CSG > Ariane Keller
epics.txt · Last modified: 2011/12/15 15:40 by arkeller

EPiCS

Links

- Official webpage: http://www.epics-project.eu/ [http://www.epics-project.eu/]
- EPiCS Githup top page: https://github.com/EPiCS/epics-org [https://github.com/EPiCS/epics-org]
- Wiki: https://github.com/EPiCS/epics-org/wiki/Work-Packages [https://github.com/EPiCS/epics-org/wiki/Work-Packages]
- EPiCS @ csg: http://www.csg.ethz.ch/research/projects/EPiCS [http://www.csg.ethz.ch/research/projects/EPiCS]
- Reconos official: http://reconos.de/ [http://reconos.de/]
- Reconos documentation: http://pc-techinf-4.cs.upb.de/twiki/bin/view/ReconOS/HowtosAndTutorials [http://pc-techinf-4.cs.upb.de/twiki/bin/view/ReconOS/HowtosAndTutorials]

USB Serial

- sudo modprobe usbserial
- insmod drivers/usb/serial/cp210x.ko (e.g. in eth/LANA/linux-2.6)
- start board
- entry in /dev/ttyUSB0

Mount tardis

- sudo mount -t cifs -o user=arkeller,name=homes.ee.ethz.ch \\\\homes.ee.ethz.ch\\arkeller tardis/
- besser: mount //homes.ee.ethz.ch/arkeller
- sudo mount -o remount,exec /home/arkeller/tardis

Show which process has which open files

lsof

Tools

- xmd = command line tool for downloading elf files (software) to the board
- impact = gui tool for downloading bit files (hardware) to the board
- ise = editor for hardware only FPGA designs
- edk = xps = embedded design kit, used for embedded designs where you can “click” your system together.
- planAhead = floorplanning (e.g., draw partial reconfiguration areas)
- fpga_editor = edit floorplan (for 9.1 tools) you have to do a “export DISPLAY=:0” first. Note,

slice coordinates are different from BRAM coordinates!

Linux on the ml605

- sources from http://pc-techinf-25.cs.upb.de:8088/ml605-linux/ [http://pc-techinf-25.cs.upb.de:8088/ml605-linux/]
- generate bitstream
  - cd edk
  - xilinx-12.3 -64 xps
  - menu: generate bitstream
- compile kernel
  - update PATH with path to microblaze compiler (microblaze-unknown-linux-gnu-)
  - cd linux-2.6-xlnx
  - make CROSS_COMPILE=microblaze-unknown-linux-gnu- ARCH=microblaze clean simpleImage.ml605_epics
- setup network interface
  - IP address: 192.169.30.1
  - netmask: 255.255.255.0
- setup nfs on the host system. http://wiki.ubuntuusers.de/NFS [http://wiki.ubuntuusers.de/NFS]
  - location: /exportfs/rootfs_mb
  - client IP address: 192.168.30.2
- Ensure that the ml605 board is configured for GMII mode (e.g. jumber J66 and J67 on pins 1-2) (SGMII is on pins 2-3)
- download the bitstream (edk-static/implementation/system.bit) and the kernel image (arch/microblaze/boot/simpleImage.ml605_epics) with the xilinx 12.3 tools! (12.1 does not work)
- if it doesnt work, turn off/on the board and try again - I think there is somehow a “race condition”
- make sure the IP address of the server stays configured! it might be lost during the boot of the board…

Linux on the xup

- with a ramdisk
  - put device tree from edk-static/ppc405_0/libsrc/device-tree/xilinx.dts to arch/powerpc /boot/dts/suitable_name.dts
  - compile kernel with ARCH=powerpc CROSS_COMPILE=powerpc-405-linux-gnu- simpleImage.suitable_name.dts
  - add ramfs as rootfs :
    - put ramdisk.image.gz in arch/powerpc/boot
    - compile with make ARCH=… CROSS_COMPILE=… simpleImage.initrd.suitable_name
  - download file to board dow arch/powerpc/boot/simpleImage.initrd.suitable_name.elf
  - update ramfs:
    - gunzip ramdisk.image.gz
    - mount ramdisk.image /mnt/ramdisk -t ext2 -o loop=/dev/loop0

  - edit in /mnt/ramdisk
  - umount /mnt/ramdisk
  - gzip -9 ramdisk.image
- kernel command line: console=ttyS0 root=/dev/ram ramdisk_size=100000 (how big ramfs is in blocks of 1024 bytes)

PATH TO UNISIM

/home/arkeller/tardis/ml605/reconos/reconos/support/simlibs/unisim

Path to functional blocks

/home/arkeller/fresh_reconos_githup_clone/reconos/demos/

Install Xilinx 12.3er tools on student maschines

1. copy DVD to harddisk
2. chown -hR studentname:user /opt
3. chmod a+x ./xsetup and ./bin/lin64/*
4. execute .xsetup on the copied 12.3er tools - wait…
5. in the .bashrc file add “source /opt/Xilinx/12.3/settings64.sh
6. export XILINXD_LICENSE_FILE=8181@lunghin.ee.ethz.ch
7. ln /usr/bin/make /usr/bin/gmake

Install Modelsim SE

1. http://model.com/content/modelsim-se-downloads-support [http://model.com/content/modelsim-se-downloads-support]
2. Installer only works with sun-java and not with open Java
3. aptitude purge openjdk-6-jdk openjdk-6-jre openjdk-6-jre-headless openjdk-6-jre-lib
4. aptitude install ia32-sun-java6-bin
5. export LM_LICENSE_FILE=8161@lunghin.ee.ethz.ch
6. export MGLS_LICENSE_FILE=8161@lunghin.ee.ethz.ch
7. ./install.linux
8. test modelsim: vsim

Setup ReconOS

1. git clone git@github.com:EPiCS/reconos.git
2. add to .bashrc source /path/to/reconos/setenv.sh
3. start a project, add the static file and set the correct path to your reference design
4. install ecos 2.0
5. copy the demo source from the corresponding demo (not as indicated in the user_guide_ecos)
6. install cheetah, indent, pyparsing (and of course python)

7. change the mssaddriver.py file to the old version
8. for the sw design: do not use make mrproper setup, but use
   I. (make deps)
   II. make setup.mb (mb for microblaze) make ecos.mb

Setup Ecos

1. download the ecos-install tool (this will install all the cross compilers etc.)
2. wget -passive-ftp ftp://ecos.sourceware.org/pub/ecos/ecos-install.tcl [ftp://ecos.sourceware.org/pub/ecos/ecos-install.tcl]
3. sh ecos-install.tcl (and select i386 and powerpc-eabi)
4. Download the official ecos 2.0 tree and gunzip it next to ecos 3.0
5. change the ecosenv.sh file to point to ecos-2.0 instead of ecos-3.0
6. edit .bashrc and add source /opt/ecos/ecosenv.sh and source recons/setenv.sh
7. edit .bashrc and add PATH=$PTATH:/path/to/EDK/gnu/microblaze/lin64/bin
8. Download the official ecos 2.0 tree
9. Replace the 2.0 config tool by the 3.0 config tool