Tecnoroll BMB s.r.l.
Via Olona 9, 21040 Vedano Olona - VA

www.tecnoroll.it

FLEXMod 3

modules

Quick start guide

1. The use of this product in conformity with the DVB Standard is prohibited in any case, unless the license of patents applicable to the DVB patent group is granted.
2. It's the purchaser's duty to check which patents with potentially payable author's rights are used in his/her own final project and act accordingly with the bodies in charge.
3. The contents of this manual are subject to change without notice.
4. Of the specifications described in this manual, the functions and performance the supply of which is assured with this product are only those that can be confirmed with the separate Basic Firmware attached.
5. When the product or information described in this manual is used, our company does not guarantee or grant any license of the intellectual property rights and other rights of our company or of third parties. If any problems concerning the ownership rights of third parties attributable to the above use occur, our company will not be responsible for them.
6. Since the product described in this manual was not developed specially for equipment related to human life, those who intend to use it for equipment whose failure or malfunction may directly threaten human life or endanger the human body or for special applications requiring high reliability (such as aerospace, nuclear power control, submarine repeaters, travel control, and medical equipment for life support) must consult our sales responsible personnel in advance. Our company will not be responsible for any damages that may occur due to such use without consulting us.
7. Although our company is endeavoring to improve quality and enhance reliability, semiconductor products will fail with a certain probability. Users are requested to pay due attention to safe design techniques such as redundant design, preventive measures against the spreading of fire, over-current, and malfunctions, so that failure of the products described in this manual will not result in accidents leading to injury or death, fire, or social damage.
8. The product described in this manual is not designed to be radiation-resistant.
9. This board is just an 'electronic component' that cannot work in a standalone mode, therefore all the tests and qualifications for electromagnetic compatibility, all the CE certifications and any other regulations concerning the final product implemented by the customer are to be carried out by the customer himself referring to the final product for which this electronic component has been used.

Revisions history:

Rev A, January, 28 2013, dbrini@tecnoroll.it

First release

Index

Revisions history....3

FLEXMod 3....5

The bootloader....6

Memory management....6

Setup of serial communication....8

Use of the Bootloader....10

Help....11

TAB key use....12

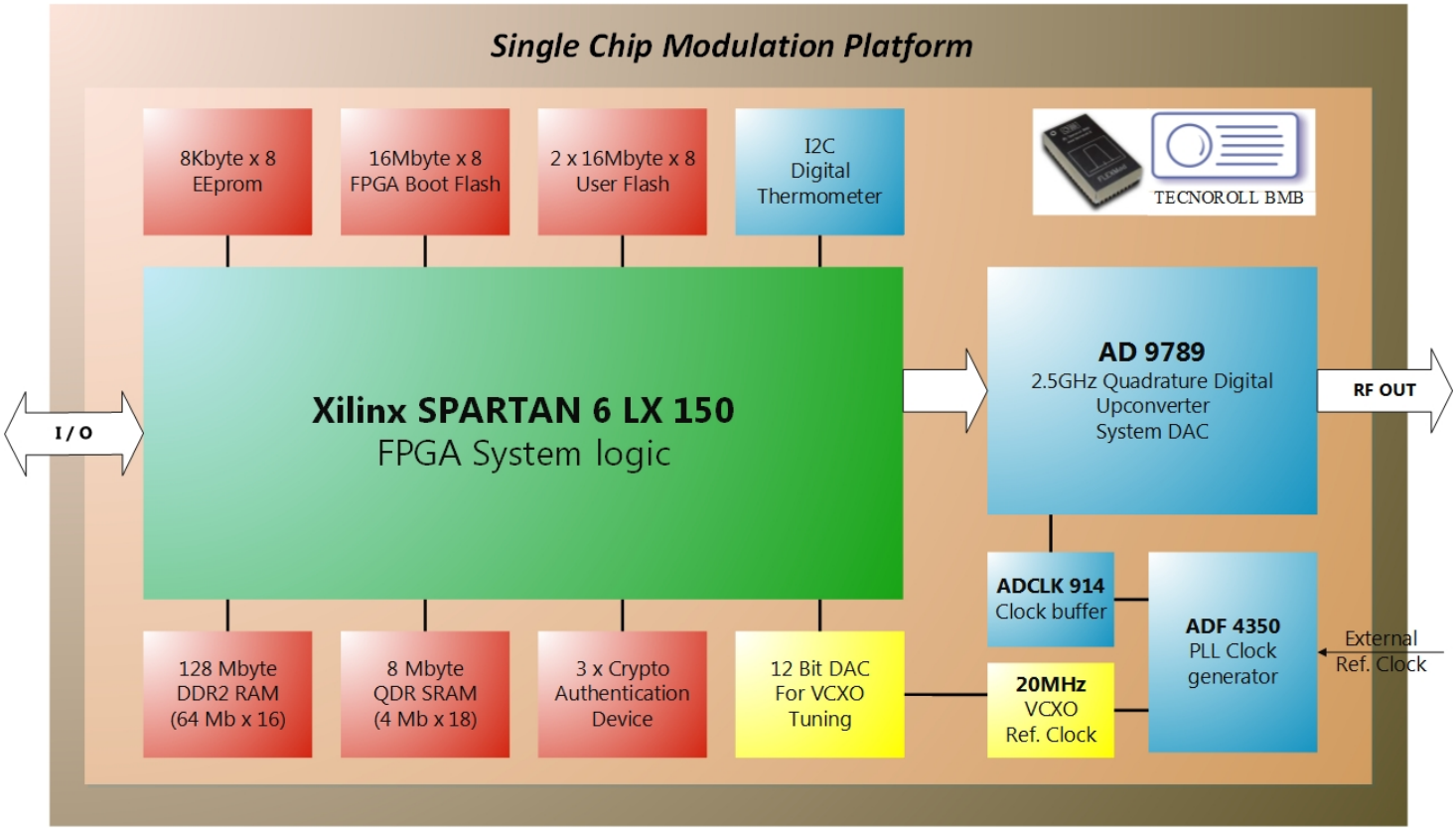
Selecting the bitstream to be launched....13

Loading a bitstream....14

Addendum for IP Providers....18

Known issue....21

FLEXMod 3 is a multi-standard modulation platform which can combine high performance with great flexibility and many types of use, either as a finished product or as hardware platform where the customer can implement his own Ips.
Tecnoroll's module is internally organized as follows:

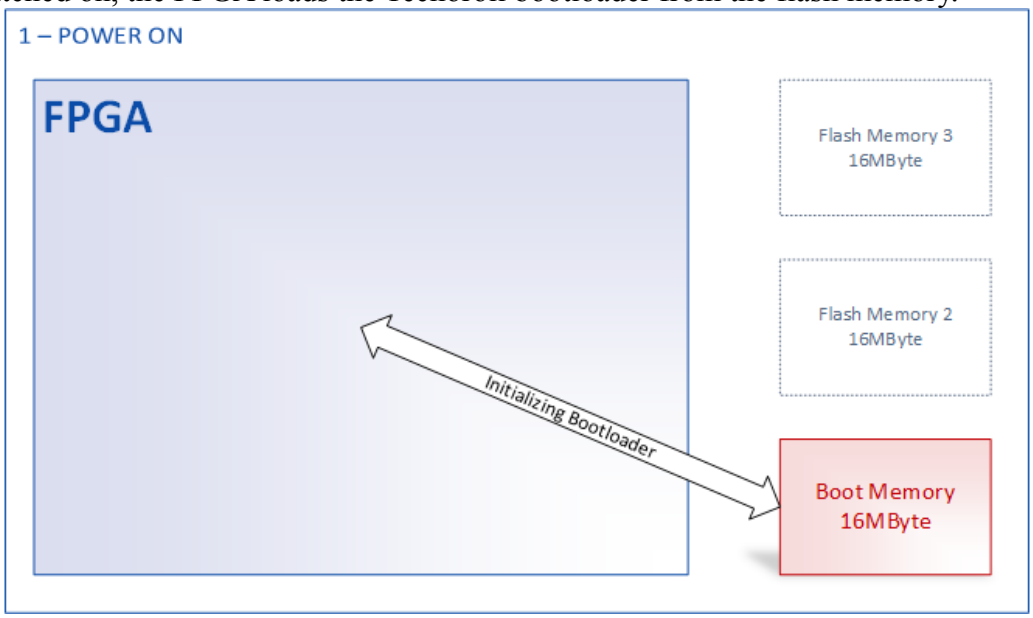


The bootloader

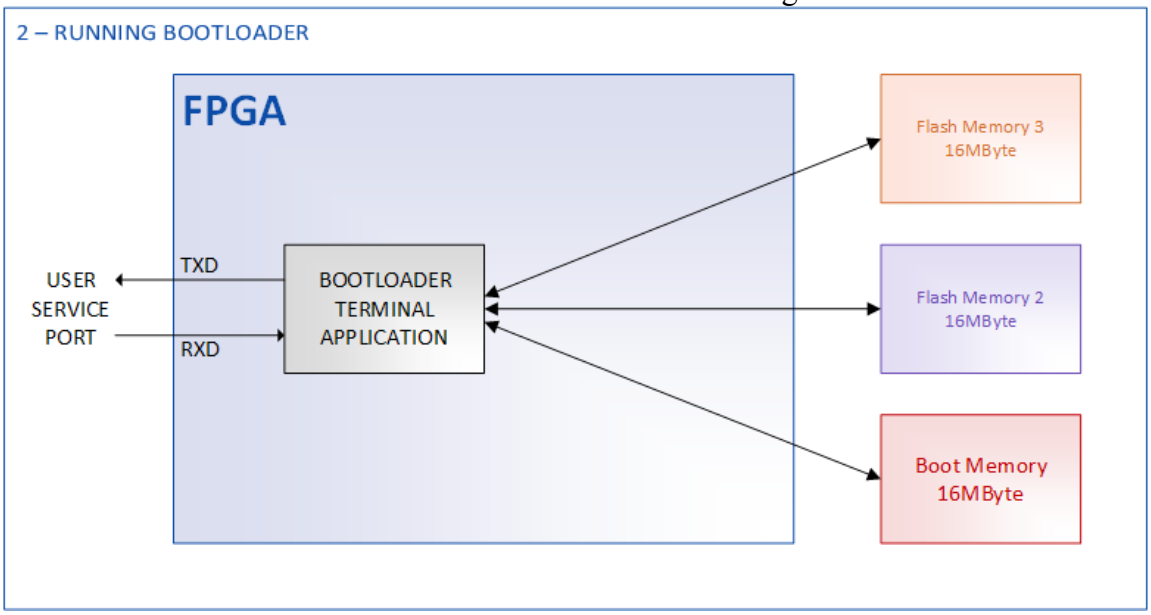
The FLEXMod 3 chipset integrates an advanced bootloader allowing to quickly load and manage several different bitstreams into the system flash memory. This is achieved thanks to an internal command interpreter accessible through the UART port and a standard terminal software such as Windows HyperTerminal.
The bootloader also takes care of the multi-bank flash memories as if they were a single virtual disk, thus allowing the user to bypass the FPGA limitation of 16MB addressing range. Thanks to this feature, the user can get access to the expanded memory in a transparent way, without having to develop any other bootloader bitstream or software. This feature allows to manage up to 16 bitstreams, which can be automatically started at power-up and can be uploaded and downloaded through a standard terminal software for easy upgrading on the field.

Memory management

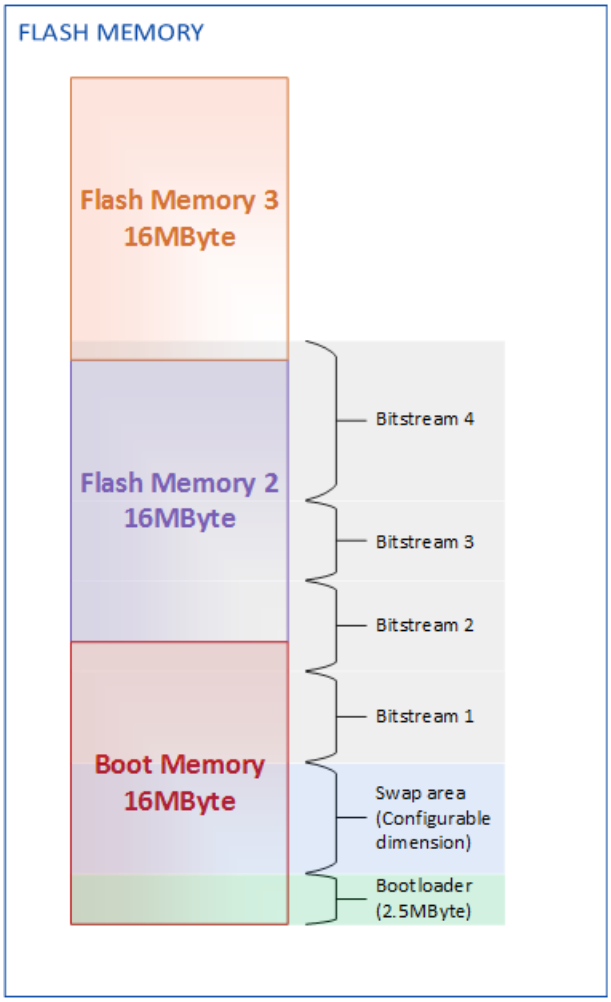
Whenever switched on, the FPGA loads the Tecnoroll bootloader from the flash memory.



Then, the bootloader makes the other flash memories visible and manages the serial user interface.



The bootloader also links the memories together, so that they are visible as a single virtual hard disk.
Example:



The partition occupied by the bootloader (2.5 MB) cannot be changed or overwritten, and is automatically loaded into the FPGA at power up.

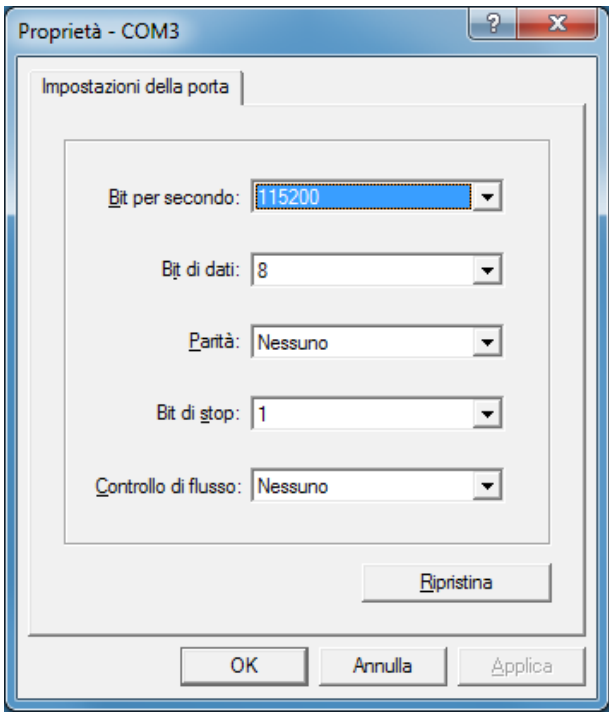
The swap area size can be configured by the user and is the partition where the bootloader automatically stores the bitstreams that can't be loaded directly by the FPGA because of the 16MB addressing limitations. When the FPGA loads the bitstream, one of the following 3 scenarioes can occur (shown as an example in the diagram above), and the bootloader automatically recognizes it and takes action accordingly:

- **Bitstream 1**: fully resident in the memory bank directly managed by the FPGA (boot memory); it is loaded directly from its residence partition.
- **Bistream 2**: resident across two different memory banks; in order to be run, it must be loaded into the swap area.
- **Bitstream 3 or 4**: fully resident in a memory bank which is not directly managed by the FPGA; in order to be run, it must be loaded into the swap area.

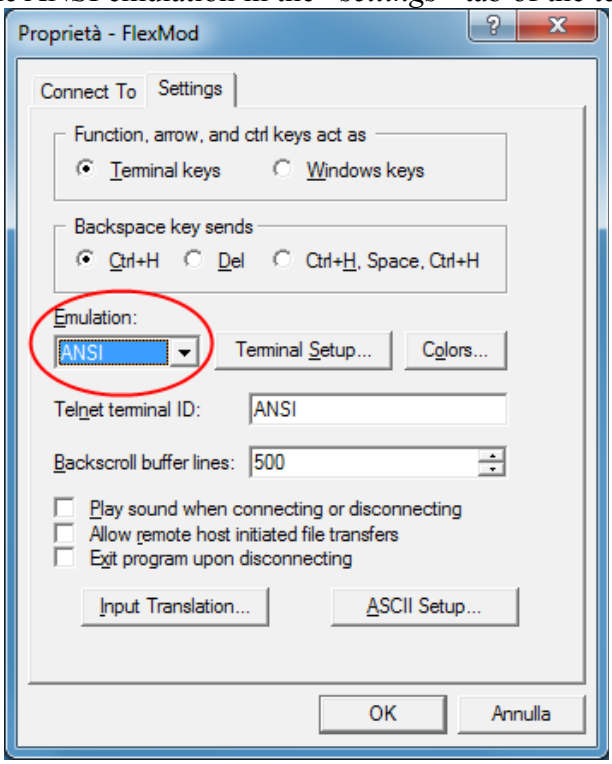
A maximum of 16 bitstreams can be saved inside the flash memory or, at least, as far as there is free space.
Please make sure that the swap area is big enough to contain the user bistreams.
If needed, the default 4.5 MBytes swap area size can be set via the `FlashFormat` command.

Setup of serial communication

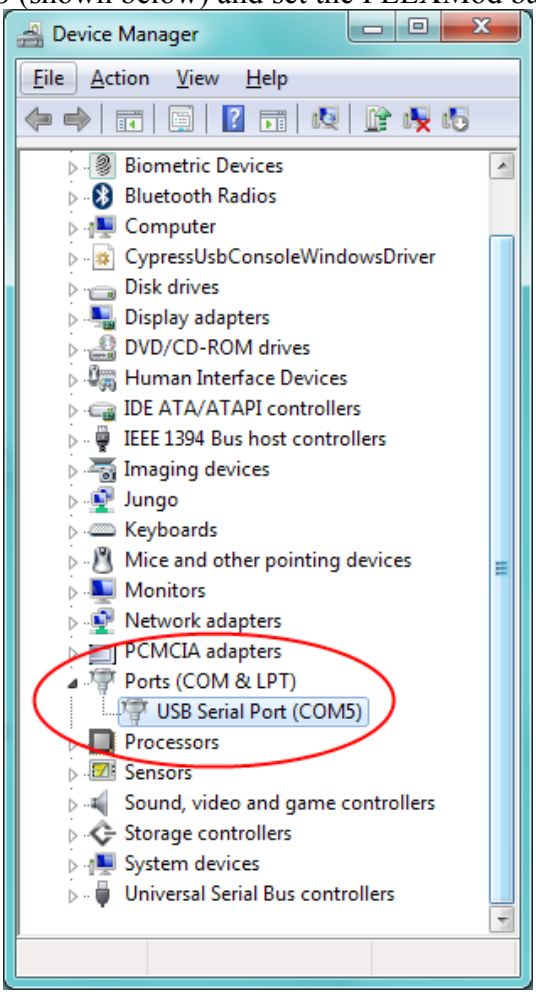
The bootloader manages the user interface via serial communication. You can use any terminal to interact with it; for example, you can use Hyper-terminal, which is already present in Windows XP, and configure it according to the following scheme:

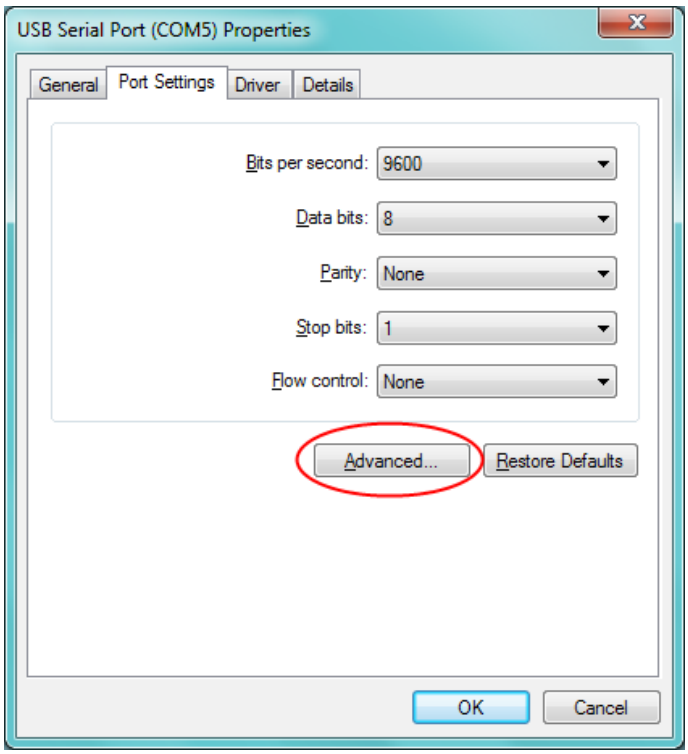


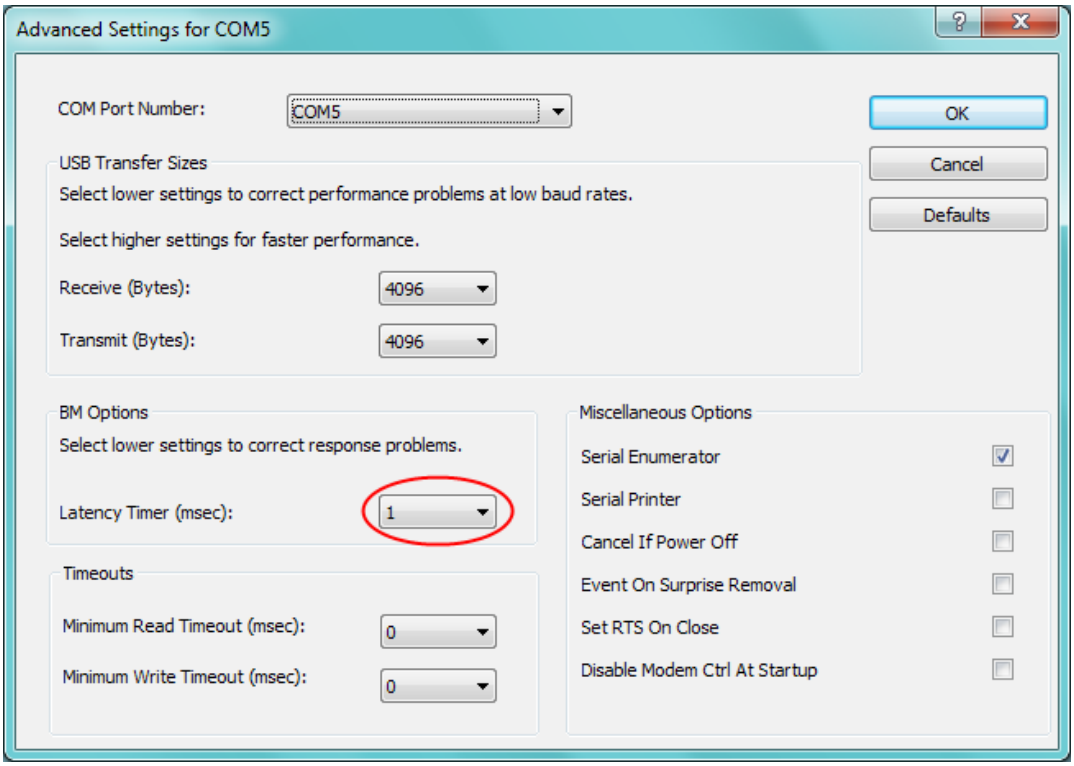
Please, also remember to set the ANSI emulation in the *"settings"* tab of the terminal connection properties:



In order to optimize the upload speed of a desired bitstream into the flash memory, set the COM port latency to 1mS (shown below) and set the FLEXMod baudrate to 921600bit/s.

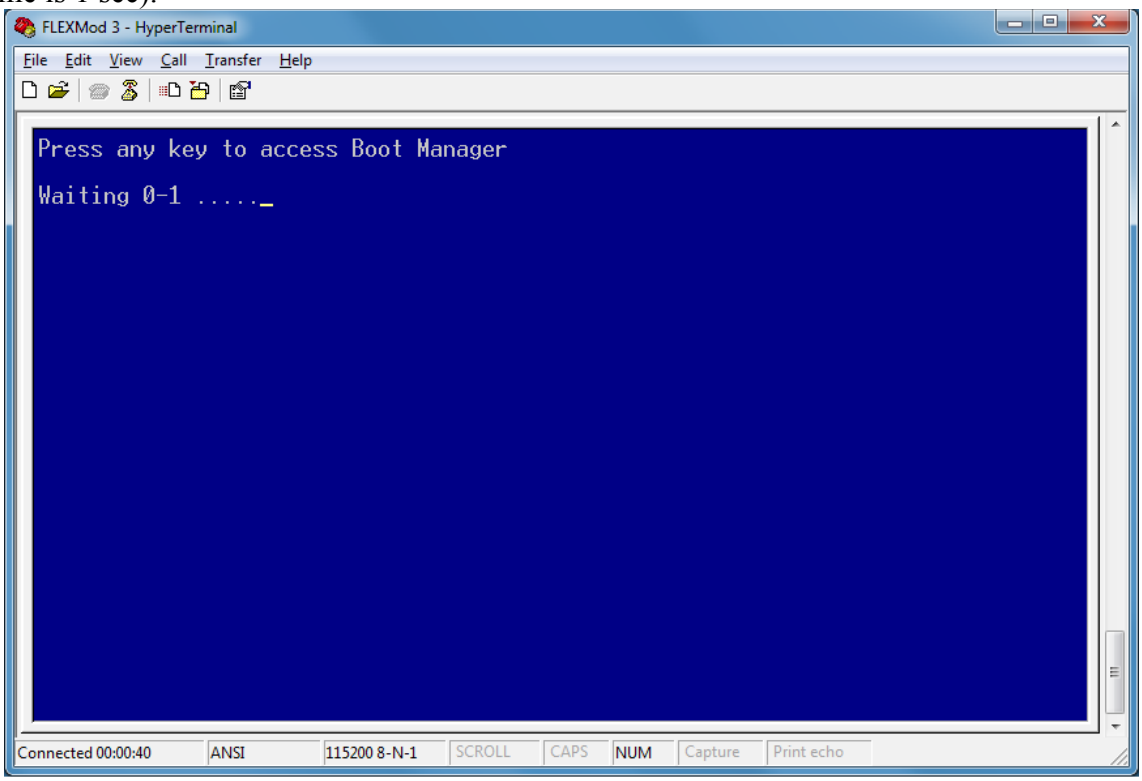




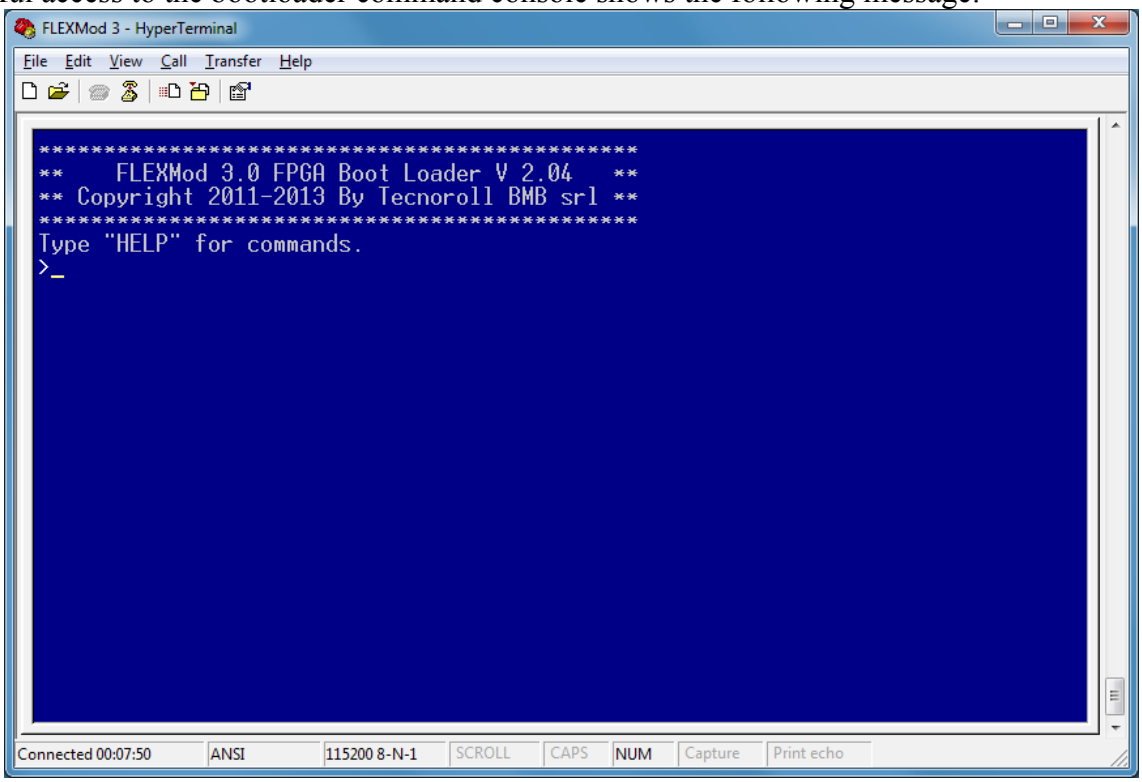


Use of the bootloader

When you power up the FLEXMod module, when you see the "waiting" dots appearing, you can stop the booting process and get access to the bootloader console by pressing any key within the given time interval (as shown in the screenshots below). The time interval can be set through the `SetBootWait` command (default time is 1 sec).

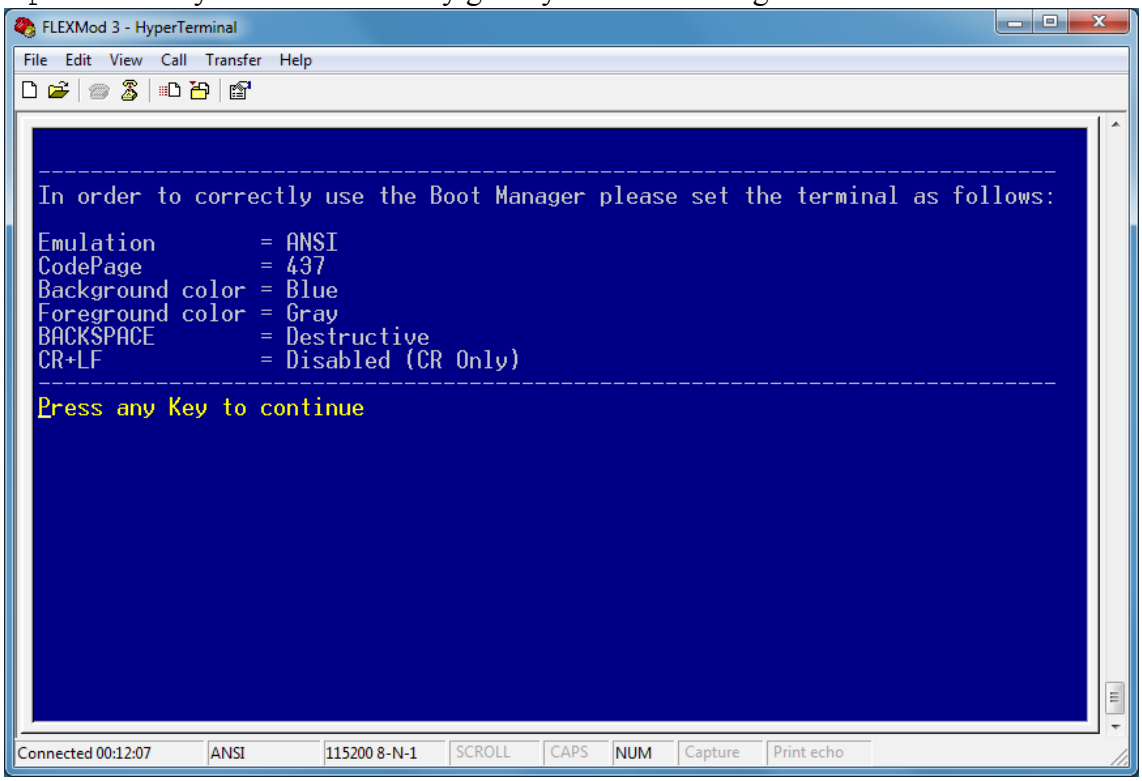


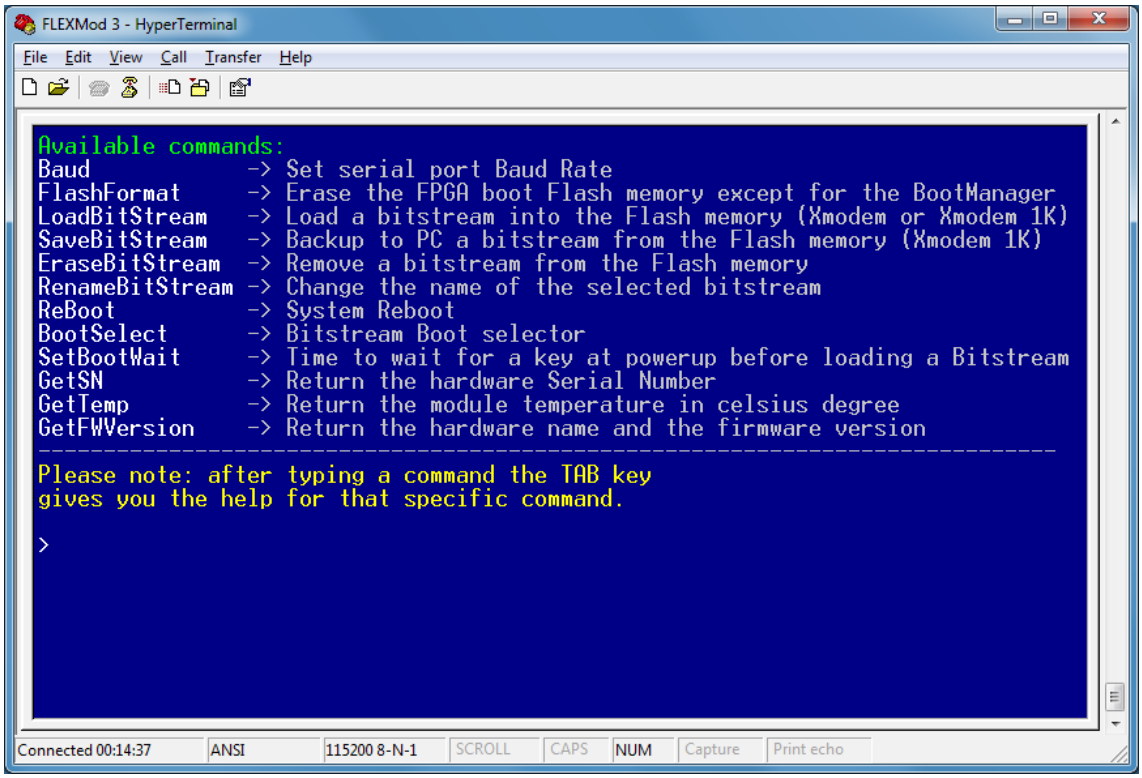
A successful access to the bootloader command console shows the following message:



Help

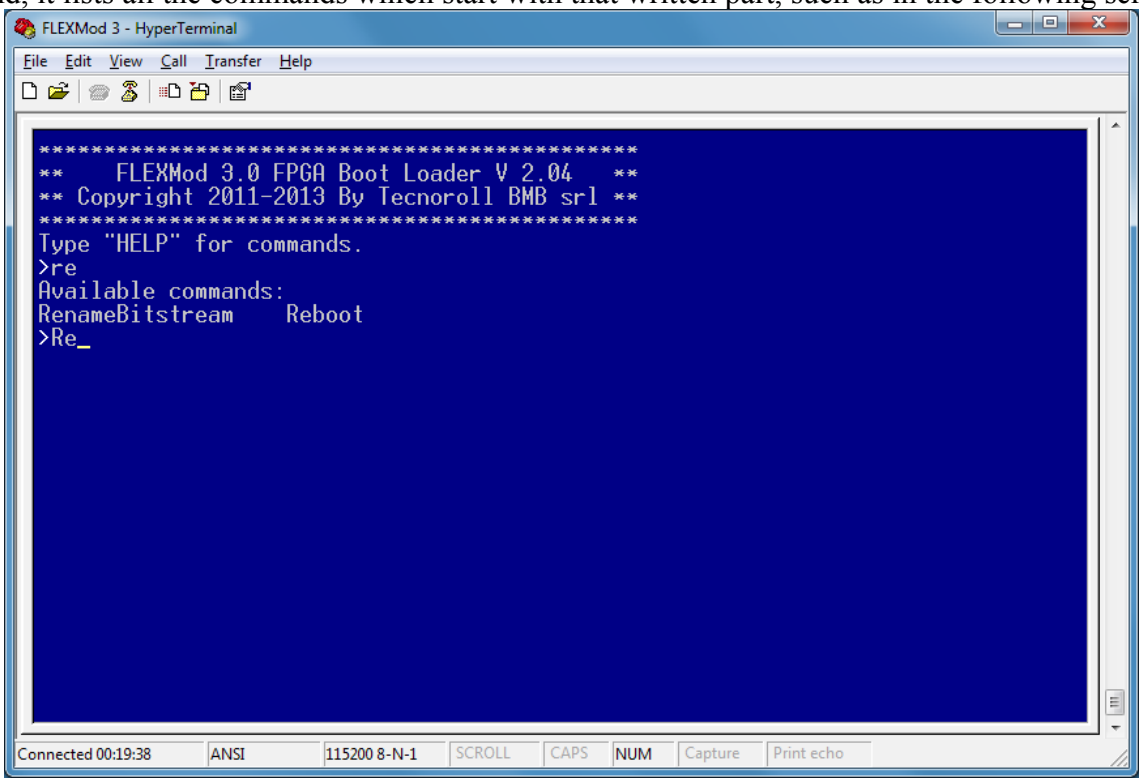
Typing `Help` followed by the <ENTER> key gives you the following result:



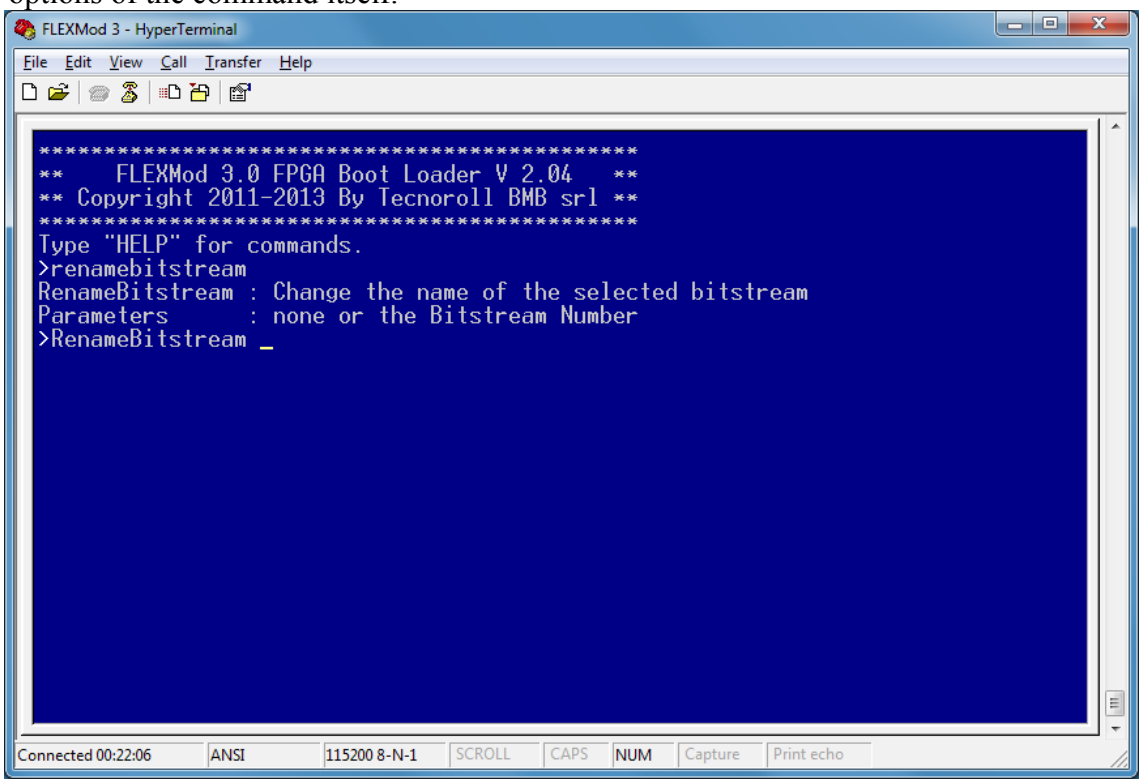


<TAB> key use

The <TAB> key is used to get specific help on the various commands: if pressed after writing only a part of a command, it lists all the commands which start with that written part, such as in the following screenshot.

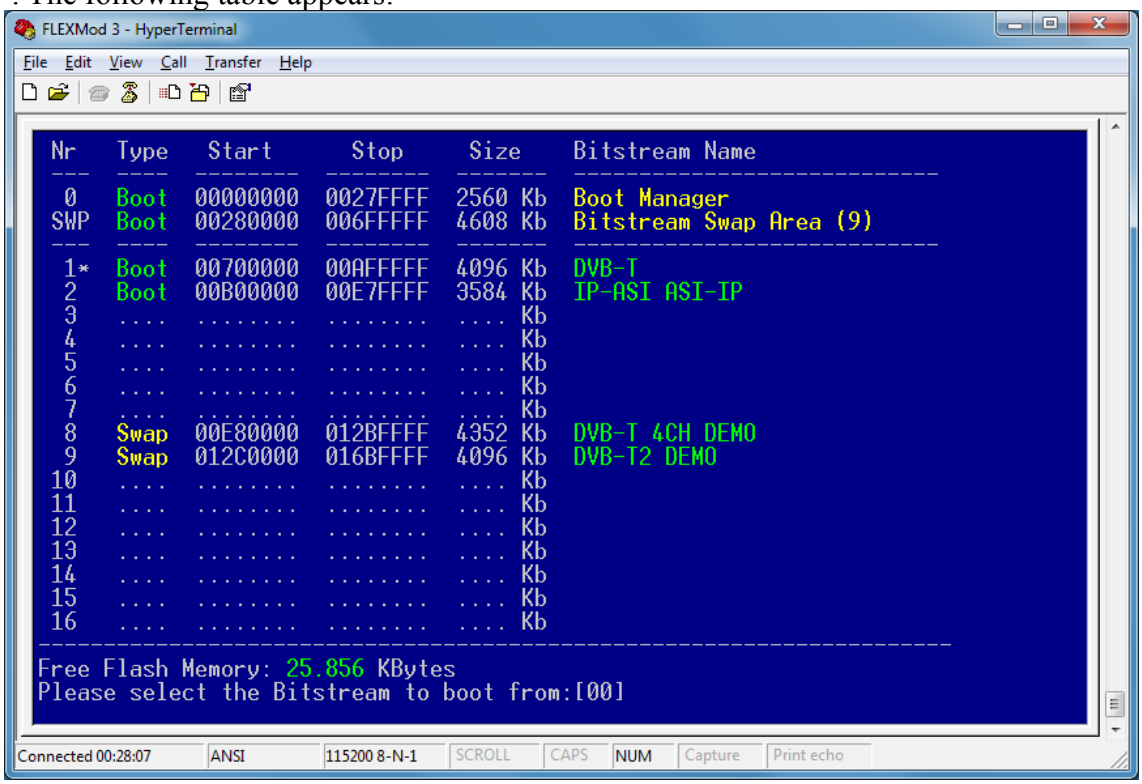


If used after writing the whole command, it provides specific help on the correct syntax and on any additional options of the command itself.



Selecting the bitstream to be launched

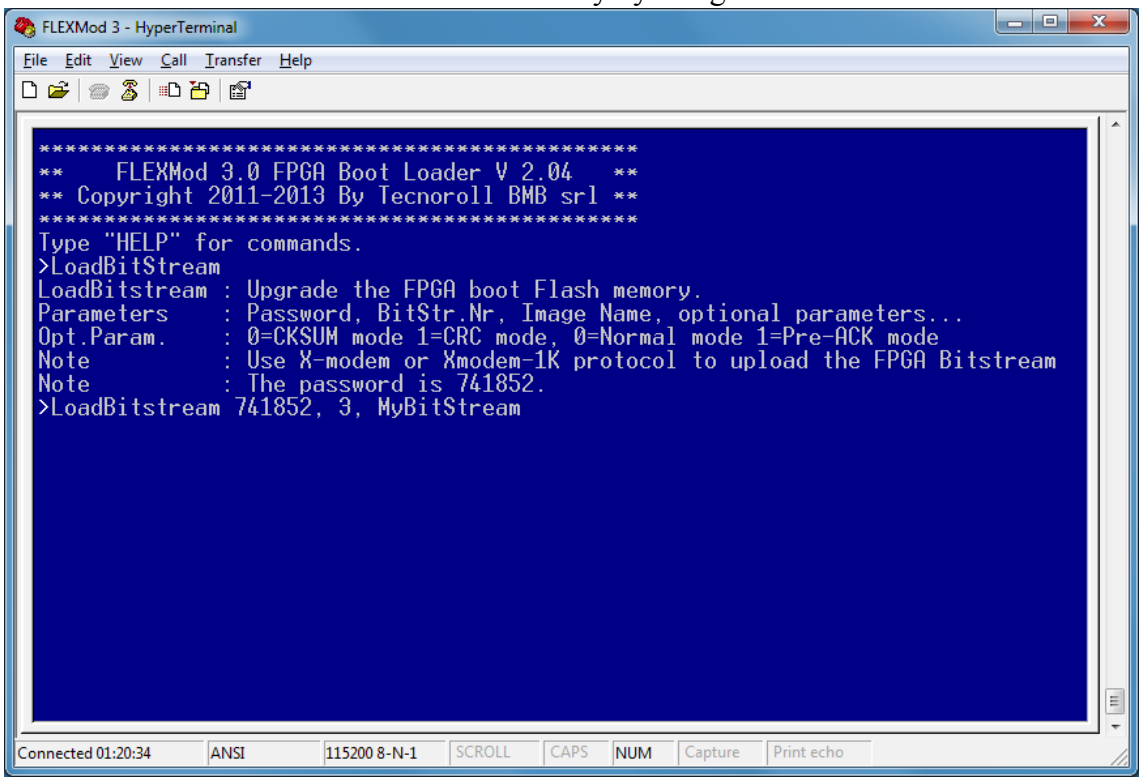
One of the most important operations you can perform thanks to the bootloader is selecting the bitstream to be loaded when the FLEXMod is powered up. The relevant command is `BootSelect` followed by <ENTER>. The following table appears:



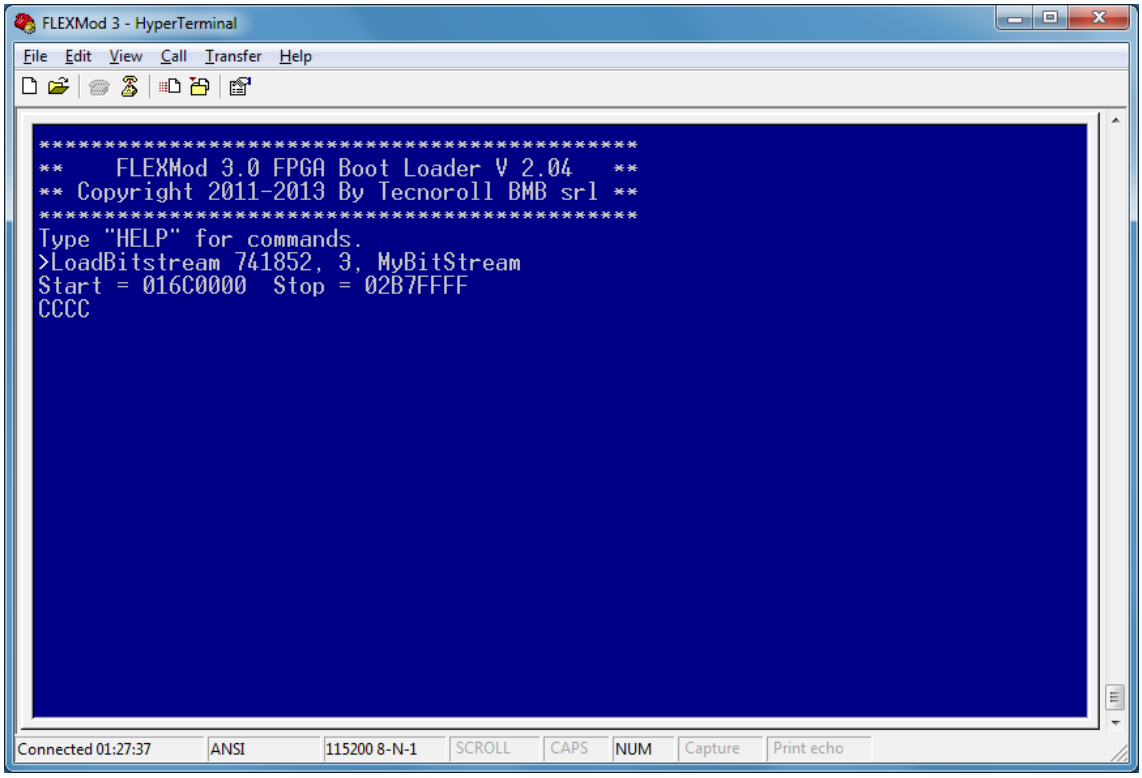
The *"Type"* column indicates whether the bitstream can be uploaded directly by the FPGA or whether it must pass though the swap area.
The *"Start"* and *"Stop"* columns indicate the start and stop addresses of the bitstream.
The *"Size"* column displays the total size occupied by the bitstream inside the flash memory.
The last column is for the bitstream name.
Note: the number of the bitstream currently present in the swap area is indicated between parentheses in the swap area row.
An asterisk beside the bitstream number (first column) indicates that this bitstream is the one selected for automatic startup when the module is powered up.
In order to select the bitstream to be launched, simply enter the corresponding number followed by the <ENTER> key. The user will be then asked whether the bitstream should be started automatically or not when powering up the device.
Note: If the FLEXMod module is managed through a string of commands from a microcontroller without passing through the ANSI terminal, it is highly recommended not to set any automatic bitstream startup options. In this case the bootloader gives you direct access to the command interpreter without needing to manually stop the booting procedure. When the booting procedure is complete and you have accessed the command interpreter, simply send the `BootSelect xx` *string (where xx is the number of bitstream that you want to start) which in this case is only a temporary choice.*

Loading a bitstream

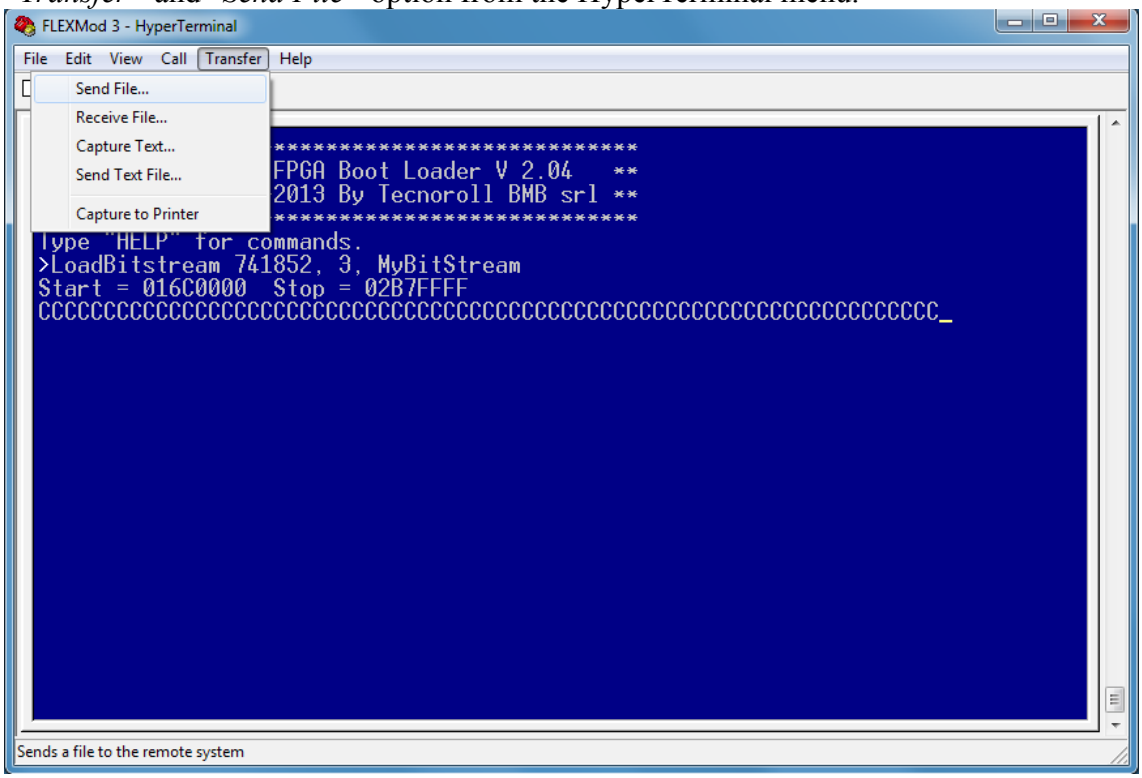
It is possible to load a bitstream into the FLASH memory by using the `LoadBitStream` command.



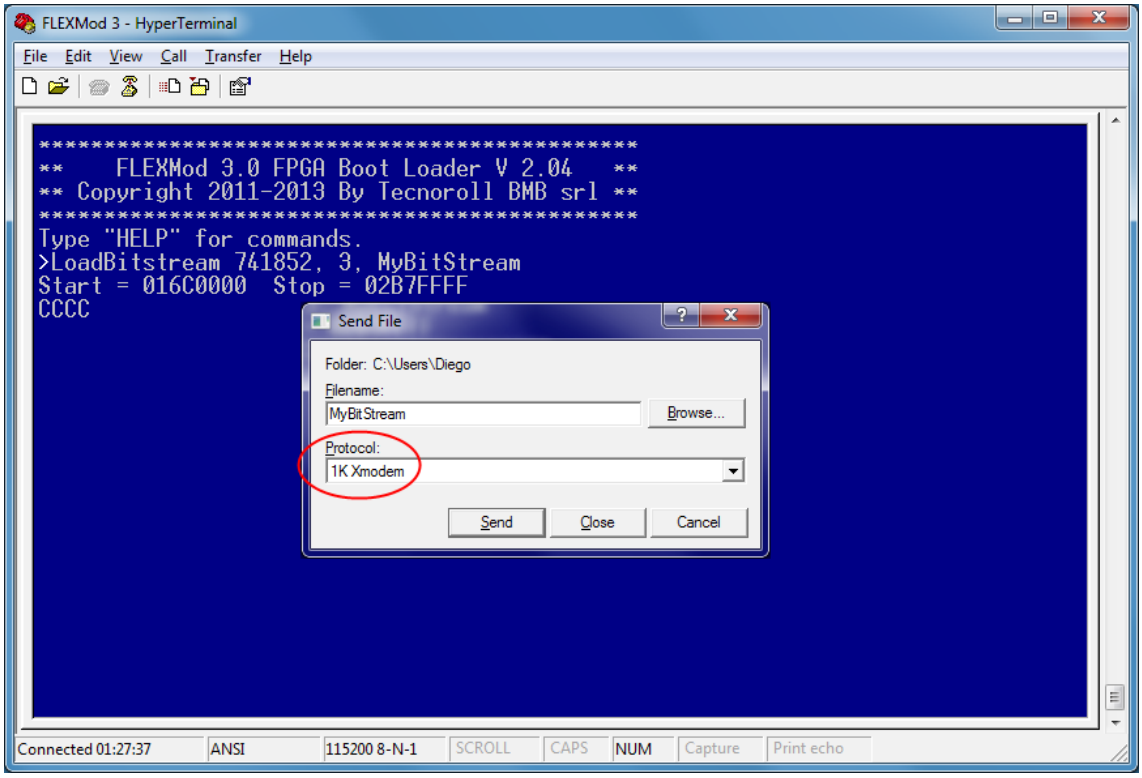
In the above screenshot you can see the command help with all the options and the typical syntax. Usually, only three parameters are enough to complete the process: the password, the location of the bitstream within the table and the bitstream name. The default transfer protocol is Xmodem-1K with CRC; if you want to change the upload mode, just add the optional corresponding parameters. Once the <ENTER> key is pressed, you will see a sequence of characters. This means that the module is ready to receive the bitstream to be loaded.



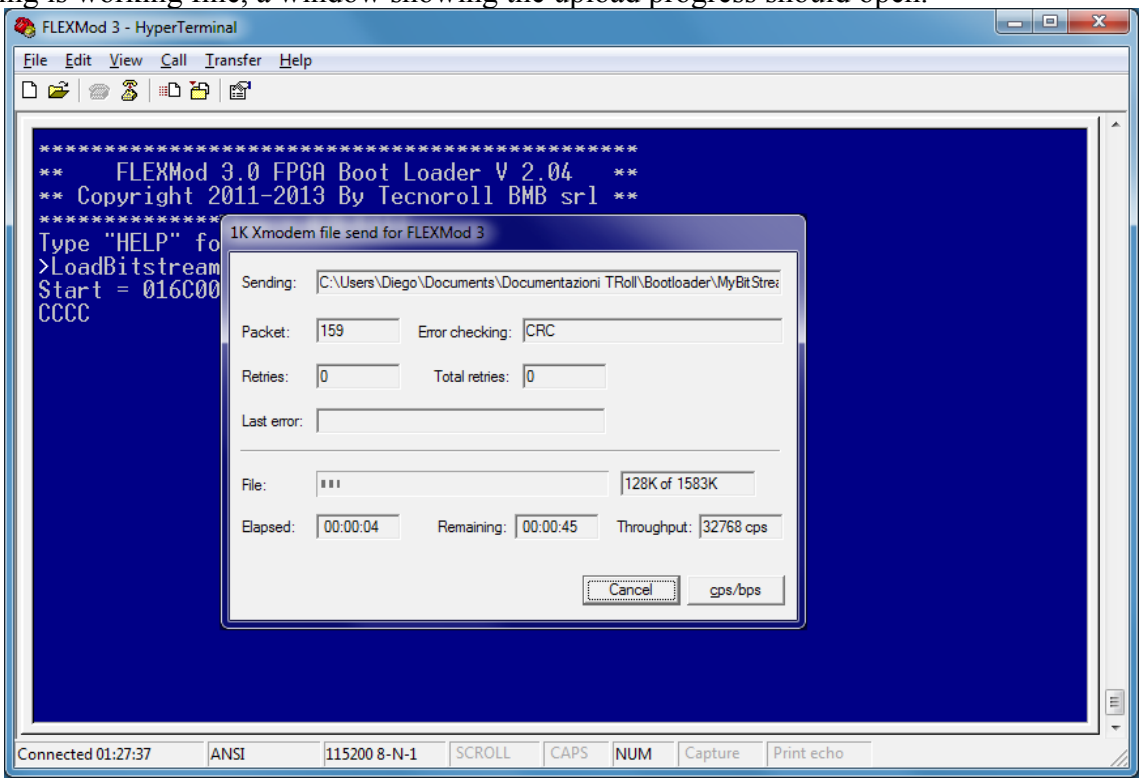
Select the "*Transfer*" and "*Send File*" option from the HyperTerminal menu.



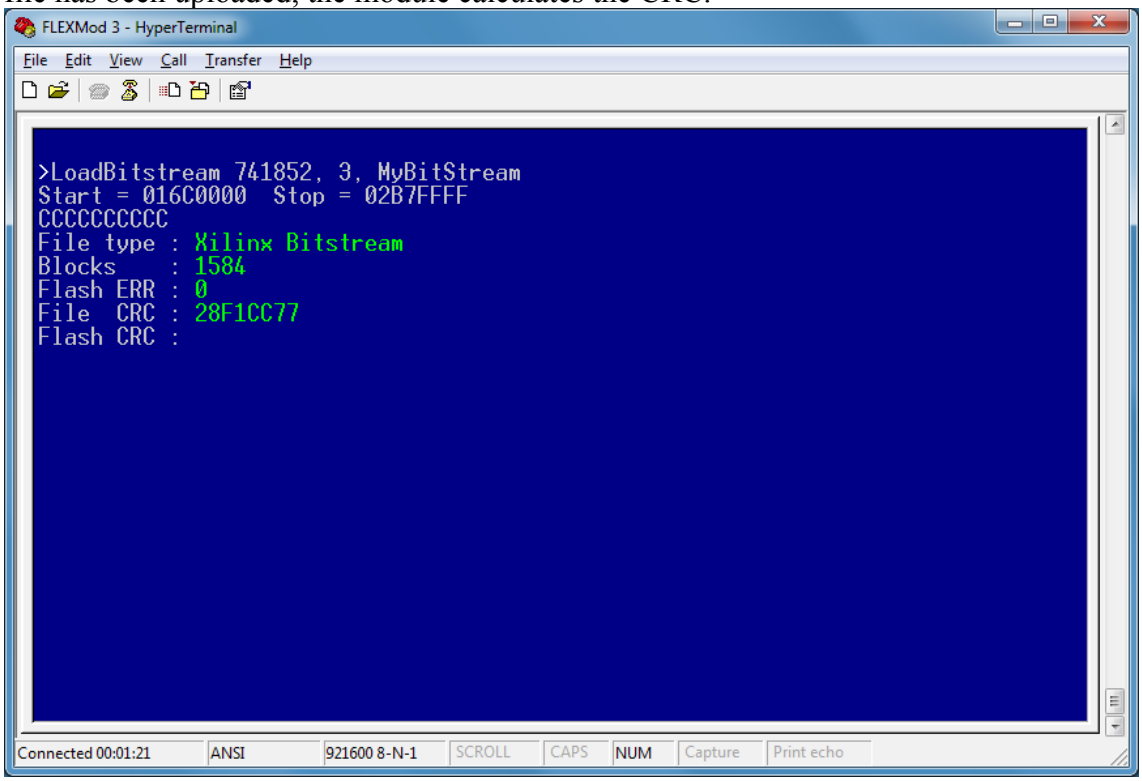
Select the file you want to upload and set the *"protocol"* to 1K Xmodem, then press the *"Send"* button.



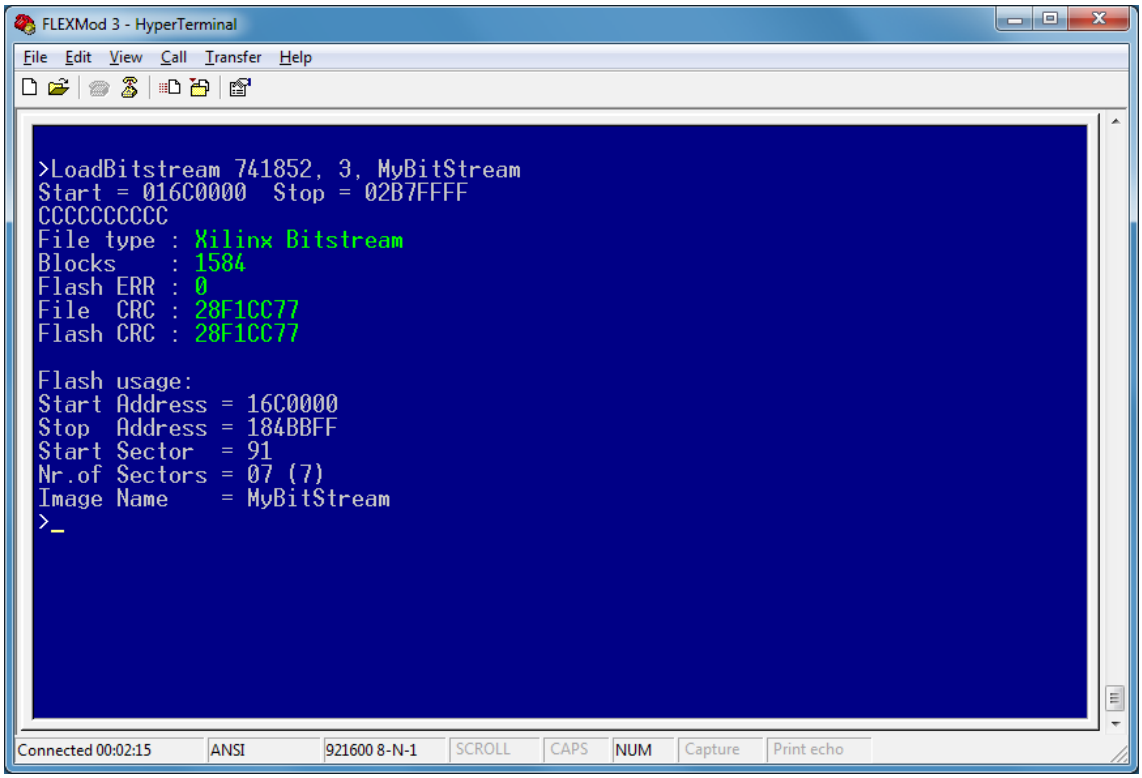
If everything is working fine, a window showing the upload progress should open.



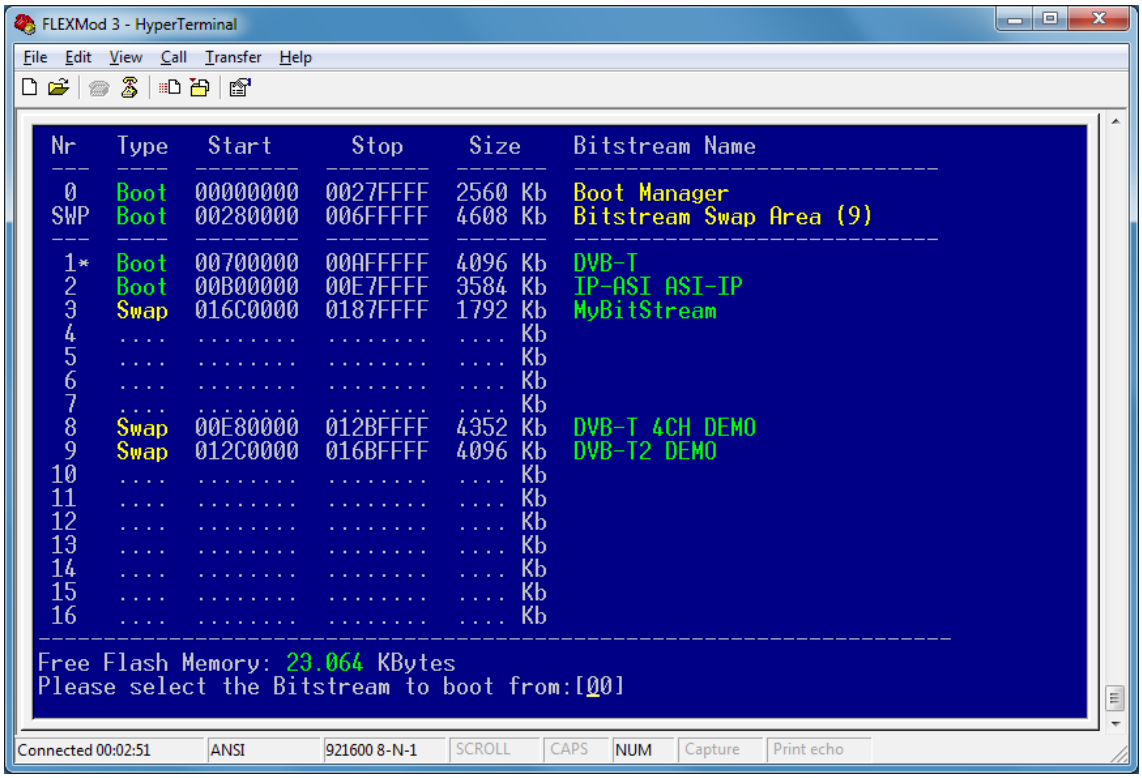
When the file has been uploaded, the module calculates the CRC.



After a few seconds, a report on the use of the flash memory is also displayed. The bitstream upload is now complete.

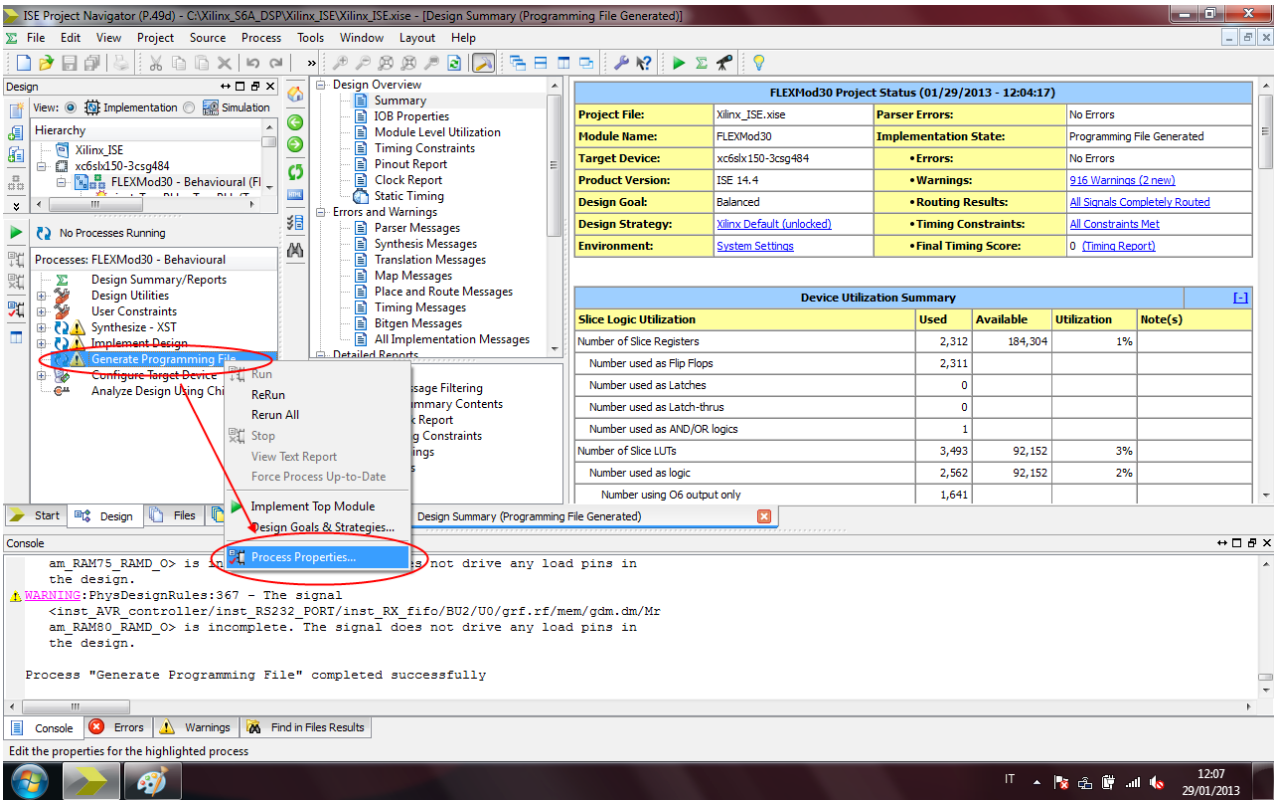


The loaded bitstream is now available in the table:

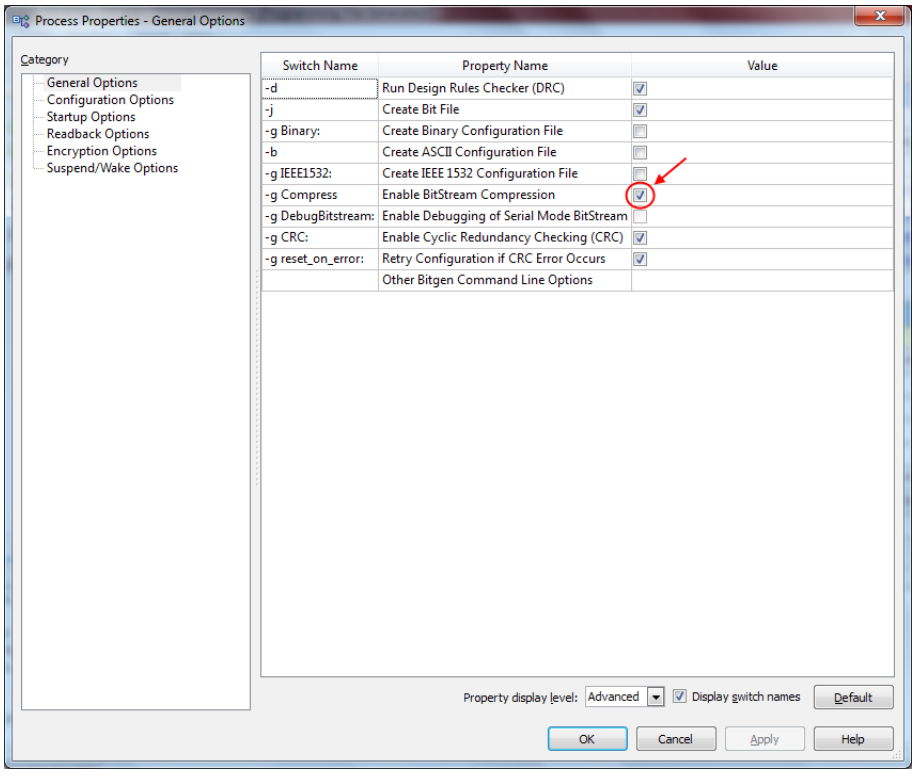


Addendum for IP Providers - Generating the bitstream file to be loaded into the FLEXMod module

The following screenshots show the correct way to generate the bitstream file to be loaded into the flash memory of the FLEXMod module.
In the Xilinx ISE Project Navigator, inside the *"Processes"* panel, select *"Generate Programming File"* and then *"Process Properties"*.

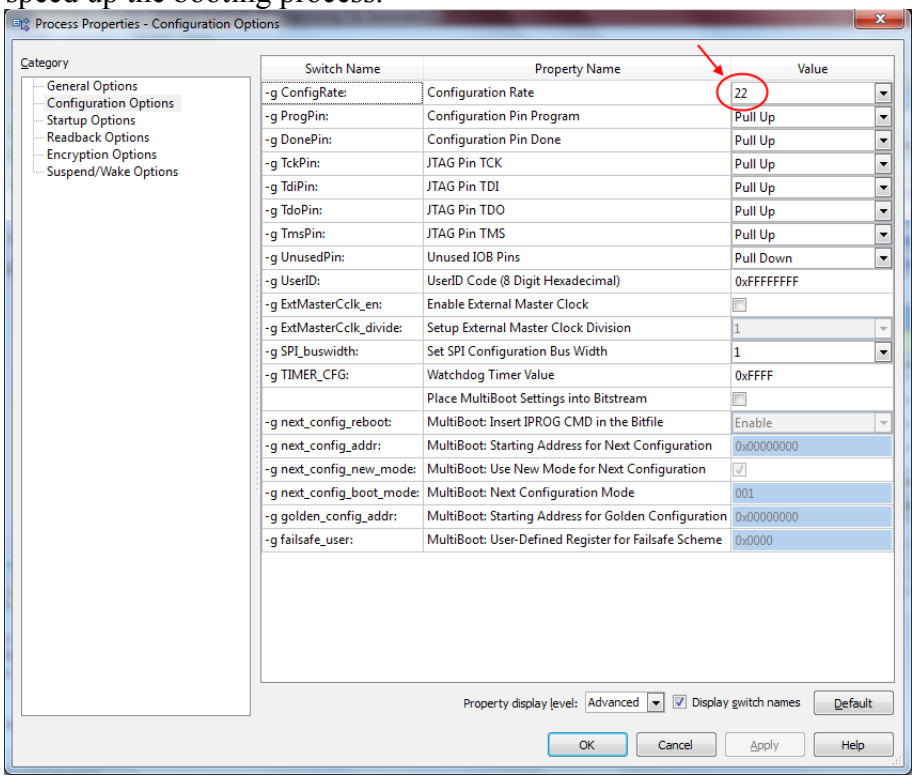


The *"Process Properties - General Options"* window opens, and we suggest you tick the "*Enable Bitstream Compression"* option to reduce the total dimensions of the file.

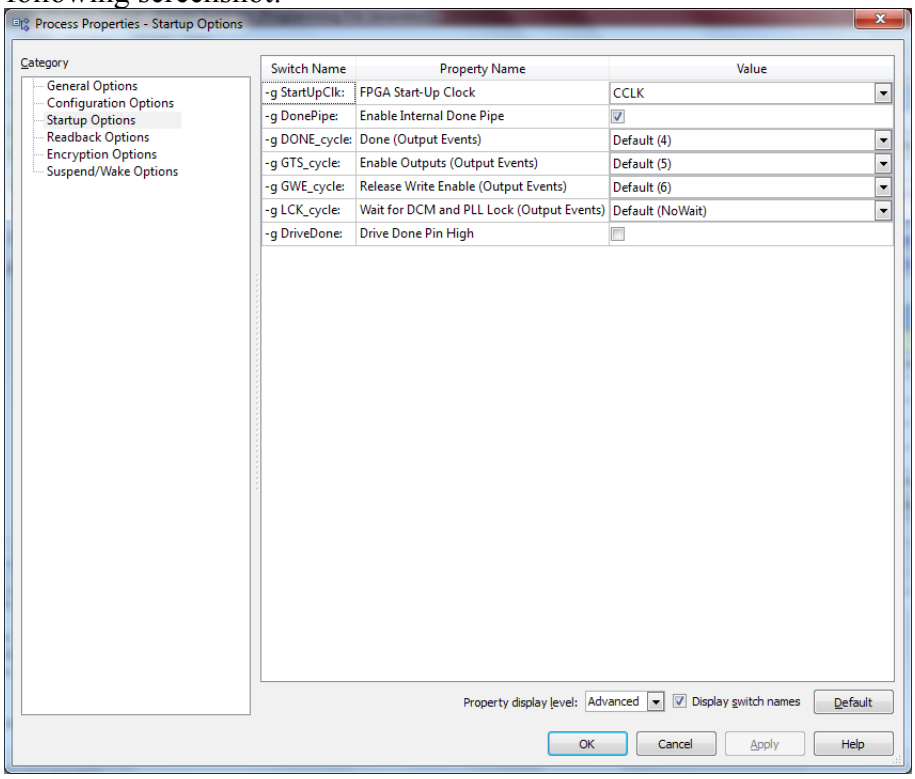


In the same window you should also select the *"Configuration Options"* and set the "*Configuration rate"* to

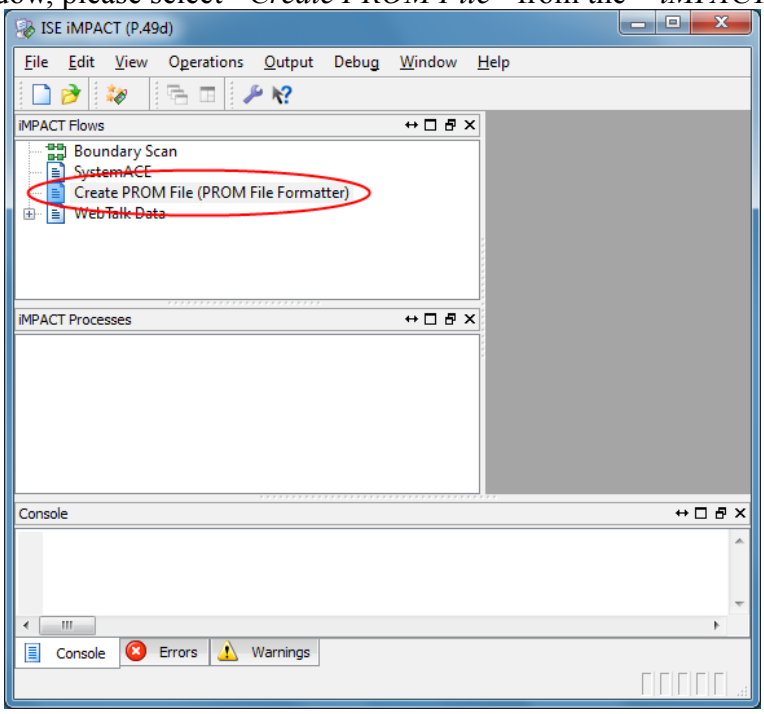
22MHz in order to speed up the booting process.



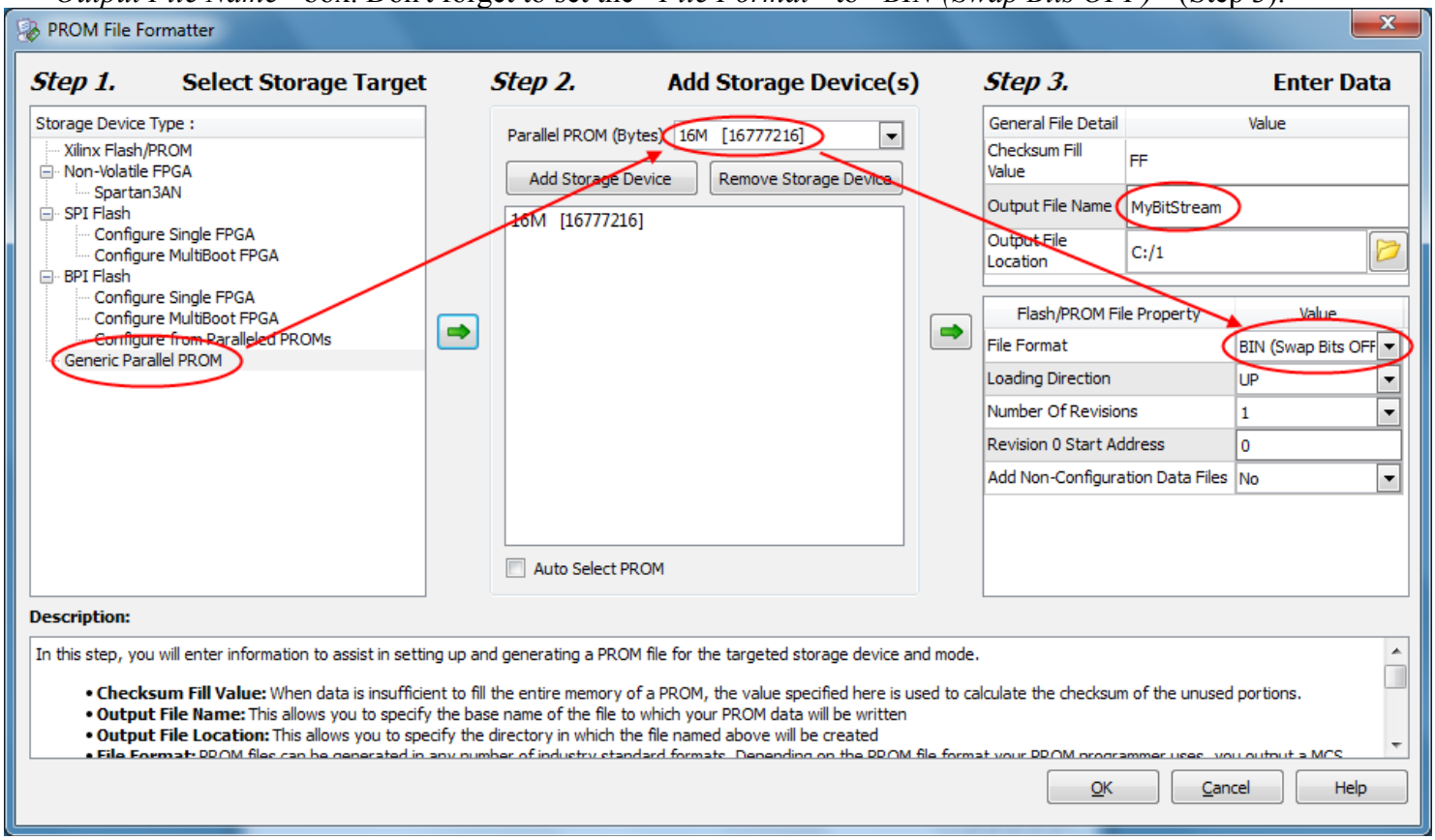
In *"Process Configurations"*, please select the *"Startup Options"* and make sure that the settings match those shown in the following screenshot:



Once you have generated the **.bit** file in ISE, use the ISE iMPACT software to generate the final bitstream. In the ISE iMPACT window, please select *"Create PROM File"* from the *"iMPACT Flows panel"*.



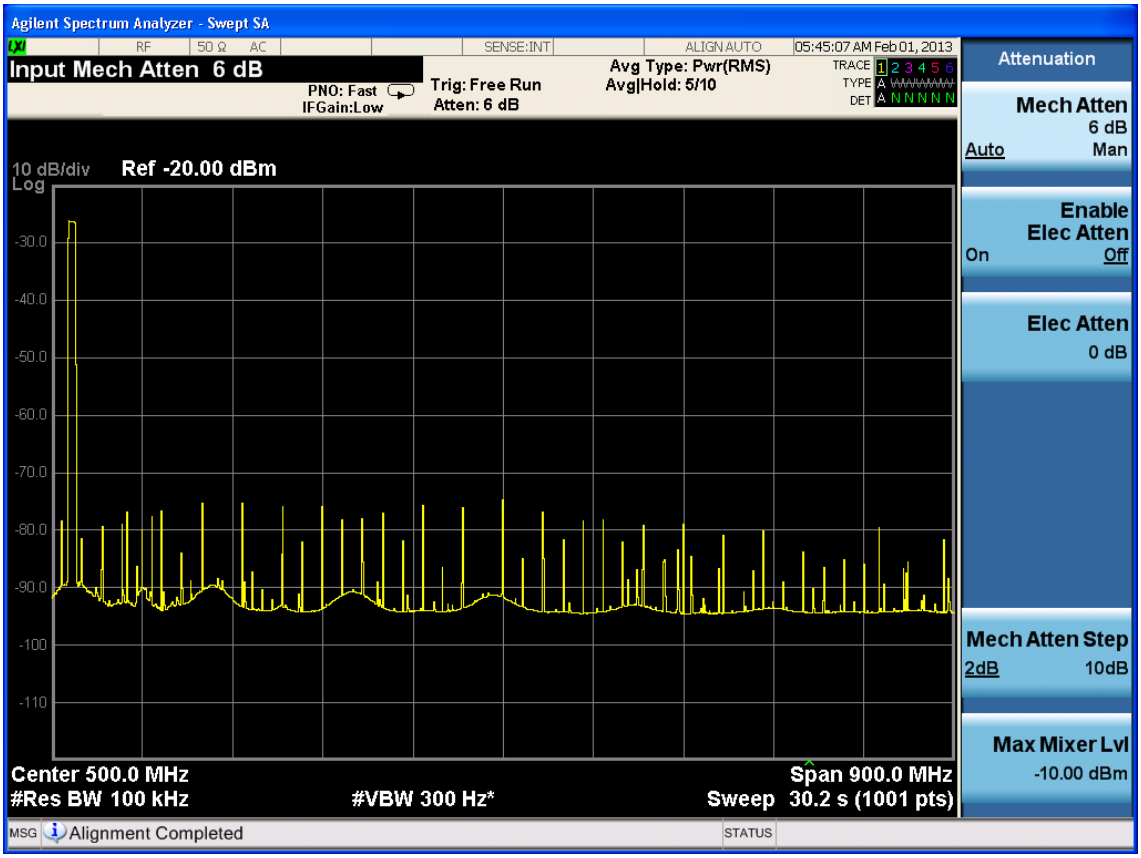
In the "*PROM File Formatter"* window, select "*Generic Parallel PROM*" (Step 1), then set *"16M [16777216]"* in the multiple-choice window (Step 2), and enter the name you want to give to the file in the "*Output File Name"* box. Don't forget to set the *"File Format"* to *"BIN (Swap Bits OFF)"* (Step 3).



Now you can proceed to generate the BIN file, which can now be uploaded into the module through the procedure described above (see "Loading a Bitstream").

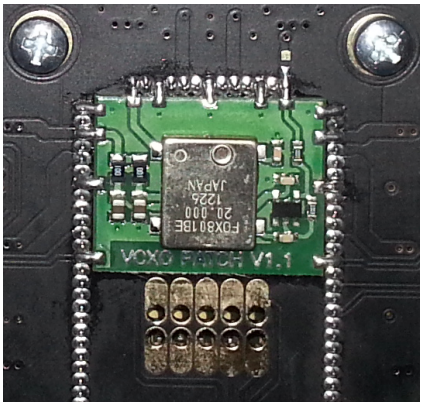
Known issue

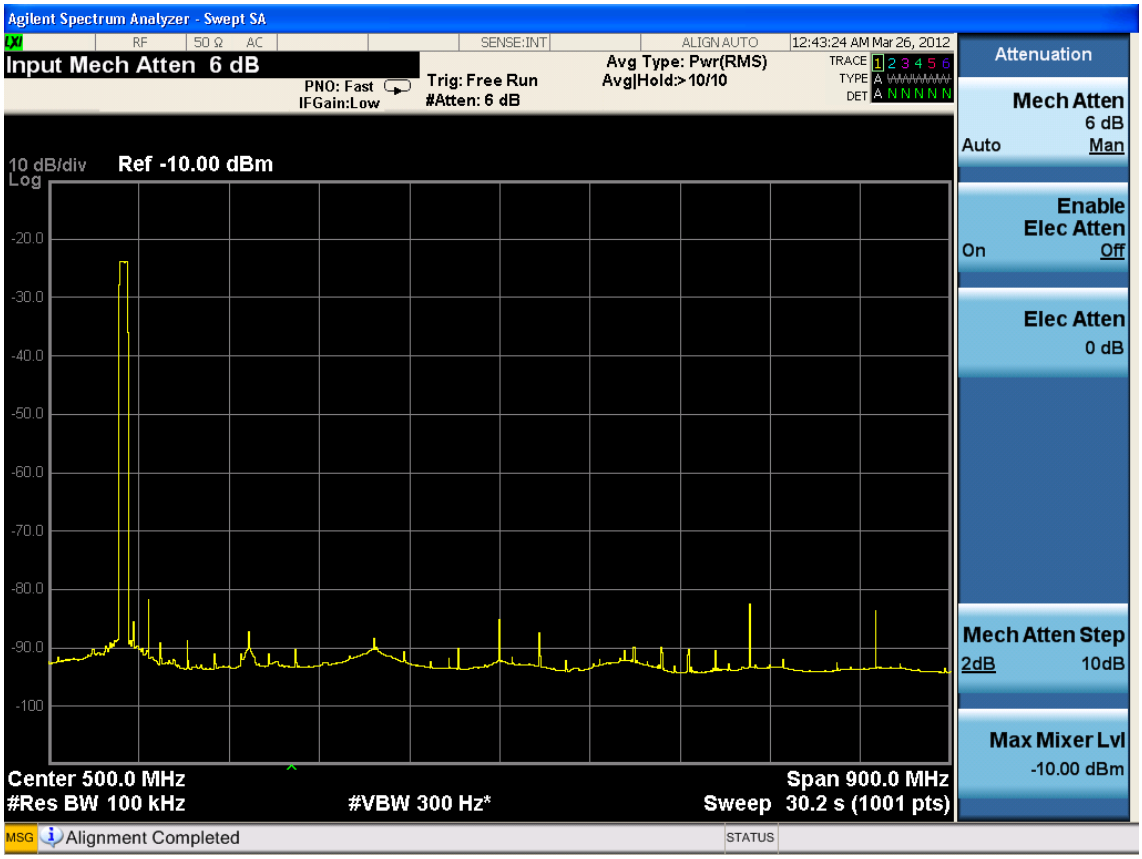
The presence of low power spurious signals in the spectrum output of the FLEXMod 3 starter kit is already known and is not due to any defects or malfunctions in the FLEXMod module.
The FLEXMod 3 chipset can work either with an internal clock source or with an external one. In order to obtain the best performance also with the DVB-T2 modulator, we have decided to mount an external oscillator (not envisaged in the original design) on the starter kit through an additional patch.
Although this patch, which is integrated in the current version of the starter kit, causes some spurious signals, these are in any case very low and have no effect whatsoever on the use of the starter kit itself.
In future versions of the starter kit, the additional oscillator will be integrated into the circuit, thus finally solving this little issue.
If the user designs his own PCB and embeds this oscillator correctly (if needed), he will obtain a much better output spectrum than that of the current version of our starter kit.



In the picture above you can see the output spectrum of the FLEXMod module with the external oscillator. The visible spurious signals are the harmonics of the reference 20 MHz oscillator mounted on the patch board.

Here below you can see the patch board with the oscillator:





In the above picture, you can see the output spectrum of the FLEXMod module without the external oscillator, or with an external oscillator properly implemented on the PCB.