AK-LINK

Reference manual

Contents

About this document

Revision history

The table below displays the revision history for the chapters in this manual.

Chapter	Date	Revision	Changes made
All	November 2011	1.0	First publication

Contact information

For the latest news, upgrades and information about Artekit products, visit the Artekit web site at http://www.artekit.eu
For technical support on this product, visit the support page at http://www.artekit.eu/support
For additional information about Artekit products, consult the sources below.

Information type	Resource
Technical support	support@artekit.eu
Literature	www.artekit.eu
Sales	sales@artekit.eu
Products forum	www.artekit.eu

Life support policy

Artekit Italy products are not indented or authorized for use as critical components in life support devices or systems without the express written approval from Artekit Italy. Those devices may include devices for supporting or sustaining life, devices for surgical implant into the body or any other device whose failure to perform correctly could result in life support failure.

Copyright information

Specifications

Product description

The Artekit AK-LINK board is a versatile, low-cost, small-footprint JTAG adapter, OpenOCD compatible. Based on the popular FTDI ® FT2232D chip, the AK-LINK JTAG is powered directly by USB,

The AK-LINK is capable of sourcing TCK to up 6 MHz, and supports 12Mb USB 2.0 (USB 1.1 compatible).

The AK-LINK autosenses the target voltage and adjust its output levels from 1.65V to 5.5V, supporting all JTAG compatible development boards. It uses a standard 20-pin JTAG connector.

You may visit the Artekit website at www.artekit.eu to download the latest documentation and software.

Main components

The AK-LINK board has the following main components:

- Mini USB connector.
- Standard 20-pin JTAG connector
- Power ON LED, Activity yellow LED and Activity green LED

Environmental requirements

The AK-LINK board must be stored between -40° C and +100° C. The recommended operating temperature is between 0° C and +70° C.

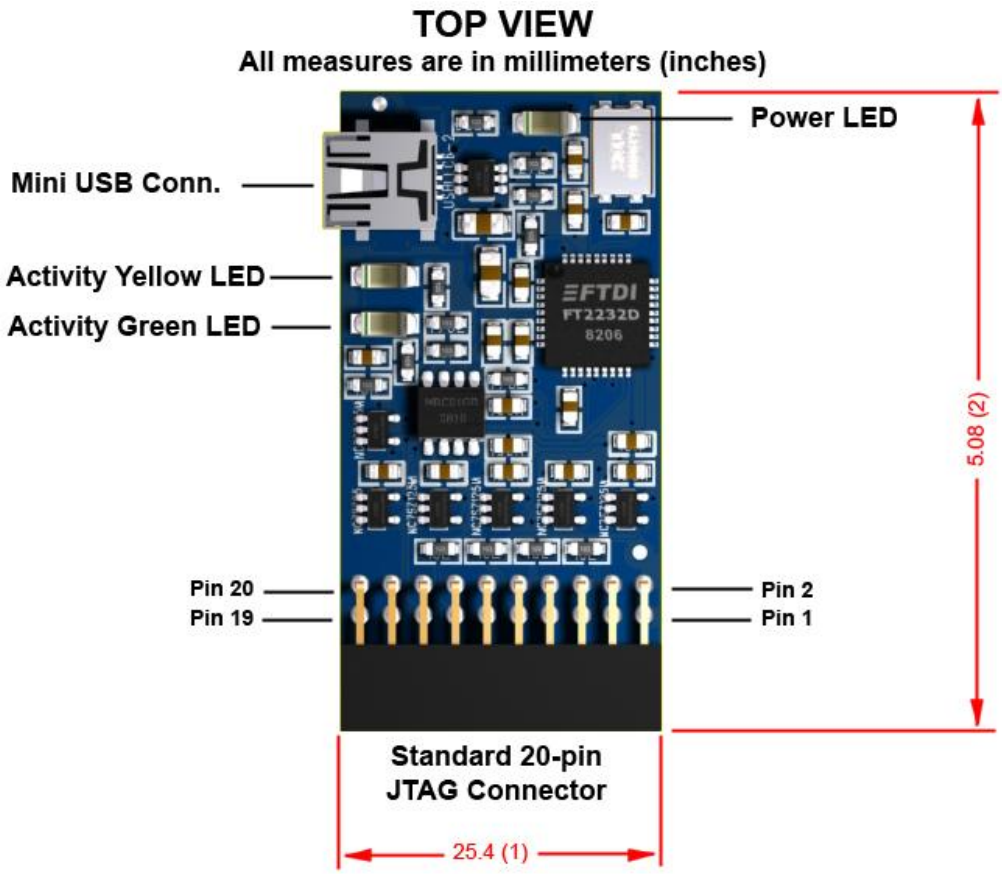
The AK-LINK board may be damaged without proper anti-static handling.

Handling the board

When handling the board, it is important to observer the following precaution:

Static discharge precaution – Without proper anti-static handling the board can be damaged. Therefore, take anti-static precautions when handling the board.

Board overview



Pin description

1	VCC	Vcc from Target.
2	VCC	Vcc from Target.
3	NC	No connection (TRST in standard JTAG connector).
4	GND	Ground.
5	OUTPUT	JTAG TDI.
6	GND	Ground.
7	OUTPUT	JTAG TMS.
8	GND	Ground.
9	OUTPUT	JTAG TCLK.
10	GND	Ground.
11	NC	No connection (RTCK in standard JTAG connector).
12	GND	Ground.
13	INPUT	JTAG TDO.
14	GND	Ground.
15	OUTPUT	Target reset.
16	GND	Ground.
17	NC	No connection.
18	GND	Ground.
19	NC	No connection.
20	GND	Ground.

Electrical characteristics

Test conditions

Unless specified, all voltages are referenced to GND.

Minimum and maximum values

Unless otherwise specified, the minimum and maximum values are guaranteed in the worst conditions of ambient temperature, supply voltage and frequencies by tests in production on the 100% of the devices with an ambient temperature `TA = 25 °C`.

Typical values

Unless otherwise specified, typical data are based on TA = 25 °C, VCC = 5V. They are given only as design guidelines and are not tested.

Absolute maximum ratings

WARNING Exceeding values beyond these absolute maximum values may cause permanent damage to the device and/or board. Operating at absolute maximum rating conditions for extended periods may affect the device reliability.

Symbol	Ratings	Min.	Max.	Unit
VCC-GND	External main supply voltage.	-0.5	6	V
Vin	Voltage on input pins.	GND	6	

Current consumption in operating mode

Symbol	Parameter	Max.	Unit
Icc	Supply current	50	mA

Normal operating parameters

Symbol	Parameter	Value	Unit
Vcc	Power supply applied to VCC through USB cable	5	V

Software

AK-LINK OpenOCD Installation (Windows)

Download the latest version of the OpenOCD debugger at the Artekit website (http://www.artekit.eu).

The Artekit AK-LINK installation comprises:

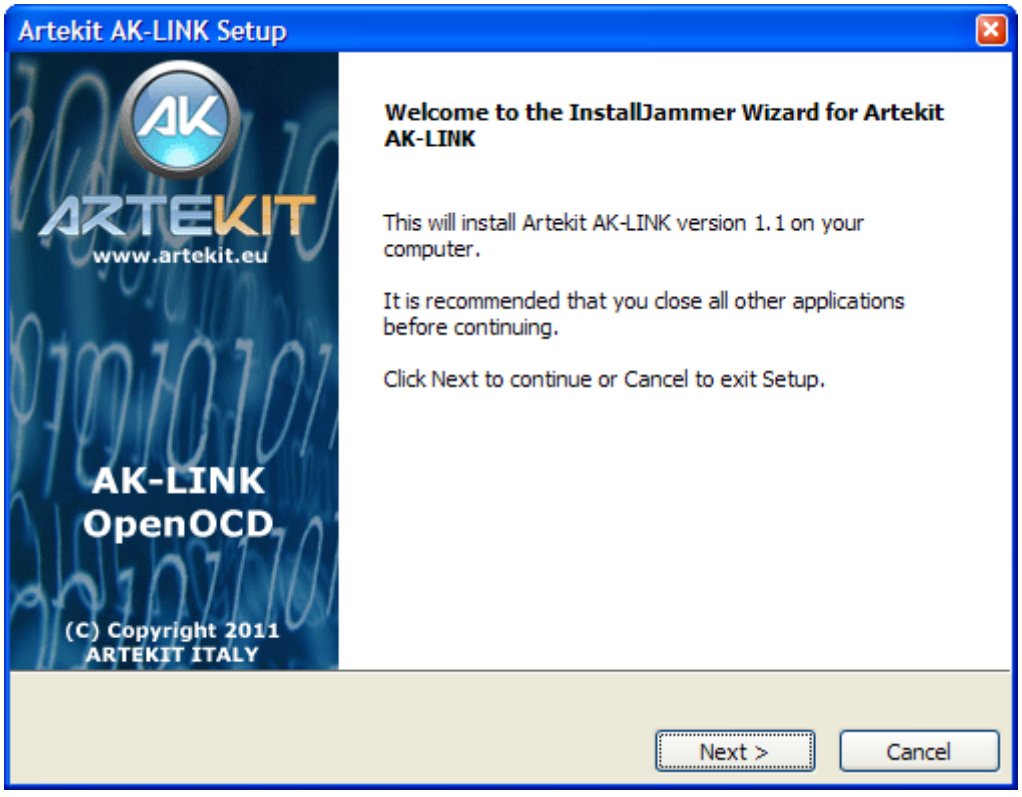
- OpenOCD version 6, for Windows.
- OpenOCD version 5, for Windows.

In order to drive the communication between the PC and the AK-LINK, you must download the latest FTDI D2XX driver for you platform, from the FTDI website (http://www.ftdichip.com/FTDrivers.htm).

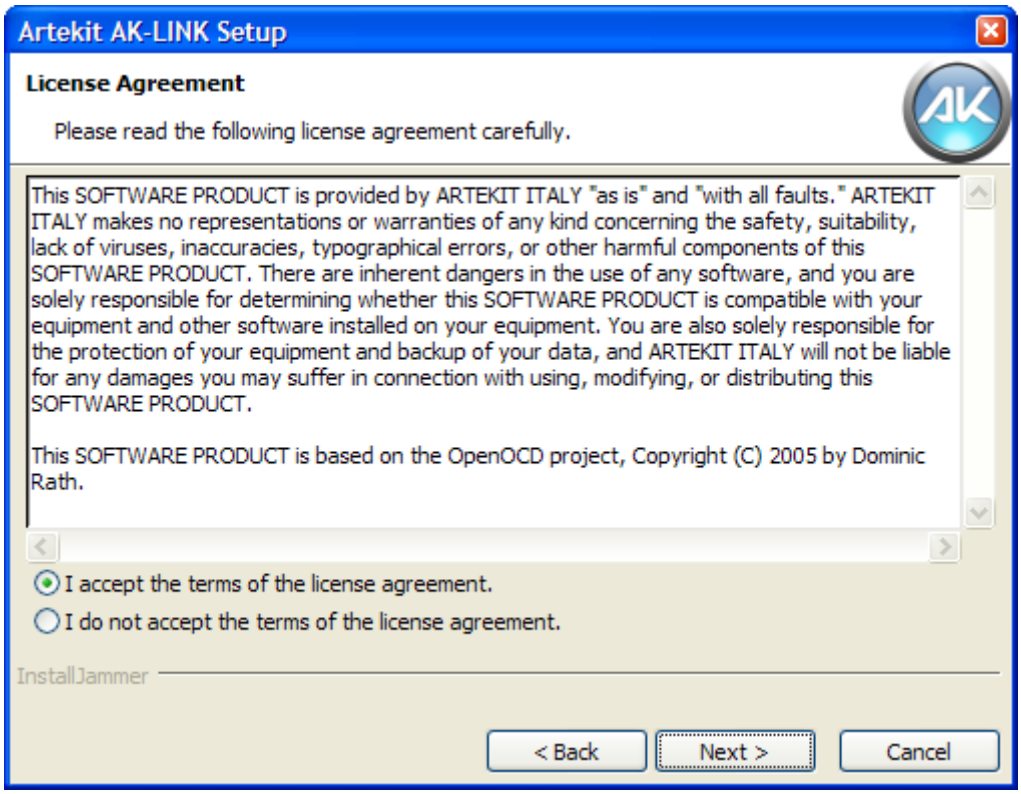
Note that the FTDI driver is not included in the AK-LINK installation distribution and must be downloaded separately. Follow the FTDI installation procedure to install the FTDI D2XX drivers.

After installing the FTDI drivers double click the downloaded AK-LINK installer and follow the installation steps.

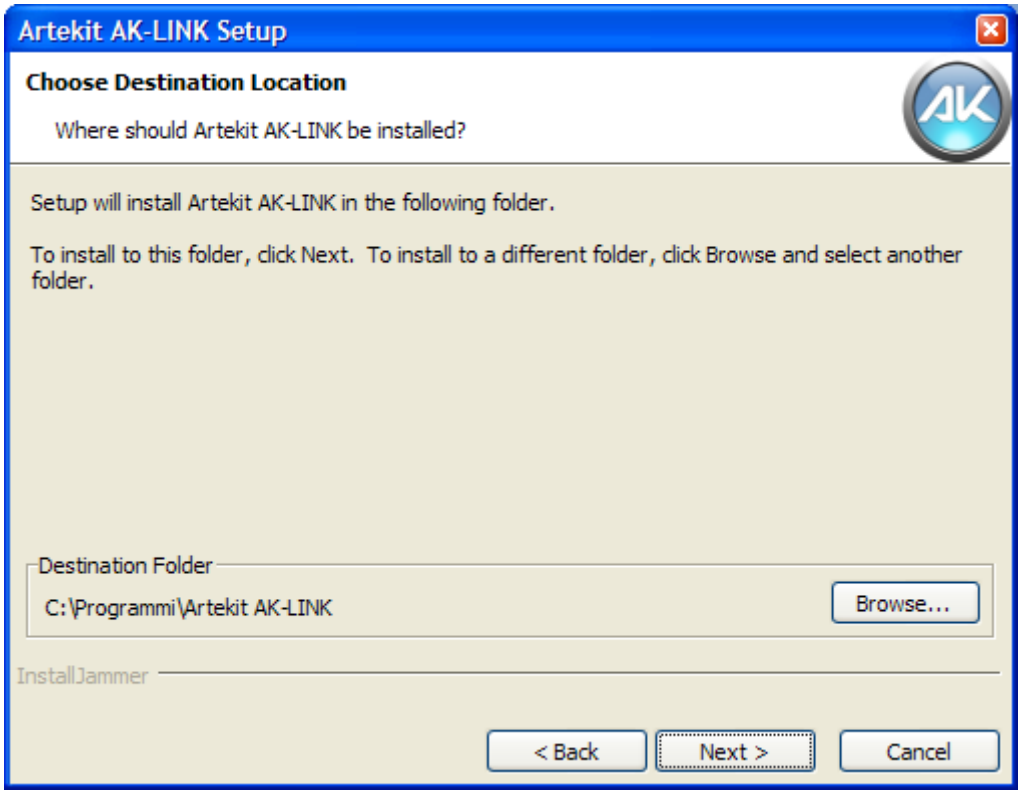
- Click next to continue.



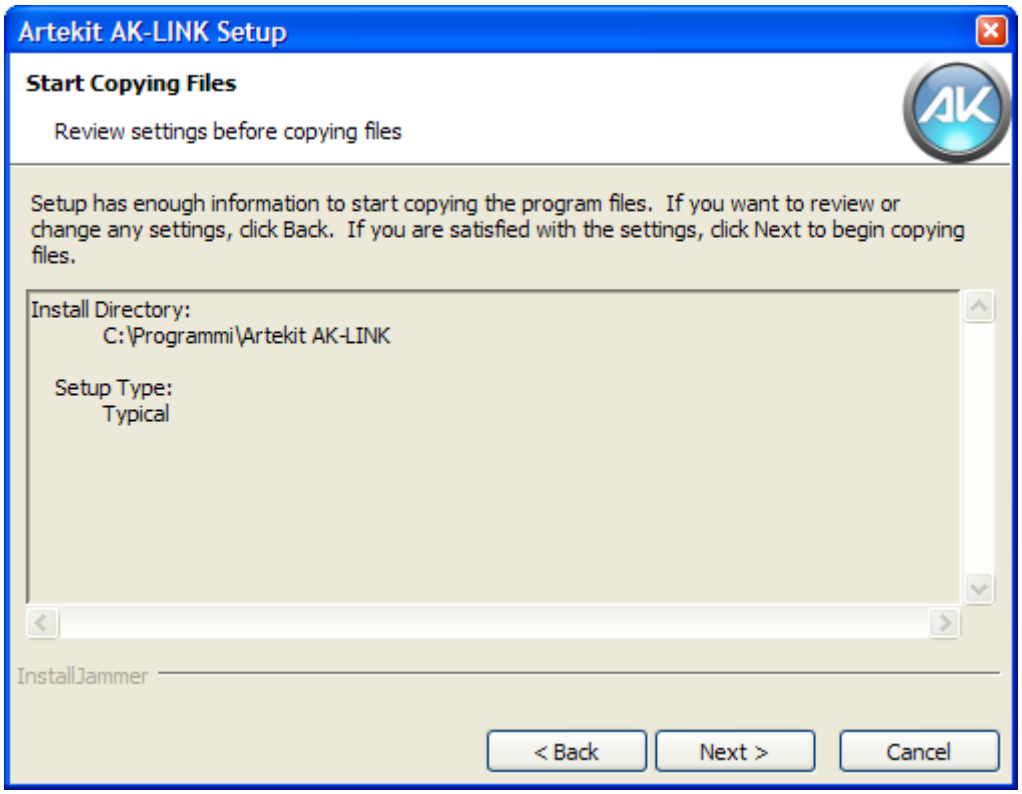
- Read and accept the license agreement, then click “Next”



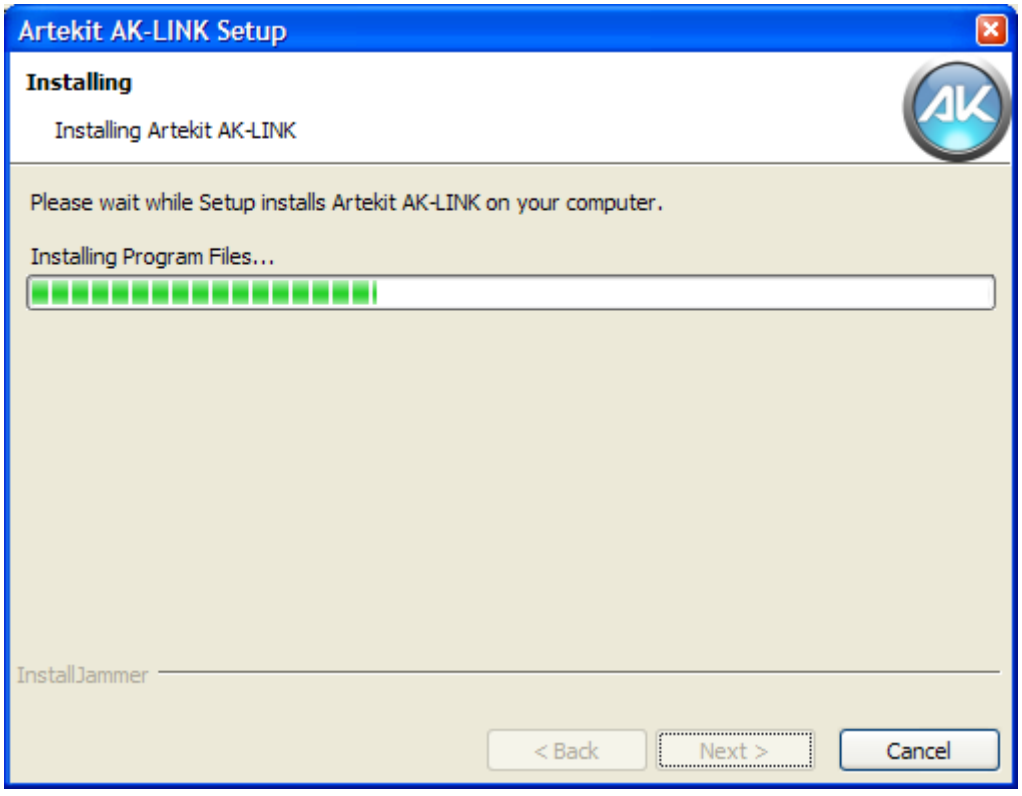
- Choose a destination folder to place the AK-LINK software, then click “Next”



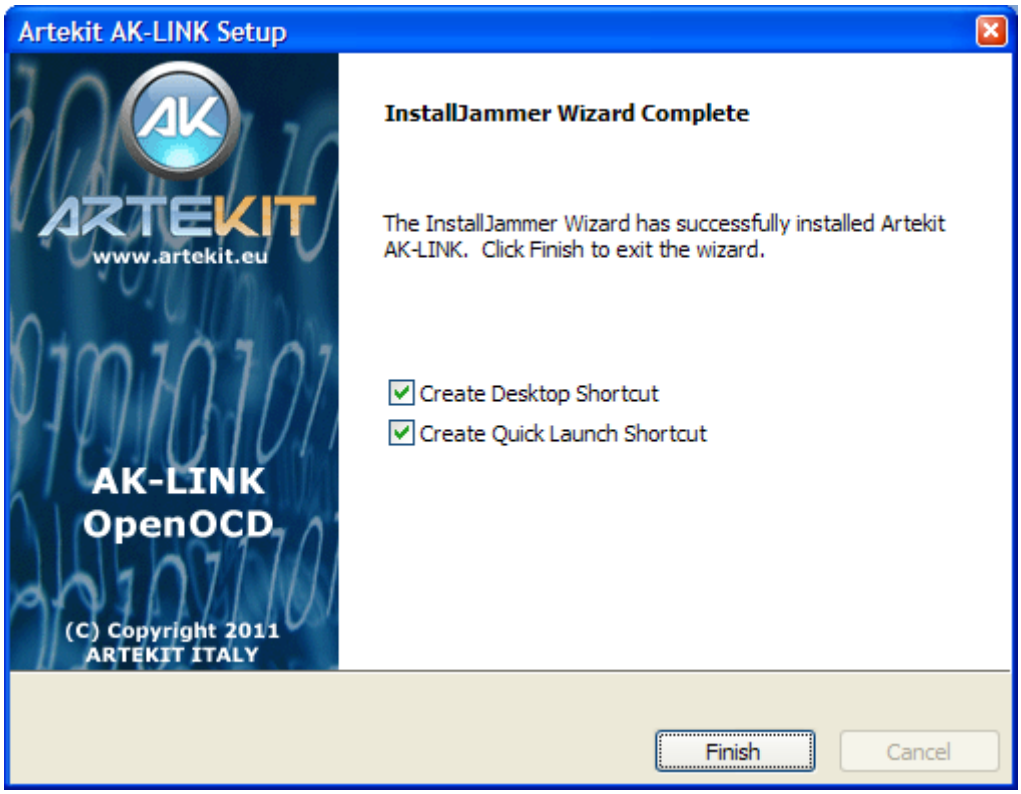
- Verify the installation options, and then click “Next”.



- Wait for the installation to finish. It should take just a couple of seconds.

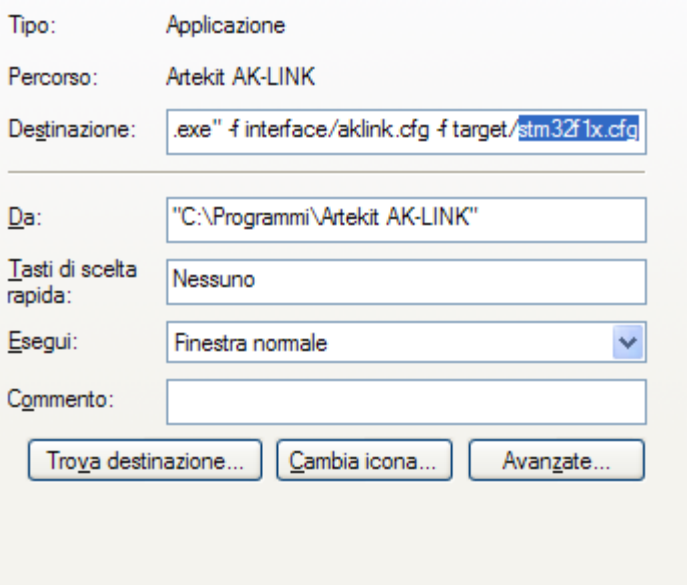


- Choose whether to create a desktop and/or a quick launch icon, then click “Finish” to finish the installation.



Running OpenOCD

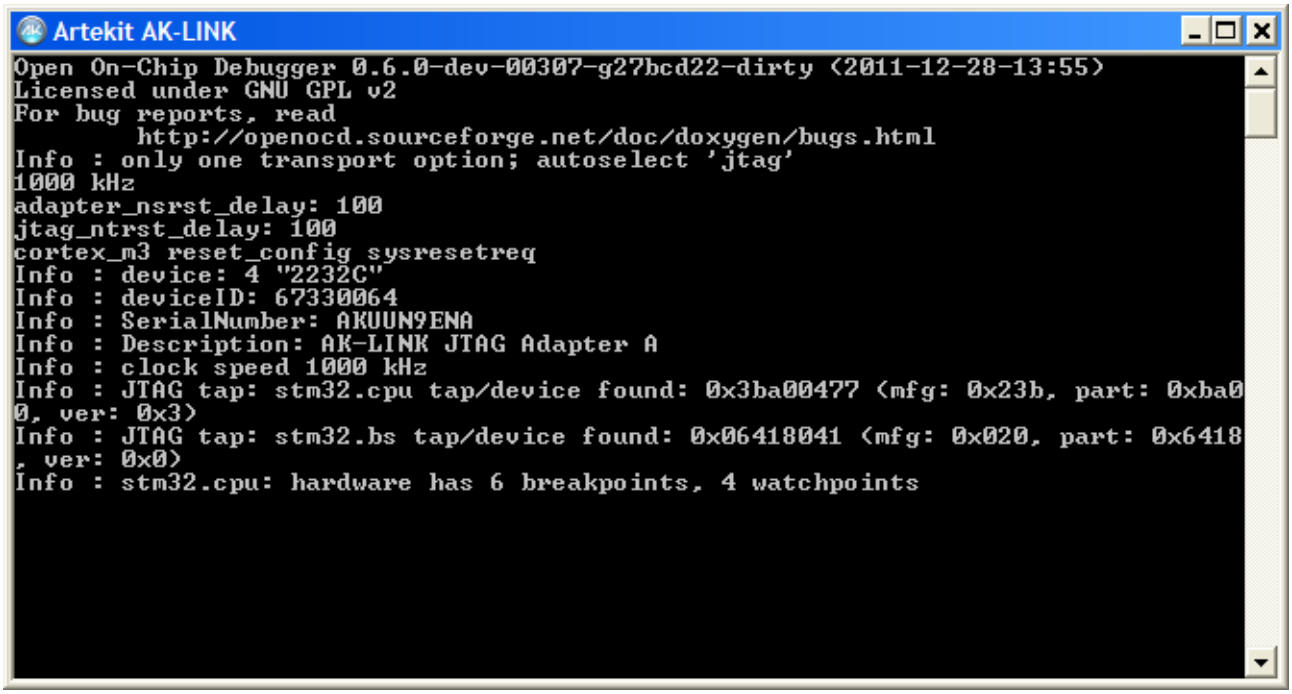
By default the installer will create the desktop, quick-launch and program menu icons to use the AK-LINK to debug a STM32F1x device. Whenever you may want to change the target, just edit the shortcuts in the following way:



Right-click the shortcut and choose “Properties”, then change the text highlighted in the picture above with the name of a file located in the “target” folder, in the installation directory. For example: stm32f2x.cfg.

Now you can plug the AK-LINK to your target, power on your board and connect the AK-LINK to the PC through a USB cable.

Launch the Artekit AK-LINK OpenOCD application. A window like the following should appear, indicating that the target is now attached.



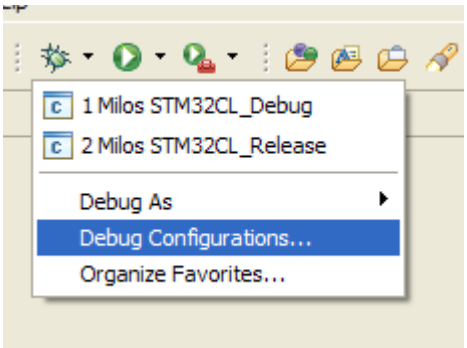
By default, the shortcuts point to the version 6 of OpenOCD. Whenever you may find incompatibilities, a version 5 of OpenOCD is also provided. Check on the Programs menu, under the Artekit AK-LINK group of programs.

Example debug session with Eclipse

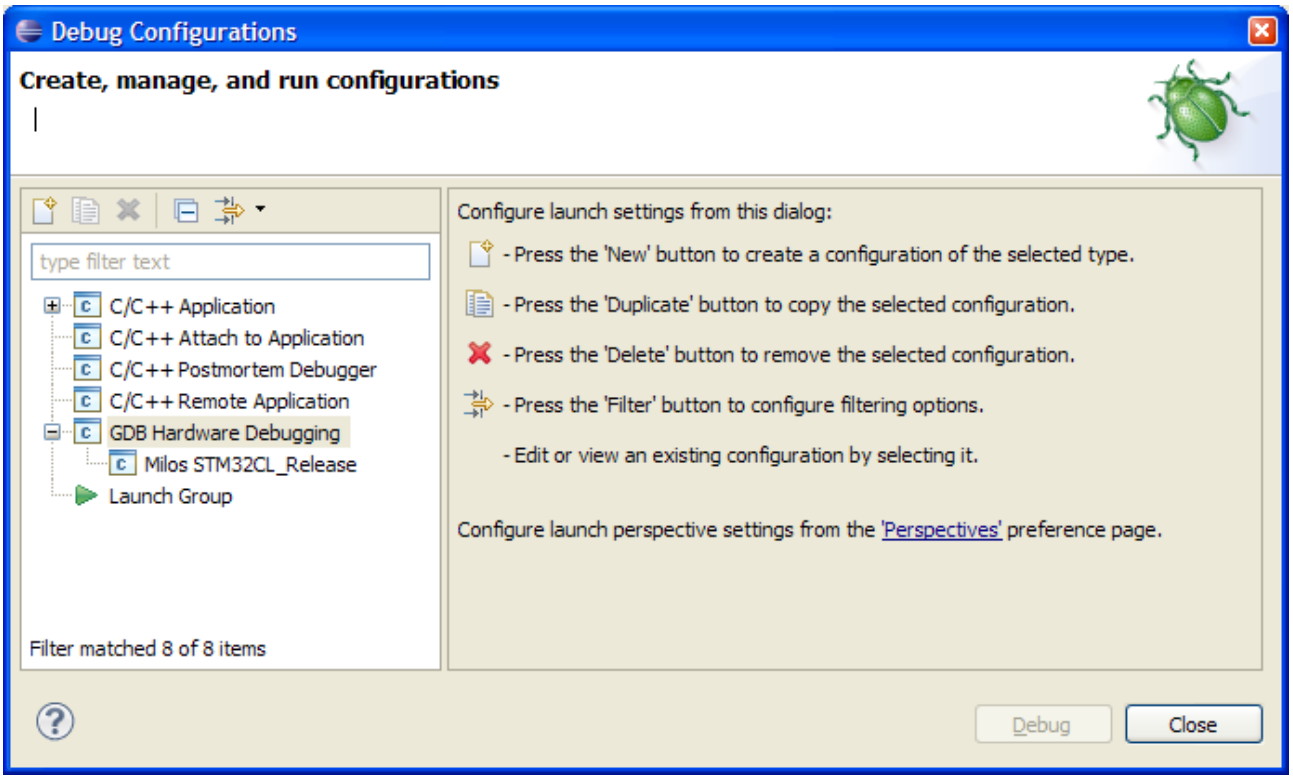
You need an ARM toolchain with GDB support, like YAGARTO or Codesourcery, together working with Eclipse.

In the following example we will be using the Eclipse environment to debug the Milos RTOS.

To configure a debug session click on the Debug menu and choose “Debug Configurations...”.

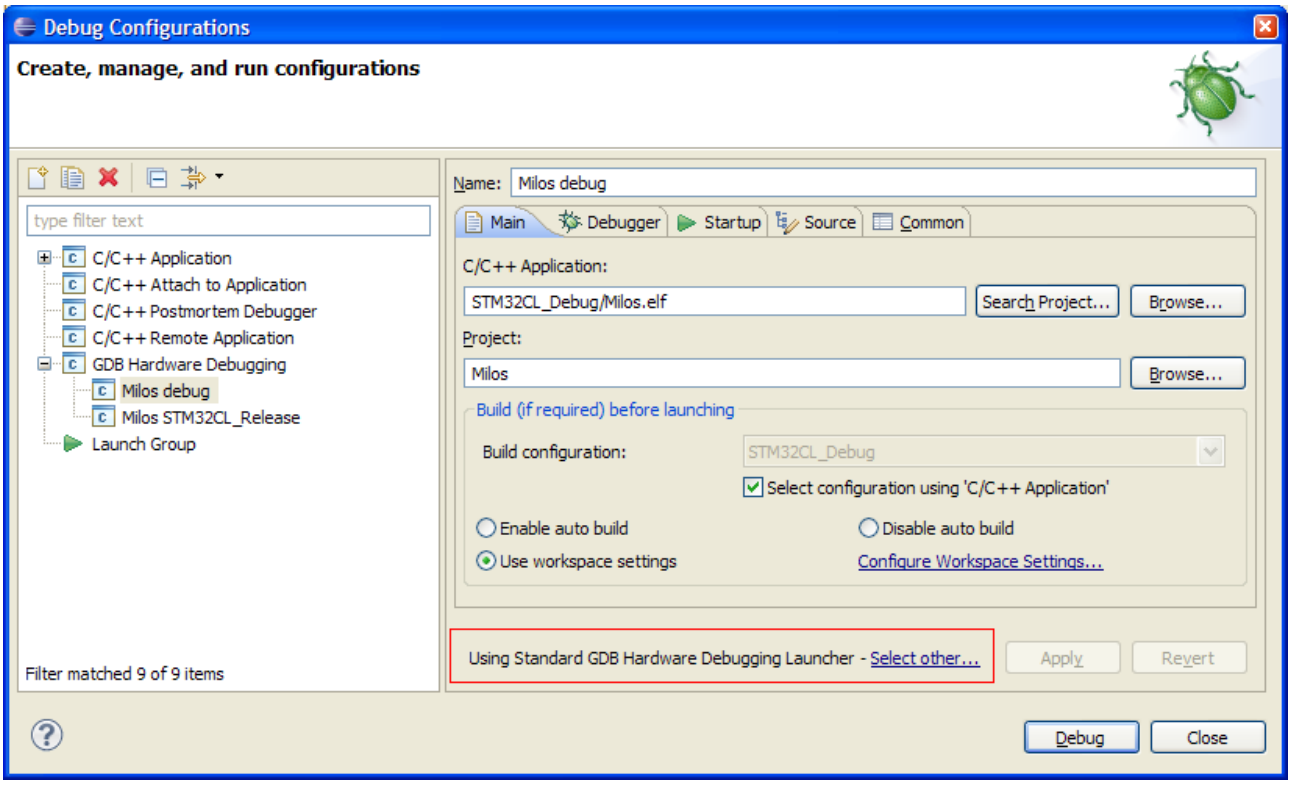


The following window allows you to create a Debug configuration.



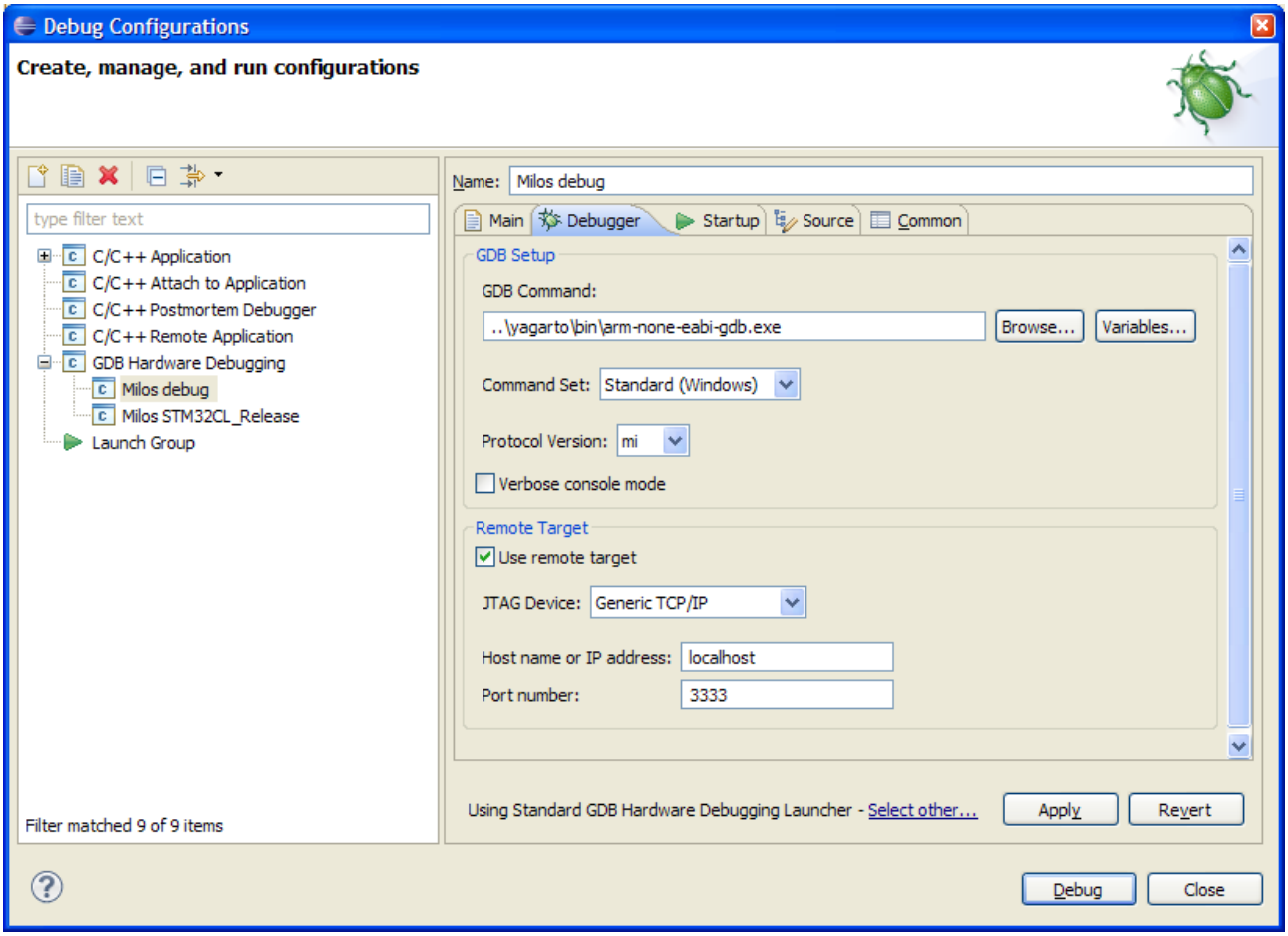
Create a new Debug Configuration by right-clicking the “GDB Hardware Debugging” tree and clicking on “New”. Since we will be creating a Debug Configuration for the debug version of Milos, name it “Milos debug”.

Select the project you want to debug (in this case is “Milos”), and the binary to load on the target (Milos.elf).



Change the GDB launcher to “Using Standard GDB Hardware Debugging Launcher”, by clicking on the “Select other…” option, as shown in the picture above (enclosed by the red rectangle).

Now in the “Debugger” tab we will configure the AK-LINK OpenOCD address and port. If you are running the AK-LINK OpenOCD on the same computer you are launching the debug session, you may leave “localhost” as the remote target address. Change the Port number to 3333 (the default server port for OpenOCD), like the following picture:

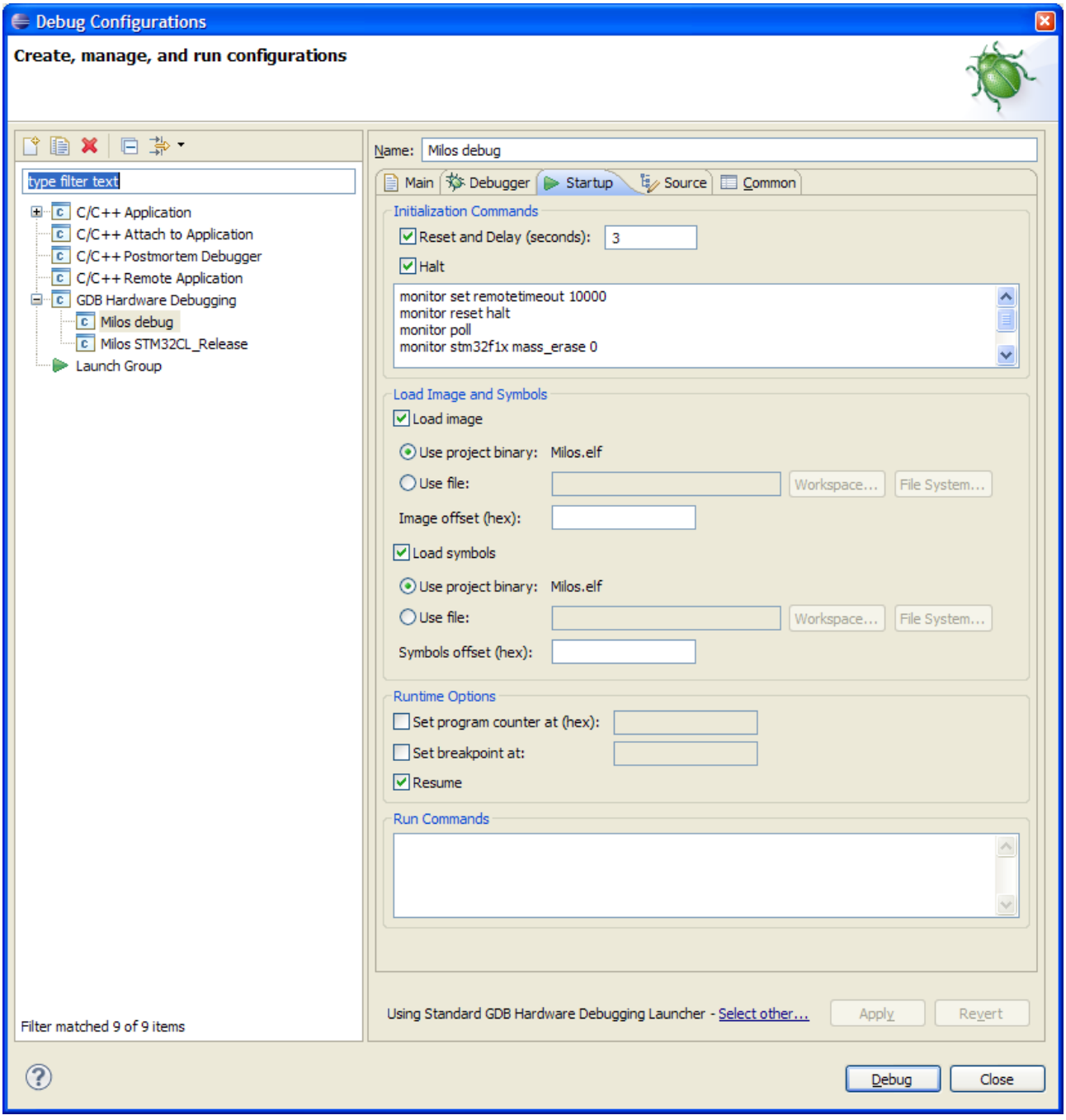


In the GDB Command field, choose your toolchain GDB binary. In this case we will use the YAGARTO GDB client.

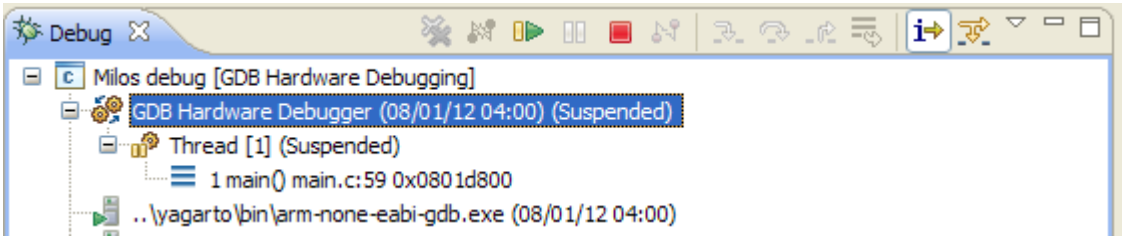
Now in the “Startup” tab, we will configure the startup commands and the binary download into the target. Copy and paste the following text into the Initialization Commands field:

monitor set remotetimeout 10000
monitor reset halt
monitor poll
monitor stm32f1x mass_erase 0
break main

The rest of the parameters should be set as the following picture:



Now the environment is ready to launch a debug session. Click on Apply and the on Debug. After the image is downloaded you can control the debug session from the Debug panel in the Eclipse IDE.



By default the debug session will start from your main function. After that you are ready to debug:

```
45 /*!
46  * @brief Application entry point.
47  */
48 __VOID appEntry(__VOID)
49 {
50     __threadCreate("test1", threadTest1, 80, 1024, 1, __NULL);
51     __threadCreate("test2", threadTest2, 81, 512, 1, __NULL);
52 }
53
54 /*!
55  * @brief Program entry point.
56  */
57 int main(void)
58 {
59     __systemInit(appEntry);
60     return 0;
61 }
```

Quick key shortcuts

- F5 Step into.
- F6 Step.
- F8 Run.