SpartanMC

SpartanMC-SoC-Kit

Table of Contents

List of Figures

List of Tables

MANPAGE – SPARTANMC(7)

NAME

spartanmc – Toolkit for easy implementation of custom SoCs (System-on-Chip) on Xilinx FPGAs

SYNOPSIS

Global targets:

make cablesetup [**+group**=*GROUP*]

make integrity_check +target=*PLATFORM*

make man [*MANPAGE*]

make newproject +path=*PATH*

make reconfigure

make setup

make unconfigure

make [**what**]

Project targets:

DESCRIPTION

The SpartanMC-SoC-Kit is a set of tools for implementing FPGA-based SoCs. The implementation process does not require knowledge about hardware description languages such as Verilog or VHDL.

Based on the 18-bit SpartanMC microprocessor core the SoC-Kit provides a toolchain to allow easy implementation of a custom System-on-Chip (SoC) on a Xilinx FPGA. The frontend used is GNU *make* to invoke a number of underlying backend tools.

To compose an SoC, the GUI-based system builder *jConfig* will generate a set of hardware source code and configuration files based on the users configuration choices. The SoC then is synthesized from this set of files by invoking the corresponding tools from Xilinx ISE Suite. The whole process is driven by *make* which finally generates a bitfile that can be downloaded to your target FPGA.

The configuration also includes system firmware which is written in C and will be embedded into the design during synthesis. An optional bootloader allows for later update of the software components without re-synthesizing the hardware.

MAKE TARGETS

All steps to design a SoC are triggered by *make*. This section describes all available operations implemented as make targets.

Global targets

Global targets are available from the installation directory of the SpartanMC-SoC-Kit. This directory is called *SPARTANMC_ROOT* .

cablesetup   Creates a rules file (*.rules) understood by udev to ensure proper operation of the Xilinx programming tool *impact*. The rules will make sure our system loads the correct USB-firmware to enable the cable driver to detect your cable. If your system has restricted USB access, the option *GROUP* specifies which user group is granted access to the USB programming cable. If ommitted, the default group 'xilinx' is used. After running this target you have to copy the generated file to the proper place for udev-rules in order to take effect.

integrity_check   Performs an integrity check on the SpartanMC installation. This will run an automated sequence covering most of the functionality of the SpartanMC-SoC-Kit. The sequence will start with calling **make unconfigure** to get a clean installation directory. The next actions cover all setup steps followed by the creation of a test project. Finally, this project will be synthesized. If *PLATFORM* specifies any other value than 'nohw', the design will be implemented on the corresponding target platform. Otherwise, the sequence will be complete after bitfile generation. To get a list of supported platforms, call **make integrity_check** without any option. If the described sequence completes, the SoC-Kit most likely is properly installed and configured. If not, there may be a configuration problem or a functional issue concerning the toolchain. The test sequence will abort with an error message in that case.

man   Displays the SpartanMC manpage denoted by *MANPAGE*. Ommitting *MANPAGE* is equal to **make man spartanmc** and will show this manpage.

reconfigure   Runs **configure** with the same options and relevant environment variables as the last time the configure was explicitely invoked via the command line.

setup   Builds or updates all required components of the SpartanMC-SoC-Kit from the corresponding sources. Components that any other make targets depend on are automatically built when invoking that target (e.g. manpages are generated from the users manual sources when invoking **make man**).

unconfigure   Removes all files generated by **make** and **configure**. After running **make unconfigure** , your SpartanMC installation will be left in the same state as just after a fresh install.

what   Shows a list containing all currently available targets and a short description. The availability of some targets may depend on your host system configuration or the current state of your SpartanMC installation or current project.

SEE ALSO