

NOVAembed User Manual

Index

1 : Introduction.....	3
1 : Introduction.....	3
2 : System requirements	4
3 : Launch NOVAembed	4
4 : Welcome.....	4
5 : Boot, Kernel and File System, the BKF tab	6
5.1 The Board drop down.....	6
5.2 Boot Loader area	6
5.3 Kernel area	6
5.4 File System area.....	7
8 : Contacts.....	11
9 : Document revisions, references and notes.....	12
9.1 Document revisions.....	12
9.2 External references	12
9.3 Notes	12

Figures

Figures	2
Figure 1 : The Welcome screen	4
Figure 2 : The BKF screen	6
Figure 3 : Select an existing file system.....	7
Figure 3 : Create a new file system	8
Figure 4 : The BSP Factory screen for a P board.....	9
Figure 5 : The tools windows.....	11

1 : Introduction

This manual explains how to use NOVAembed. NOVAembed is an application that covers most of the features involved in basic setup for the software development on NOVAsom board.

Graphical user interface makes it extremely user friendly and easy to use, even if the user has little experience in the embedded world.

Each operation described below can be accomplished via command line too.

See application notes at www.novasomindustries.com for details.

Further updates, support, software and documentation available at www.novasomindustries.com

Note that :

- Contents and illustrations may differ from your device, depending on the software version, OS version or product improvements that NOVA**som** Industries judges important, and are subject to change without prior notice. Always stay updated visiting www.novasomindustries.com .
- Descriptions are based on the device default settings.
- Modifying the device, the device's operating system or installing software from unofficial sources may damage the device itself and lead to data corruption or data loss, or worst, hardware damage. Such actions will violate your NOVA**som** Industries license agreement and void your warranty.
- Always use genuine NOVA**som** Industries accessories. The supplied items are designed only for this device and may not be compatible with other devices. To have further information on this specific item visit www.novasomindustries.com .
- Default applications on the device are subject to updates, and support for these applications may be withdrawn without prior notice. If you have any questions about an application provided with the device, please contact NOVA**som** Industries at www.novasomindustries.com .
- Software, audio, wallpaper, images, and other media supplied with your device or found in the appropriate SDK are licensed for limited use. If you extract and use these materials for commercial or other purposes, you may be infringing copyright laws. As a user, you are fully responsible for the illegal use of media.

2 : System requirements

NOVAembedded requires to run on a virtual machine with all the basic development packages are installed. The solution adopted is to distribute an entire virtual machine, already configured to run NOVAembedded and all the associated tools.

This document is valid for the following board:

P-line

U-line

3 : Launch NOVAembedded



Clicking onth NOVAembedded icon will launch NOVAembedded.

4 : Welcome

When NOVAembedded starts you will get the following windows.

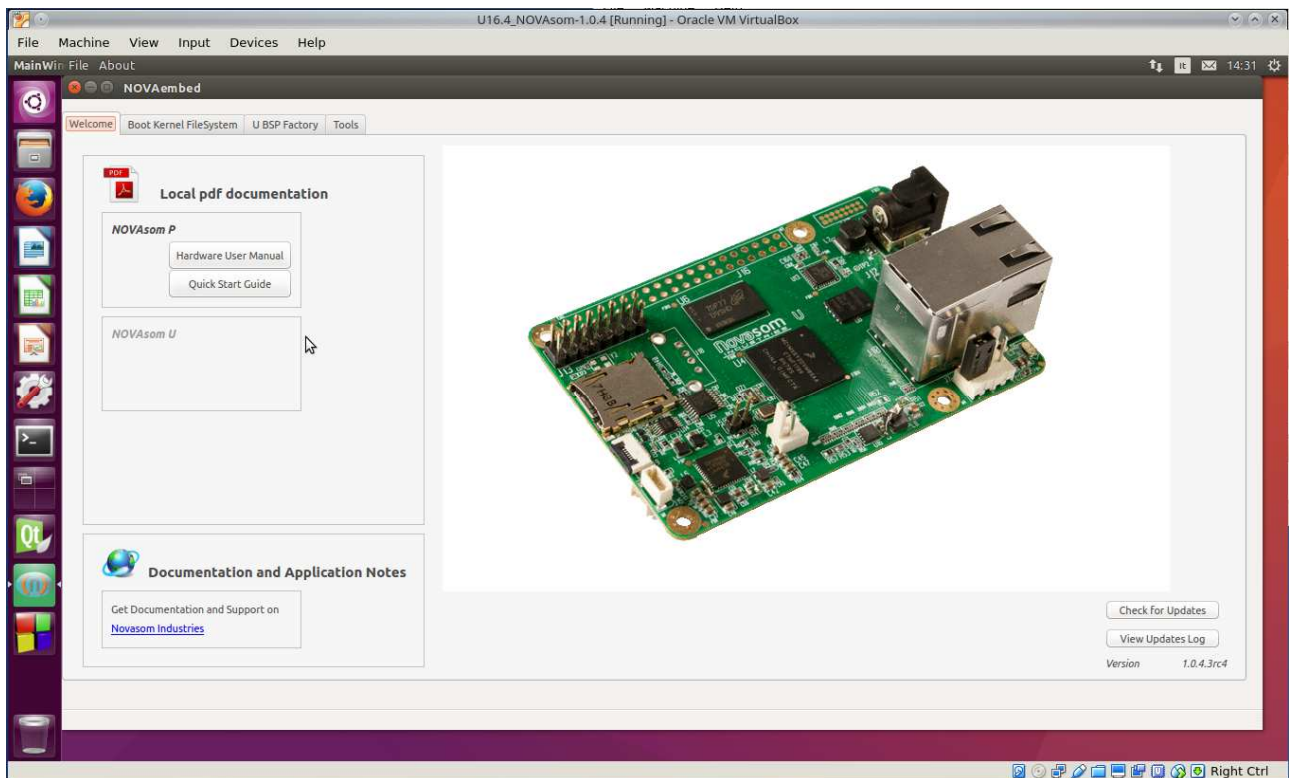


Figure 1 : The Welcome screen

Local pdf documentation

There is some documentation on the initial system. Clicking on the appropriate button a pdf viewer will be started.

In any case the source for the documentation is the link pointed at **Documentation and Application Notes** below.

Check for updates button

Always keep your sdk updated: click this button to check for updates.

A script will run on terminal, if updates are available, confirm to download and install (hit: Y).

The terminal will be closed automatically at the end of the procedure. In case of NOVAembed updates the NOVAembed application must be restarted.

View Updates Log button

You can also check the updates that has been applied to your system clicking on View Updates Log button. In case of malfunction this can be useful to identify the problem.

Version field

The **Version** field indicates the current version. When a version changes there are strong reason you need to recompile the whole system (e.g. boot, kernel and file system), but normally is a matter of some seconds, as the system has to maintain the sync between different versions. This will be explained better in th **BKF Tab** paragraph below.

5 : Boot, Kernel and File System, the BKF tab

Clicking on tab **Boot Kernel FileSystem** will display the following page:

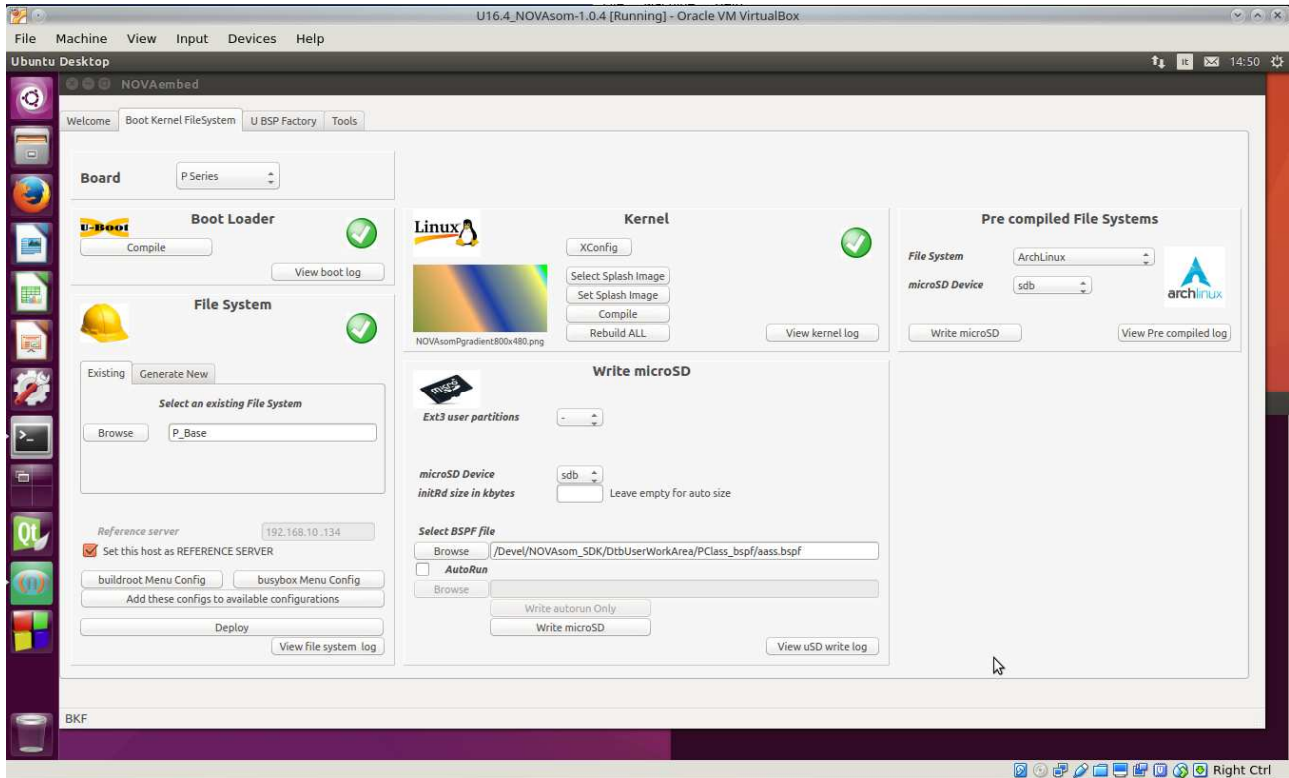


Figure 2 : The BKF screen

The **Boot Kernel FileSystem** tab (or **BKF** tab) is a quite complex windows.

This windows is divided in five main areas : The **Boot loader** area, the **Kernel** area, the **File System** are, the **Pre compiled File Systems** area and the **WritemicroSD** area.

5.1 The Board drop down

The Board drop down selects between the NOVAasis boards. Each time the Board dropdown is changed the compilation of the boot loader, the kernel and the file system is invalidated, and the green sign in Figure 2 will be red. Only when all the signs are red the uSD write area will be enabled.

5.2 Boot Loader area

The **Boot loader** area is where the boot loader is compiled. This is the simpler area, as only one button is present (**Compile**) and no further actions can be performed.

After a successful compilation there will be the green sign in this area. In case of failure the sign will be red, and the **WritemicroSD** area will be disabled.

5.3 Kernel area

The **Kernel** area is where the kernel can be compiled. There are some options in this area:

- Xconfig : starts the usual xconfig for linux kernel

- Select splash image : selects which image will be on the selected panel at boot. Note : this is not the splash image at power up that is managed from the boot loader.
- Set splash image : actually compiles the selected splash image in the kernel. Despite seems a simple process, the image will be translated, color reduced, compressed, transformed in c code and stored in the appropriate location of the kernel. Immediately after this process the kernel is compiled.
- Compile : compiles the kernel, and normally this is enough
- Rebuild all : if happens that the kernel is too messed this will clean up the kernel and rebuild it again. On a decent machine (2 cores with 4GB of ram) this process lasts for 10 minutes or less.

After a successful compilation there will be the green sign in this area. In case of failure the sign will be red, and the **WritemicroSD** area will be disabled.

5.4 File System area

This is the more complex area, and probably the most used ones, together with the **Write microSD** area below. This is due there are more iterations on the correct composition of the file system than all the other steps.

The first operation that should be done is to decide if a current file system is the one that will be used on your application or if you need to create a new one.

The tab under the **File System** area title does this : selecting **Existing** an already present file system can be selected.

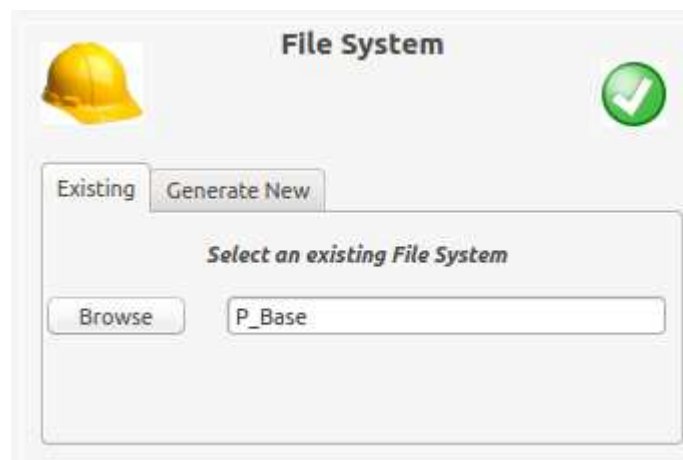


Figure 3 : Select an existing file system

In this case there was already a compiled file system on the sdk, so this has been selected.

Selecting **Generate New** opens a different tab composition :

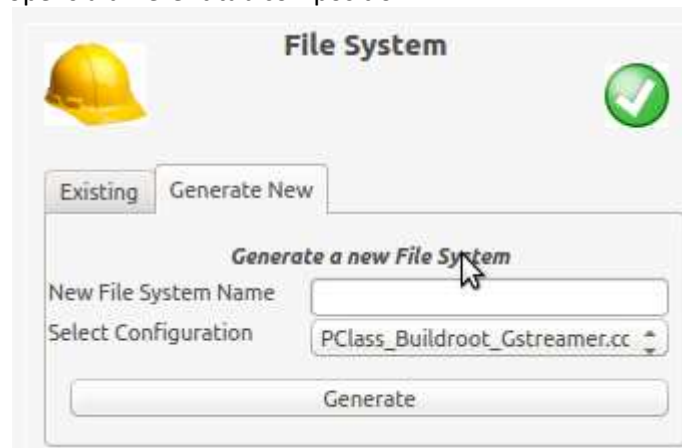


Figure 3 : Create a new file system

In this case the field **New File System Name** must be filled with an appropriate name and the **Select Configuration** drop down can be used to select a basic configuration.

In the example above the configuration is for a Gstreamer application.

The **Generate** button will create a new file system that can be compiled with the Deploy button described below.

Note that the configuration file for a new file system can be saved and will be present on the drop down menu, so the user can save all the configurations he needs and switch between them in seconds.

Reference Server

This is the ip from which the NOVA-som board will download files, upload kernel and file system.

This IP is the IP configured on the NOVA-som board as default.

It is possible to set the reference server ip on the microSD card at any time.

Launch menuconfig

Launch buildroot menuconfig to modify the .config file. This operation will run only if a valid existing file system is selected. Browse the menu and select the tools that will be added to the file system, then **Save** and **Exit**.

Launch busybox-menuconfig

Launch busybox menuconfig to modify the .config file. This operation will run only if a valid existing file system is selected. Browse the menu and select the tools that will be added to the file system, then **Save** and **Exit**.

Deploy

Starts the file system compile.

After a successful compilation there will be the green sign in this area. In case of failure the sign will be red, and the **WritemicroSD** area will be disabled.

6 : BSP Factory

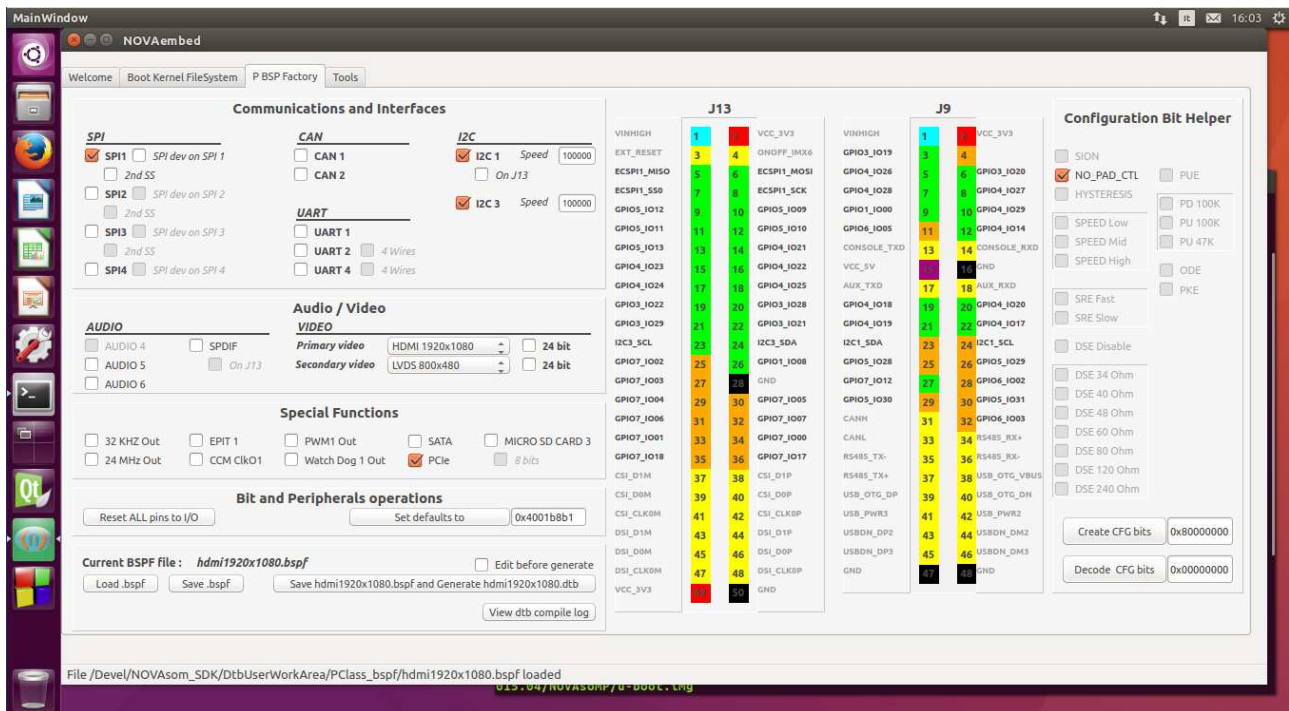


Figure 4 : The BSP Factory screen for a P board

In the BSP Factory screen the user can select the behavior of it's own machine, configuring almost everything there is on the I/O connectors down to the existence of PCI, SATA or other things. The system used peripherals are already configured, and can't be modified, so a very bare system can have everything in the GPIO state.

All the peripherals are in the form of check boxes, and selecting a peripheral can disable some other, depending on the i.MX6 multiplexing : in the example above, the SPI#2 cannot have a second slave select signal because is shared with the SPI#1 and the SPI#1 uses it. The appropriate **Hardware User Manual** of the board can explain better and in details what configurations can be applied.

In the right part of the screen there are the connector pins, and changing or adding a function will change the pin on the connector symbol.

As per the **Hardware User Manual** of the P board, for example, there are colors that indicate the pin function, where some are fixed, some other are power supply and some other are GPIO at fixed voltage or variable voltage. For the details look at the **Hardware User Manual** of the selected board.

A useful helper is on the right of the screen, the **Configuration Bit Helper**. It can take bit configuration and decode them, or can encode the pin characteristic compiled above in hex numbers, exactly what i.MX6 needs to configure properly a pin.

The buttons in this screen are:

- **Reset all pins to I/O** : when everything is too messed up clicking on this pin set everything to I/O function.
- **Set defaults to** : sets the pin I/O characteristic to a specified value. This value is stored and all subsequent operations will use this value.
- **Load .bspf and Save .bspf** : load and save a .bsp file (an intermediate language to describe the pin usage) , used for the configuration of the items in this screen.
- **Save <.bspf_file_name> and generate dtb** : when configuration is finished the tool will generate a .dtb file, usable for flashing or update. Clicking the checkbox above this button opens a gedit session with the .dts file before generating the .dtb file. Please note that this file can be changed but subsequent runs will overwrite the dts file, effectively deleting all the modifications.

7 : Tools

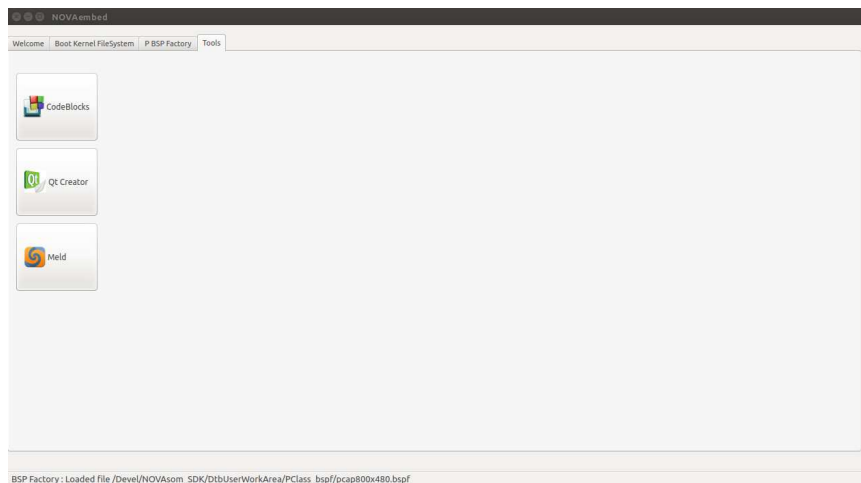


Figure 5 : The tools windows

CodeBlocks

Open CodeBlocks IDE. Info about CodeBlocks available at www.codeblocks.org .
See application notes about CodeBlocks IDE configuration.

Meld

Compare two files with meld tool. Click **Browse** to select the files to compare, then choose **Start meld** .
Meld support available at meldmerge.org .

Qt Creator

Launch QtCreator. See application notes about QtCreator IDE configuration.

8 : Contacts

Novasom Industries Italy

Headquarter:

Via Orbassano 2/a 10048 Vinovo (TO)

Phone (+39) 011/9004003

Fax (+39) 011/9653274

mail info@novasomindustries.com

Web page : www.novasomindustries.com .

9 : Document revisions, references and notes

9.1 Document revisions

AN001-A- V1.0	02/03/2017 Update
---------------	-------------------

9.2 External references

See the NOVASom Industries official website: www.novasomindustries.com

9.3 Notes

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates.

It is your responsibility to ensure that your application meets with your specifications.

NOVASom Industries MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. NOVASom Industries disclaims all liability arising from this information and its use.

Use of NOVASom Industries devices and software in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless NOVASom Industries from any and all damages, claims, suits, or expenses resulting from such use.

No licenses are conveyed, implicitly or otherwise, under any NOVASom Industries intellectual property rights.