slurm_hardware_run.py

```
import argparse
import os
from subprocess import call

# example call: python3 slurm_hardware_run.py jacobi3d

# grab all arguments
parser = argparse.ArgumentParser()
parser.add_argument("name")  # name of the design we want to synthesis
parser.add_argument("-N", default=1)  # nodes, default: 1
parser.add_argument("-n", default=1)  # task, default: 1
parser.add_argument("-c", default=1)  # cores, default: 1
parser.add_argument("--mem", default=32768)  # memory, default: 131072MB=128GB
parser.add_argument("-o", default="outfile")  # stdout, default: saved to outfile_NAME
parser.add_argument("-e", default="errfile")  # stderr, default: saved to errfile_NAME
parser.add_argument("-t", default="01:00:00")  # time requested hh:mm:ss, default: 24h
parser.add_argument("--partition", default="fpga")  # cluster queue, default: long
args = parser.parse_args()

if args.name is None:
    raise Exception("No design name specified, exit.")

header = [("-N", args.N), ("-n", args.n), ("-c", args.c), ("--mem", args.mem),
          ("-o", "{}_{}_run".format(args.o, args.name)), ("-e", "{}_{}_run".format(args.e, args.name)), ("-t", args.t),
          ("--partition", args.partition)]
home_dir = os.path.expanduser("~")

_SDK_PATH = "source /apps/ault/intelFPGA_pro/19.1/hld/init_stratix.sh\n"
_PYTHON_PATH = "module load python/3.7.2\n"
_CMAKE_PATH = "module load cmake/3.14.0\n"
_GCC_PATH = "module load gcc/8.3.0\n"

# generate slurm batch job file
with open("{}/{}_run.sh".format(home_dir, args.name), "w") as f:
    f.write("#!/bin/sh\n")
    for item in header:
        f.write("#SBATCH {} {}\n".format(item[0], item[1]))
    f.write(_SDK_PATH)
    f.write(_PYTHON_PATH)
    f.write(_CMAKE_PATH)
    f.write(_GCC_PATH)
    f.write("python3 code/run_program.py stencils/{}.json hardware -log-level 0\n".format(args.name))

call(["sbatch", "{}/{}_run.sh".format(home_dir, args.name)])
```