NEURAghe Instructions

September 24, 2019

NEEDED:

- **sudo apt-get install libc6:i386 libstdc++6:i386**
- Vivado SDK
- Mentor QuestaSim

CONFIGURE ENVIRONMENT:

- select the target architecture (NEU_ARCH) on top of neu_env_conf.sh
- **source neu_env_conf.sh**
- modify paths in order to get the right Vivado and Vivado SDK location
- modify path to reach target board
- read also the NEURAghe_README

RUN SIMULATION:

from sw/nuraghe_MW/ $->$ cd $MW (path defined as environment variable)

- nuraghe_MW.c : uncomment "#define SIM" at the top of file
- nuraghe_MW.h : select the target board and architecture
- default_settings.h : set the desired parameters
- convoluzioni_SW/convoluzione.h : set parameters as in default_settings.h and uncomment #NEU_TCN

- ddr_emulation/settings_(ARCH).txt : set the right parameters
- [only for 2.0] neuraghe_(ARCH)/fe/rtl/includes/ulpsoc_defines.sv : uncomment ‘define SIM
- make build-rtl $->$ needed only for hardware changes
- make ddr_(ARCH)_for_sim $->$ compiles and executes the convoluzione.c and post_training.py to create synthetic activation and weight files for sim
- make ddr_(ARCH)_16_bit $->$ arranges simulation files
- make clean all l2size=2097152 stackSize=8192 nbPe=1 scmSize=1024 l1Size=31744 $->$ to compile MW including config settings
- make run gui=1 $->$ runs QuestaSim
- after simulation rerun "make ddr_(ARCH)_for_sim" to get hw and sw convolution comparision.

SYNTHESIS:

- [only for 2.0] neuraghe_(ARCH)/fe/rtl/includes/ulpsoc_defines.sv : comment ‘define SIM
- fe/rtl/ulpsoc $->$ in ulpsoc.sv select the target board
- from project_folder/fpga:

  source board_setup.sh && make synth-soc $->$ synthetize neuraghe soc to be included in the global project
- open vivado to run synthesis and implementation for the whole project
- generate bitstream with .bin file

EXECUTION ON TARGET BOARD:

- nuraghe_MW.c : comment "#define SIM" at the top of file
- nuraghe_MW.h : select the target board and architecture
- cd neuragheconvnet/

- test/conv_test/settings_(ARCH) $->$ edit
- types.h $->$ modify QF to be the same as in settings
- arrange synth weights (post training) and compile test to run in the target board:

  arch 2.0: **make comp_mw clean_pt pt_tcn clean all load NEURAGHE=1 FIXED=1 TEST=tests/conv_test TIME=1 CI=1**

  OR

  arch 1.9: **make comp_mw clean_pt pt clean all load NEURAGHE=1 FIXED=1 TEST= tests/conv_test TIME=1 CI=1**
- copy the architecture related bitstream to the board
- from the NEURAGHE_INSTALL_DIR in the board:

  sudo ./main -c weights -i ./Imgs -p 3 -h 224 -w 224 -o 1000 -b vivado_synth_bitstream.bin