Thermal Model
ESESC Tutorial

Speaker: Elnaz Ebrahimi

ESESC

Department of Computer Engineering,
University of California, Santa Cruz
http://masc.soe.ucsc.edu

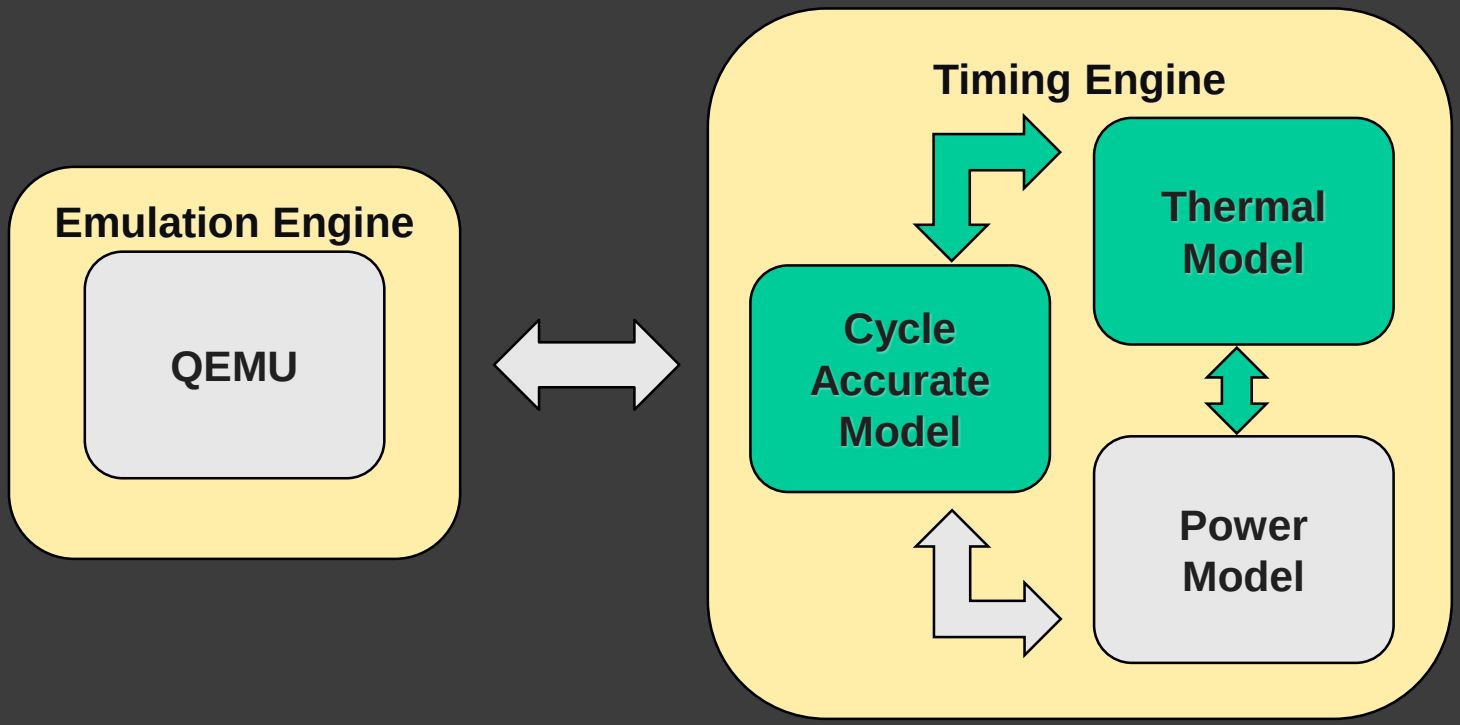
UC SANTA CRUZ

Thermal Model

- You will learn:
  - High level view of ESESC thermal model
  - How to setup and run the thermal model
  - How to generate a new floorplan
  - How to generate a heatmap

Outline

- Thermal Model
- Running Thermal Model (Demo 1)
- Changing the Floorplan
- Running Thermal Floorplanner (Demo 2)
- Enabling Thermal Map Graphics


Emulation Engine
QEMU
Timing Engine
Cycle Accurate Model
Thermal Model
Power Model

Thermal Model

- A Modified version of SescTherm [1]
- Thermal characterization such as scaling leakage based on
  - Temperature
  - Device Properties

[1] J. N.-Battilana and J. Renau, "*SOI, Interconnect, Package, and Mainboard Thermal Characterization*," in Proceedings of the 14th ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED), 2009, pp. 327–330.

SescTherm Main Files

- `SescTherm.cpp`
  - `esesc/pwth/libsesctherm/SescTherm.cpp`

  - Computes the temperature of floorplan blocks
  - Dumps temperature trace per floorplan block

SescTherm Main Files

- `ThermTrace.cpp`
  - `esesc/pwth/libsesctherm/ThermTrace.cpp`

  - Reads floorplan mapping
  - Reads energy numbers
  - Scales leakage based on temperature

SescTherm Main Files

- `ThermModel.cpp`
  - `esesc/pwth/libsesctherm/ThermModel.cpp`

  - Extracts layer information from `pwth.conf`
  - Partitions the floorplan
  - Creates solution matrices
  - Re-computes material properties

SescTherm Main Files

- `ChipFloorplan.cpp`
  - `esesc/pwth/libsesctherm/ChipFloorplan.cpp`
  - Reads and processes the floorplan based on floorplan information specified in `pwth.conf`

Thermal Model Requirements

- Power
- Performance
- Floorplan information and configuration
- Package information
- Thermal management policy

Thermal Model Config. File

`pwth.conf`

- Floorplan
- Layers (transistor, interconnect,…)
- Model config (temp and equation solver)
- Cooling Solution (air, oil)
- Package Configuration/Dimension
- Graphical thermal map
- Other layer configurations

Outline

- Thermal Model
- Running Thermal Model (Demo 1)
- Changing the Floorplan
- Running Thermal Floorplanner (Demo 2)
- Enabling Thermal Map Graphics

Main Thermal Settings

- In ~/build/release/run
  - esesc.conf
    - enablePower = true
    - enableTherm = true
    - thermTT   = 373
  - pwth.conf
    - enableTurbo = false
    - dumpPower   = true #dump leakage

Main Thermal Settings

- pwth.conf
  - [SescTherm] #section
  - floorplan[0] = ‘floorplan_1C’
  - layoutDescr[0] = ‘layoutDescr_1C’
- flp.conf
  - floorplan_1C
  - layoutDescr_1C

Thermal Model Output Files

- Thermal model related output files in `~/build/release/run`
  - ESESC configurations and statistics
    - `esesc_iscademo.?????`
  - Temperature trace
    - `temp_esesc_iscademo.?????`
  - Total Power
    - `totalpTh_esesc_iscademo.?????`

Thermal Model Reports

- Overall chip thermal related statistics
  - Dynamic power
  - Leakage Power
  - Gradient Temperature Across Chip
  - Average Temperature
  - Maximum Temperature
  - etc.
- Temperature per block
- Total dynamic + scaled leakage power

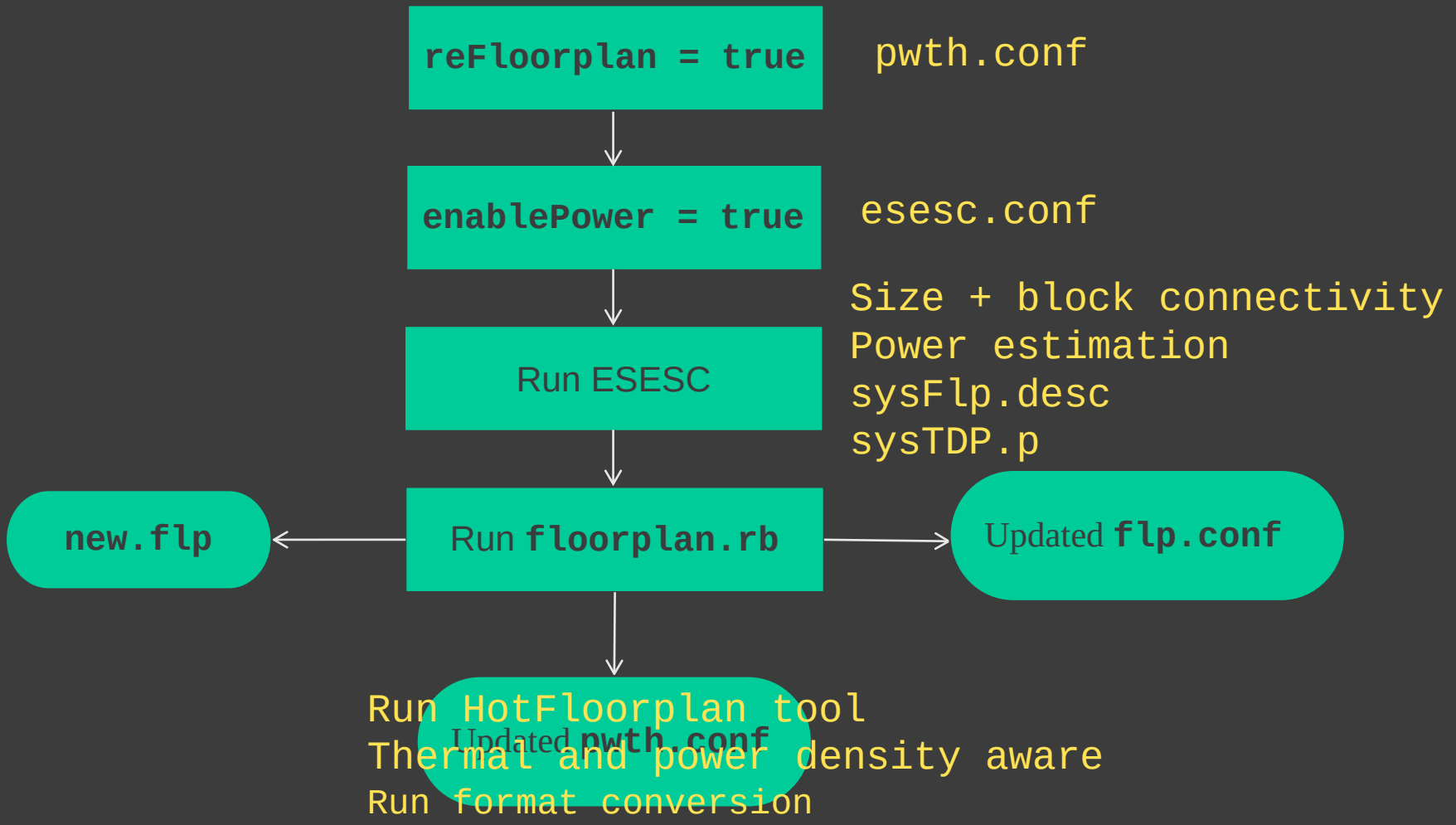
Demo 1

- Assume floorplanning and device parameters are set
- Enable power and thermal
- Setup thermal throttling
- Full thermal run with Crafty benchmark
- Extract thermal statistics
- Explain thermal related output files

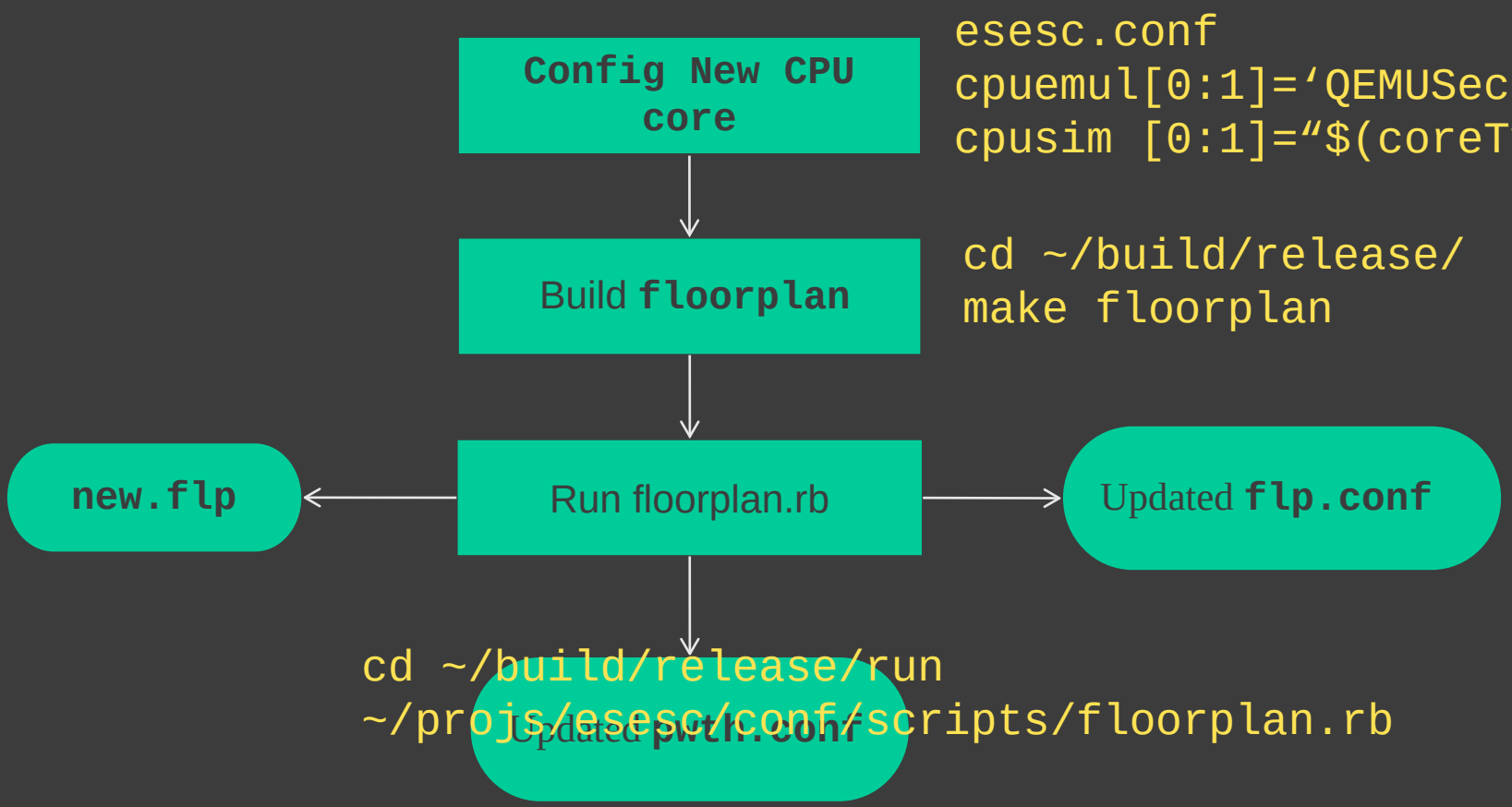
Outline

- Thermal Model
- Running Thermal Model (Demo 1)
- Changing the Floorplan
- Running Thermal Floorplanner (Demo 2)
- Enabling Thermal Map Graphics

Floorplan Tool



How to Generate a New Floorplan



Outline

- Thermal Model
- Running Thermal Model (Demo 1)
- Changing the Floorplan
- Running Thermal Floorplanner (Demo 2)
- Enabling Thermal Map Graphics

DEMO 2

- Change core config.
- Running floorplan tool for the new core config.
- Check the new floorplan settings
  - In `~/build/release/run/pwth.conf`
  - In `~/build/release/run/flp.conf`

Outline

- Thermal Model
- Running Thermal Model (Demo 1)
- Changing the Floorplan
- Running Thermal Floorplanner (Demo 2)
- Enabling Thermal Map Graphics

Temperature Map Graphics

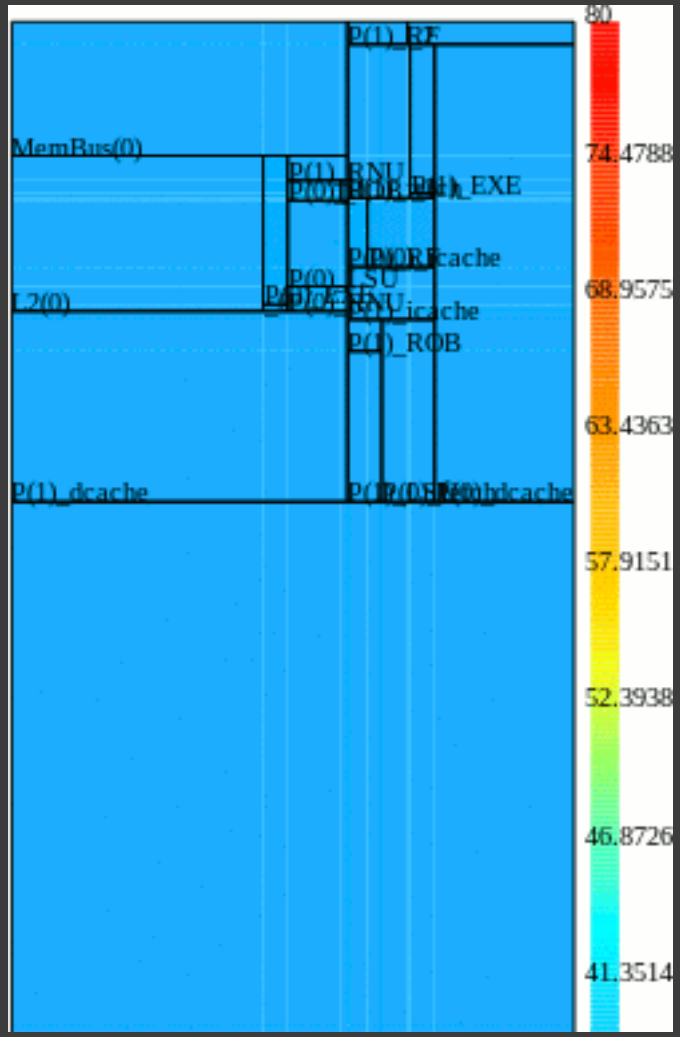
In `pwth.conf`

- `[graphics_config]`
  - Enable thermal map image dump
    - `enableGraphics = true`
  - Set the image resolution
    - `resolution_x  = 1024 #1440x900`
    - `resolution_y  = 768`
  - Link the floorplan layer
    - `grpahics_floorplan_layer = 2`

Temperature Map Graphics

- Convert the floorplan thermal map snapshots to gif
  - In `~/build/release/run`
  - `convert -delay 0.3 lcomp-NORM_layer-2_smpltype-CUR_0.* crafty.gif`

Thermal Map Example



Summary

- You will learn:
  - High level view of ESESC thermal model
  - How to setup and run the thermal model
  - How to generate a new floorplan
  - How to generate a heatmap

Backup Slides

Floorplan Tool References

[2] Ardestani, E.K.; Ziabari, A.; Shakouri, Ali; Renau, J., "Enabling power density and thermal-aware floorplanning," *Semiconductor Thermal Measurement and Management Symposium (SEMI-THERM), 2012 28th Annual IEEE* , vol., no., pp.302,307, 18-22 March 2012

[3] Hung, W. -L; Xie, Y.; Vijaykrishnan, N.; Addo-Quaye, C.; Theocharides, T.; Irwin, M.J., "Thermal-aware floorplanning using genetic algorithms," *Quality of Electronic Design, 2005. ISQED 2005. Sixth International Symposium on* , vol., no., pp.634,639, 21-23 March 2005

Generate Floorplan

- `floorplan.rb`
  - Change power, thermal, refloorplan flags
  - Run esesc to generate block connectivity and power estimation
  - Run `hotfloorplan` to generate floorplan
  - Convert the format for `pwth.conf`
  - Update `pwth.conf` with new floorplan
  - Update `esesc.conf` with floorplan link

How to Generate a New Floorplan

- Change single core to dual core
  - `esesc.conf`
    - `cpuemul[0:1]  = ‘QEMUSectionCPU’`
    - `cpusim [0:1]  = “$(coreType)”`
- In build directory
  - `~/build/release/`
- Run
  - `make floorplan`

How to Generate a New Floorplan

- In run directory
  - `~/build/release/run`
- Run
  - `~/projs/esesc/conf/scripts/floorplan.rb BuildDir_Path SrcDir_Path RunDir_Path NameMangle`

How to Generate a New Floorplan

Example command:

```
~/projs/esesc/conf/scripts/floorplan.rb
    ~/projs/build/release/
    ~/projs/esesc/
    ~/projs/build/release/run/
    2C
```

How to Generate a New Floorplan

- New links in `pwth.conf`
  - `floorplan[0]    = 'floorplan2C'`
  - `layoutDescr[0] = 'layoutDescr2C'`
- New layout/floorplan in `flp.conf`
  - `[layoutDescr2C]` …
  - `[floorplan2C]` …
- New floorplan also saved in `new.flp`

Detailed Package Configurations

- Defining chip layers
  - Add or define layers in `pwth.conf`

```
[SescTherm]
layer[0]= ‘mainboard0’       #mainboard
layer[1]= ‘interconnect0’    #metal
layer[2]= ‘die_transistor0’ #transistor
layer[3]= ‘bulk_silicon0’    #substrate
layer[4]= ‘air_layer0’       #air
```

Detailed Package Configurations

- [die_transistor0] #power layer
  - granularity = ‘x’ #(m)
  - floorplan = 2 #layer index
  - lock_temp = -1
- [air_layer0]
  - lock_temp = 25+273.15 #ambient T
  - floorplan = -1
- floorplan = -1
  - for all layers except die_transistor0

Detailed Package Configurations

- Package specific configuration sections
  - Model configuration
    - `Model = 'model_config'`
  - Thermal map image dump
    - `Graphics = 'graphics_config'`
  - Air or oil cooling solution
    - `Cooling = 'air_cooling_config`
  - Chip and package size and dimensions
    - `Chip = 'chip_config'`

Detailed Package Configurations

- `[model_config]`
  - matrix solver
    - `useRK4 = true`
  - initial temperature
    - `initialTemp = 35+273.15`
  - ambient temperature
    - `ambientTemp = 35+273.15`

Detailed Package Configurations

- `[chip_config]`
  - Chip dimensions: based on based on floorplan information (x, y)
    - `chip_width`
    - `chip_height`
    - `chip_thickness`
  - Package size: architectural decision
    - `package_height`
    - `package_width`
    - `package_thickness`

Detailed Package Configurations

- Cooling solutions
  - `[air_cooling_config]`
  - `[oil_cooling_config]`
  - Related code
    - `esesc/pwth/libsesctherm/ChipMaterial.cpp`
- For other `pwth.conf` configurations
  - Compare with default `pwth.conf` settings
  - Check source code

Turbo Mode

- Frequency changes based on temperature

~/projs/esesc/simu/libsampler/PowerModel.cpp

```
int PowerModel::updateFreqTurbo()
…
// Decide on the actual turbo frequency based on temperature
  if (maxT > K(100)) {
    turboFreq = getFreq();
    state = 4;
  } else if (maxT > K(90)) {
    turboFreq = maxF - 3*(maxF - getFreq())/4;
    state = 3;
  } else if (maxT > K(80)) {
    turboFreq = maxF - 2*(maxF - getFreq())/4;
    state = 2;
  } else ….
```