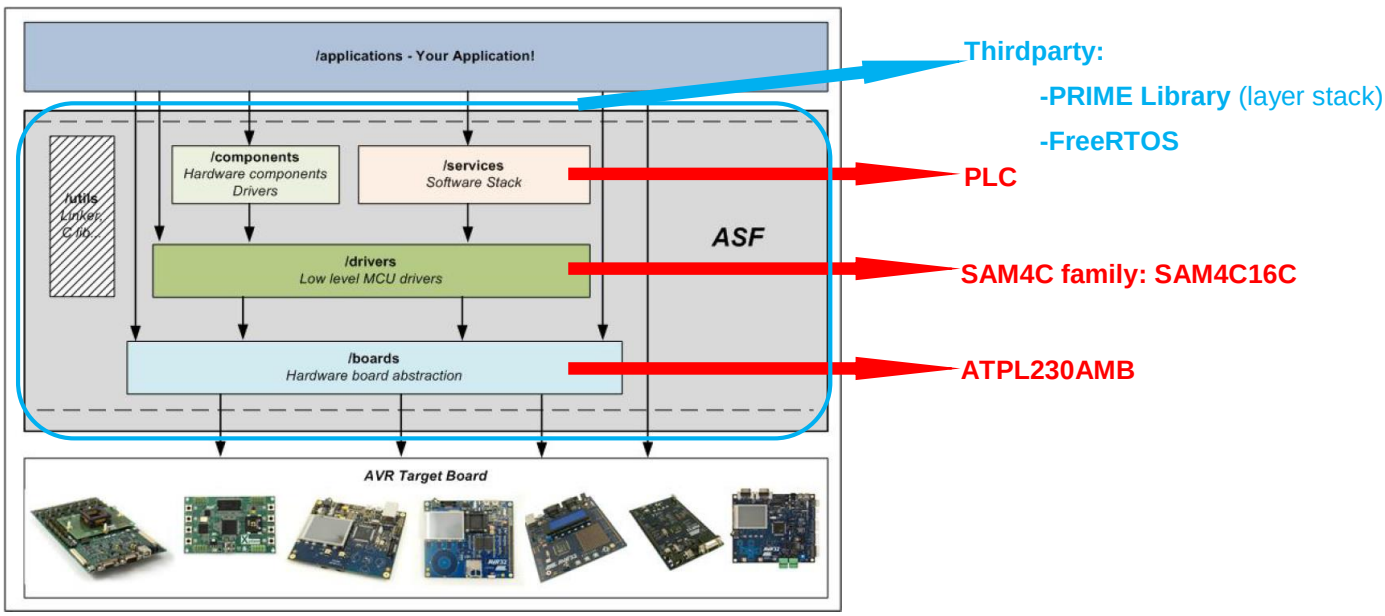
FreeRTOS is a real-time kernel (or real-time scheduler) on top of which Cortex-M3/M4 microcontroller applications can be built to meet their hard real-time requirements. It allows Cortex-M3/M4 microcontroller applications to be organized as a collection of independent tasks to be executed. The kernel decides which task should be executed by examining the priority assigned to each by the application designer. In the simplest case, the application designer could assign higher priorities to tasks that implement hard real-time requirements, and lower priorities to tasks that implement soft real-time requirements. This would ensure that hard real-time tasks are always executed ahead of soft real-time one.

Thanks to the FreeRTOS scheduler we are able to optimize PRIME code and memory usage. Although the SAM4C16C has two cores, we will run the PRIME project only in the core 0.

6.5.2 ASF Integration

As it was explained before, ASF has a defined structure. ASF root folder contains the *common/* directory, the *sam/* directory and the *thirdparty/* directory. The components contents of thirdparty directory are showed in the following figure. That is the way to integrate the whole platform in this structure (PRIME, PLC, SAM4C and FreeRTOS).

Figure 6-66. SAM4C & PRIME Integration in thirdparty folder.



We integrate the different parts according to the ASF structure:

- Boards: The ATPL230AMB board hardware mapping is defined here.
- Drivers: The drivers for the SAM4C Family.
- Services. We offer the PLC modem as a service.
- ThirdParty: We add in this point the PRIME and FreeRTOS libraries.

It happens that last version of the Atmel Software Framework provided version in the web link at this moment - release ASF 3.27 (September 2015) - does not coincide with the PLC libraries of the projects from the kit's Software folder. PLC libraries of the kit are an above version that ASF.

In the release notes document about the PRIME version appears the individual version layers of PHY, MAC and SSCS432

Take into account, previous to download futures releases of ASF, if it is supported by these kit's version boards.

In case you do not know the ASF version downloaded in Atmel Studio, go to Help>Atmel Studio. Select in the combo box of the new window the component: *Atmel Software Framework*. After that, all the versions installed are showed.

6.5.3 Atmel PRIME Stack Structure

The PRIME specification currently describes the following architecture from bottom to top:

- **PHY layer** capable of achieving rates of uncoded 128 kbps. It transmits and receives MPDUs (MAC Protocol Data Units) between Neighbor Nodes using orthogonal frequency division multiplexing (OFDM).
- **MAC layer** for the power line environment. It provides core MAC functionalities of system access, bandwidth allocation, connection establishment/maintenance and topology resolution.
- **Convergence layer** for adapting several specific services. It classifies traffic associating it with its proper MAC connection; this layer performs the mapping of any kind of traffic to be properly included in MSDUs (MAC Service Data Units). It may also include header compression functions. Several SSCSs (Service Specific Convergence Sublayer) are defined to accommodate different kinds of traffic into MSDUs.
- **Management Plane** enables a local or remote control entity to perform actions on a Node. In the Management Plane, there are two modules – called MLME (MAC Layer Management Entity) and PLME (PHY layer Management Entity) – that allow the external configuration of the MAC and PHY layers respectively. This module also includes the firmware upgrade capability.

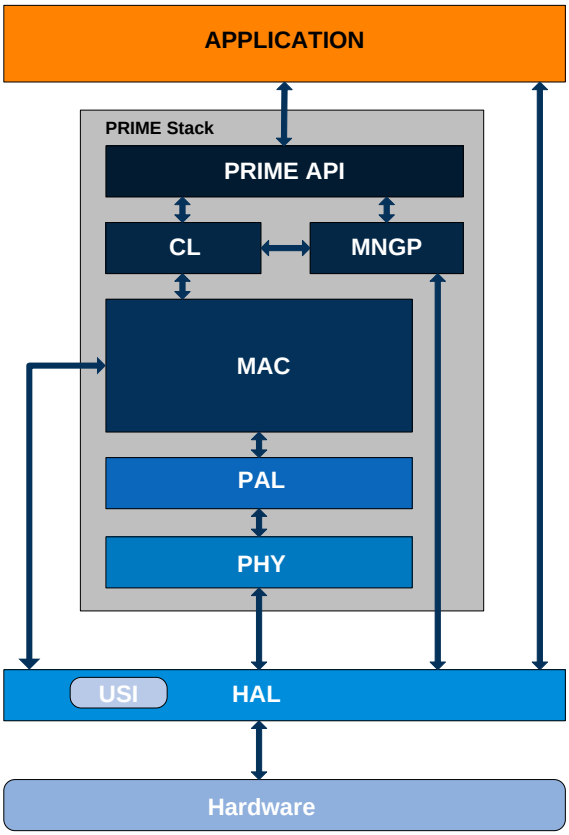
The Atmel PRIME Firmware stack is able to run on a system with an Operative System, or without it, running in microcontroller mode. The OSS intends to transform the microcontroller-mode operation into a task-mode operation typical of operating systems. In order to do that, it creates and manages a single task where all active layers and interfaces are included. The user does not need to take care of controlling how the PRIME stack is running and can create their applications normally. The current implementation of the OSS is based on FreeRTOS but the user could modify it appropriately to use any other RTOS.

The Service Node project provided in the kit only provides the Atmel PRIME layers. The Physical layer provided is in source code. The Figure 6-67 shows the Atmel PRIME Stack structure and the Figure 6-68 shows the PRIME FW Stack project structure for the SAM4C16C according to the ASF structure.

The Atmel PRIME FW stack modules are from the bottom up:

- Physical Layer (PHY).
- Physical Abstraction Layer (PAL).
- Medium Access Control (MAC) layer.
- Convergence Layer (CL).
- Management Plane (MNGP)

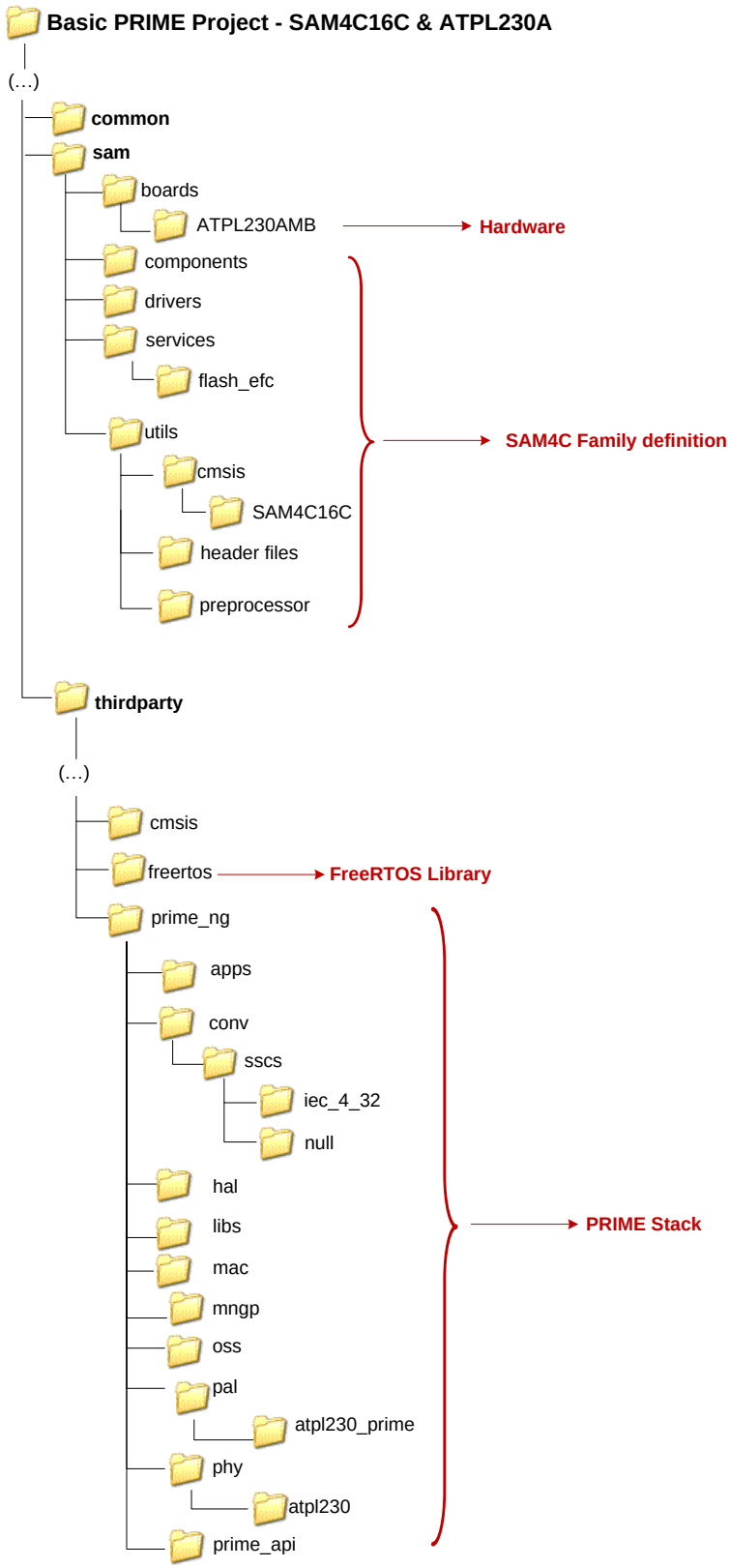
Figure 6-67. Atmel PRIME Firmware structure.



As it can be seen in the previous figure, the only entry point to the PRIME Firmware stack is through the PRIME API, which contains the interfaces defined in the PRIME specification as well as stack control functions. Furthermore, it must be noticed that the Hardware Abstraction Layer (HAL), including the Atmel Universal Serial Interface (USI), is not part of the PRIME FW stack.

For more information about the Atmel PRIME software Stack see the doc43085 and the PRIME specification.

Figure 6-68. Atmel PRIME Service Node Project.



Note: This figure is only to give a general idea about the distribution of the folders and the libraries in a basic Atmel PRIME Service node project.

Where *prime_ng* directory contains the following folders:

- *apps*: it contains the workspace with all the example applications provided by Atmel, including the configuration files and the project to generate the PRIME FW stack binary file. New user applications must be developed in this directory, too.
- *libs*: it contains the library file that implements the complete PRIME FW stack.
- *hal*: header and code files for the HAL.
- *phy*: header and code files of the PHY layer for the PLC modem.
- *pal*: header and code files of the PAL to interface PRIME with the PHY layer.
- *prime_api*: header and code files to manage the PRIME API.
- *mac*: header files with functions, definitions and PIB attributes for the MAC layer.
- *mngp*: header file with functions for the Management Plane.
- *conv*: header files with functions and definitions for the Convergence Layer, including the Null SSCS and the IEC 4-32 SSCS.
- *oss*: header and code files for OSS (if using an OS).

Users must not modify any of the provided files except from the configuration files, the HAL and the OSS. Users are free to use the example applications as templates to create new user applications. Users can also develop their own applications in the apps directory.

The first important thing to notice is that the user application and the PRIME FW stack are integrated as separated software applications. Therefore, Atmel provides two independent projects which generate two different files, one binary file for the PRIME FW stack and another one for the user application. They are:

- PRIME FW stack project, *prime_service_bin.zip*.
- User application projects:
  - A PRIME user application project (DLMS application). It is an application example that shows how the PRIME API should be used. This application configures the ATPL230AMB board as a Service Node with DLMS capabilities and simulates the data exchange between the Base Node and the Service Node. The Service Node responds dummy DLMS messages after receiving data requests from the Base Node. For this example, a PRIME Concentrator is required. Depending on the operation mode (as a Real Operating System or as Microcontroller) there are two projects: *prime_service_dlmsemu_fi.atpl230amb.zip* file for OSS based on FreeRTOS and *prime_service_dlmsemu_ui.atpl230amb.zip* file.
  - A PRIME user application project (modem application). This application configures the ATPL230AMB board as a Service Node. It is an application example that shows how to serialize the PRIME API when the user application is running in an external device. See *prime_service_modem.atpl230amb.zip* file.

With this architecture, both parts can be updated separately even when they are running in the same board. Furthermore, this means that the memory must be correctly managed in order to be able to allocate all binary files.

6.5.3.1 PRIME FW stack

The PRIME FW stack project, *APPS_PRIME_SERVICE_BIN*, contains the PRIME library together with the configuration files. This project is required in order to generate the PRIME binary file that is later loaded into the board.

Remember that, PRIME FW Stack project is contained in the following folders depending on the IDE tools used:
".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_servic

e_bin\sam4cp16b_sam4cp16bmb_gcc\as5_arm".
".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_service_bin\sam4cp16b_sam4cp16bmb_iar\iar".

This project can be reused whenever a new PRIME library is available – it would only be necessary to exchange the library file in the project and update the PRIME firmware version in the corresponding configuration file.

In this project users must only modify the available configuration files according to their application needs:

- Management Plane configuration: *conf_mngp.h*.
- PAL configuration: *conf_pal.h*.
- PHY configuration: *conf_phy.h*.
- PRIME Stack configuration: *conf_prime_stack.h*.

6.5.3.2 User application

This firmware stack has been intended to hold the application code developed by the user. So the user can integrate his application code in the firmware package delivered by Atmel.

The user application project is independent from the PRIME FW stack project. It contains only header and configuration files related to the PRIME FW stack so that users can develop their applications and later load them into the board at the allocated address without disturbing the PRIME FW stack.

The HAL is also part of the user application project and users can allocate it at any address within their region. The pointer to the HAL functions will be passed to the PRIME FW stack at initialization. Users are also responsible for initializing, starting and running the HAL.

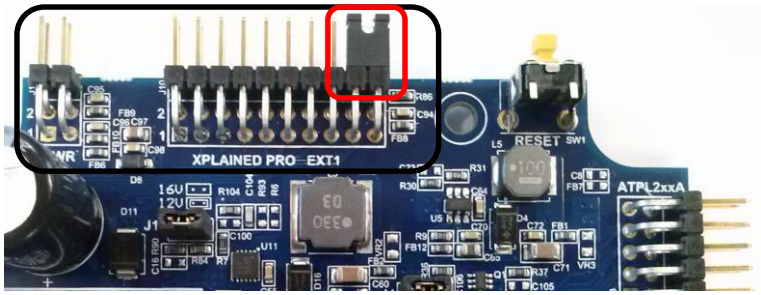
There are two PRIME user application projects:

- DLMS application: This application configures the ATPL230AMB board as a Service node with DLMS capabilities and simulates the data exchange between the PRIME Base Node and the Service Node. The Service Node responds dummy DLMS messages after receiving data requests from the Base Node. It is an application example that shows how the PRIME API should be used. Depending on the operation mode (as a Real Operating System or as Microcontroller) there are two projects:
  - *APPS_PRIME_SERVICE_DLMSEMU_UI* running as microcontroller mode.
  - *APPS_PRIME_SERVICE_DLMSEMU_FI* running as OSS mode.

Remember that, DLMS application project running as microcontroller mode is contained in the following folders depending on the IDE tools used:
".\Software\PRIME_vaa.bb.cc.dd\prime_service_dlmsemu_ui.atpl230amb\thirdparty\prime_ng\apps\prime_service_dlmsemu_ui\sam4c16c_atpl230amb\as5_arm".
".\Software\PRIME_vaa.bb.cc.dd\prime_service_dlmsemu_ui.atpl230amb\thirdparty\prime_ng\apps\prime_service_dlmsemu_ui\sam4c16c_atpl230amb\as5_arm\iar".

DLMS application project also implements the AppEmu (Application Emulation) –application for PRIME certification-. This application is required for certification purposes, see Test Cases of PRIME Certification document. To enable the AppEmu, PC3 pin should be at 0 volts. So, previous to supply the board, you must set a jumper between the PC3 pin and ground, GND. For that, set a jumper in J10 connector, XPLAINED PRO, as is showed in the following figure.

Figure 6-69. Jumper to enable AppEmu.



Once Service node opens a connection and Base node accepts it, the exchange data test starts up automatically and the Service node shows the statistics of the test by UART1. If you want to see the Service node statistics in a serial interface tool (i.e., PuTTy terminal), you have to enable the `define APP_EMU_DEBUG_ENABLE` in *app_emu.h* file (see Figure 6-70).

Figure 6-70. Enabling statics in UART1.

```
58  /**INDENT-ON**/
59  /** @endcond */
60  /* debug mode printf app */
61  /* #define APP_EMU_DEBUG_ENABLE */
62
63  /** APP Emu Task Rate */
64  #define APP_EMU_TASK_RATE          (100 / portTICK_RATE_MS)
65
66  /** APP Emu Task Stack priority */
67  #define TASK_APP_EMU_PRIO          (tskIDLE_PRIORITY + 2)
68
69  /** APP Emu Task Stack definition */
70  #define TASK_APP_EMU_STACK         (configMINIMAL_STACK_SIZE * 5)
```

Connect the USB cable to the micro-B USB connector, J9, and now configure the terminal. Remember to select the COM port number assigned to the Standard COM Port, so that, UART1. After that, set 115200 in the *Speed* field and in the *Serial Category* change the Flow Control to *None*. The other fields should already be correctly configured. Finally, click *Open*.

This application is only to configure the ATPL230AMB board as Service node, Atmel does not provide the Base node application.

- Modem application: This application configures the ATPL230AMB board as a Service node. It is an application example that shows how to serialize the PRIME API when the user application is running in an external device. Open *prime_service_modem.atpl230amb.zip* file and select the *APPS_PRIME_SERVICE_MODEM* project.

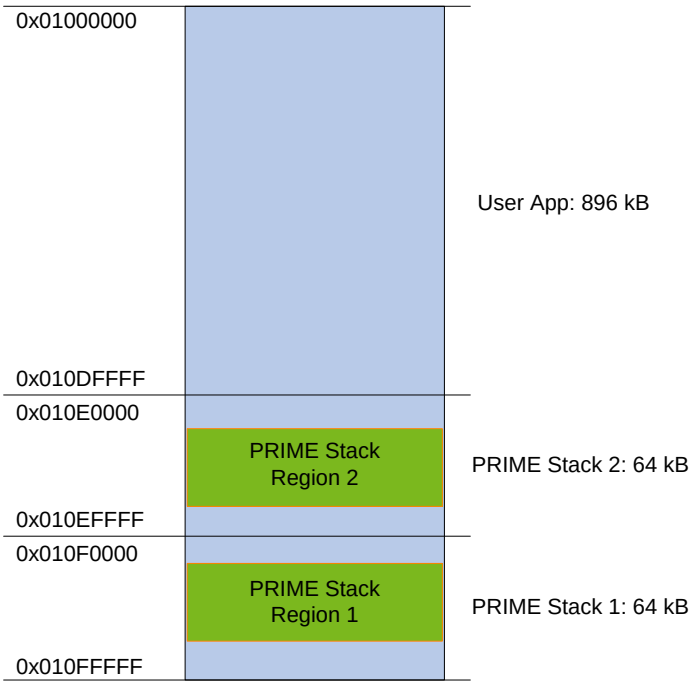
> Remember that, Modem application project is contained in the following folders depending on the IDE tools used:
> *".\Software\PRIME_vaa.bb.cc.dd\prime_service_modem.atpl230amb\thirdparty\prime_ng\apps\prime_service_modem\sam4c16c_atpl230amb\as5_arm".*
> *".\Software\PRIME_vaa.bb.cc.dd\prime_service_modem.atpl230amb\thirdparty\prime_ng\apps\prime_service_modem\sam4c16c_atpl230amb\iar".*

Basic procedure for performing this integration (firmware configuration) is commented widely in the Atmel PRIME Firmware Stack, doc43085. Basic requirements are:

- Memory allocation. The allocation address of the PRIME FW stack is allocated at the highest address and uses the shown addresses in Figure 6-71. This memory distribution has been chosen to ease the firmware upgrade process. Addresses can be changed to fit the user's needs as long as they respect the indicated sizes. There are two regions reserved for the PRIME FW stack: one

to run the stack and the other one to store an old/new stack version. The user application is responsible for managing the pointer to the correct region.

Figure 6-71. FLASH memory allocation for projects compiled with IAR IDE.



Note: Different memory allocation of PRIME Stacks for projects compiled with AS.

- The PRIME API is the only one that users need to use to develop their PRIME applications.
- Initialization. In order to start using the Atmel PRIME firmware stack, it is necessary to initialize different parameters and to call the corresponding initialization functions. These actions are independent of the operation mode.
- Operation modes. It is up to the user to decide the operation mode to run their application. There are two different modes: as Microcontroller or as a Real Operating System. Atmel provides an Operating System Support (OSS) based on FreeRTOS.
- Configuration files. The configuration files in the user application project allow the application to configure its own resources.
- Firmware upgrade management. In order to upgrade the PRIME FW stack, two regions are reserved in the memory: one for the current running stack and the other one to store the new stack. The PRIME FW stack manages the FU process as described in the PRIME specification, whereas users are responsible for handling the pointers to these regions and controlling the PRIME FW stack version running according to the indications received in the HAL. By just updating the pointer, a system restart is avoided and the user application can continue execution although the PRIME FW stack has changed.

6.6 PLC application 5 – PLC Network

In this chapter the example proposed is used to show the capabilities of the ATPL230A in a network of smart devices. One ATPL230AMB board acts as a Base Node, i.e. the device that controls the whole network, whereas the other one ATPL230AMB board acts as Service Node.

A PC tool is used to monitor and manage the Base Lite node called *Atmel PRIME Manager v1.a.b*. This tool allows you to monitor data traffic on PRIME networks and gather information of a PRIME network.

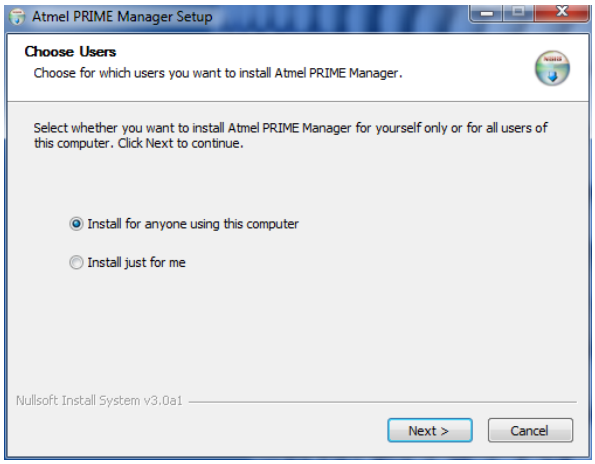
And even displays information about the devices connected to the network (MAC address, logical address and status) and the evolution of the network.

Following sections explain to you how to install the PC tool, select the projects, supplying the boards, select the COM ports to communicate with the ATPL230A and run the application.

6.6.1 Atmel PRIME Manager tool installation

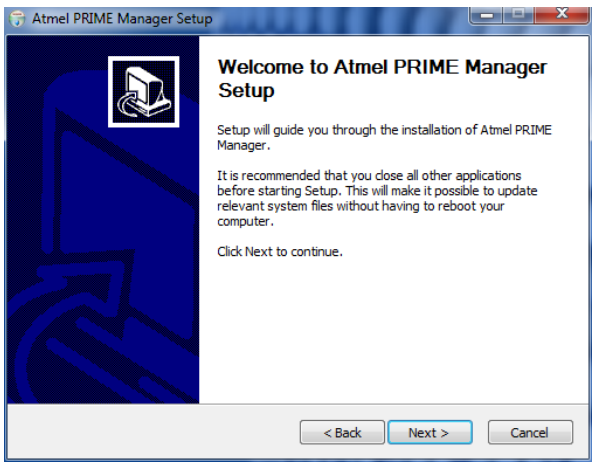
To install Atmel PRIME Manager v1.a.b tool in a Windows Operating System, execute the provided installer in the *PCTools* folder *"\PCTools\Atmel_PRIME_Manager\BN\"* and follow the installation wizard. The installer wizard should open. To follow the installation, click Next.

Figure 6-72. Atmel PRIME Manager Installation process.



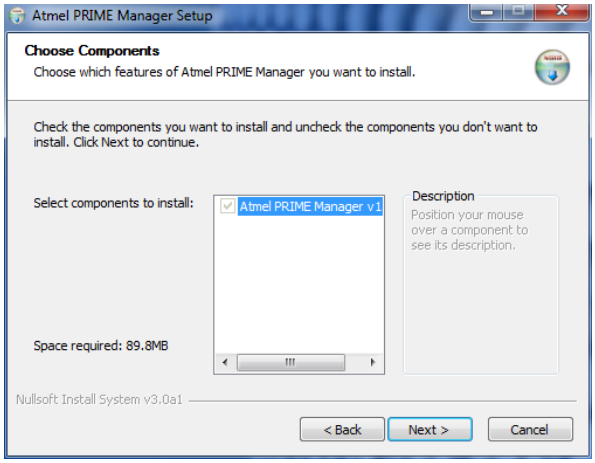
Select the users' permissions and click Next.

Figure 6-73. Atmel PRIME Manager Installation process.



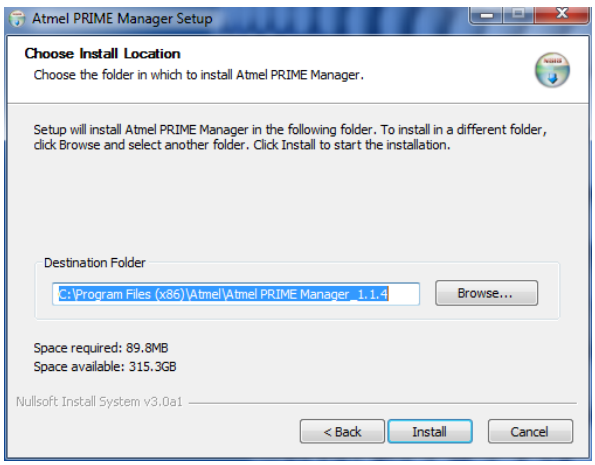
Click Next to continue.

Figure 6-74. Atmel PRIME Manager Installation process.



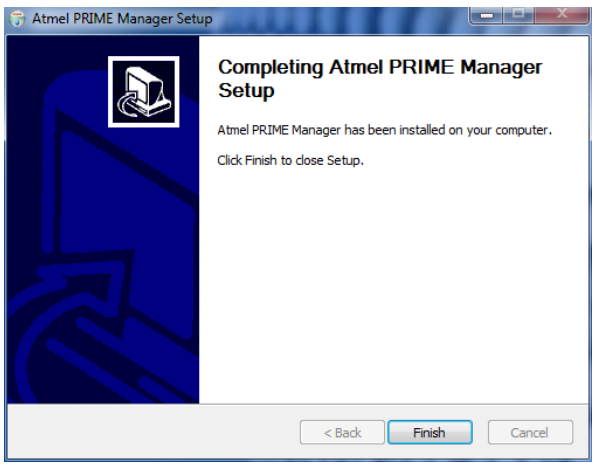
Click Next to continue.

Figure 6-75. Atmel PRIME Manager Installation process.



Setup will install the program in the Destination Folder. To install in a different folder, click Browse and select your destination folder. Click Install to start the installation process.

Figure 6-76. Atmel PRIME Manager Installation process.



Click Finish.

Now the program is installed in your computer and a shortcut should have been created in your desktop.

Note: Take into account that for this example, you have to use Atmel PRIME Manager version 1.a.b and not later versions, as 2.a.b.

6.6.2 Supplying the boards

Please refer to 6.2.2 in order to know how to supply the ATPL230AMB boards.

6.6.3 USB connection

Please refer to 6.2.3 in order to know how to connect the micro USB cable with the ATPL230AMB board.

6.6.4 Programming the embedded files

It is commented in section 6.2.4 the way to program a board. To program the board as Service node, process should be the same: building the IDE projects and downloading into the board.

Open the IDE tool used and select the PRIME FW Stack for AS and IAR IDE, so *APPS_PRIME_SERVICE_BIN.atsln* or *APPS_PRIME_SERVICE_BIN.eww* projects of *prime_service_bin*. And now build it to generate the output file. As it is commented in section 6.5.3.2, and it is showed in Figure 6-71, the Flash address to store the program should be 0x010F0000.

Open the IDE tool used, Atmel Studio or IAR Embedded Workbench. Select the PRIME user application (select the project running as Microcontroller operation mode), *APPS_PRIME_SERVICE_DLMSEMU_UI.atsln* or *apps_prime_service_dlmsemu_ui.eww*, and now build it to generate the output file. As it is commented in section 6.5.3.2, and it is showed in Figure 6-71, the Flash address to store the program should be 0x01000000 or 0x 010EC000 (AS).

Note that kits do not provide a J-Link ARM or SAM-ICE JTAG probe in order to connect to the user's host PC and the boards to download and debug the projects.

Remember that, PRIME FW Stack project is contained in the following folders depending on the IDE tools used:
".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_service_bin\sam4cp16b_sam4cp16bmb_gcc\as5_arm".
".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_service_bin\sam4cp16b_sam4cp16bmb_iar\iar".

Remember that, DLMS application project running as microcontroller mode is contained in the following folders depending on the IDE tools used:
".\Software\PRIME_vaa.bb.cc.dd\prime_service_dlmsemu_ui.atpl230amb\thirdparty\prime_ng\apps\prime_service_dlmsemu_ui\sam4c16c_atpl230amb\as5_arm".
".\Software\PRIME_vaa.bb.cc.dd\prime_service_dlmsemu_ui.atpl230amb\thirdparty\prime_ng\apps\prime_service_dlmsemu_ui\sam4c16c_atpl230amb\as5_arm\iar.

Remember that the J-Link USB drivers must have been downloaded previously from the Segger webpage (see section 6.1.4) and they depend on your operating system.

An alternative process to load the Service node project and Base Lite node files should be as is explained below:

1. Place the JTAG connector of the J-Link or SAM-ICE in the J13, JTAG connector of the board. Check pin number 1 of J13 connector to place the cable in the right position.
2. Switch on the power supply of the board.

3. Download the binary file using a command script file (see section 6.1.5). To do easier to load the bin file, Atmel provides you a script for Service node, *program_bin.bat*, and Base Lite node, *program_bin.bat*, which lets you download the .bin files in the right flash memory position. You can find them in the following directories: *".\Software_vaa.bb.cc.dd\Scripts\SN\"* and *".\Software_vaa.bb.cc.dd\Scripts\BN\"*. This script loads the files and shows an error when it falls. A typical error could be when the J-Link tool is a different version of the written in the script or in a different path. To solve it, edit the path according to your installation folder in the *.bat* file. See section 6.1.5 for more information.

Please, make sure all nodes (Base and Service) have valid MAC addresses. An invalid MAC makes a node unable to register. Take into account that the projects store a MAC address in the Base Lite Node and another one in the Service Node. If you have more Service Nodes, the same MAC address is stored in all of them. Since the MAC address must be unique for each node, you should be careful in this situation because MAC must be unique.

By default, every board has a MAC number preprogrammed which coincides with the serial number of the board label fixed in the enclosure. Anyway, if you want to change it, see section 6.6.4.1.

6.6.4.1 Setting MAC number

In the Service Node project, MAC address is defined but the user is free to change it. A way, could be: configure the board in Manufacturing Test mode and sending a PIB to write the MAC. Process should be: setting the board in MTP mode (send the PIB *MTP_PHY_ENABLE, 0x808E*), and send the PIB *macEUI48, 0x8100*, with the desired MAC number. You can use the Atmel PRIME Manager Tool, go to PRIME management view and select MFG Test tab. Once the Manufacturing mode has been enabled, write the MAC number in the box and press Set Mac button. This action is showed in section 6.7.5.5.

6.6.5 Running the PLC application example 6

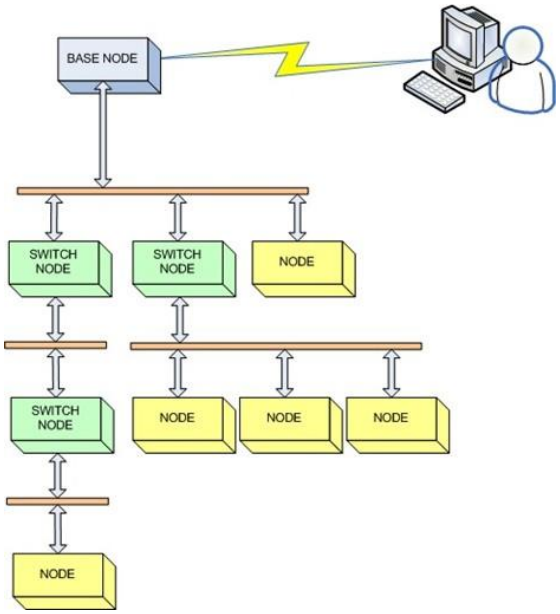
As you can see in Figure 6-77, the boards are plugged into the same power line. In this PLC example, one board is the Base Lite Node and the other one is the Service Node. And users have to execute an instance of the Atmel PRIME Manager tool – which has been previously installed in the host PC – in order to enable communication between the Base Node board and the PC. In this example, please note that it is only necessary to establish a serial connection between the board acting as a Base Node and the host PC, so only one instance of the PC tool is required.

The Atmel PRIME Manager tool allows you to gather information of a PRIME network. This tool displays information about the devices connected to the network (MAC address, logical address, firmware version, vendor identity, etc.) and the amount of registered nodes over time. The PRIME Network Manager tool can also upgrade the firmware of all devices connected to the Base Node and it is used to monitor data traffic on and manage the PRIME network.

This PC application can be found in the PC Tools folder: *"…\PCTools\Atmel PRIME Manager\BN"*.

Please remember that the provided PRIME Base Node is a lite version, i.e. it is limited to manage up to 10 Service Node connections. Default coupling board configuration for the projects is configured for ATPLCOUP001v1 coupling board.

Figure 6-77. PRIME Network concept.



Once the Base Lite node board is powered, the green led D5, LED0, is blinking. The Base Node Lite communicates with the Atmel PRIME Manager tool by means of a serial interface and it retrieves information about the network structure and data traffic. The main window of the Atmel PRIME Manager PC interface is shown in Figure 6-78.

Figure 6-78. Atmel PRIME Manager tool window.

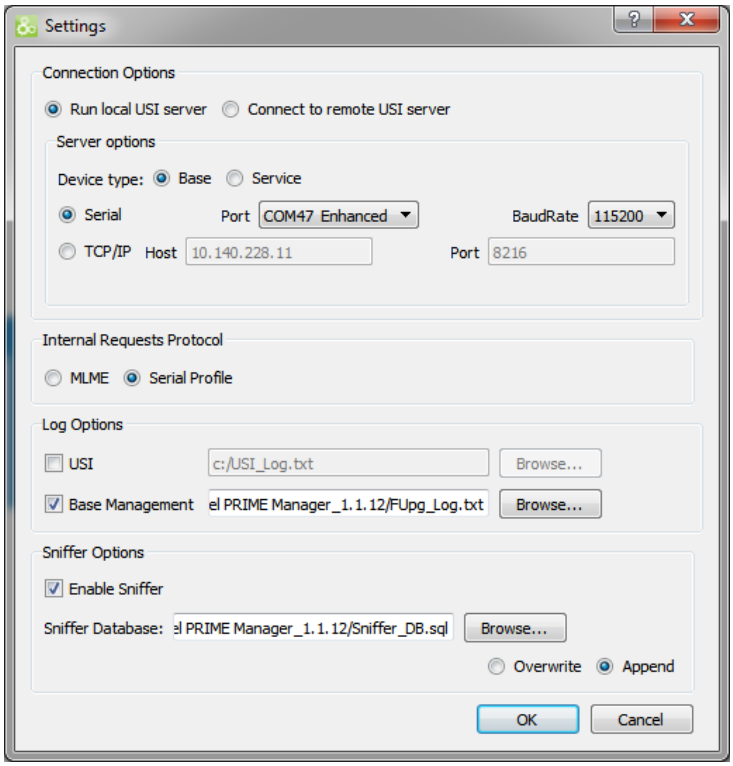
Once the application is launched, the COM port for the board needs to be configured. The COM port selection window is available by choosing File>SettingsInput. A new window *Settings* will appear as shown in Figure 6-79. Select the connection options:

- Click Run local USI server option.
- Select device type as Base.

- Click the Serial option and select the COM port (it can be checked in the Windows Device Manager Com/Port section, select the enhanced port -UART0-) and set the speed. Default speed for this application is 115200 bauds.

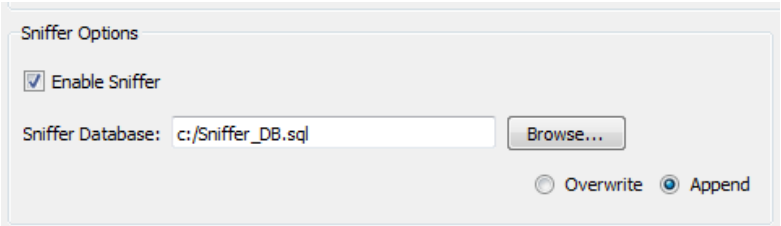
Also, this tool is able to connect to a remote device through the TCP/IP protocol. This option requires the IP address of the server and the port opened by the server.

Figure 6-79. Settings window.



In case the embedded sniffer is enabled, the database file to store the traffic must be configured. So in the Settings window select the database file name and the location to store the sniffer log and click OK button (Figure 6-80). Furthermore, overwrite or append option can be selected. Database files can hold longer logs without having to split them in pieces. Also log stored files can be opened to review the file.

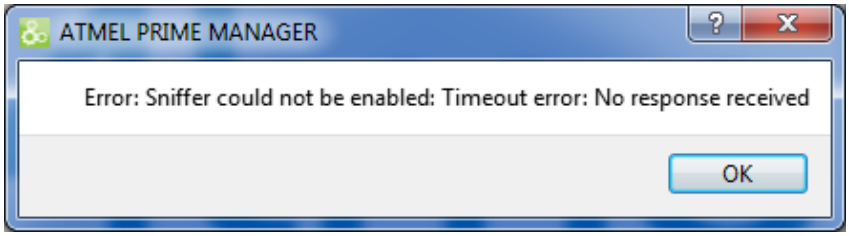
Figure 6-80. Database Settings.



At this point, the tool is ready to start capturing data. If board is not powered, this is the point to supply it.

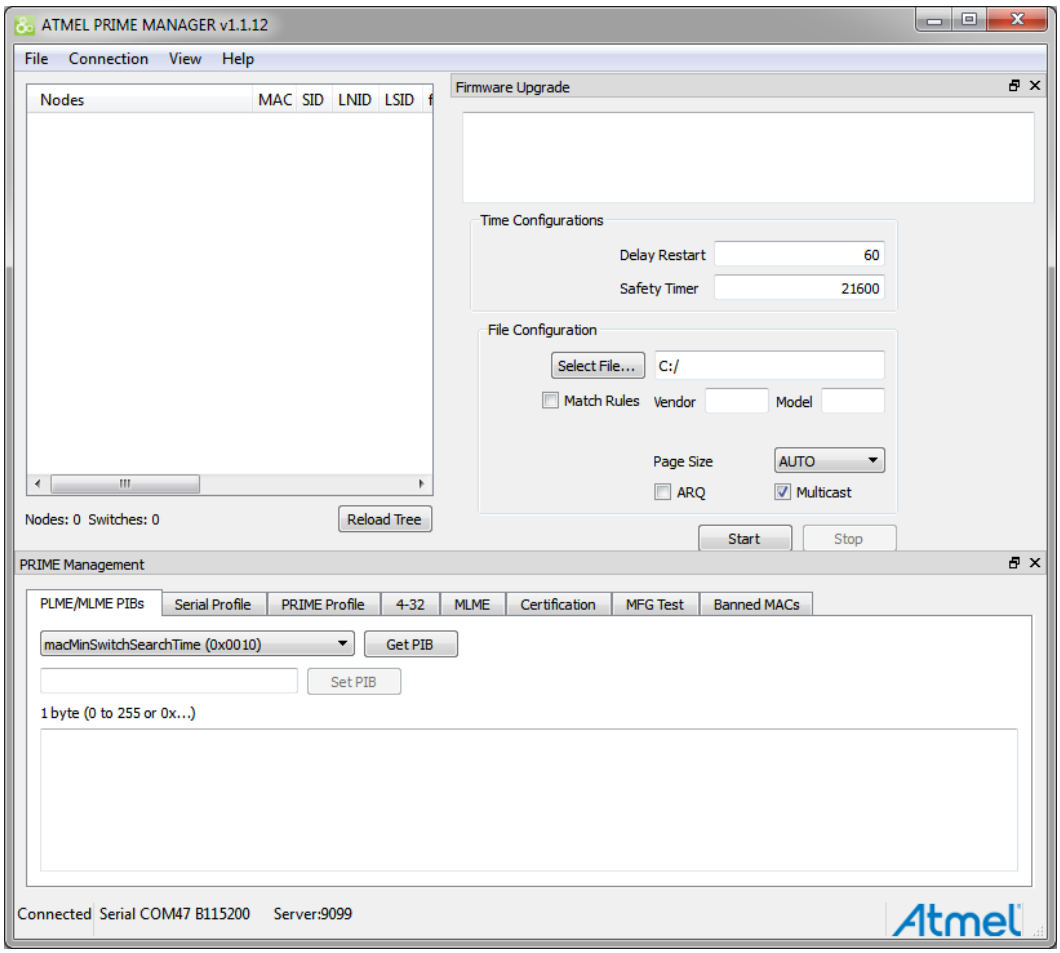
Click on the menu Connection>Connect to begin logging data. In case the serial COM port is not the proper, or board is not powered, tool shows an error window as the figure.

Figure 6-81. Error window.



If COM port is the proper, appears the main window of the tool (Figure 6-82).

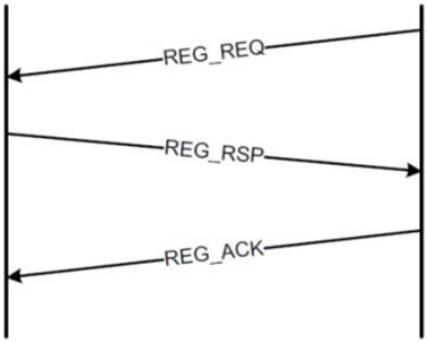
Figure 6-82. Atmel PRIME Manager tool main window.



On the other hand, once the Service node board is powered, the green led D5, LED 0, is blinking.

Once the boards are connected to the mains, the PRIME network begins to form. The Service Node listens to the Base Node beacons and starts the registration process shown. This process is shown in the Figure 6-83.

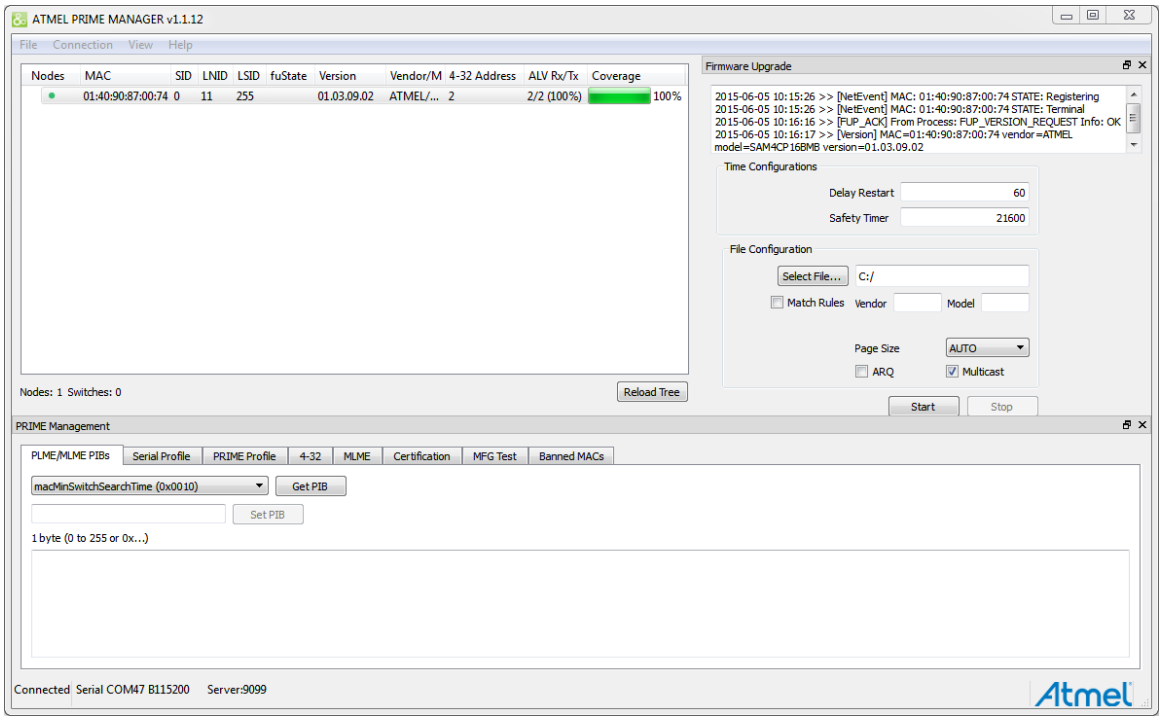
Figure 6-83. PRIME registration process.



The Service Node sends the registration request and waits for the base node respond. When the Base node sends a PLC message, the TX led of the coupling board is toggled. And when the Service node sends a message, the TX led of the coupling board is toggled. You can use them to check if boards are sending PLC messages properly.

If tool establishes the communication with the COM port of Base node, the status bar at the bottom of the window will show the current setup and status of the tool. On a PRIME network, the main window will look like as the Figure 6-84. Main window displays a table with the current log. It is updated in real time as frames are received from the hardware sniffer.

Figure 6-84. Atmel PRIME Manager tool main window when is connected to a Base Node.



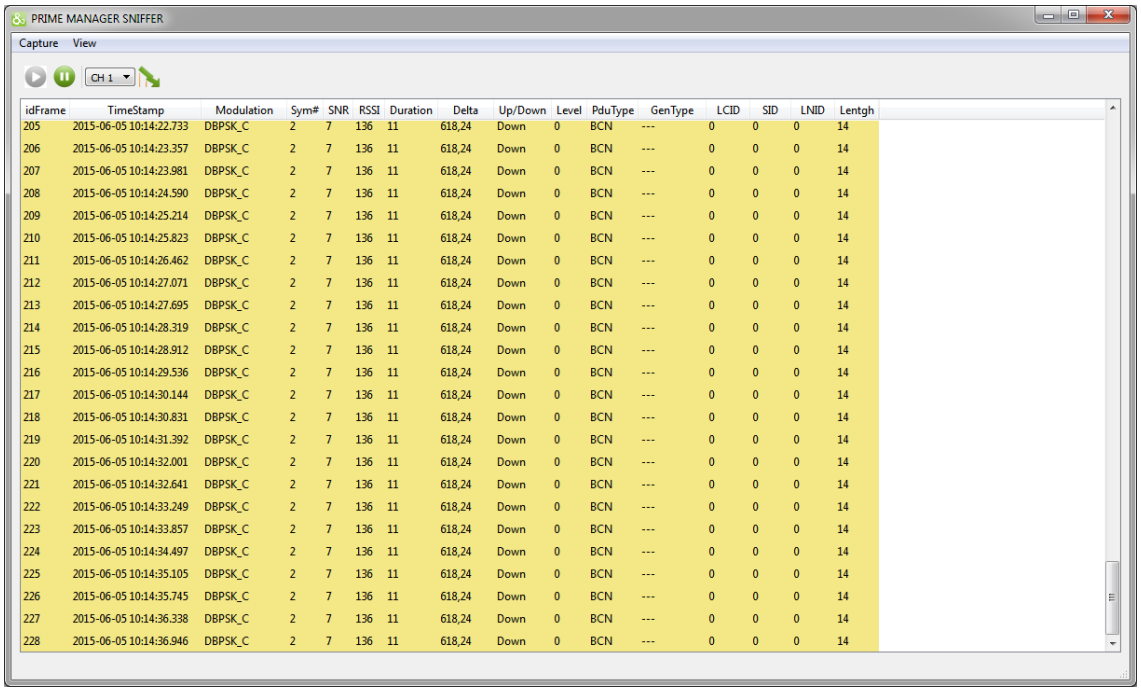
The main window is split into three different areas:

- Network Topology view. This is the basic presentation in the main application window and it is always visible when is connected to a Base Node. This view displays the logical network structure. The structure is inferred analyzing the network events received from the Base Node. In addition, this view shows node information such as MAC, LNID, LSID, SID, firmware upgrade state, firmware version, model and vendor, 4-32 address, ALV message count and coverage.
- Firmware Upgrade view. This view displays firmware upgrade options and information about the current upgrade process. It is always visible when is connected to a Base Node.

- PRIME Management. This view allows requesting information and other functionalities provided by the different protocols available in the tabs. It is available in both Base and Service Node connection, but they are different.

Besides this main window, a new window is shown with the Sniffer. The aspect and working are the same as the ATPL Multiprotocol Sniffer Tool. See section 6.4.5 for more information.

Figure 6-85. ATPL Multiprotocol Sniffer tool main window.



idFrame	TimeStamp	Modulation	Sym#	SNR	RSSI	Duration	Delta	Up/Down	Level	PduType	GenType	LCID	SID	LNID	Lentgh
205	2015-06-05 10:14:22.733	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
206	2015-06-05 10:14:23.357	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
207	2015-06-05 10:14:23.981	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
208	2015-06-05 10:14:24.590	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
209	2015-06-05 10:14:25.214	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
210	2015-06-05 10:14:25.823	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
211	2015-06-05 10:14:26.462	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
212	2015-06-05 10:14:27.071	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
213	2015-06-05 10:14:27.695	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
214	2015-06-05 10:14:28.319	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
215	2015-06-05 10:14:28.912	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
216	2015-06-05 10:14:29.536	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
217	2015-06-05 10:14:30.144	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
218	2015-06-05 10:14:30.831	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
219	2015-06-05 10:14:31.392	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
220	2015-06-05 10:14:32.001	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
221	2015-06-05 10:14:32.641	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
222	2015-06-05 10:14:33.249	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
223	2015-06-05 10:14:33.857	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
224	2015-06-05 10:14:34.497	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
225	2015-06-05 10:14:35.105	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
226	2015-06-05 10:14:35.745	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
227	2015-06-05 10:14:36.338	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14
228	2015-06-05 10:14:36.946	DBPSK_C	2	7	136	11	618,24	Down	0	BCN	---	0	0	0	14

The capture window has a tool bar with four commands:

- Pause command will stop the update of the scroll view, while the logging process will continue.
- To restart showing the live stream of PDUs, click Play button.
- Channel combo box allows selecting the PRIME channel to listen. Obviously the compatible PLC coupling board must be used.
- Thunder button will set the CRC configuration on the hardware device. If it is enabled, the hardware device will calculate the CRC on all the frames and discard frame errors. If it is enabled, all frames received will be sent to the PC software.

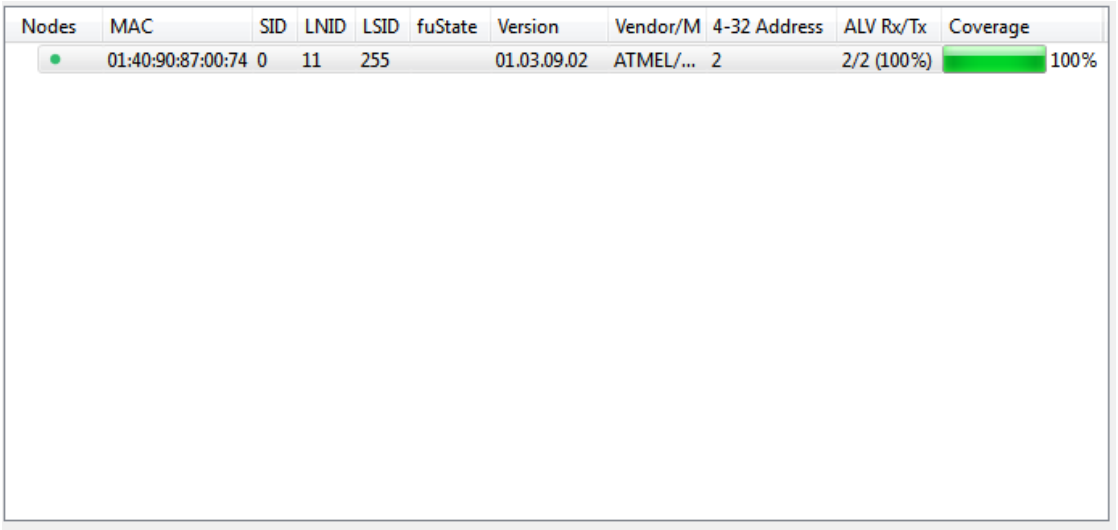
While the PLC traffic is logged into a database, the software tries to infer the PLC network structure and status as seen by the Base Node. This information is shown in several docking views. They are available on the menu *View*:

- Hexa view shows the hexadecimal display of the selected frame in the main view.
- Packet view shows the disassembled data of the selected frame in the main window. All the specified fields on the PRIME specification are shown.
- Network view shows the current status of the inferred network by the software. It is refreshed every time a change in the network is detected.
- Nodes/Switches plot shows the evolution of the network as seen by the sniffer. It plots the number of active nodes and switches on the network versus time. It is useful to detect problems of stability on the PLC network.
- Filter view allows selecting the frames shown in the main view table.

6.6.5.1 Network Topology view

The Network Topology View looks like the following figure.

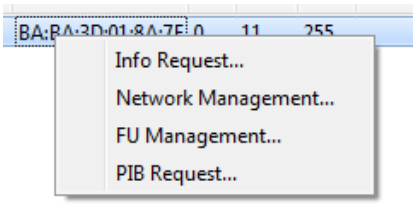
Figure 6-86. Network Topology view.



This view shows the current status of the network inferred by the tool. It is automatically refreshed every time a change in the network is detected, for example a new device is registered or promoted to switch. It is also possible to refresh this view with the Reload Tree button, which requests the registered nodes to the Base Node in order to rebuild the network structure.

There is a right-click menu in this view with further options as shown in the following figure.

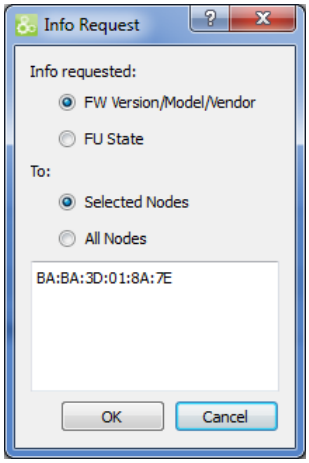
Figure 6-87. Right-Click menu.



Info Request dialog

This dialog allows requesting information from the Service Nodes by means of the Base Management Protocol. Available information to be requested is firmware version, model and vendor, and firmware state. The firmware state is only returned when a firmware upgrade process is running.

Figure 6-88. Info Request dialog.



The information can be requested to all registered nodes or just the selected nodes (multiple selections are enabled in Windows OS mode).

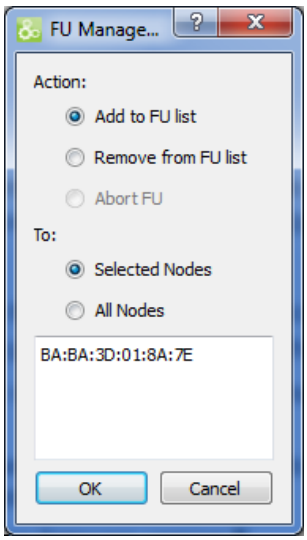
Network Management dialog

This dialog lets the user to manage the network with available functions Unregister, Promote, Demote and Reboot.

Firmware Upgrade Management dialog

The available options in this dialog depend on whether a firmware upgrade process is running or not. If not running, nodes can be added to and removed from the list. When the process starts, the Remove option is disabled but the Abort option is enabled so that the upgrade process can be cancelled any time. It is always possible to add nodes to the list.

Figure 6-89. FU dialog.



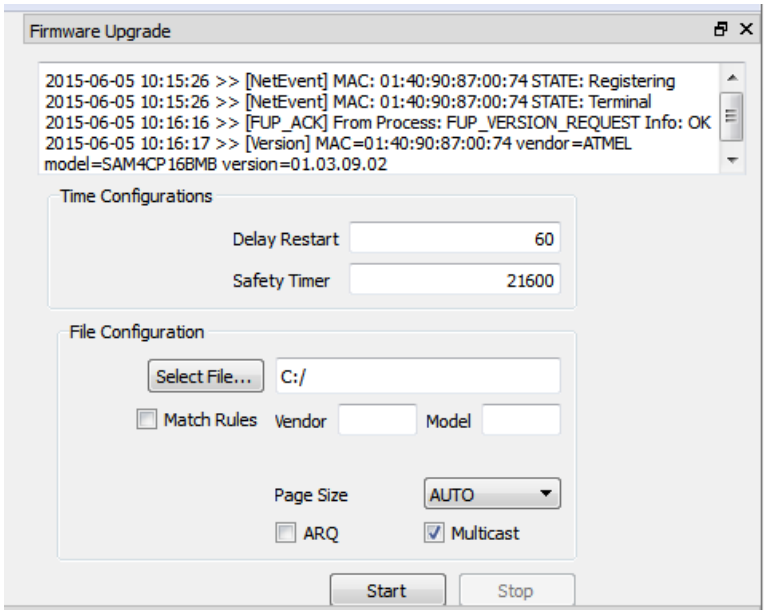
This dialog only adds the nodes to the Firmware Upgrade, but the process does not start when the *OK* button is clicked. The next section describes how to start the Firmware Upgrade Process.

6.6.5.2 Firmware Upgrade

In case you want to upgrade the firmware of the Service nodes from the Base node, you have to use the Firmware Upgrade tab. There are some parameters to configure the FUP:

- Delay Restart. Time in seconds that a device must wait before performing a restart after receiving the new firmware.
- Safety Timer. Time in seconds that a device waits to get back to the previous firmware version if the upgraded node is not able to register.
- File. Filename of the new firmware version.
- Match Rules. When marked, the Base Node checks that vendor and model firmware of the Service Node match those of the new firmware. If they do not match, the Base Node will not upgrade this node.
- Page Size. Size of the data packets sent through the PLC line. By default, the *AUTO* option is 64 bytes.
- ARQ. Enable or disable the ARQ protocol in the Base Node.
- Multicast. Enable or disable PRIME multicast capabilities to transfer the firmware to a list of devices.

Figure 6-90. Firmware Upgrade view.



The Firmware Upgrade tab is used to update the PLC version of the Service Nodes. For that:

- Click on the Firmware Update tab.
- Select the .bin file that it has to be upgraded (vendor and model must be the same between different versions).
- Keep the default values for all the parameters.
- Select the Service Nodes that need to be upgraded.
- Click the Start button. To cancel a FU process, use the Stop button or abort it from the Network Topology view.

The firmware upgrade process consists of two phases:

- Transferring the .bin file from PRIME Network Manager to the Base Node by UART0.
- Transferring the .bin file from the Base Node to the Service Nodes by PLC communication.

Once the firmware upgrade process has finished, get the new PRIME version (right-click on the Service Node) and compare with the previous PRIME version.

Atmel Studio projects provided in the kit do not support firmware upgrade process, only IAR projects.

To uninstall the Atmel PRIME Manager tool from your computer, go to Start > All Programs > ATMEL > Atmel PRIME Manager vX.Y.Z > Uninstall.

For further information of the Atmel PRIME Manager tool, please refer to the tool's embedded help (in the menu bar).

6.7 PLC application 6 – ATMEL PRIME Manager tool

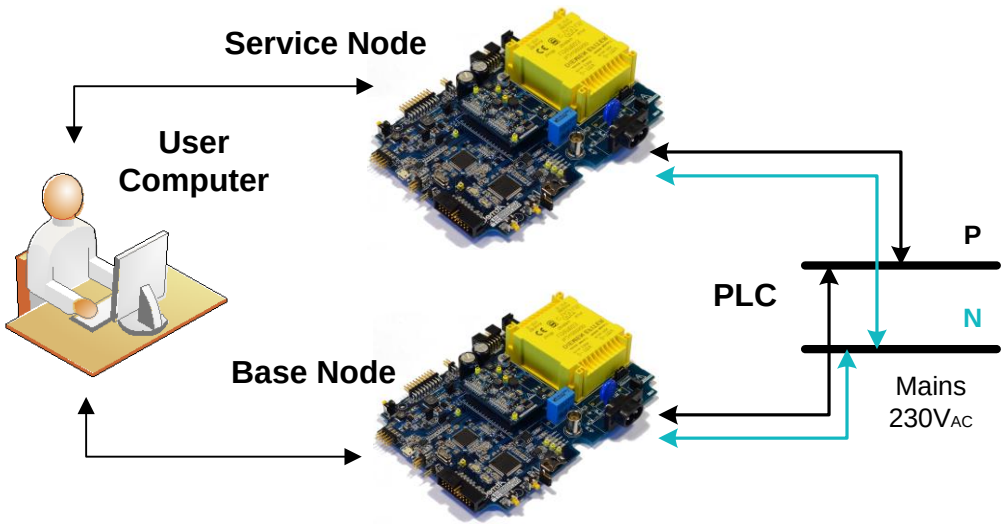
The Atmel Universal Serial Interface (USI) is a peripheral in the HAL that enables the handling of the different serial interfaces described in the PRIME specification through one or more serial ports (always an UART). It can also handle the serialization of the PRIME API in order to facilitate communication between an external application and the PRIME FW stack.

In this chapter we are going to describe the Atmel PRIME Manager tool for Service node. Tool has the following tabs:

- PLME/MLME PIBs.
- Serial Profile.
- MAC.
- 4-32.
- PLME.
- MLME.
- Certification.
- ManuFacturinG (MFG) Test.

As the figure below shows, the setup is composed of three different parts: one ATPL230AMB board acts as a Base Lite node, the other one ATPL230AMB board acts as Service node and the User Computer.

Figure 6-91. Atmel PRIME Manager setup.

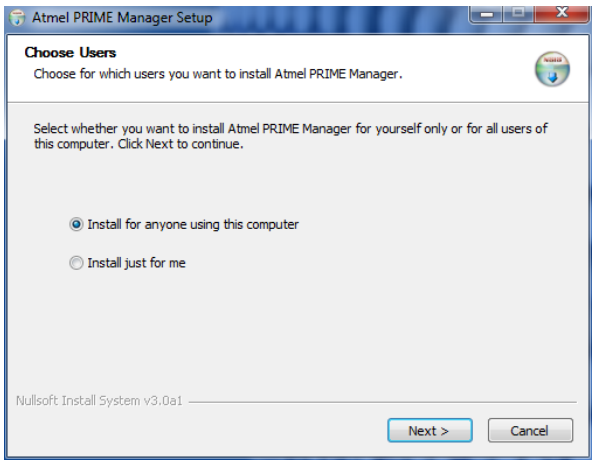


Following sections explain to you how to install the PC tool, select the projects, supplying the boards, select the COM ports to communicate with the ATPL230A and run the application.

6.7.1 Atmel PRIME Manager tool installation

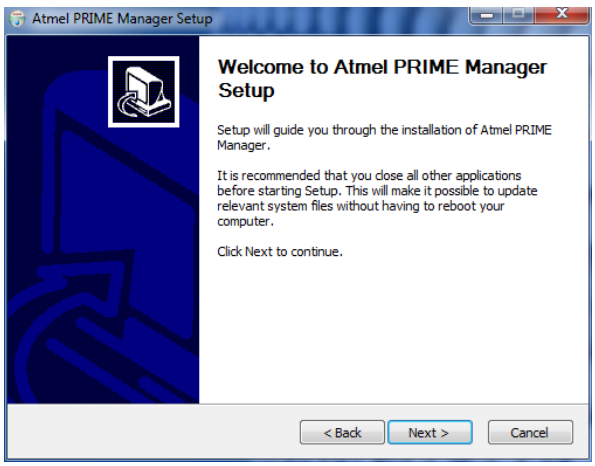
To install Atmel PRIME Manager v2.a.b tool in a Windows Operating System, execute the provided installer in the *PCTools* folder *"\PCTools\Atmel_PRIME_Manager\SN\"* and follow the installation wizard. The installer wizard should open. To follow the installation, click Next.

Figure 6-92. Atmel PRIME Manager Installation process.



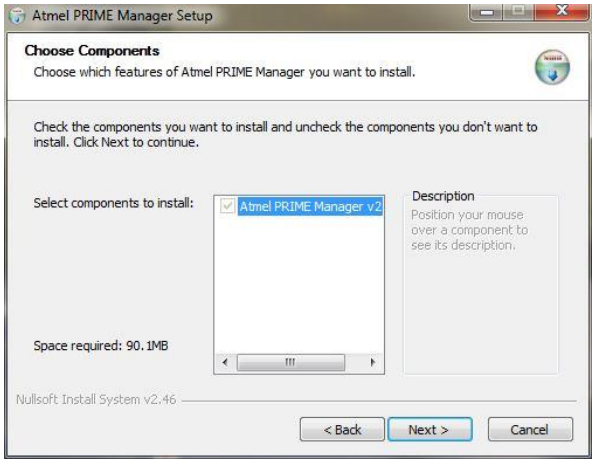
Select the users' permissions and click Next.

Figure 6-93. Atmel PRIME Manager Installation process.



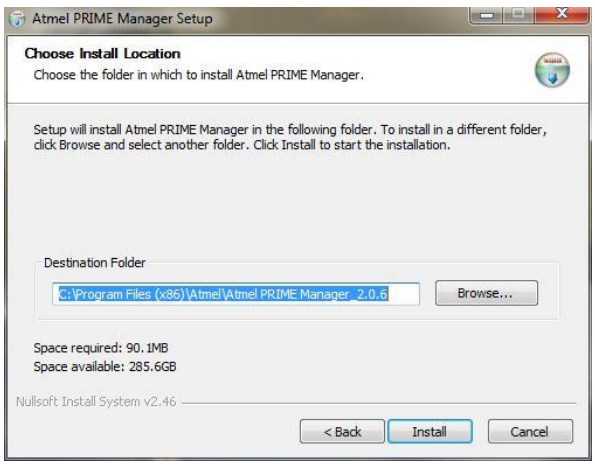
Click Next to continue.

Figure 6-94. Atmel PRIME Manager Installation process.



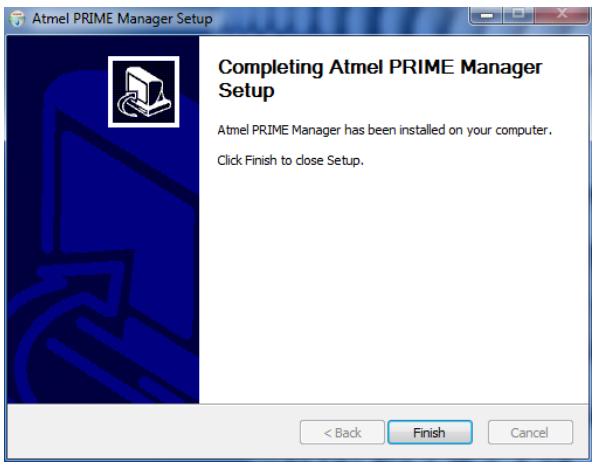
Click Next to continue.

Figure 6-95. Atmel PRIME Manager Installation process.



Setup will install the program in the Destination Folder. To install in a different folder, click Browse and select your destination folder. Click Install to start the installation process.

Figure 6-96. Atmel PRIME Manager Installation process.



Click Finish.

Now the program is installed in your computer and a shortcut should have been created in your desktop.

6.7.2 Supplying the boards

Please refer to 6.2.2 in order to know how to supply the ATPL230AMB boards.

6.7.3 USB connection

Please refer to 6.2.3 in order to know how to connect the micro USB cable with the ATPL230AMB board.

6.7.4 Programming the embedded files

It is commented in section 6.6.4 the way to program a board. To program the board as Service node, process should be the same: building the IDE projects and downloading into the board.

Note that kits do not provide a J-Link ARM or SAM-ICE JTAG probe in order to connect to the user's host PC and the boards to download and debug the projects.

> Remember that, PRIME FW Stack project is contained in the following folders depending on the IDE tools used:
> *".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_service_bin\sam4cp16b_sam4cp16bmb_gcc\as5_arm".*
> *".\Software\PRIME_vaa.bb.cc.dd\prime_service_bin\thirdparty\prime_ng\apps\prime_service_bin\sam4cp16b_sam4cp16bmb_iar\iar".*
>
> Remember that, modem application project running as microcontroller mode is contained in the following folders depending on the IDE tools used:
> *"./Software/PRIME_vaa.bb.cc.dd\prime_service_modem.atpl230amb\thirdparty\prime_ng\apps\prime_service_modem\sam4c16c_atpl230amb\as5_arm".*
> *"./Software/PRIME_vaa.bb.cc.dd\prime_service_modem.atpl230amb\thirdparty\prime_ng\apps\prime_service_modem\sam4c16c_atpl230amb\iar".*

Remember that the J-Link USB drivers must have been downloaded previously from the Segger webpage (see section 6.1.4) and they depend on your operating system.

An alternative process to load the Service node project and Base Lite node files should be as is explained below:

1. Place the JTAG connector of the J-Link or SAM-ICE in the J13, JTAG connector of the board. Check pin number 1 of J13 connector to place the cable in the right position.
2. Switch on the power supply of the board.
3. Go to script folder of Service Node *"…\Software_vaa.bb.cc.dd\Scripts\SN"*, and edit the file *program_bin.jlink*. Uncomment the *apps_prime_service_modem_flash.bin* file and comment the *apps_prime_service_dlmsemu_ui_flash.bin* file. After that, save and close it.
4. Download the binary file using a command script file (see section 6.1.5). To do easier to load the bin file, Atmel provides you a script for Service node, *program_bin.bat*, and Base Lite node, *program_bin.bat*, which lets you download the .bin files in the right flash memory position. You can find them in the following directories: *"…\Software_vaa.bb.cc.dd\Scripts\SN"* and *"…\Software_vaa.bb.cc.dd\Scripts\BN"*. These scripts load the files and show an error when programming process falls. A typical error could be when the J-Link tool is a different version of the written in the script or in a different path. To solve it, edit the path according to your installation folder in the *.bat* file.

Please, make sure all nodes (Base and Service) have valid MAC addresses. An invalid MAC makes a node unable to register. Take into account that the projects store a MAC address in the Base Lite Node and another one in the Service Node. If you have more Service Nodes, the same MAC address is stored in all of them. Since the MAC address must be unique for each node, you should be careful in this situation because MAC must be unique.

By default, every board has a MAC number preprogrammed which coincides with the serial number of the board label fixed in the enclosure. Anyway, if you want to change it, see section 6.6.4.1.

6.7.4.1 Setting MAC number

In the Service Node project, MAC address is defined but the user is free to change it. A way, could be: configure the board in Manufacturing Test mode and sending a PIB to write the MAC. Process should be: setting the board in MTP mode (send the PIB *MTP_PHY_ENABLE, 0x808E*), and send the PIB *macEUI48, 0x8100*, with the desired MAC number. You can use the Atmel PRIME Manager Tool, go to PRIME management view and select MFG Test tab. Once the Manufacturing mode has been enabled, write the MAC number in the box and press Set Mac button. This action is showed in section 6.7.5.5.

6.7.5 Running the PLC application example 7

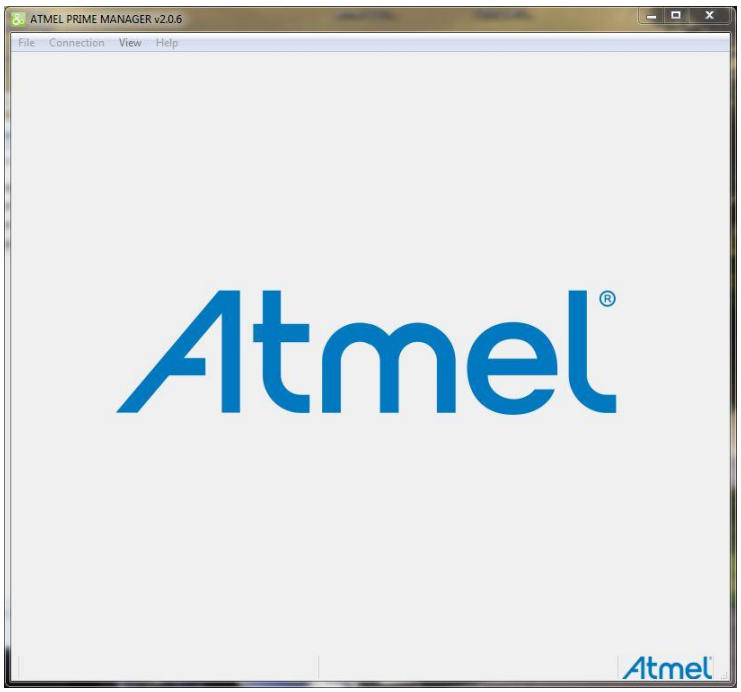
As you can see in Figure 6-91, the example's boards are plugged into the same power line. In this PLC example, one board is the Base Lite Node and the other one is the Service Node. And users have to execute an instance of the Atmel PRIME Manager tool – which has been previously installed in the host PC – in order to enable communication between the Service node board and the PC. In this example, please note that it is only necessary to establish a serial connection between the board acting as a Service node and the host PC, so only one instance of the PC tool is required.

This PC application can be found in the PC Tools folder: *"…\PCTools\Atmel PRIME Manager\SN"*.

Please remember that the provided PRIME Base Node is a lite version, i.e. it is limited to manage up to 10 Service Node connections. Default coupling board configuration for the projects is configured for ATPLCOUP001v1 coupling board.

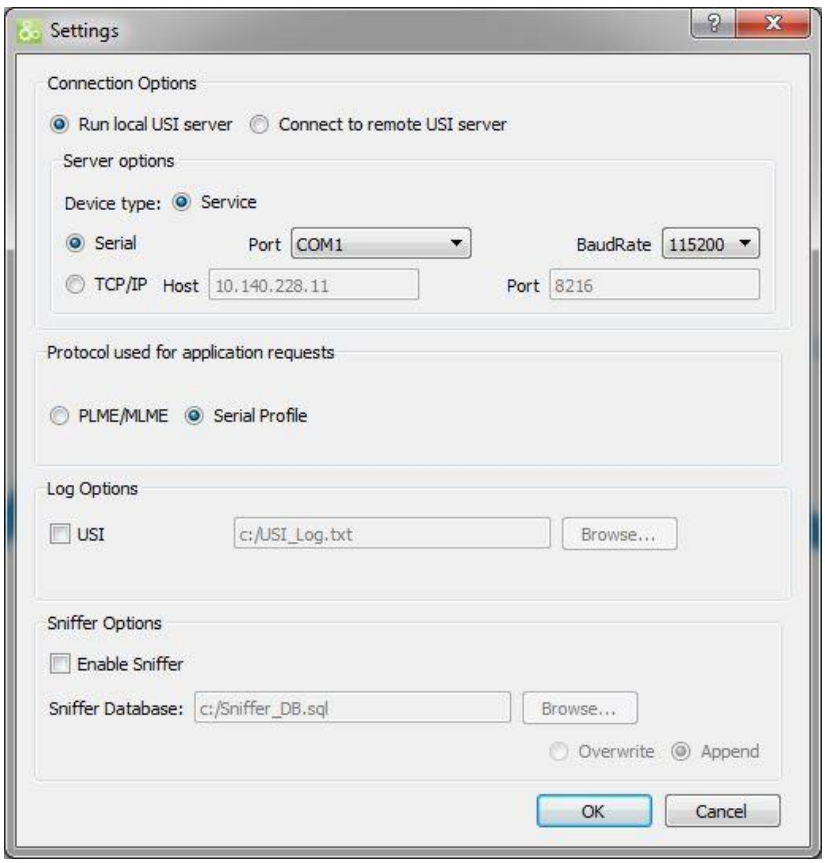
When the Service node board is powered, the green led D5, LED0, is blinking. The Service node communicates with the Atmel PRIME Manager tool by means of a serial interface. The main window of the Atmel PRIME Manager PC interface is shown in Figure 6-97.

Figure 6-97. PRIME Management view.



Once the application is launched, the COM port for the board needs to be configured. The COM port selection window is available by choosing File>SettingsInput. A new window *Settings* will appear as shown in Figure 6-98.

Figure 6-98. Settings window.



Select the connection options:

- Click Run local USI server option.
- Click the Serial option and select the COM port (it can be checked in the Windows Device Manager Com/Port section, select the enhanced port -UART0-) and set the speed. Default speed for this application is 115200 bauds.

Also, this tool is able to connect to a remote device through the TCP/IP protocol. This option requires the IP address of the server and the port opened by the server.

This tool could use different protocols for the requests. Depending on the application loaded in the service node board, it uses kind of protocol. For example, the *prime_service_modem_atpl230amb* uses Serial Profile and PLME/MLME and *prime_service_dlmsemu_ui.atpl230amb* project uses Serial Profile. It involves that different tabs of the tool run with different applications. See following tables. In this example, we will use the modem example, so *prime_service_modem_atpl230amb* project.

Table 6-4. **Protocols for Software projects.**

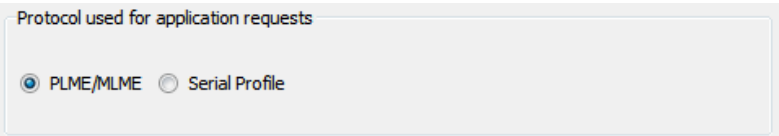
Project example	Serial Profile	PLME/MLME
thirdparty.apps.prime_service_prime_ng.dlmsemu_fi.atpl230amb	X	-
thirdparty.apps.prime_service_prime_ng.dlmsemu_ui.atpl230amb	X	-
thirdparty.prime_ng.apps.prime_service_modem_230	X	X

Table 6-5. **Tool tabs and protocols.**

Protocol used	PLME/MLME PIBs	Serial Profile	MAC	4-32	PLME	MLME	Certification	MFG Test
Serial Profile	-	X	-	-	-	-	X	X
PLME/MLME	X	-	X	X	X	X	-	-

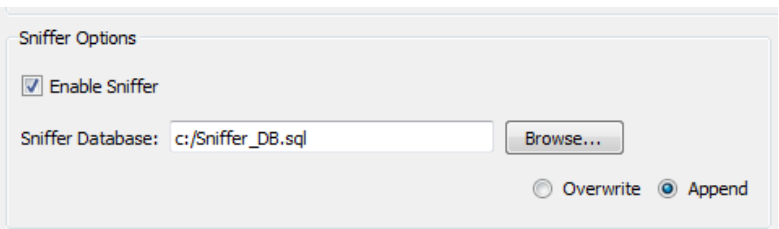
So in window *Settings* select the option PLME/MLME protocol for the examples 6.7.5.1, 6.7.5.2 and 6.7.5.3.

Figure 6-99. Protocol PLME/MLME selected.



In case the embedded sniffer is enabled, the database file to store the traffic must be configured. So in the Settings window select the database file name and the location to store the sniffer log and click OK button (Figure 6-100). Furthermore, overwrite or append option can be selected. Database files can hold longer logs without having to split them in pieces. Also log stored files can be opened to review the file.

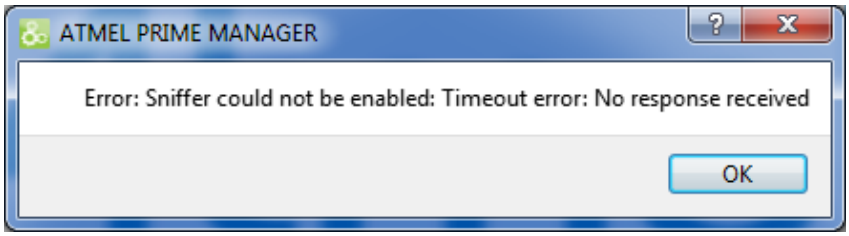
Figure 6-100. Database Settings.



At this point, the tool is ready to start capturing data. If board is not powered, this is the point to supply it.

Click on the menu Connection>Connect to begin logging data. In case the serial COM port is not the proper, or board is not powered, tool shows an error window as the figure.

Figure 6-101. Error window.



If tool establishes the communication with the COM port of Service node, the status bar at the bottom of the window will show the current setup and status of the tool.

On the other hand, once the Base Lite node board is powered, the green led D5, LED0, is blinking.

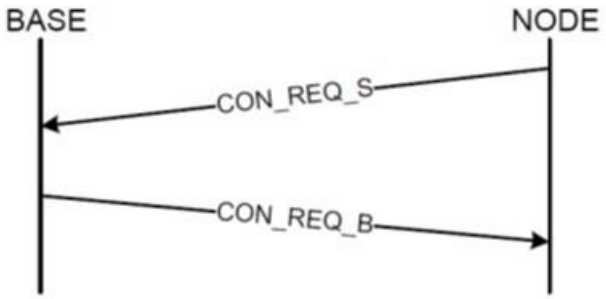
Once the boards are connected to the mains, the PRIME network begins to form. The Service Node listens to the Base Node beacons and starts the registration process shown. This process is shown in the Figure 6-83.

The Service Node sends the registration request and waits for the base node respond. When the Base node sends a PLC message, the TX led of the coupling board is toggled. And when the Service node sends a message, the TX led of the coupling board is toggled. You can use them to check if boards are sending PLC messages properly.

6.7.5.1 Opening a 4-32 connection

PRIME specifies that is the service node which opens the 4-32 connection. The process is the showed in Figure 6-102:

Figure 6-102. PRIME 4-32 connection request.



Note: For this example, the Service node requires PLME/MLME protocol, so it has been programed with *prime_service_modem_atpl230amb* project.

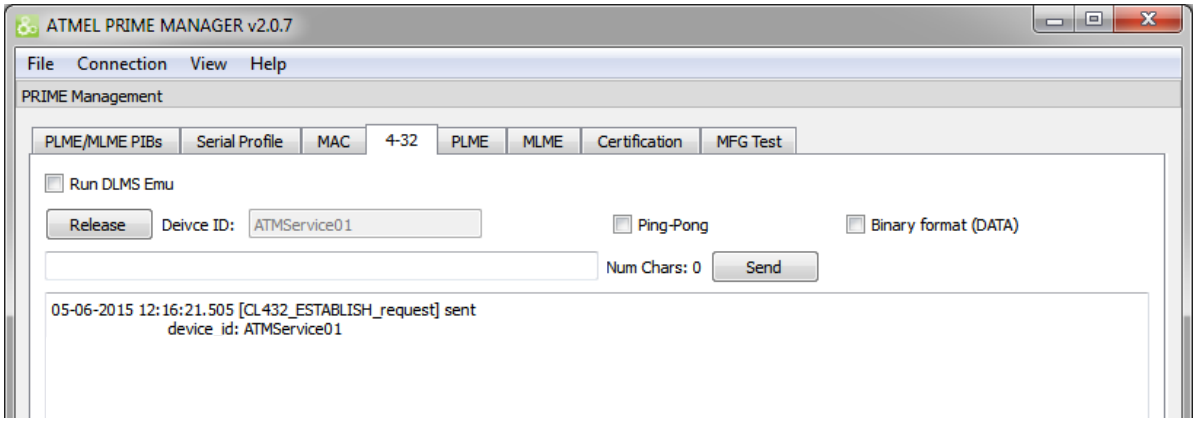
Process to open a 4-32 connection with the Atmel PRIME Manager tool in the Service node should be:

- Select the 4-32 tab.
- Click Establish.
- The executed operations appear in the Log output windows of both the Atmel PRIME Manager tool in the Service and Base nodes instances. See the figures below.

Note: The Service Node must be registered in the Base Node to perform this operation.

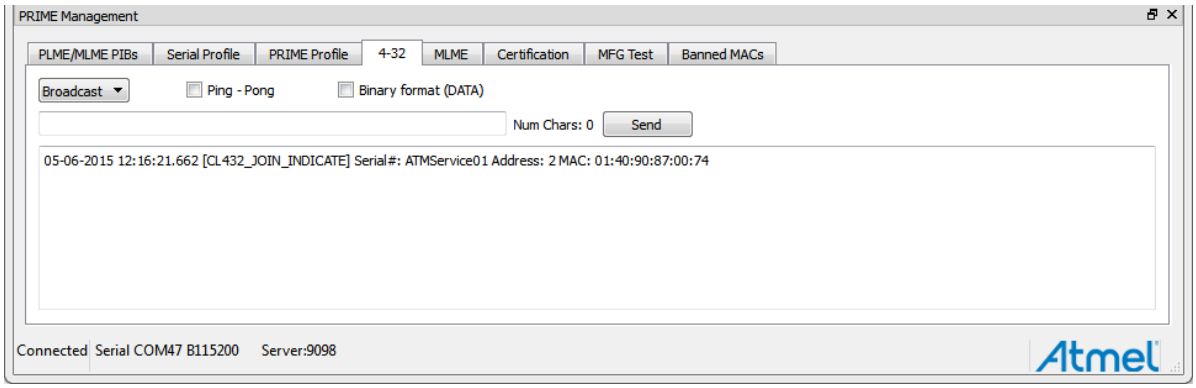
Service Node sends the request and when the 4-32 connection is opened, the service node receives a confirmation.

Figure 6-103. Service node tool application window.



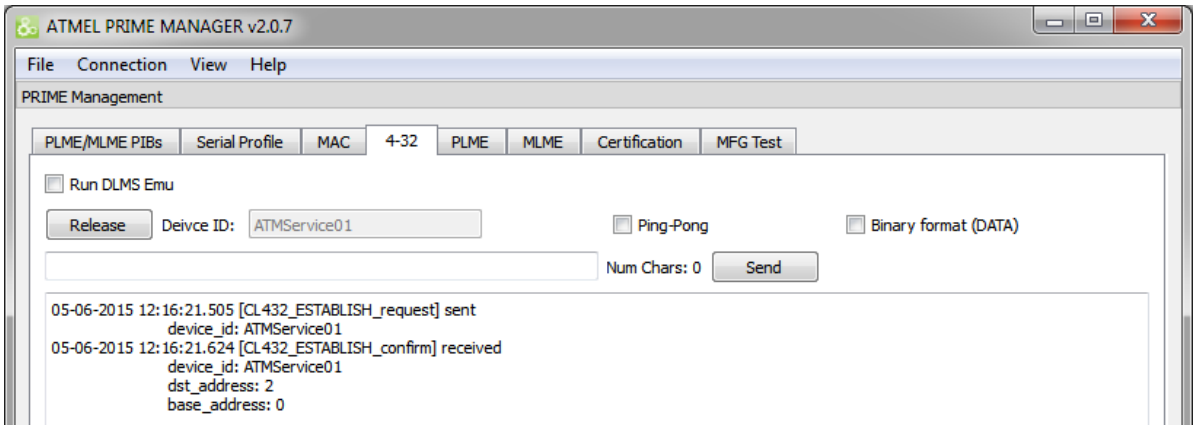
At the same time, you will receive the 4-32 notification in the Base node Tool instance.

Figure 6-104. Base node tool application window.



When the 4-32 connection is opened, the Service Node receives a confirmation.

Figure 6-105. Service node tool application window.



6.7.5.2 Sending messages

Once the communication is established, we can send data between Service node and Base node. For example, in this case we send a message from Service node to Base node. Process should be:

- In the 4-32 tab, type a message as: *Atmel Enabling unlimited possibilities*.
- Click Send.
- In the Log output of the Base node instance we receive the sent data.

Note: The Service Node must be registered in the Base Node to perform this operation.

Maximum size of the 4-32 packet that we are able to manage is 1000 bytes.

When the option Binary format (DATA) is marked, the messages must be introduced in hexadecimal format, otherwise ASCII format is used. The received messages are also shown in such format.

Figure 6-106. Service node tool application window.

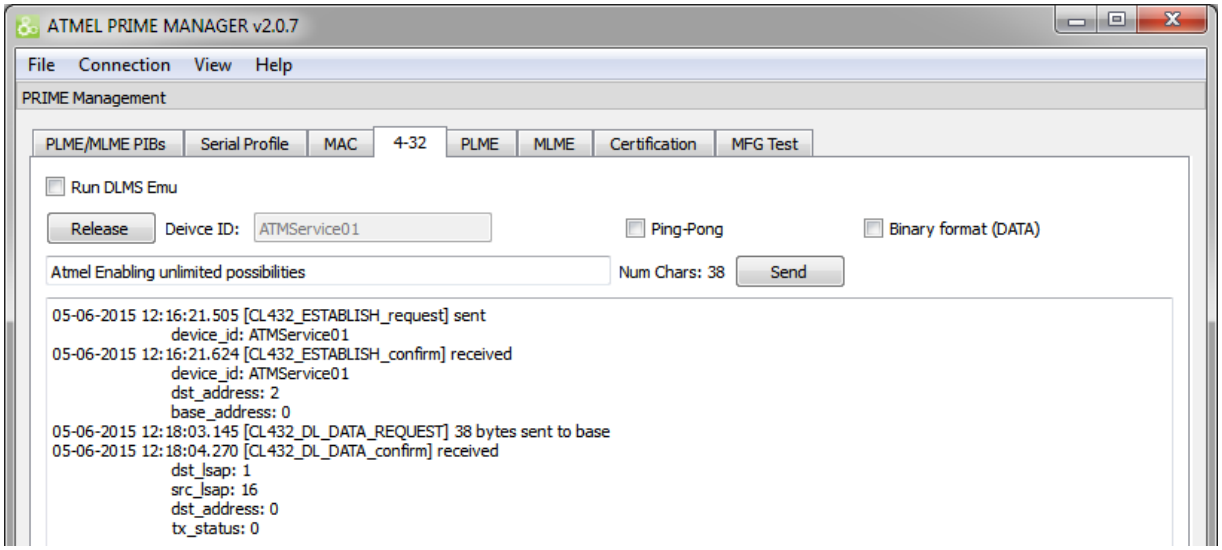
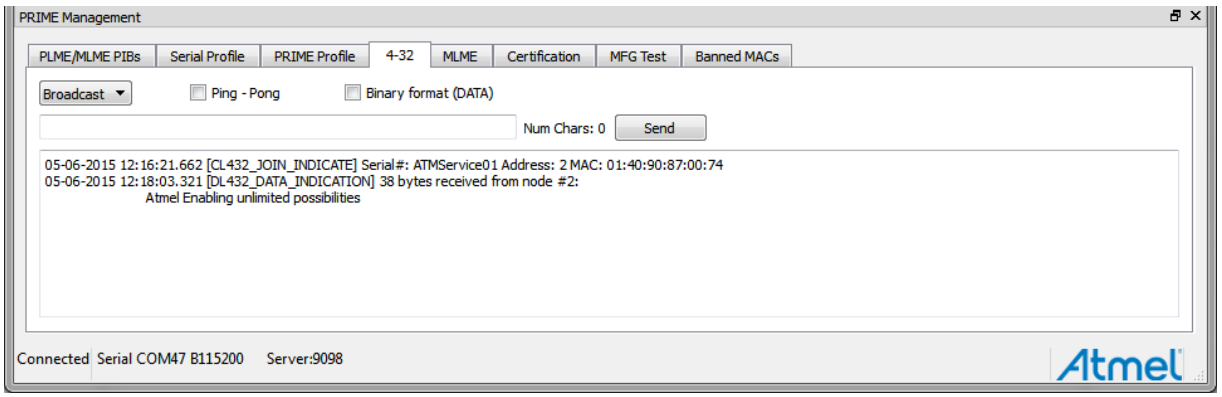


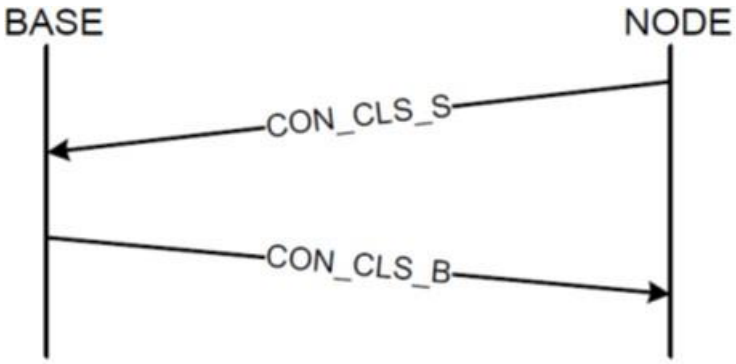
Figure 6-107. Base node tool application window.



6.7.5.3 Closing a 4-32 connection

After the interchange of messages is time to close the 4-32 connection. The PRIME process to close the 4-32 connection is shown in the Figure 6-108.

Figure 6-108. PRIME 4-32 connection close.



Remember, for this example, the Service node requires PLME/MLME protocol, so it has been programed with *prime_service_modem_atpl230amb* project.

Process to close a 4-32 connection with the Service node tool instance should be:

- Select the 4-32 tab.
- Click Release.
- The executed operations appear in the Log output windows of both the Service and Base nodes tool instances. See the figures below.
- If you see the Base node tool instance, you have received the leave indication message and in the Service node window appears the release confirm.

Note: The Service Node must be registered in the Base Node to perform this operation.

Figure 6-109. Service node tool application window.

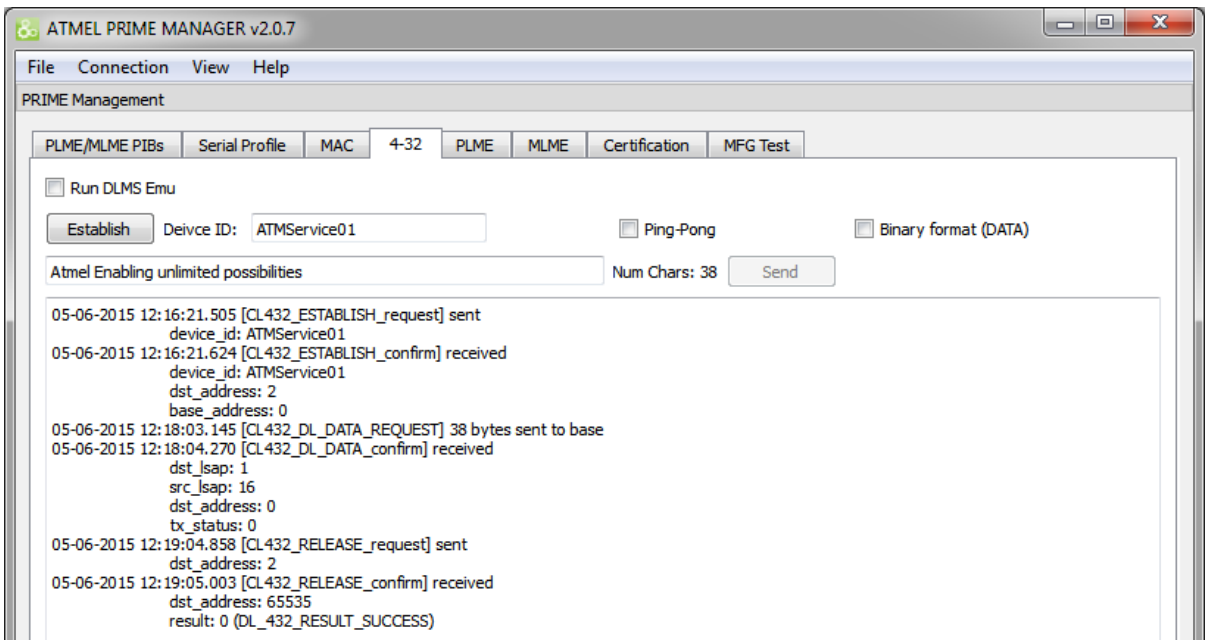
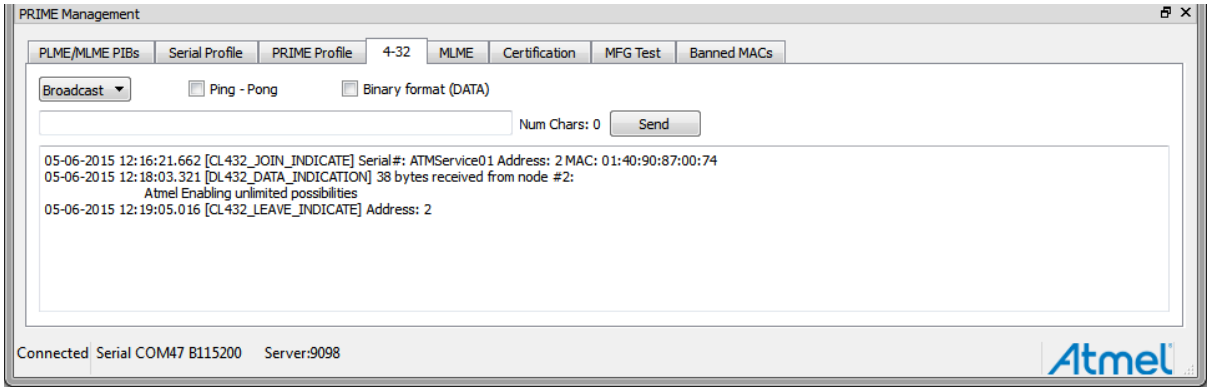


Figure 6-110. Base node tool application window.



6.7.5.4 Using the Certification features tab

Process to use the certification mode with the Atmel PRIME Manager tool for Base Node and Service Node should be:

- Select the Certification tab of Base node instance.
- Select the *PHY* in the combo box.
- Click Set CERT mode.
- Select the Certification tab of Service node instance.
- Select *PHY v1.3.6* in the combo box.
- Click Set CERT mode.
- Select *D8PSK* in the Modulation combo box.
- You can modify the number of messages, attenuation… In this example, we do not modify anything.
- Click Start Transmission.
- The executed operations appear in the Log output windows of both instances.

Notes: 1. The Service Node must be registered in the Base node to perform this operation.

2. For this example, the Service node requires Serial protocol, so it has been programed with *prime_service_modem_atpl230amb* project or *prime_service_dlmsemu_ui.atpl230amb* project.

Figure 6-111. Service node tool application window.

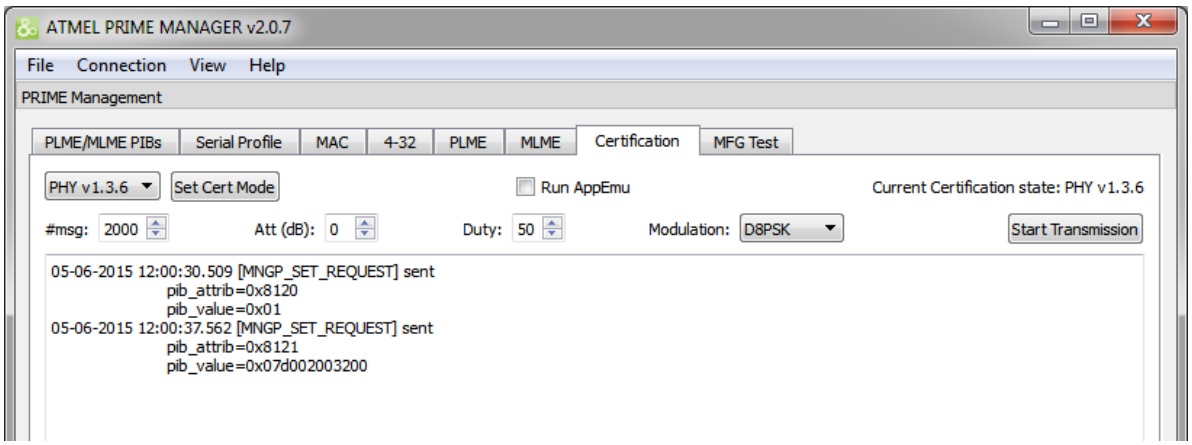
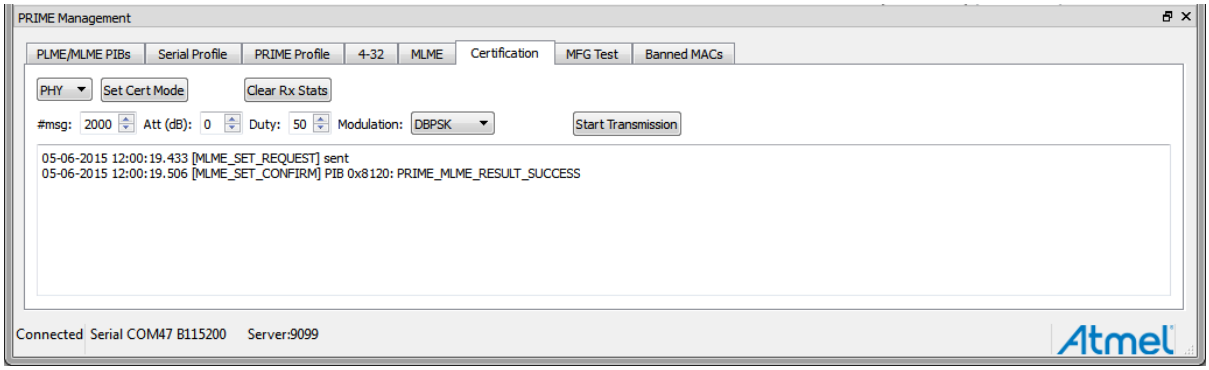


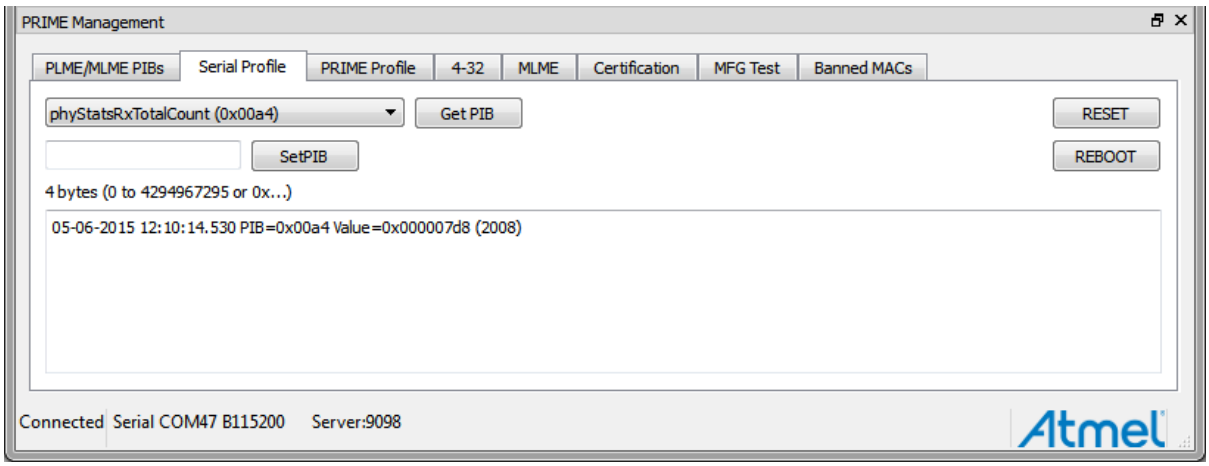
Figure 6-112. Base Node tool window.



Once 2000 frames have been sent (during every transmission, TX led and green led LED0 of Service node board are on) we have to ask to Base node the number of frames received. Process should be:

- Select the Serial profile tab of Base node tool.
- Select the *phyStatsRxTotalCount (0x00a4)* in the combo box.
- Click Get PIB.
- The value of the PIB appears in the Log output window.

Figure 6-113. Base node results.



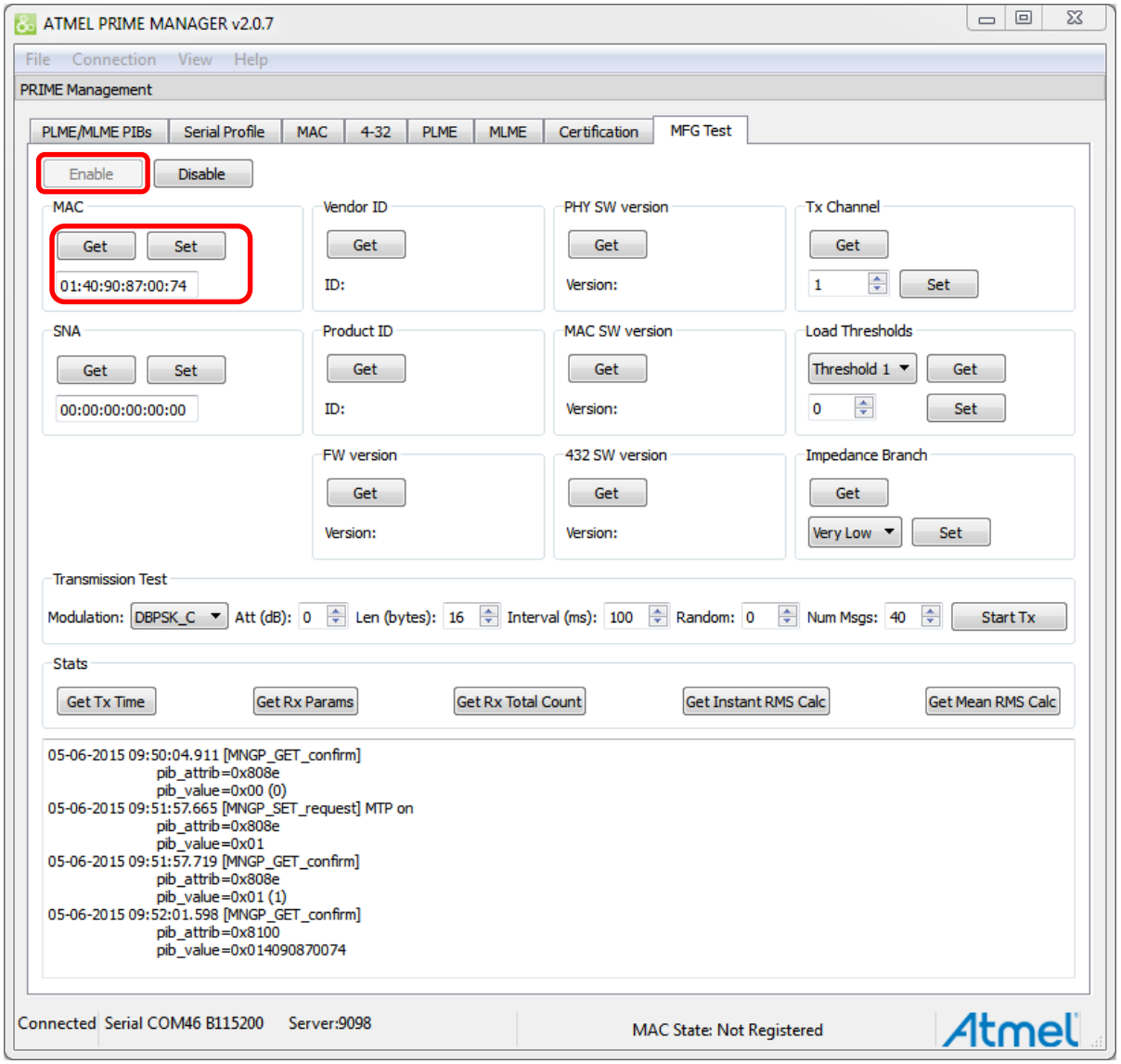
6.7.5.5 Using the Manufacturing Test features tab

Process to use the Manufacturing test mode with the Atmel PRIME Manager tool for Service node should be:

- Select the MFG Test tab.
- Click Enable.
- Click Get MAC.
- The MAC number appears in the MAC box and the executed operation appears in Log output window.
- Write a new MAC number in the box.
- Click Set MAC.
- The executed operation appears in Log output window. See the figure below.
- To finish, click Disable.

Note: For this example, the Service node requires Serial protocol, so it has been programed with *prime_service_modem_atpl230amb* project or *prime_service_dlmsemu_ui.atpl230amb* project.

Figure 6-114. Setting the MAC with the Service node tool application.



To uninstall the Atmel PRIME Manager tool from your computer, go to Start > All Programs > ATMEL > Atmel PRIME Manager vX.Y.Z > Uninstall.

For further information of the Atmel PRIME Manager tool, please refer to the tool's embedded help (in the menu bar).

7. References

[1] CENELEC, EN 50065-1. Signaling on low-voltage electrical installations in the frequency range 3 kHz to 148.5 kHz.

[2] FCC Part 15 Subpart B.

[3] doc43053: ATPL230A Datasheet, 2015.

[4] doc11102: SAM4C Series Datasheet, 2014.

[5] doc43085: Atmel PRIME Firmware Stack User Guide, 2015.

[6] PRIME Specification: PRIME-Spec_R1.3.6.1, 2014.

[7] doc43052: PLC coupling reference designs, 2015

[8] doc43072: PHY Performance Verification, 2015.

Appendix A. Board schemes

A.1 ATPL230AMBv4 schemes

This section contains the schemes of the ATPL230AMB multipurpose board:

- Top level scheme.
- Power supply scheme.
- ATPL230A transceiver.
- PLC Coupling transmission scheme.
- PLC Coupling reception scheme.
- ATSAM4C16C MCU.
- Peripherals.
- Interface.
- Components location in top layer.

Figure A-1. ATPL230AMB Top level scheme.

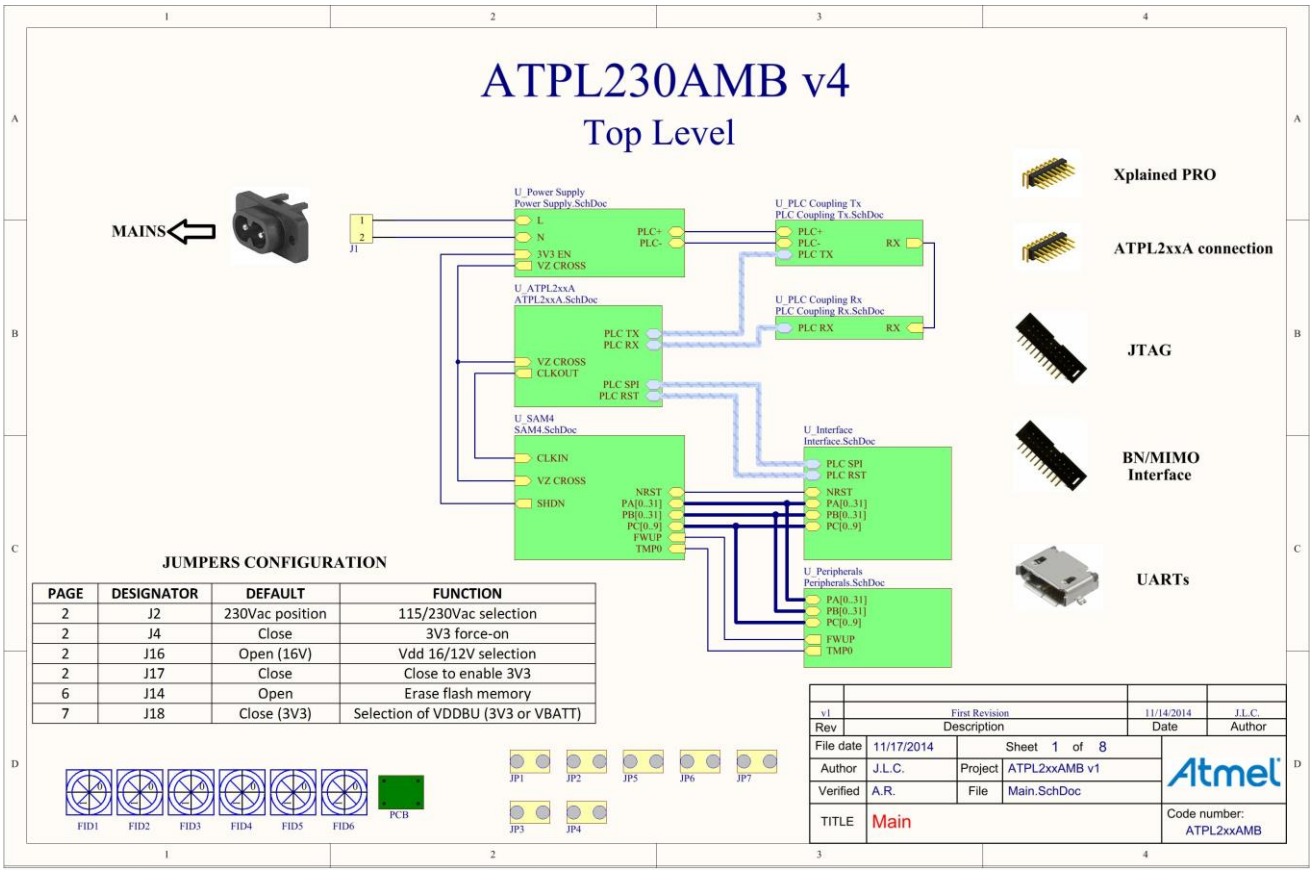


Figure A-2. ATPL230AMB Power supply scheme.

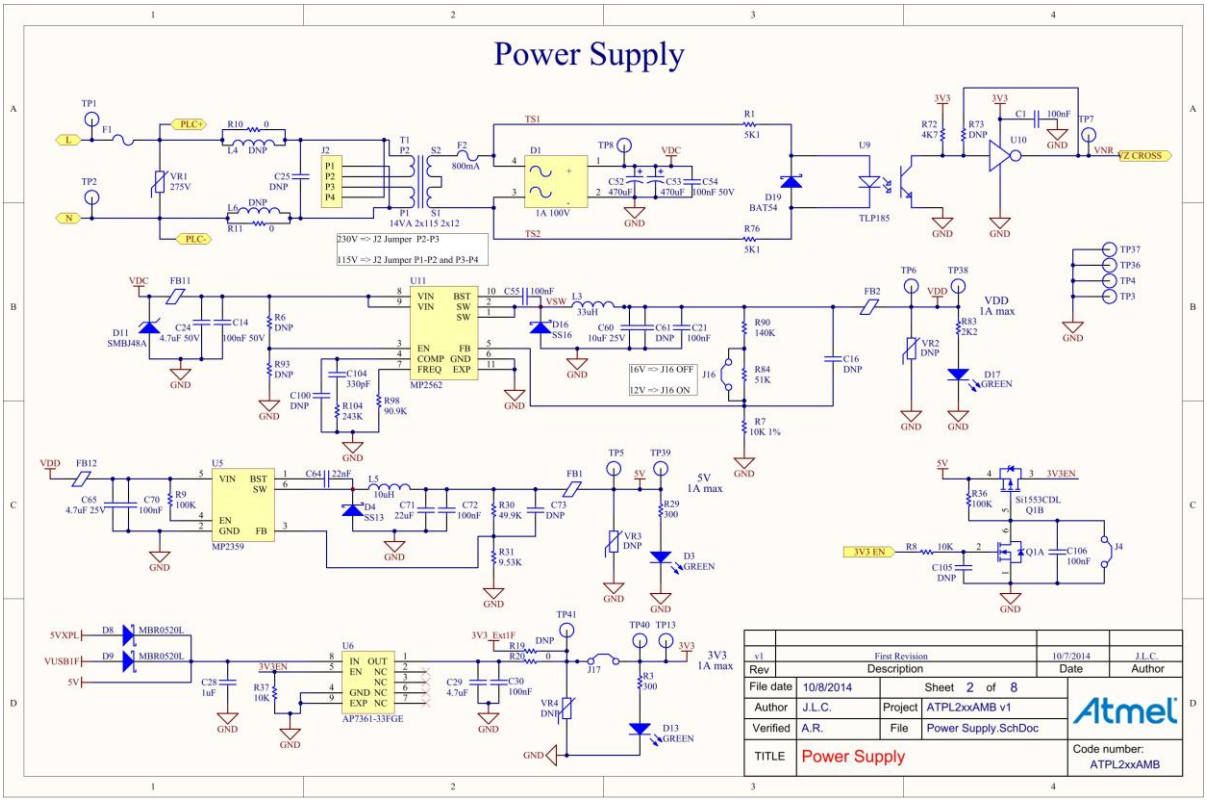


Figure A-3. ATPL230AMB Transceiver.

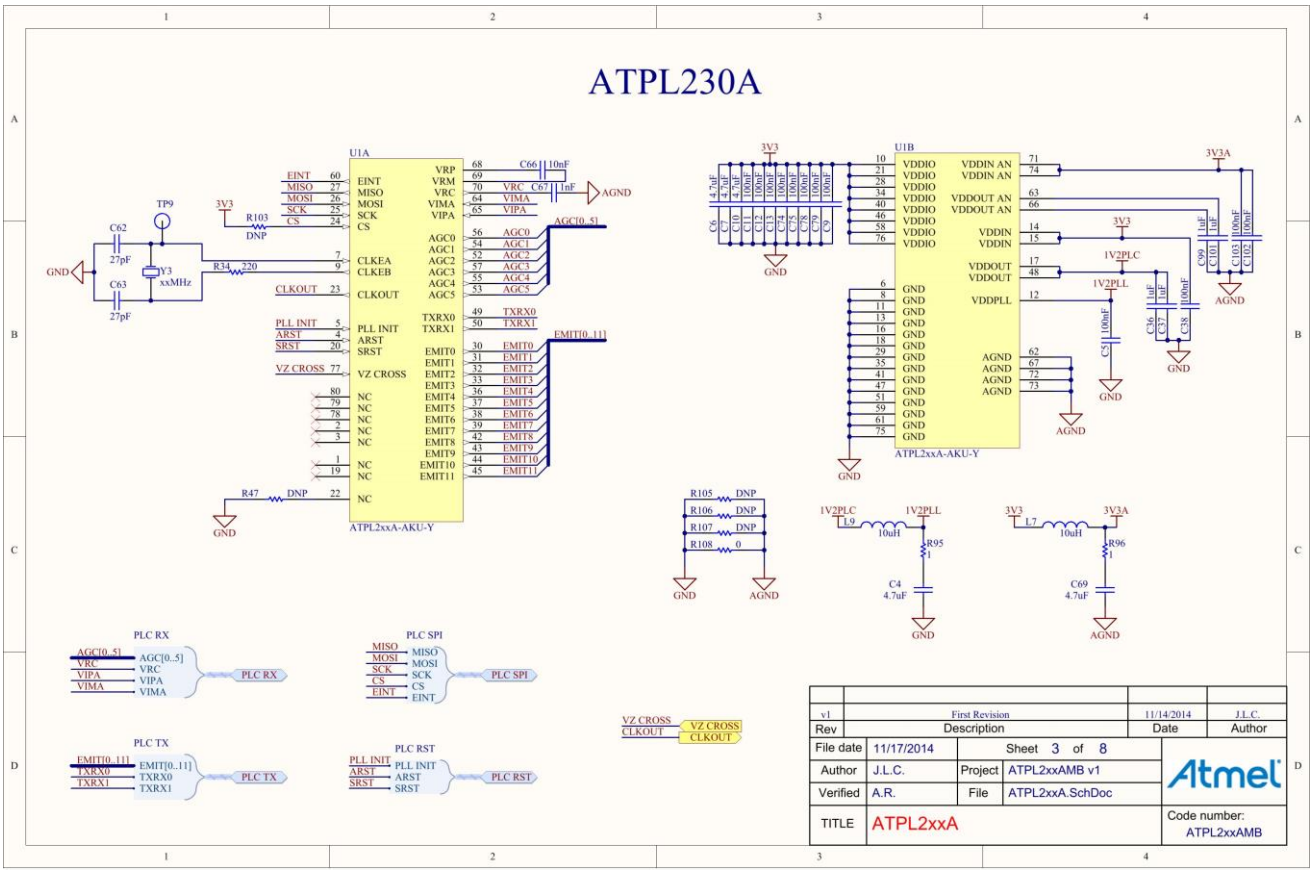


Figure A-4. ATPL230AMB PLC Coupling Transmission scheme.

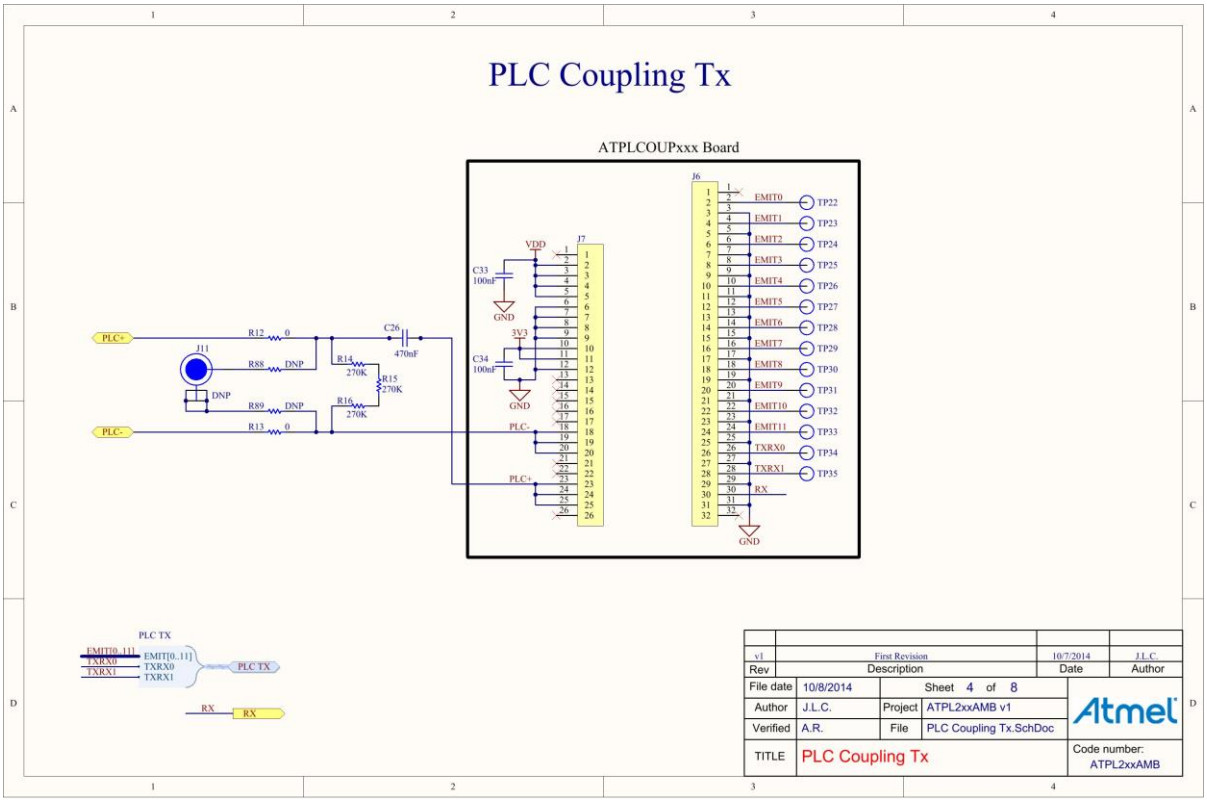


Figure A-5. ATPL230AMB PLC Coupling Reception scheme.

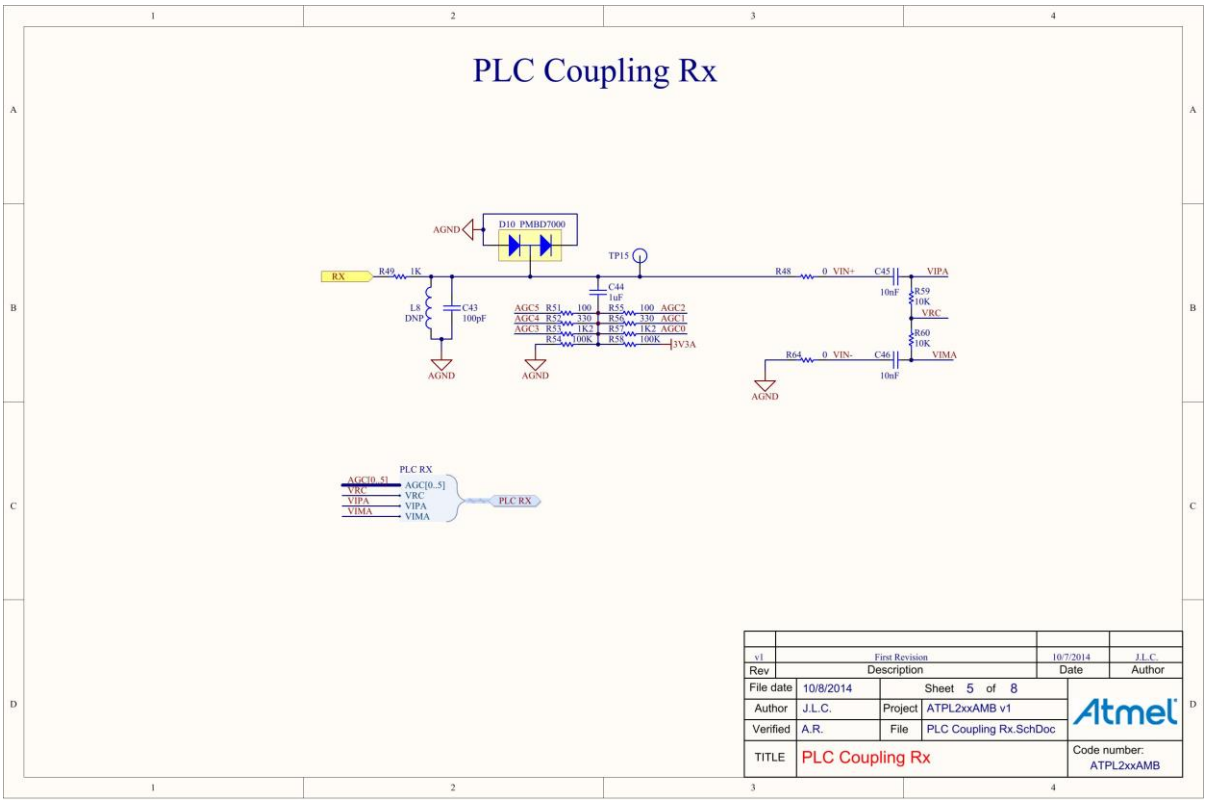


Figure A-6. SAM4C MCU.

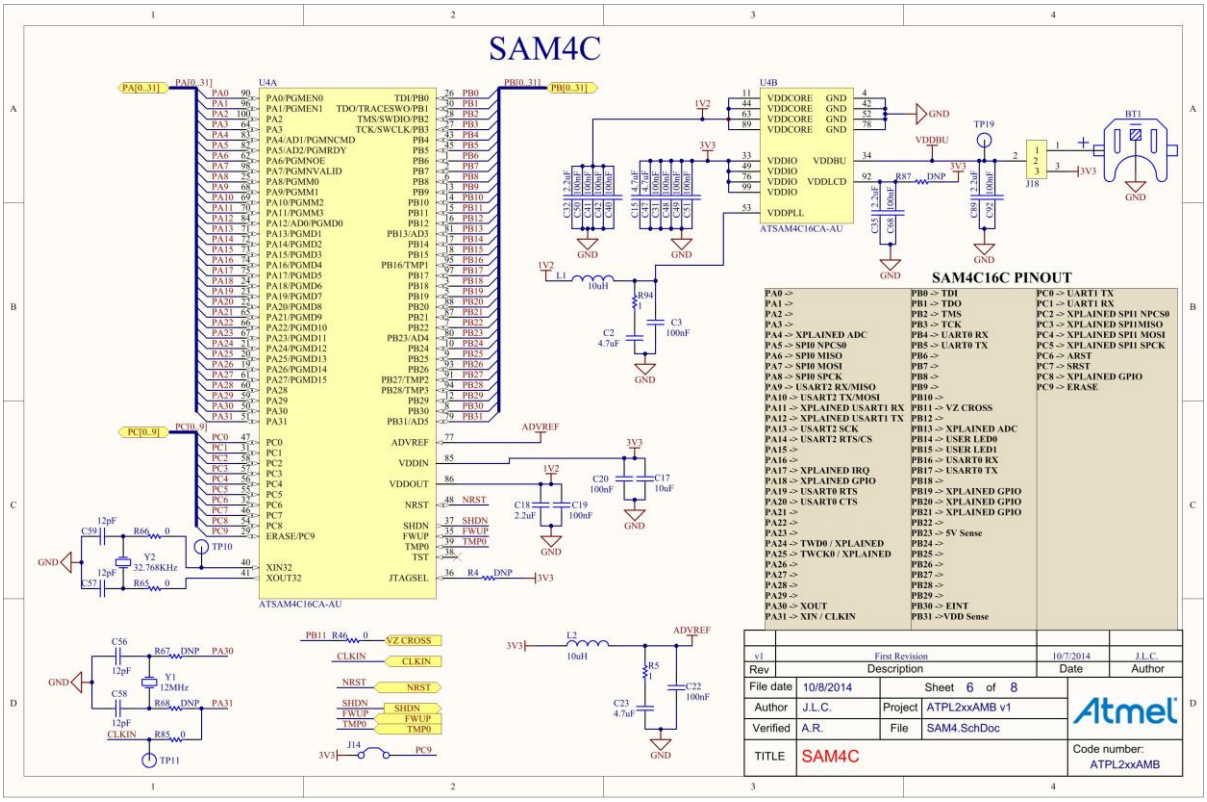


Figure A-7. ATPL230AMB Peripherals.

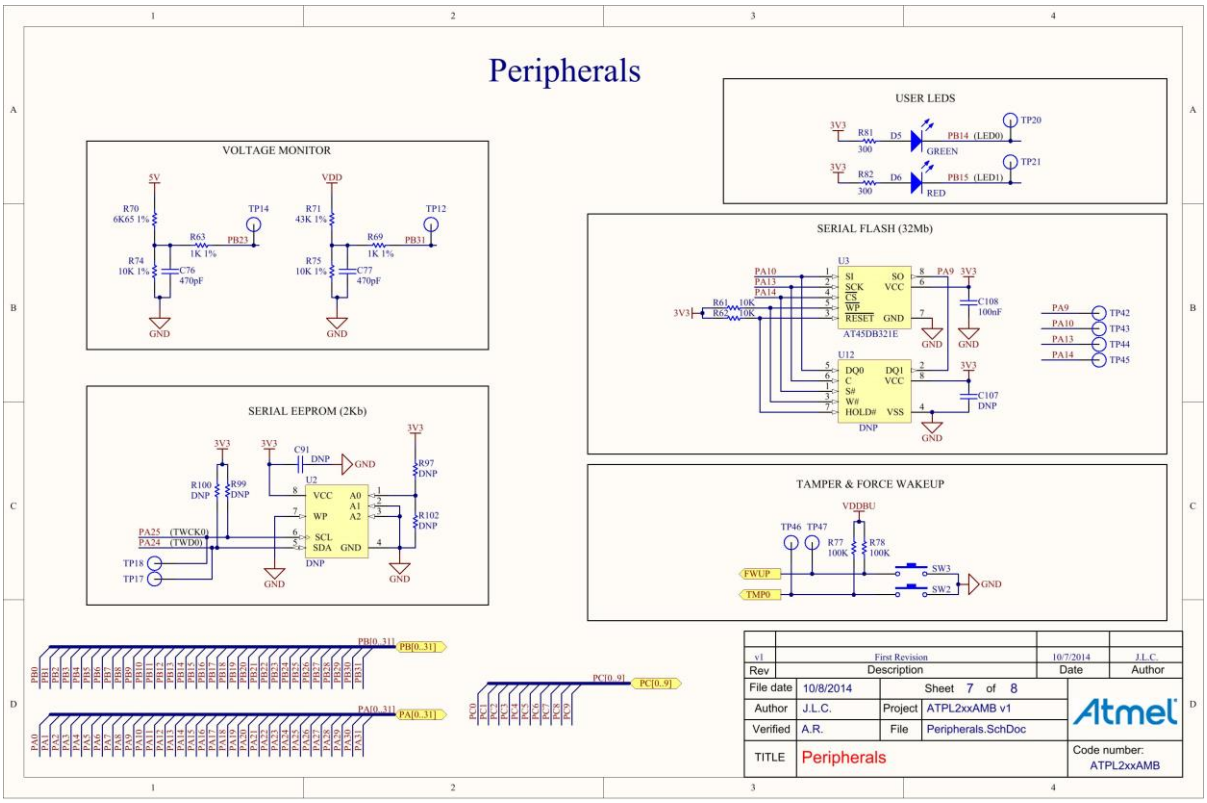


Figure A-8. ATPL230AMB Interface.

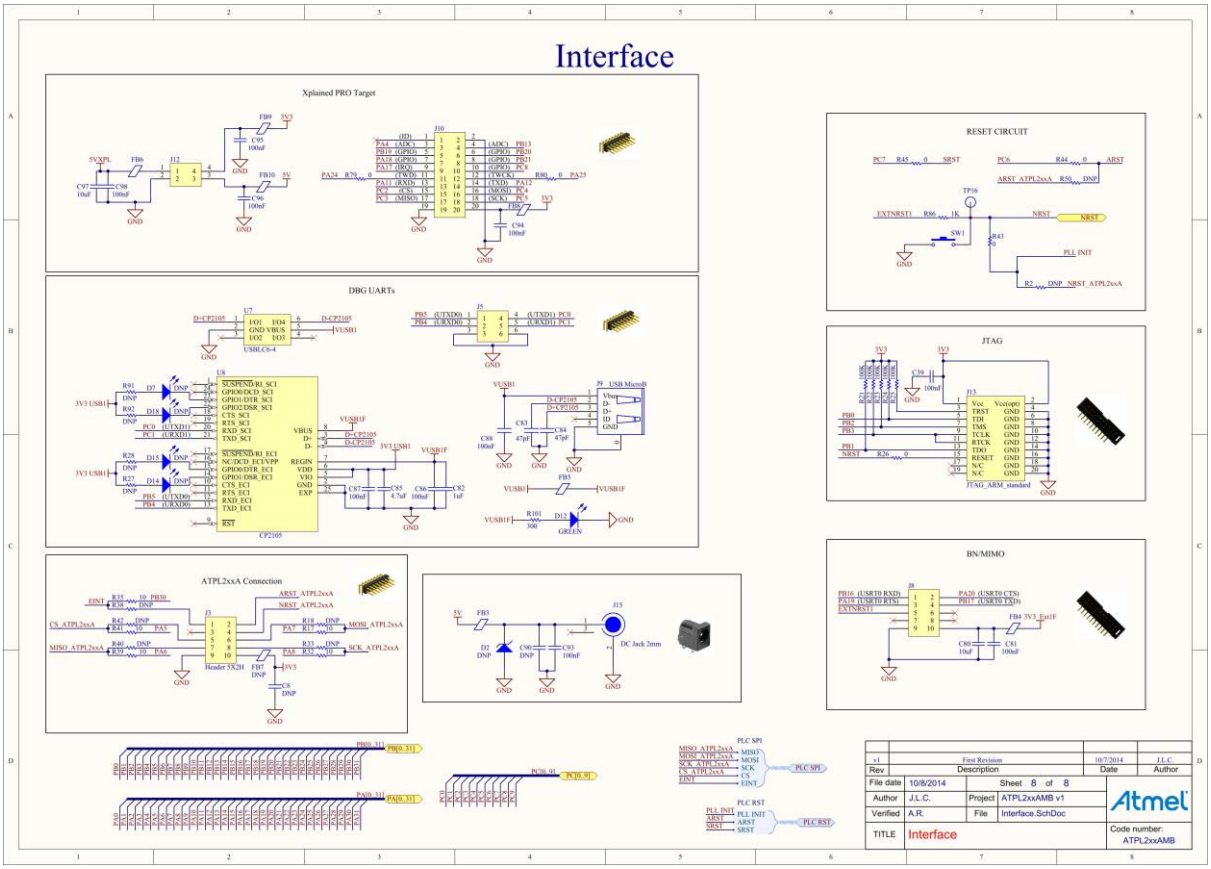
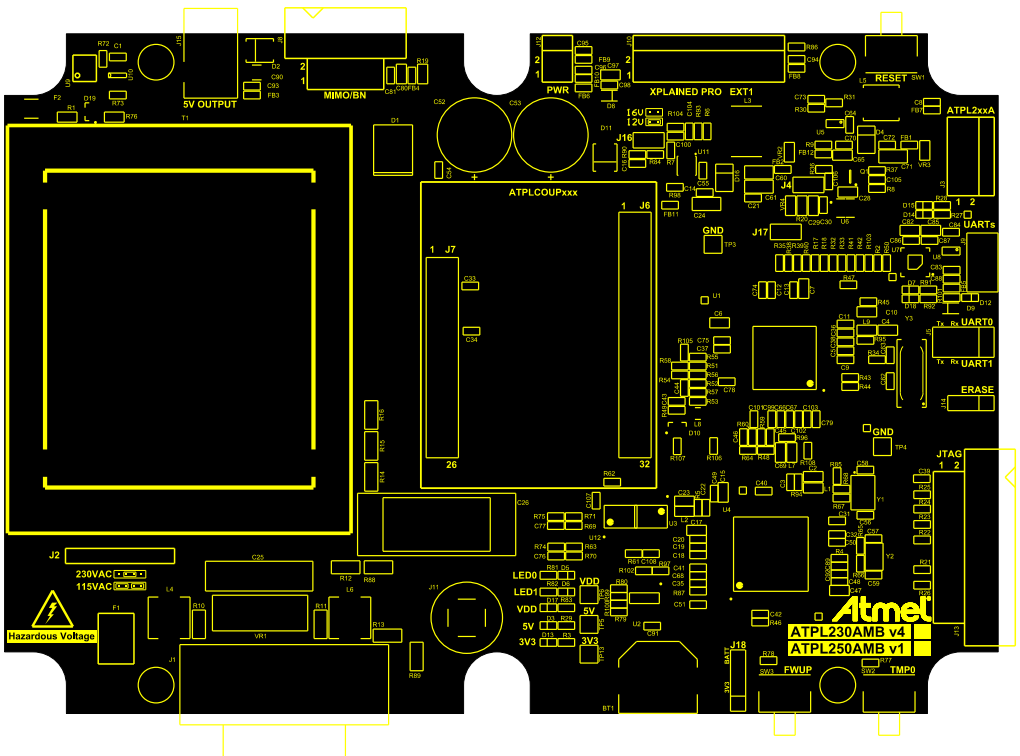


Figure A-9. ATPL230AMB components location in top layer.



A.2 ATPLCOUP001v1 schemes

This section contains the schemes of the ATPLCOUP001 PLC Coupling board:

- PLC Coupling transmission scheme.
- Components location in top and bottom layers.

Figure A-10. ATPLCOUP001 PLC Coupling transmission scheme.

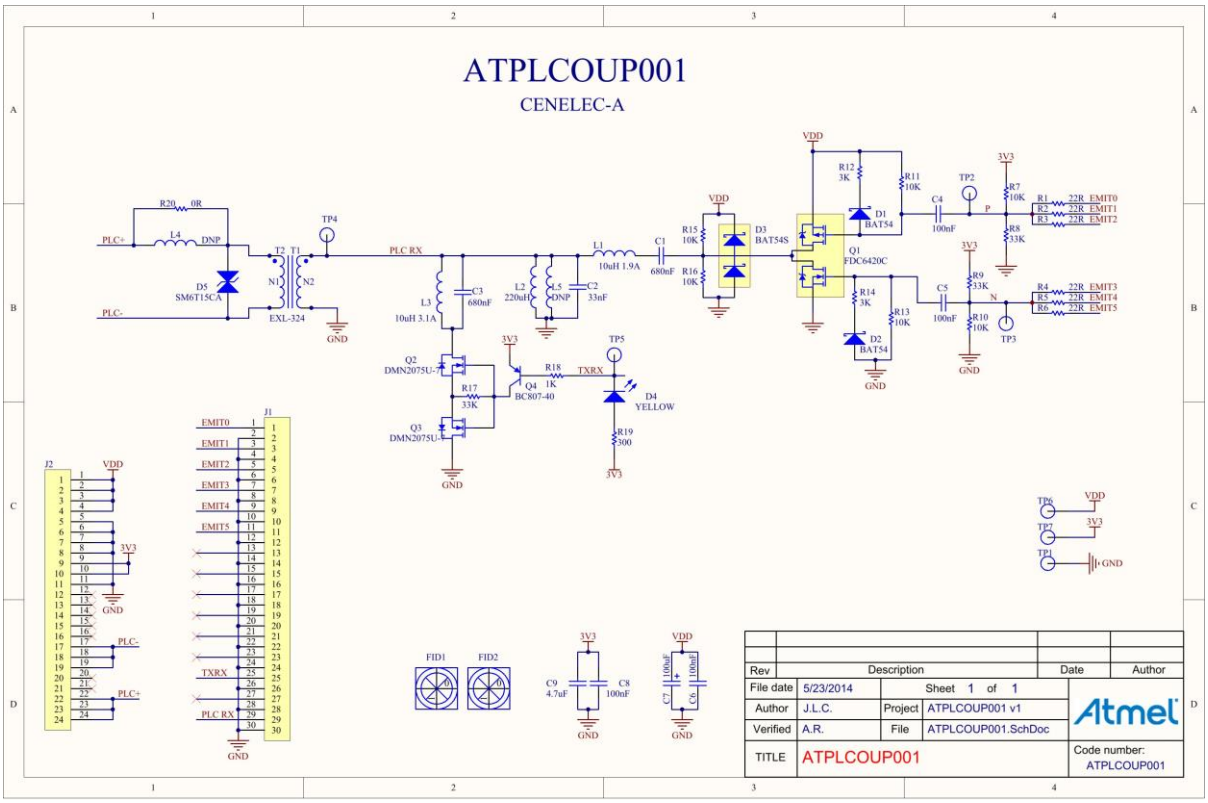
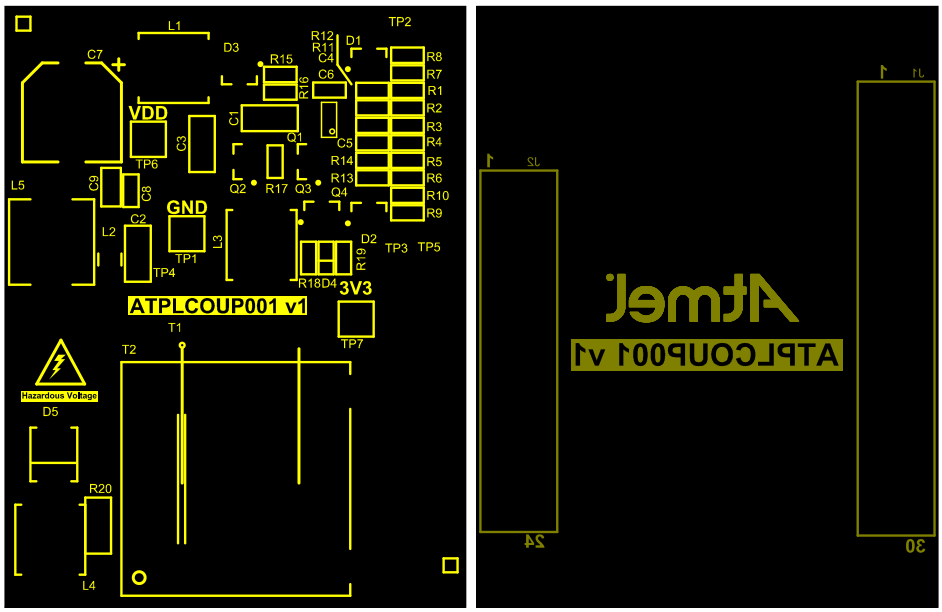


Figure A-11. ATPLCOUP001 component location in top and bottom layers.



ATMEL EVALUATION BOARD/KIT IMPORTANT NOTICE AND DISCLAIMER

This evaluation board/kit is intended for user's internal development and evaluation purposes only. It is not a finished product and may not comply with technical or legal requirements that are applicable to finished products, including, without limitation, directives or regulations relating to electromagnetic compatibility, recycling (WEE), FCC, CE or UL. Atmel is providing this evaluation board/kit “AS IS” without any warranties or indemnities. The user assumes all responsibility and liability for handling and use of the evaluation board/kit including, without limitation, the responsibility to take any and all appropriate precautions with regard to electrostatic discharge and other technical issues. User indemnifies Atmel from any claim arising from user's handling or use of this evaluation board/kit. Except for the limited purpose of internal development and evaluation as specified above, no license, express or implied, by estoppel or otherwise, to any Atmel intellectual property right is granted hereunder. ATMEL SHALL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RELATING TO USE OF THIS EVALUATION BOARD/KIT.

ATMEL CORPORATION
1600 Technology Drive
San Jose, CA 95110
USA

Revision History

Doc Rev.	Date	Comments
43075E	10/2015	Updating PRIME software version.
43075D	06/2015	Updating PRIME software version and data information.
43075C	03/2015	Updating kit's software project examples.
43075B	11/2014	Updating kit's boards.
43075A	04/2014	Initial document release.

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**