Automated Integration Of Tools for System Power Profiling and Temperature Modeling

Tools:- gem5,McPAT,Hotspot

SAURAV MALLA

Advisor:- Dr. Arnab Sarkar

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

Declaration

I hereby declare that except where specific reference is made to the work of others,the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Saurav Malla

July,2019

Acknowledgements

I would like to Thank Dr. Arnab Sarkar (Associate Professor) and Mr. Piyoosh(Research Scholar) and Mr. Sanjay Moulik(Research Scholar) for their guidence throughout the project. I would Also Like to thank the IITG management without their cooperation for the equipments and services the project would not have been possible. I would also like to thank my friends and parents for having belief in me. I would also thank those who have been directly or directly help me in this time at IITG as Summer Intern.

Saurav Malla

July,2019

Abstract

This Document is all about the Integration and Automation of Three Tools(gem5+McPAT+Hotspot) using Python Scripting and the Installation of all these Three tools. Document also talks about How One can Run His Own C Program in gem5 by creating its Binary. Document Shows the Performance of PARSEC benchmark in the integration script(Python). It Uses the simsmall.rcS Script file for all the PARSEC Benchmark Programs which is having more workload than test.rcS and simdev.rcS workloads.

Contents

Topic Name **Page Number**

1.Introduction

1.1) Gem5:-

GEM5 (http://www.m5sim.org/Main_Page) is a computer system simulation platform. Unlike the processor architecture simulator SimpleScalar, GEM5 can perform simulations for a complete multicore platform (that is processor, memory system, operating system itself). For processors, GEM5 is capable of simulating a number of ISAs, including ALPHA, ARM, MIPS, and X86.

There are two modes for GEM5 to simulate a system:

- **Syscall Emulation (SE)**: in this mode there is no operation system (OS). All the system calls in the application is emulated by GEM5.
- **Full System (FS)**: In this mode a complete system is simulated, including the OS and all peripherals. This is what we'll mostly use.

In GEM5 SE mode, the support for multithreaded applications is limited. Therefore, we are going to use the FS mode in this assignment and the next. In addition, we'll use the Alpha and X86 ISA's as the target architectures for the Integration of the Tools(mentioned above).

1.2) McPAT:-

McPAT (Multicore Power, Area, and Timing, http://www.hpl.hp.com/research/mcpat/) is an integrated power, area,and timing modeling framework for multithreaded, multicore, and manycore architectures. Basically, what McPAT does is to read in (micro-)architectural parameters and event statistics and to estimate the area, timing and power figures for each component of the system. It has models for different technology nodes from 90nm to 22nm. The accuracy depends on the level of details provided by the input.

1.3) HotSpot:-

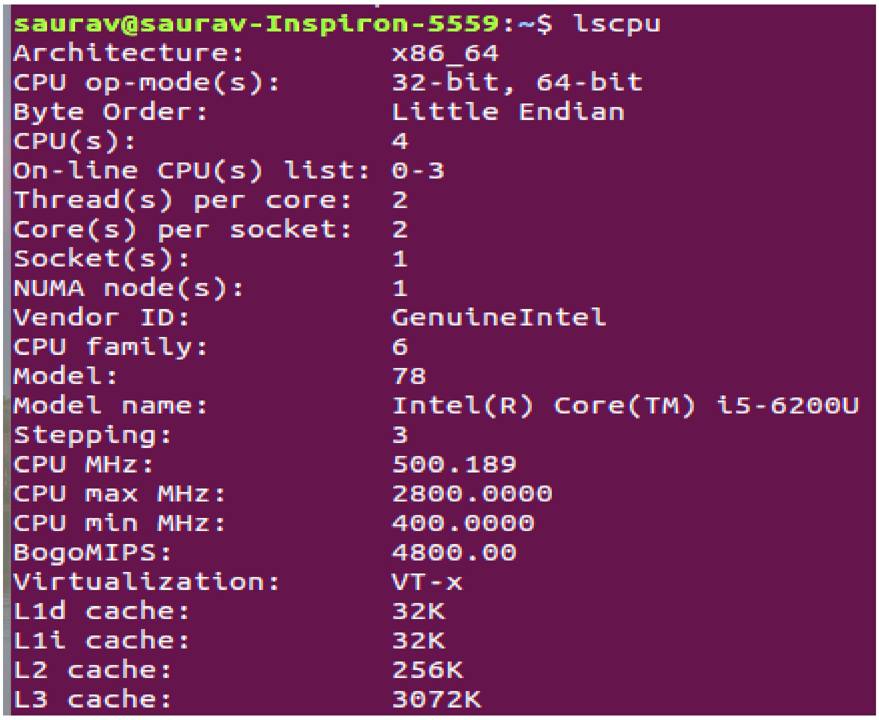
HotSpot is an accurate and fast thermal model suitable for use in architectural studies. It is based on an equivalent circuit of thermal resistances and capacitances that correspond to microarchitecture blocks and essential aspects of the thermal package. The model has been validated using finite element simulation. HotSpot has a simple set of interfaces and hence can be integrated with most power-performance simulators like Wattch. The chief advantage of HotSpot is that it is compatible with the kinds of power/performance models used in the computer-architecture community, requiring no detailed design or synthesis description. HotSpot makes it possible to study thermal evolution over long periods of real, full-length applications.

2) Installation:-

2.1) Gem5:-

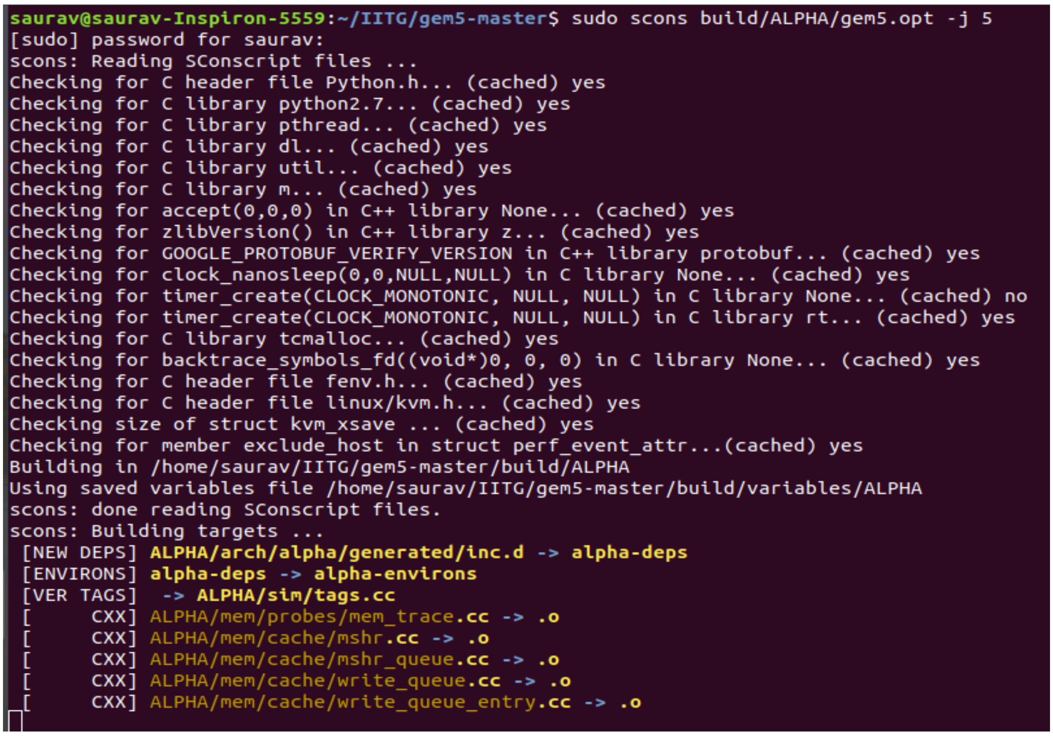
- To Install Gem5 you need to install some of the dependencies it required like **swig,scons,m4,python-dev** etc by running the following command in your terminal.
    - **sudo apt-get install build-essential python-dev scons swig m4**
- Then One More Dependency Needs to be Downloaded separately i,e **zliblg-dev** by running the following command in your terminal
    - **sudo apt-get install zliblg-dev**
- Download the gem5 Respiratory by Running the Following command in your Terminal:-
    - **sudo `git clone` [`https://gem5.googlesource.com/public/gem5`](https://gem5.googlesource.com/public/gem5)**
- For Building the Target Architectures in Your PC after dowloading the gem5 Respiratory there is several way.
    - First Change Your directory in the Terminal to the gem5 directory which you have just now Downloaded.
    - Then Run the Following command to build different Architectures in your PC.
        - **sudo scons build/<ISA>/<Binary> -j no. Of CPU’s +1**
        - ISA Available:
            - ALPHA, ARM, X86, MIPS, SPARC, RISC-V, NULL
        - Binaries Available:
            - gem5.opt, gem5.debug, gem5.prof, gem5.perf, gem5.fast
            - For Our Purpose We Have Used gem5.opt which is more widely used.
        - To know the no. Of CPU of your pc then you can run the following Command to know the Number of CPU’s in your PC.
            - **lscpu (See The Screenshot 2.1)**
        - Example:- Lets say we want to build the ALPHA and X86 ISA Using gem5.opt we Should run Following two commands respectively.
            - sudo scons build/ALPHA/gem5.opt -j 5
            - sudo scons build/X86/gem5.opt -j 5
            - For My PC the Number of CPU is 4 that’s why I have Used 5 after -j.

- **Screenshot 2.1)** For Checking the Number of CPU’s in PC



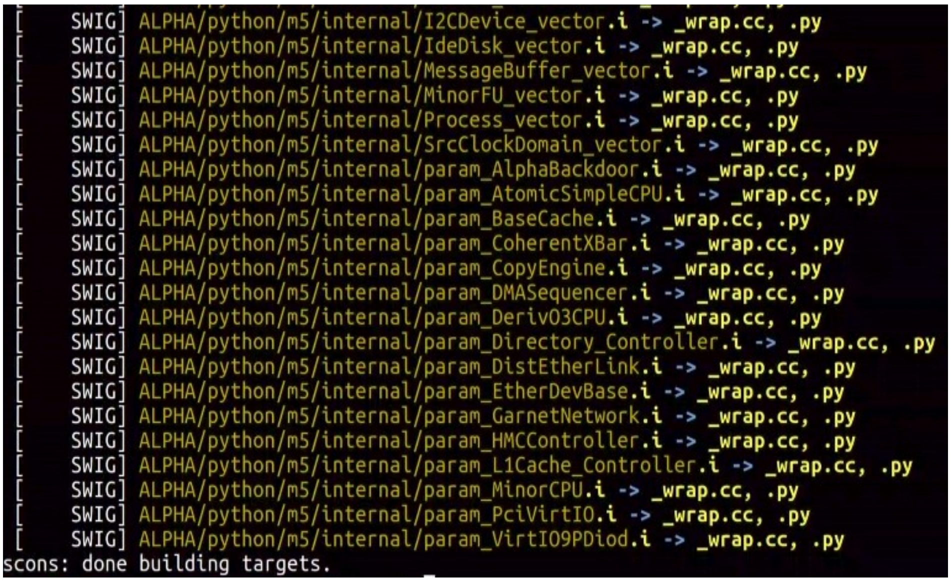
```
saurav@saurav-Inspiron-5559:~$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                4
On-line CPU(s) list:   0-3
Thread(s) per core:    2
Core(s) per socket:    2
Socket(s):             1
NUMA node(s):          1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 78
Model name:            Intel(R) Core(TM) i5-6200U
Stepping:              3
CPU MHz:               500.189
CPU max MHz:           2800.0000
CPU min MHz:           400.0000
BogoMIPS:              4800.00
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              3072K
```

- **CPU(S):4** -> So Number of CPU’s in my PC is 4. So check Your’s Respectively.
- After Running the Command to Build the ALPHA Architecture for gem5 you will see the Output in Terminal as Follows.(**See Screenshot 2.2**)
- Screeshot 2.2)



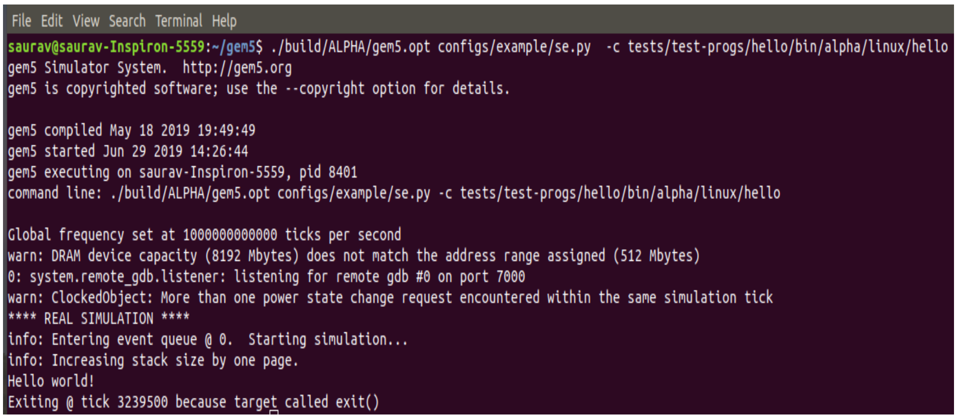
```
saurav@saurav-Inspiron-5559:~/IITG/gem5-master$ sudo scons build/ALPHA/gem5.opt -j 5
[sudo] password for saurav:
scons: Reading SConscript files ...
Checking for C header file Python.h... (cached) yes
Checking for C library python2.7... (cached) yes
Checking for C library pthread... (cached) yes
Checking for C library dl... (cached) yes
Checking for C library util... (cached) yes
Checking for C library m... (cached) yes
Checking for accept(0,0,0) in C++ library None... (cached) yes
Checking for zlibVersion() in C++ library z... (cached) yes
Checking for GOOGLE_PROTOBUF_VERIFY_VERSION in C++ library protobuf... (cached) yes
Checking for clock_nanosleep(0,0,NULL,NULL) in C library None... (cached) yes
Checking for timer_create(CLOCK_MONOTONIC, NULL, NULL) in C library None... (cached) no
Checking for timer_create(CLOCK_MONOTONIC, NULL, NULL) in C library rt... (cached) yes
Checking for C library tcmalloc... (cached) yes
Checking for backtrace_symbols_fd((void*)0, 0, 0) in C library None... (cached) yes
Checking for C header file fenv.h... (cached) yes
Checking for C header file linux/kvm.h... (cached) yes
Checking size of struct kvm_xsave ... (cached) yes
Checking for member exclude_host in struct perf_event_attr...(cached) yes
Building in /home/saurav/IITG/gem5-master/build/ALPHA
Using saved variables file /home/saurav/IITG/gem5-master/build/variables/ALPHA
scons: done reading SConscript files.
scons: Building targets ...
 [NEW DEPS] ALPHA/arch/alpha/generated/inc.d -> alpha-deps
 [ENVIRONS] alpha-deps -> alpha-environs
 [VER TAGS]  -> ALPHA/sim/tags.cc
 [     CXX] ALPHA/mem/probes/mem_trace.cc -> .o
 [     CXX] ALPHA/mem/cache/mshr.cc -> .o
 [     CXX] ALPHA/mem/cache/mshr_queue.cc -> .o
 [     CXX] ALPHA/mem/cache/write_queue.cc -> .o
 [     CXX] ALPHA/mem/cache/write_queue_entry.cc -> .o
```

- It will take 25-30 Minute to build the whole ALPHA ISA in gem5.
- After Completion of the Building targets in ALPHA Sucessfully You Should Get the Following in your Terminal(**See Screenshot 2.3**)
- **Screenshot 2.3)**



```
[     SWIG] ALPHA/python/m5/internal/I2CDevice_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/IdeDisk_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/MessageBuffer_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/MinorFU_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/Process_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/SrcClockDomain_vector.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_AlphaBackdoor.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_AtomicSimpleCPU.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_BaseCache.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_CoherentXBar.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_CopyEngine.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_DMASequencer.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_DerivO3CPU.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_Directory_Controller.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_DistEtherLink.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_EtherDevBase.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_GarnetNetwork.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_HMCController.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_L1Cache_Controller.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_MinorCPU.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_PciVirtIO.i -> _wrap.cc, .py
[     SWIG] ALPHA/python/m5/internal/param_VirtIO9PDiod.i -> _wrap.cc, .py
scons: done building targets.
```

- The Message **scons: done building targets** assures you that you have Succesfully build your targets for any particular ISA like ALPHA,X86,ARM etc.
- To check Whether the ISA you build just now is working Properly you can run the Simple Hello World Program as Follows:-
  - **./build/ALPHA/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/alpha/linux/hello**
  - This Will give you Output as Follows:-(**See Screenshot 2.4**)
    - Screenshot 2.4



```
saurav@saurav-Inspiron-5559:~/gem5$ ./build/ALPHA/gem5.opt configs/example/se.py  -c tests/test-progs/hello/bin/alpha/linux/hello
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled May 18 2019 19:49:49
gem5 started Jun 29 2019 14:26:44
gem5 executing on saurav-Inspiron-5559, pid 8401
command line: ./build/ALPHA/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/alpha/linux/hello

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
0: system.remote_gdb.listener: listening for remote gdb #0 on port 7000
warn: ClockedObject: More than one power state change request encountered within the same simulation tick
**** REAL SIMULATION ****
info: Entering event queue @ 0.  Starting simulation...
info: Increasing stack size by one page.
Hello world!
Exiting @ tick 3239500 because target called exit()
```

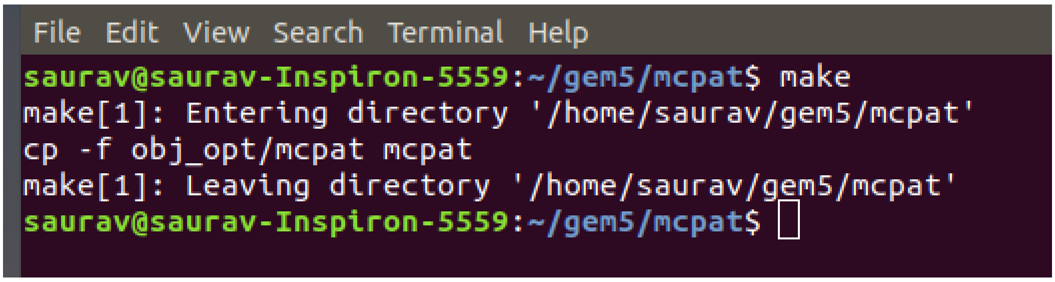
- The Message **Hello world!** Ensures that your gem5 simulation Running Successfully.
- The Output file you can see at **m5out** directory inside the gem5 directory where you will get 3 files as follows
  - **stats.txt, config.ini, config.json**
  - As Name Suggests stats.txt contains all the simulation Statistics.
  - Config.ini Contains all the Objects formed during the Simulation.
  - Config.json contains same result as config.ini file but as .json format.

  - The Same Way You can build the other Architecture like X86,MIPS,SPARC etc.

- **This Completes the Installation and building of gem5 Tool !!!**

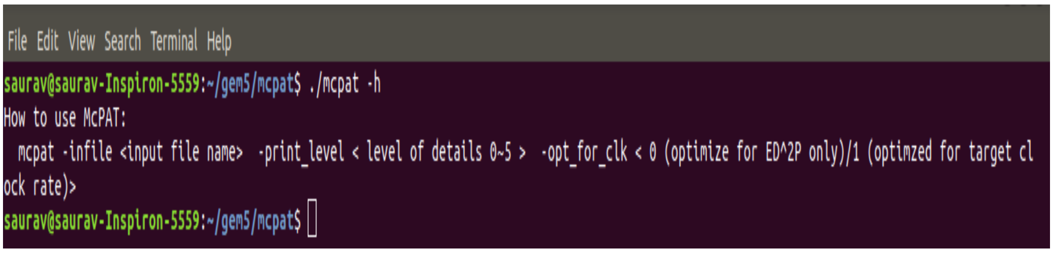
2.2) McPAT:-

- To Install the McPAT toll You need to Download the Respiratory from the Official Website
  - https://www.hpl.hp.com/research/mcpat/
- Or You can Directly Download the Respiratory from this Website
  - https://code.google.com/archive/p/mcpat/downloads
  - Download the First File i,e McPAT_v1.0.tar.gz
  - Unzip the File and That is Your McPAT Directory.
- Now to build or Install the McPAT just do as Following in your Terminal
  - Go to the McPAT directory Just Now You have Downloaded in terminal
  - run the **make** command in the terminal.
  - You Are Done With Installation and You will get the Output in terminal as follows(see Screenshot 2.5)
  - Screenshot 2.5



```
saurav@saurav-Inspiron-5559:~/gem5/mcpat$ make
make[1]: Entering directory '/home/saurav/gem5/mcpat'
cp -f obj_opt/mcpat mcpat
make[1]: Leaving directory '/home/saurav/gem5/mcpat'
saurav@saurav-Inspiron-5559:~/gem5/mcpat$
```

- To check if McPAT is Running Properly You Just need to run the Following Command
  - ./mcpat -h
  - This Will Show the Output as Following(see Screenshot 2.6)
  - Screenshot 2.6

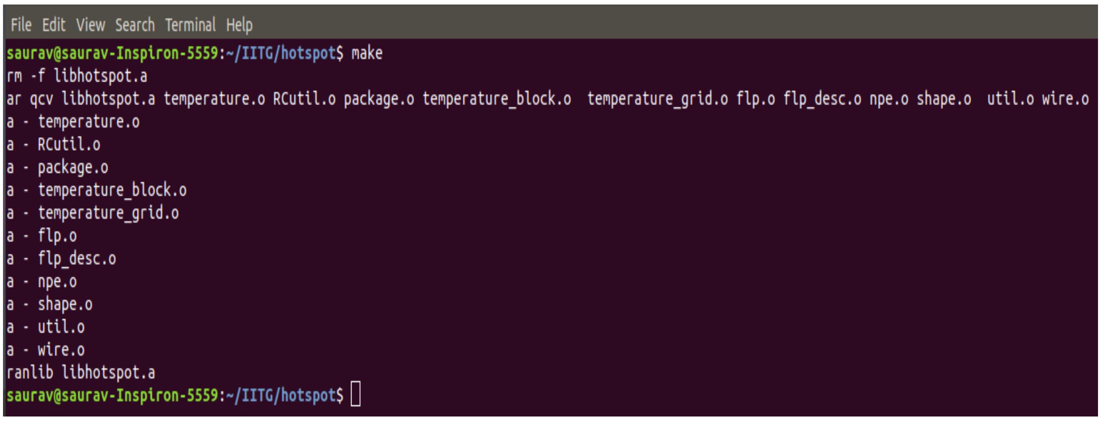


```
saurav@saurav-Inspiron-5559:~/gem5/mcpat$ ./mcpat -h
How to use McPAT:
  mcpat -infile <input file name>  -print_level < level of details 0~5 >  -opt_for_clk < 0 (optimize for ED^2P only)/1 (optimzed for target cl
ock rate)>
saurav@saurav-Inspiron-5559:~/gem5/mcpat$ 
```

- **This Completes Your Installation of McPAT Tool !!!**

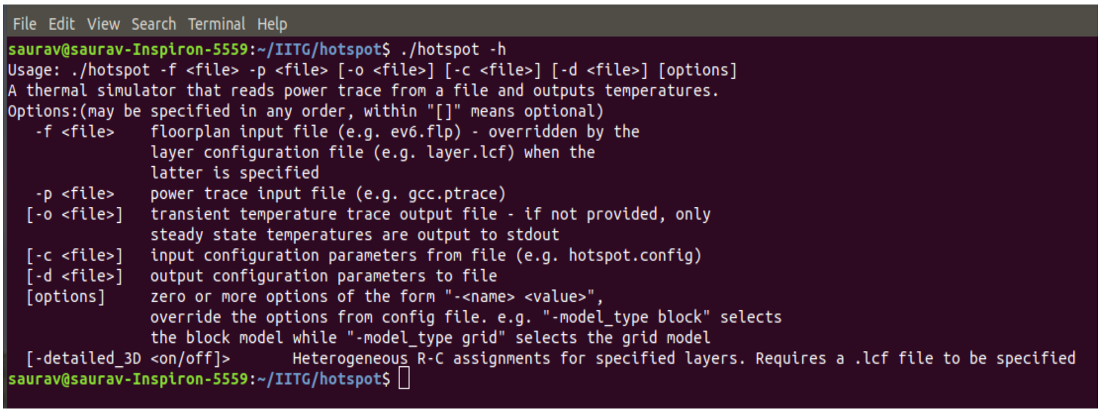
2.3) HotSpot:-

- To Install HotSpot You need to download the Hotspot Respiratory from the Following Official Website.
  - http://lava.cs.virginia.edu/HotSpot/download_form2.html
- Or You can download the Hotspot Respiratory directly from the following website
  - https://github.com/uvahotspot/hotspot
- After Downloading Unzip the file and that is your hotspot directory.
- Now go to the Hotspot Directory in the terminal.
- Now to install the Hotspot or Build the Hotspot You should run the make Command in the terminal.
  - Run **make** Command in the terminal.
  - You will get the Following in the Terminal.(see Screenshot 2.7)
  - Screenshot 2.7



```
saurav@saurav-Inspiron-5559:~/IITG/hotspot$ make
rm -f libhotspot.a
ar qcv libhotspot.a temperature.o RCutil.o package.o temperature_block.o  temperature_grid.o flp.o flp_desc.o npe.o shape.o  util.o wire.o
a - temperature.o
a - RCutil.o
a - package.o
a - temperature_block.o
a - temperature_grid.o
a - flp.o
a - flp_desc.o
a - npe.o
a - shape.o
a - util.o
a - wire.o
ranlib libhotspot.a
saurav@saurav-Inspiron-5559:~/IITG/hotspot$ 
```

- To check Whether the HotSpot is Running Properly Ypu can run the following command in the terminal
  - **./hotspot -h**
- You will get the following in your terminal(see Screenshot 2.8)
- Screenshot 2.8



```
saurav@saurav-Inspiron-5559:~/IITG/hotspot$ ./hotspot -h
Usage: ./hotspot -f <file> -p <file> [-o <file>] [-c <file>] [-d <file>] [options]
A thermal simulator that reads power trace from a file and outputs temperatures.
Options:(may be specified in any order, within "[]" means optional)
   -f <file>    floorplan input file (e.g. ev6.flp) - overridden by the
                layer configuration file (e.g. layer.lcf) when the
                latter is specified
   -p <file>    power trace input file (e.g. gcc.ptrace)
  [-o <file>]   transient temperature trace output file - if not provided, only
                steady state temperatures are output to stdout
  [-c <file>]   input configuration parameters from file (e.g. hotspot.config)
  [-d <file>]   output configuration parameters to file
  [options]     zero or more options of the form "-<name> <value>",
                override the options from config file. e.g. "-model_type block" selects
                the block model while "-model_type grid" selects the grid model
  [-detailed_3D <on/off]>       Heterogeneous R-C assignments for specified layers. Requires a .lcf file to be specified
saurav@saurav-Inspiron-5559:~/IITG/hotspot$ 
```

- **This Completes the installation of HotSpot Tool !!!**

3) A Brief About I/O of the Tools:-

3.1) Gem5:-

- **Input**
  - For Syscall Emulation(SE) Mode:-
    - A pre Compiled Binary for any task or Program like in Hello World Program.
  - For Full System Simulation(FS) Mode:-
    - A Script File (.rcS) file from which the Simulation to be run.
- **Output**
  - The Output is same for both the Mode it gives the Output in the **m5out** directory inside the gem5 directory.
  - Gem5 Produces three files for Output .
    - stats.txt

        - config.json
        - config.ini

3.2) McPAT:-

- **Input**
    - It takes the .xml file as input and the print_level for showing output as input also a option for clock.
    - We may give the print_level from 1 to 5 to see the output in different level.
    - Print_level 5 is preffered for detailed Output
    - opt_for_clk has two option 0 for normal ED^2P calculation and Option 1 for matching the target clock.
- **Output**
    - It gives the Output in the Terminal Only. We can Extract that into a txt file

3.3) HotSpot:-

- **Input**
    - It takes the Input of 2 files generally an floorplan file and a ptrace file
    - In Other mode like grid mode it takes a gri.steady file as to give the output in steady mode.
- **Output**
    - It gives the Output as steady temperature file in Normal Mode.
    - It gives the Output as grid.steady file in grid mode with a steady temperature file.
    - It also gives the Output in the terminal also for steady Temperature.

4) PARSEC Benchmark:-

4.1) Introduction:-

We Will work in the integrated model of these three tools(gem5,mcpat,hospot) with Parsec Benchmark. PARSEC Benchmark Suite is a Benchmark which Satisfies the following properties to become an efficient Benchmark for Monitoring the performance of the System. It Supports,

- **Multi-threaded Applications:-**

  Shared-memory CMPs are already ubiquitous. The trend for future processors is to deliver large performance improvements through increasing core counts on CMPs while only providing modest serial performance improvements. Consequently, applications that require additional processing power will need to be parallel.

- **Emerging Workloads:-**

  Rapidly increasing processing power is enabling a new class of applications whose computational requirements were beyond the capabilities of the earlier generation of processors. Such applications are significantly different from earlier applications . Future processors will be designed to meet the demands of these emerging applications and a benchmark suite should represent them.

- **Diverse:-**

  Applications are increasingly diverse, run on a variety of platforms and accommodate different usage models. They include both interactive applications like computer games, offline applications like data mining programs and programs with different parallelization models. Specialized collections of benchmarks can be used to study some of these areas in more detail,but decisions about general-purpose processors should be based on a diverse set of applications. While a truly representative suite is impossible to create, reasonable effort should be made to maximize the diversity of the program selection. The number of benchmarks must be large enough to capture a sufficient amount of characteristics of the target application space.

- **Employ State-of-Art Techniques:-**

  A number of application domains have changed dramatically over the last decade and use very different algorithms and techniques. Visual applications for example have startedto increasingly integrate physics simulations to generate more realistic animations. A benchmark should not only represent emerging applications but also use state-of-art techniques.

- **Support Research:-**

  A benchmark suite intended for research has additional requirements compared to one used for benchmarking real machines alone. Benchmark suites intended for research usually go beyond pure scoring systems and provide infrastructure to instrument, manipulate, and perform detailed simulations of the included programs in an efficient manner.

4.2) Workloads Or Input Sets of PARSEC Benchmark:-

PARSEC benchmark has 5 types Workloads in the Script File it Generates.The Five Type of Workloads Or Input Sets are:-

- test:-

  A very small input set to test the basic functionality of the program.

- simdev:-

  A very small input set which guarantees basic program behavior similar to the real behavior, intended for simulator test and development.

- simsmall:-

  Input sets of different sizes suitable for microarchitectural studies with simulators.

- simmedium:-

  Input sets of different sizes suitable for microarchitectural studies with simulators.

- simlarge:-

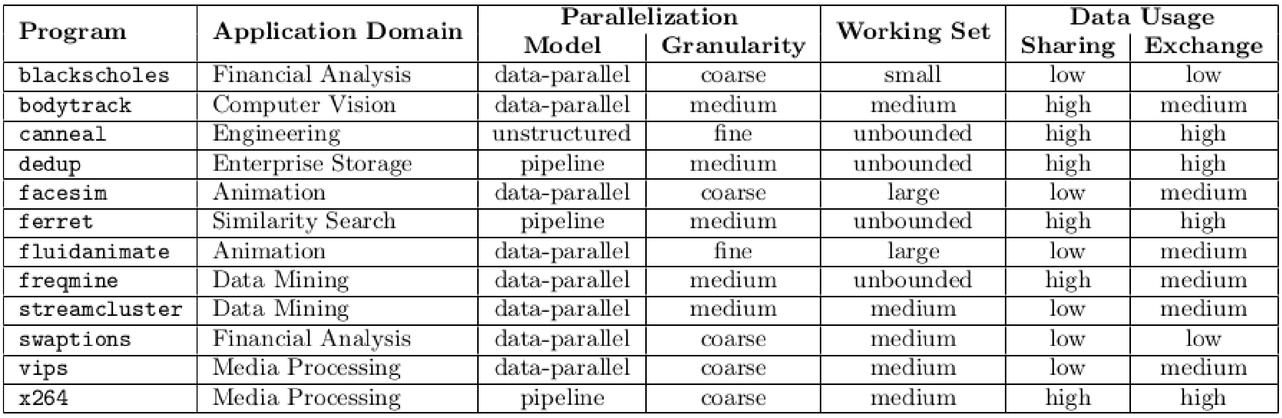
  Input sets of different sizes suitable for microarchitectural studies with simulators.

The test and simdev scripts has very small input sets intended to only use for testing purpose should not be used for scientific and simulation purpose.

For Other Three types of Input sets simlarge has the highest number of data and Parllelism. Simsmall has the Lowest among these three. More we increse the input set size more we get the detailed Output and Parllelism.

4.3) PARSEC Benchmark Programs:-

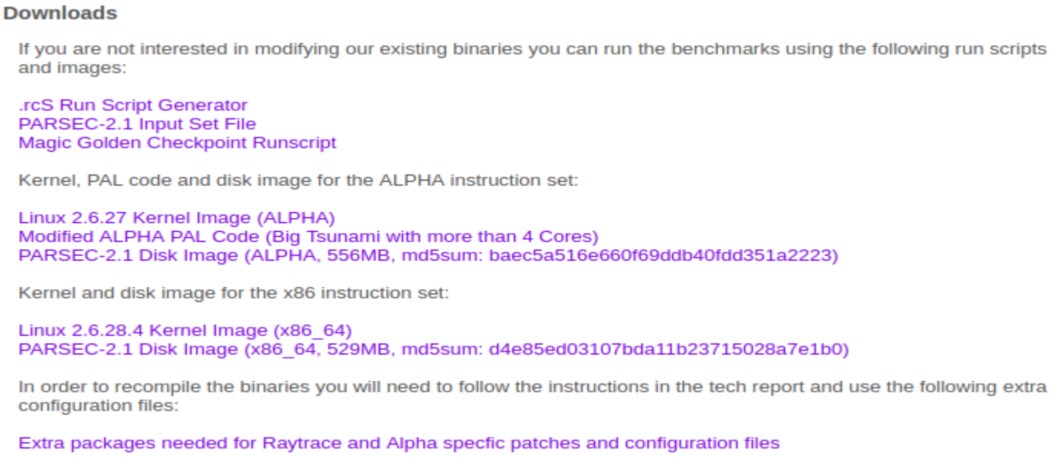
The PARSEC Benchmark has the Following set of Programs:-



Program	Application Domain	Parallelization		Working Set	Data Usage	
		Model	Granularity		Sharing	Exchange
blackscholes	Financial Analysis	data-parallel	coarse	small	low	low
bodytrack	Computer Vision	data-parallel	medium	medium	high	medium
canneal	Engineering	unstructured	fine	unbounded	high	high
dedup	Enterprise Storage	pipeline	medium	unbounded	high	high
facesim	Animation	data-parallel	coarse	large	low	medium
ferret	Similarity Search	pipeline	medium	unbounded	high	high
fluidanimate	Animation	data-parallel	fine	large	low	medium
freqmine	Data Mining	data-parallel	medium	unbounded	high	medium
streamcluster	Data Mining	data-parallel	medium	medium	low	medium
swaptions	Financial Analysis	data-parallel	coarse	medium	low	low
vips	Media Processing	data-parallel	coarse	medium	low	medium
x264	Media Processing	pipeline	coarse	medium	high	high

4.4) Downloading PARSEC Benchmark:-

- We can Download the PARSEC Benchmark 2.1 from the Official Website given below
  - **http://www.cs.utexas.edu/~cart/parsec_m5/**
  - Go to the Download Section of the Page where You will get Different Section of ISA's inside the Downllolad section.(See Screenshot 4.1)
  - Download Different PAL code,Kernal and Disk Image for PARSEC for different ISA's like ALPHA and X86 and move them to a directory as per your convenience.
  - Hope this Snap from the website Helps.
  - Screenshot 4.1



Downloads

If you are not interested in modifying our existing binaries you can run the benchmarks using the following run scripts and images:

.rcS Run Script Generator
PARSEC-2.1 Input Set File
Magic Golden Checkpoint Runscript

Kernel, PAL code and disk image for the ALPHA instruction set:

Linux 2.6.27 Kernel Image (ALPHA)
Modified ALPHA PAL Code (Big Tsunami with more than 4 Cores)
PARSEC-2.1 Disk Image (ALPHA, 556MB, md5sum: baec5a516e660f69ddb40fdd351a2223)

Kernel and disk image for the x86 instruction set:

Linux 2.6.28.4 Kernel Image (x86_64)
PARSEC-2.1 Disk Image (x86_64, 529MB, md5sum: d4e85ed03107bda11b23715028a7e1b0)

In order to recompile the binaries you will need to follow the instructions in the tech report and use the following extra configuration files:

Extra packages needed for Raytrace and Alpha specfic patches and configuration files

- You may face a problem while downloading the disk image in some case, just change the link to **https:// form   http://** in the link.

5) Running a C Program in Gem5:-

In this Section We Should Discuss How We Can run Our Own Written C Program in gem5. We can run the C Program in the Syscall Emulation Mode For Now. We Should Create the Linux Bianary of the C Program First. To run a C Program We should Follow the Following Steps for X86 Architeture(You can Use Other ISA's Also):-

- First Download the Cross Compiler for the X86 Architecture with following command in your terminal.
  - **sudo apt-get install build-essential**
- Then go to the directory where your C program is present(**be Sure that it Present inside the gem5 Directory**) in terminal.
- Then Cross Compile the C program with the following command in terminal.
  - **`gcc -static  -o binaryNameYouwantToCreate.x86 CprogramFileName.c`**
- Now We have Our Own Binary of Our Own C Program for running in gem5
- Run the gem5 in Syscall Emulation Mode for the Binary we have just created.
- Be Sure that Binary and the C Program is in a Same Directory and that Directory is Also inside the gem5 directory.
- Run gem5 in Syscall by Following Command.
  - **./build/X86/gem5.opt configs/example/se.py -c PathInsidegem5toBinary**

6) Running PARSEC in gem5:-

6.1) Downloading the Full System Files:-

This section discusses about the full system files of different ISA's for running the PARSEC benchmark.

- Download the Full System Files from Following Website
  - http://gem5.org/Download
  - Go to the ISA Section and Download the full system files.
  - Or You can Download From the Direct Links Given Below.
  - For ALPHA:- http://www.m5sim.org/dist/current/m5_system_2.0b3.tar.bz2
  - For X86:-

    http://www.m5sim.org/dist/current/x86/x86-system.tar.bz2
  - Unzip the files and You will get your Full system directory.

6.2) Swapping Files and Changing Python Scripts:-

We need to change the Disk files and Kernal and PAL code for the PARSEC with the Full system files of the different ISA. Else we cannot run the PARSEC Benchmark. Below is the explaination.

- **For ALPHA:-**
  - Make Directory named full_sys_simu (Or any name you want) and move the ALPHA Full system directory("**system**" for my PC) to that directory.
  - Inside the ALPHA Full system Directory You will get two Other Directories with following contents:-
    - binaries
      - console
      - ts_osfpal
      - vmlinux
    - disks
      - linux-bigswap2.img
      - linux-latest.img
  - We have Already Downloaded the PAL code ,kernal and disk image for PARSEC for diffrent ISA's as mentioned in 4.4 .

  - Now Copy the PAL code, kernal of ALPHA PARSEC that we have Previously Downloaded as in(4.4) and paste them in **full_sys_simu/system/binaries**(for my system,may be different for yours as you have made directories to store)
  - Delete the Previous file ts_osfpal and vmliunx file in that **full_sys_simu/system/binaries** directory.
  - Rename the PARSEC ALPHA PAL code ,kernal to **ts_osfpal** and **vmlinux** respectively in that directory.
  - Copy the PARSEC disk image for ALPHA and Paste it into the **full_sys_simu/system/disks** directory.
  - We need to change two python Files of gem5 located at gem5/configs/common
    - SysPaths.py
      - We need to change the line

        ```
        path = [ '/dist/m5/system', '/n/poolfs/z/dist/m5/system' ]
        ```

        to

        ```
        path = [ '/dist/m5/system', 'path_to_your_disk_and_binaries' ]
        ```

      - For My Case it will be like

        ```
        path = [ '/dist/m5/system','/home/saurav/IITG/full_sys_simu/system' ]
        ```

      - Every time You change the ISA You need to Change this File for Paths of the Disk and Binaries of the Full System Files of that ISA.
    - Benchmarks.py
      - We need to change

        ```
        elif buildEnv['TARGET_ISA'] == 'alpha':
                return env.get('LINUX_IMAGE', disk('linux-latest.img'))
        ```

        to

        ```
        elif buildEnv['TARGET_ISA'] == 'alpha':
            return env.get('LINUX_IMAGE', disk('linux-parsec-2-1-m5-with-test-inputs.img'))
        ```

      - Everytime you change the Benchmark for simulation you need to change this file.

- **<u>For X86:-</u>**
  - Make Directory named full_sys_simu (Or any name you want) and move the X86 Full system directory(“**system**” for my PC) to that directory.
  - Inside the X86 Full system Directory You will get two Other Directories with following contents:-

        - binaries
            - x86_64-vmlinux-2.6.22.9
        - disks
            - x86root-parsec.img
            - linux-latest.img
    - We need to get the **linux-bigswap2.img** file from the ALPHA full system file to the **full_sys_simu/x86/disks** directory.
    - We have Already Downloaded the PAL code ,kernal and disk image for PARSEC for diffrent ISA’s as mentioned in 4.4 .
    - Now Copy the kernal of X86 PARSEC that we have Previously Downloaded as in(4.4) and paste them in **full_sys_simu/x86/binaries**(for my system, may be different for yours as you have made directories to store)
    - Paste the Kernal to the Directory **full_sys_simu/x86/disks**
    - Copy the PARSEC disk image for X86 and Paste it into the **full_sys_simu/x86/disks** directory.
    - We need to change two python Files of gem5 located at gem5/configs/common
        - SysPaths.py
            - We need to change the line

              ```
              path = [ '/dist/m5/system', '/n/poolfs/z/dist/m5/system' ]
              ```

              to

              ```
              path = [ '/dist/m5/system', 'path_to_your_disk_and_binaries' ]
              ```

            - For My Case it will be like

              ```
              path = [ '/dist/m5/system','/home/saurav/IITG/full_sys_simu/x86' ]
              ```

            - You need to change this file everytime you change the ISA.
        - Benchmarks.py
            - We need to change

              ```
              elif buildEnv['TARGET_ISA'] == 'x86':
                  return env.get('LINUX_IMAGE', disk('x86root.img'))
              ```

              to

              ```
              elif buildEnv['TARGET_ISA'] == 'x86':
                  return env.get('LINUX_IMAGE', disk('x86root-parsec.img'))
              ```

            - You need to change this file everytime you change the Benchmark.

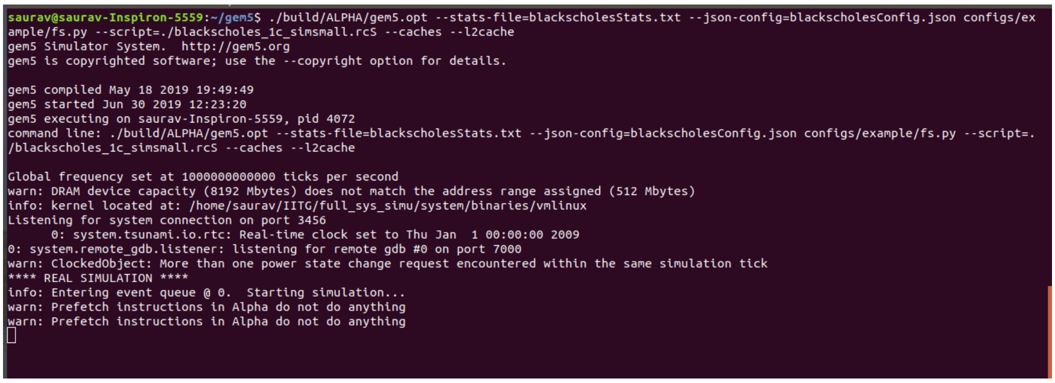
6.3) Generating Script Files For PARSEC:-

- For Running the PARSEC benchmark Programs(mentioned in 4.3) in gem5 we need to Generate the Script files i,e .rcS files.
- For Generating The Script Files of PARSEC we need to download a Script Generating Directory from the Following website
    - http://www.cs.utexas.edu/~parsec_m5/TR-09-32-parsec-2.1-alpha-files.tar.gz
- After Downloading the Zip file from the above direct link We can Unzip the file and Obtian Our Script Generating Directory.
- In that Directory You Will get a file named **writescripts.pl,** and that is the file by which we will generate the script files for all the PARSEC Program.
- Now Go to the Script Generating directory in terminal.
- Since **writescripts.pl** is a perl script, we need to take the Permission from OS to run this Script, for taking permission from the OS run the Following Command in the Terminal.
    - **chmod 777 writescripts.pl**
- After making the writescripts.pl executable we can generate the Script file by following command
    - **./writescripts.pl <Parsec Bench Program Name> <No. Of threads of Program>**
    - For Example for generating the scripts for blackscholes program of PARSEC with 1 thread we need to run following Command in terminal.
        - **./writescripts.pl blackscholes 1**
- After Running this Command the perl scripts will generate 5 .rcS file as following
    - blackscholes_1c_test.rcS
    - blackscholes_1c_simdev.rcS
    - blackscholes_1c_simsmall.rcS
    - blackscholes_1c_simmedium.rcS
    - blackscholes_1c_simlarge.rcS
- You can Generate the .rcS file for other Programs of PARSEC.
- We Will Work with simsmall .rcS files so We Have Copied the simsmall files for Some Programs and Pasted them in the gem5 directory.
- You are free to generate any script for any program.

6.4) Run PARSEC in Gem5:-

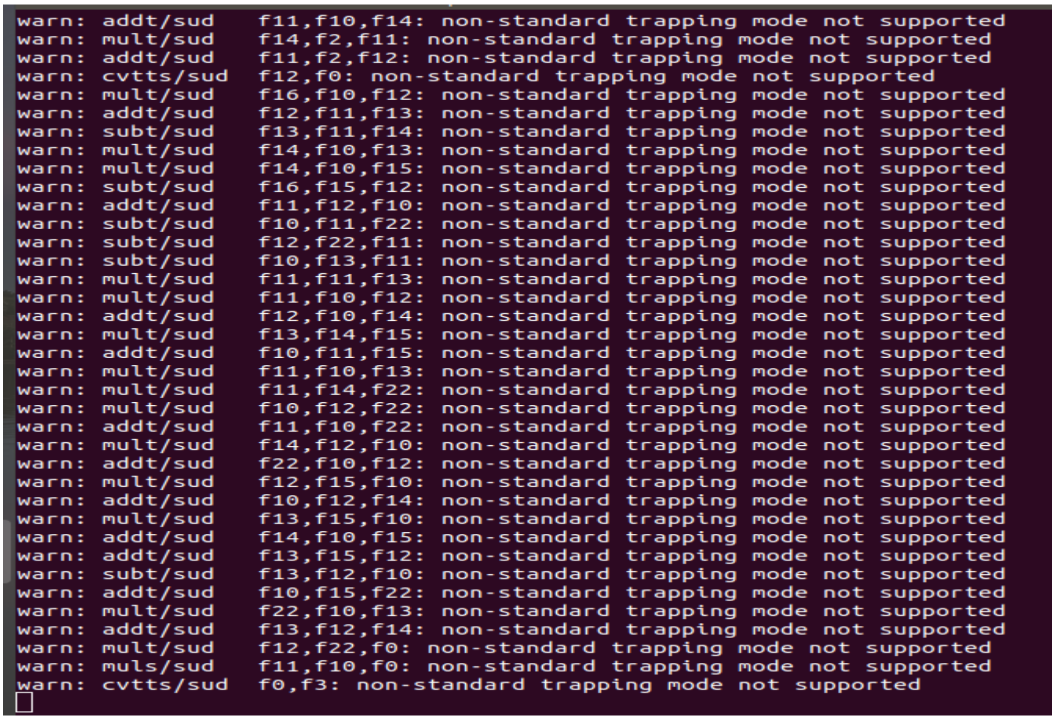
- We will run the PARSEC Benchmark Programs in Full System Simulation mode.
- We have Already copied the Script file(simsmall for my case) in the gem5 directory from script generating directory.
- Before Running the Script file in gem5 , you need to comment a line in .rcS file , and the line is
  - /sbin/m5 switchcpu

- We now can run the PARSEC programs by following command in ALPHA ISA
  - **./build/ALPHA/gem5.opt configs/example/fs.py --script=./blackscholes_1c_simsmall.rcS --caches –l2cache**
  - We Can Take the Output in our own stats.txt file and config.json file by running the following command.
    - **./build/ALPHA/gem5.opt --stats-file=blackscholesStats.txt --json-config=blackscholesConfig.json configs/example/fs.py --script=./blackscholes_1c_simsmall.rcS --caches –l2cache**
    - By default Output goes to stats.txt and config.json file.
    - Run following command for more options of Customization
      - **./build/ALPHA/gem5.opt -h**
- You Will get Following as Output (see Screenshot 6.1).
- Screenshot 6.1



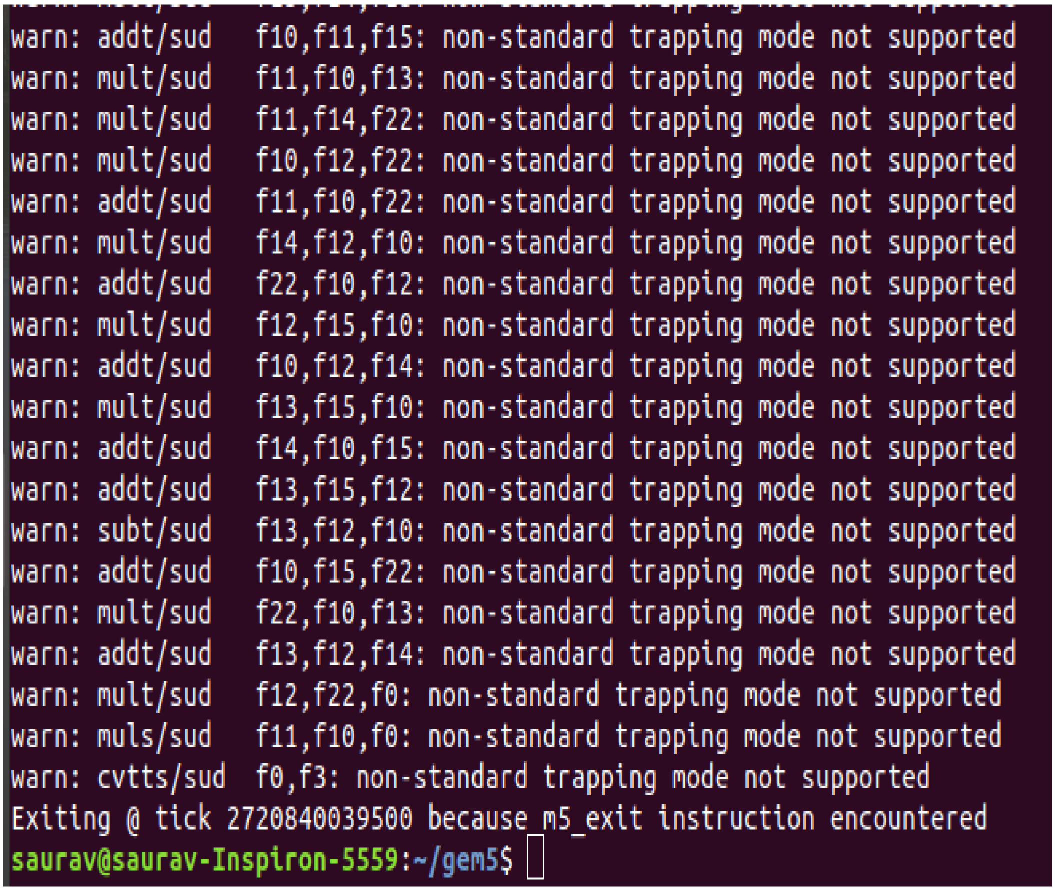
```
saurav@saurav-Inspiron-5559:~/gem5$ ./build/ALPHA/gem5.opt --stats-file=blackscholesStats.txt --json-config=blackscholesConfig.json configs/ex
ample/fs.py --script=./blackscholes_1c_simsmall.rcS --caches --l2cache
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled May 18 2019 19:49:49
gem5 started Jun 30 2019 12:23:20
gem5 executing on saurav-Inspiron-5559, pid 4072
command line: ./build/ALPHA/gem5.opt --stats-file=blackscholesStats.txt --json-config=blackscholesConfig.json configs/example/fs.py --script=.
/blackscholes_1c_simsmall.rcS --caches --l2cache

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
info: kernel located at: /home/saurav/IITG/full_sys_simu/system/binaries/vmlinux
Listening for system connection on port 3456
      0: system.tsunami.io.rtc: Real-time clock set to Thu Jan  1 00:00:00 2009
0: system.remote_gdb.listener: listening for remote gdb #0 on port 7000
warn: ClockedObject: More than one power state change request encountered within the same simulation tick
**** REAL SIMULATION ****
info: Entering event queue @ 0.  Starting simulation...
warn: Prefetch instructions in Alpha do not do anything
warn: Prefetch instructions in Alpha do not do anything
```

- In between the Simulation you will get as following(see Screenshot 6.2)
- Screenshot 6.2



```
warn: addt/sud   f11,f10,f14: non-standard trapping mode not supported
warn: mult/sud   f14,f2,f11: non-standard trapping mode not supported
warn: addt/sud   f11,f2,f12: non-standard trapping mode not supported
warn: cvtts/sud  f12,f0: non-standard trapping mode not supported
warn: mult/sud   f16,f10,f12: non-standard trapping mode not supported
warn: addt/sud   f12,f11,f13: non-standard trapping mode not supported
warn: subt/sud   f13,f11,f14: non-standard trapping mode not supported
warn: mult/sud   f14,f10,f13: non-standard trapping mode not supported
warn: mult/sud   f14,f10,f15: non-standard trapping mode not supported
warn: subt/sud   f16,f15,f12: non-standard trapping mode not supported
warn: addt/sud   f11,f12,f10: non-standard trapping mode not supported
warn: subt/sud   f10,f11,f22: non-standard trapping mode not supported
warn: subt/sud   f12,f22,f11: non-standard trapping mode not supported
warn: subt/sud   f10,f13,f11: non-standard trapping mode not supported
warn: mult/sud   f11,f11,f13: non-standard trapping mode not supported
warn: mult/sud   f11,f10,f12: non-standard trapping mode not supported
warn: addt/sud   f12,f10,f14: non-standard trapping mode not supported
warn: mult/sud   f13,f14,f15: non-standard trapping mode not supported
warn: addt/sud   f10,f11,f15: non-standard trapping mode not supported
warn: mult/sud   f11,f10,f13: non-standard trapping mode not supported
warn: mult/sud   f11,f14,f22: non-standard trapping mode not supported
warn: mult/sud   f10,f12,f22: non-standard trapping mode not supported
warn: addt/sud   f11,f10,f22: non-standard trapping mode not supported
warn: mult/sud   f14,f12,f10: non-standard trapping mode not supported
warn: addt/sud   f22,f10,f12: non-standard trapping mode not supported
warn: mult/sud   f12,f15,f10: non-standard trapping mode not supported
warn: addt/sud   f10,f12,f14: non-standard trapping mode not supported
warn: mult/sud   f13,f15,f10: non-standard trapping mode not supported
warn: addt/sud   f14,f10,f15: non-standard trapping mode not supported
warn: addt/sud   f13,f15,f12: non-standard trapping mode not supported
warn: subt/sud   f13,f12,f10: non-standard trapping mode not supported
warn: addt/sud   f10,f15,f22: non-standard trapping mode not supported
warn: mult/sud   f22,f10,f13: non-standard trapping mode not supported
warn: addt/sud   f13,f12,f14: non-standard trapping mode not supported
warn: mult/sud   f12,f22,f0: non-standard trapping mode not supported
warn: muls/sud   f11,f10,f0: non-standard trapping mode not supported
warn: cvtts/sud  f0,f3: non-standard trapping mode not supported
```

- After Completion You will get the Following in terminal(see Screenshot 6.3)
- Screenshot 6.3



```
warn: addt/sud   f10,f11,f15: non-standard trapping mode not supported
warn: mult/sud   f11,f10,f13: non-standard trapping mode not supported
warn: mult/sud   f11,f14,f22: non-standard trapping mode not supported
warn: mult/sud   f10,f12,f22: non-standard trapping mode not supported
warn: addt/sud   f11,f10,f22: non-standard trapping mode not supported
warn: mult/sud   f14,f12,f10: non-standard trapping mode not supported
warn: addt/sud   f22,f10,f12: non-standard trapping mode not supported
warn: mult/sud   f12,f15,f10: non-standard trapping mode not supported
warn: addt/sud   f10,f12,f14: non-standard trapping mode not supported
warn: mult/sud   f13,f15,f10: non-standard trapping mode not supported
warn: addt/sud   f14,f10,f15: non-standard trapping mode not supported
warn: addt/sud   f13,f15,f12: non-standard trapping mode not supported
warn: subt/sud   f13,f12,f10: non-standard trapping mode not supported
warn: addt/sud   f10,f15,f22: non-standard trapping mode not supported
warn: mult/sud   f22,f10,f13: non-standard trapping mode not supported
warn: addt/sud   f13,f12,f14: non-standard trapping mode not supported
warn: mult/sud   f12,f22,f0: non-standard trapping mode not supported
warn: muls/sud   f11,f10,f0: non-standard trapping mode not supported
warn: cvtts/sud  f0,f3: non-standard trapping mode not supported
Exiting @ tick 2720840039500 because m5_exit instruction encountered
saurav@saurav-Inspiron-5559:~/gem5$ 
```

- We Will get the Output in the **blackscholesStats.txt** file mentioned in the command line for running gem5.
- The Stats file is divided into 4 section like
  - ----------- Begin Simultion Statistics ------------

    :

    :
  - ----------- End Simulation Statistics -------------
- We are only interested in the third Section which is also called as **Region of Interest(ROI)**
- The ROI contains the information about the **Parallel Processing** of the PARSEC Benchmark Programs.
- We will also use the **blackscholesConfig.json** file in later work for integrating the Three tools.

7) Connecting gem5, McPAT, HotSpot:-

7.1) Introduction:-

This is a breif introduction about what I have done to integrate this three tools. This three tools have been connected by a python script which Takes User Choice for PARSEC program and the Choice for the ISA User wants to use for running that PARSEC program. The Third Choice User has to give is whether The User Wants the **ROI** of stats file as whole or divided into Different intervals.

7.2) Parser Used:-

- Two Parser have been Used to Convert Output of One tool to the Input of Another Tool
  - The First Parser used is to convert gem5 stats.txt and config.json file to the .xml input file for McPAT
    - This Parser used takes the Basic Reference from the Following
      - https://bitbucket.org/dskhudia/gem5tomcpat/src/master/
      - The template.xml and some functions have been derived.

        - But I have to change Some of the Code because this Parser is too old and gem5 has been updated rapidly so as its output files.
        - You may also needs to Update When You are doing your own work.
  - The Sencond Parser has been used is to convert the McPAT Output in a text file to a Power Trace File.
    - This Parser is Derived from the Following
      - https://github.com/danielpalomino/mcpat-hotspot-parser
      - This Parser has Only been made to extract the power trace of the L2 and Core.
      - I have changed the Code for extracting more Components from the McPAT Output.
      - You need to chnage the Code as per your requirements if you want to use the above Parser.

7.3) The Output of the Integration Script:-

- The Integration Script that Integrates the all the Three Tools gives Output in Two Phase:-
  - The First is the Plot of a Graph between the Interval Number or Sample Number of ROI in X-Axis and The Runtime Dynamic Power of the PARSEC benchmark from McPAT Output in Y-axis.
  - You can get the Exact Values in the Following Location gem5/mcpat/Output/(Parsec-directory)/Parsec-directory.txt file for your reference.
  - For Example for blackscholes in ALPHA ISA you will get the Output in **gem5/mcpat/alpha-blackscholes/alpha-blackscholes.txt**
  - After the Graph the Simulation will go on for HotSpot and will give the Output in IITG/hotspot/Parsec-directory) location.
  - For above example with ALPHA as ISA and blackscholes as the PARSEC Program we will get the Output of Hotspot in **IITG/hotspot/alpha-blackscholes** directory.

7.4) Changing the Path for your PC:-

You need to change the Path of the Directories in the Python Automation Script if You want to run this integrated-automation script in your PC.

All the Change You Need to do is Provided in the Python Script or the Automation Script, Please Refer that.

7.5) Statistics About the Automation Script:-

In this section it is discussed that what are the times taken by different PARSEC Program to run on this Automation Script

Table 7.1) Statistics for Time taken in Automation for Sampled ROI

S.L. NO.	Program Name	ISA Used	Output Type	Time Taken(min)
1	blackscholes	ALPHA	Sampled ROI	30
2	bodytrack	ALPHA	Sampled ROI	60
3	swaptions	ALPHA	Sampled ROI	90
4	streamcluster	ALPHA	Sampled ROI	80
5	vips	ALPHA	Sampled ROI	120
6	x264	ALPHA	Sampled ROI	160
7	blackscholes	X86	Sampled ROI	45
8	bodytrack	X86	Sampled ROI	60
9	swaptions	X86	Sampled ROI	90
10	streamcluster	X86	Sampled ROI	105
11	vips	X86	Sampled ROI	150
12	x264	X86	Sampled ROI	180

Table 7.2) Statistics for Time taken in Automation for Whole ROI

S.L. NO.	Program Name	ISA Used	Output Type	Time Taken(min)
1	blackscholes	ALPHA	Whole ROI	20
2	bodytrack	ALPHA	Whole ROI	40
3	swaptions	ALPHA	Whole ROI	70
4	streamcluster	ALPHA	Whole ROI	65
5	vips	ALPHA	Whole ROI	120
6	x264	ALPHA	Whole ROI	160
7	blackscholes	X86	Whole ROI	45
8	bodytrack	X86	Whole ROI	60
9	swaptions	X86	Whole ROI	90
10	streamcluster	X86	Whole ROI	105
11	vips	X86	Whole ROI	150
12	x264	X86	Whole ROI	180

8) The Automation:-

8.1) The Present System We Have :-

- The Prsent Sysytem is we have for running the gem5, McPAT, Hotspot for Power Profiling and Temperature Modeling is as follows:-
  - Lets Say for PARSEC Benchmark Program for Power Profiling and Temperature Modelling.
  - We need to Set the Path First in Syspaths.py and Benchmark name in Benchmarks.py manually.
  - Then We need to run the Gem5 and Extract the ROI(Region Of Interest) from Ouutput of Gem5 manually.
  - Then We need to run the Parser Script to convert the Output of gem5 to McPAT input xml File.
  - After that For Running the McPAT We Need to copy the input files we have Just Generated to the McPAT directory.
  - Then We have to run the McPAT Manually again.
  - Now We Will get the Power traces output of McPAT in the terminal, we need to extract that from terminal to a text file.
  - After that we need to Convert the Text file to a HotSpot Input file using the Proper Parser Manually.
  - Then We again have to copy the File for HotSpot input to the HotSpot Directory.
  - Then We have to run the Hotspot Manually.

8.2) What’s New?:-

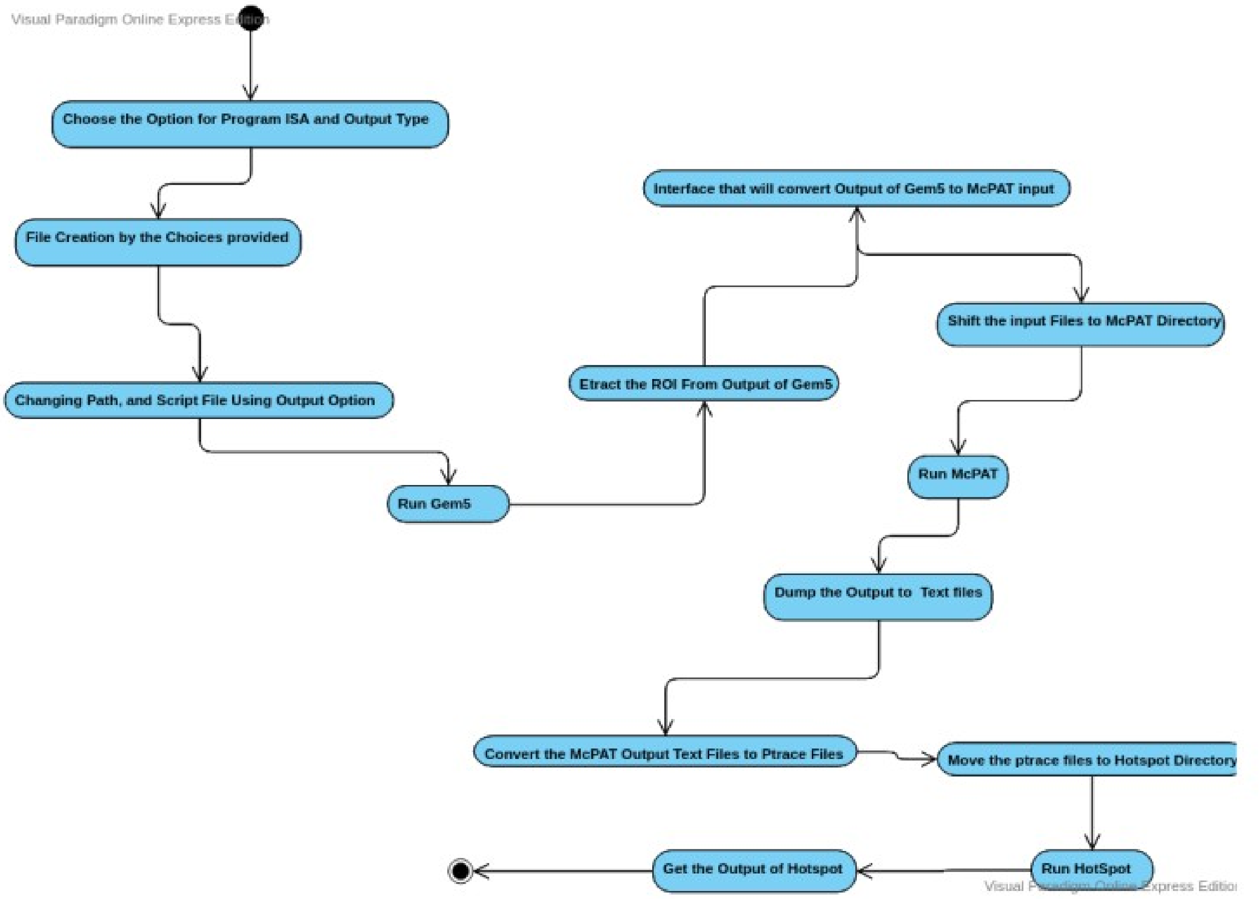
This Section discusses about the What I have Done To Modify the Present System we have for Running the gem5+McPAT+HotSpot. I have Already Discussed in the Previous section How We can do all Simulation in this tools Manually.

I have Integrated all this Tools by a Python Script Which Will run all these tools from that script and will automatically convert the Output to Input of other Tool using the Parser.

This Will reduce the Human Effort to run this all tools one by one. This Script will automatically run all these tools and will generate the Outputs in Proper Location.

8.3) Workflow of Automation:-

The Interfacing Script has the Following Activities During Its Life Cycle


Visual Paradigm Online Express Edition
Choose the Option for Program ISA and Output Type
File Creation by the Choices provided
Changing Path, and Script File Using Output Option
Run Gem5
Etract the ROI From Output of Gem5
Interface that will convert Output of Gem5 to McPAT input
Shift the input Files to McPAT Directory
Run McPAT
Dump the Output to Text files
Convert the McPAT Output Text Files to Ptrace Files
Move the ptrace files to Hotspot Directory
Run HotSpot
Get the Output of Hotspot
Visual Paradigm Online Express Edition

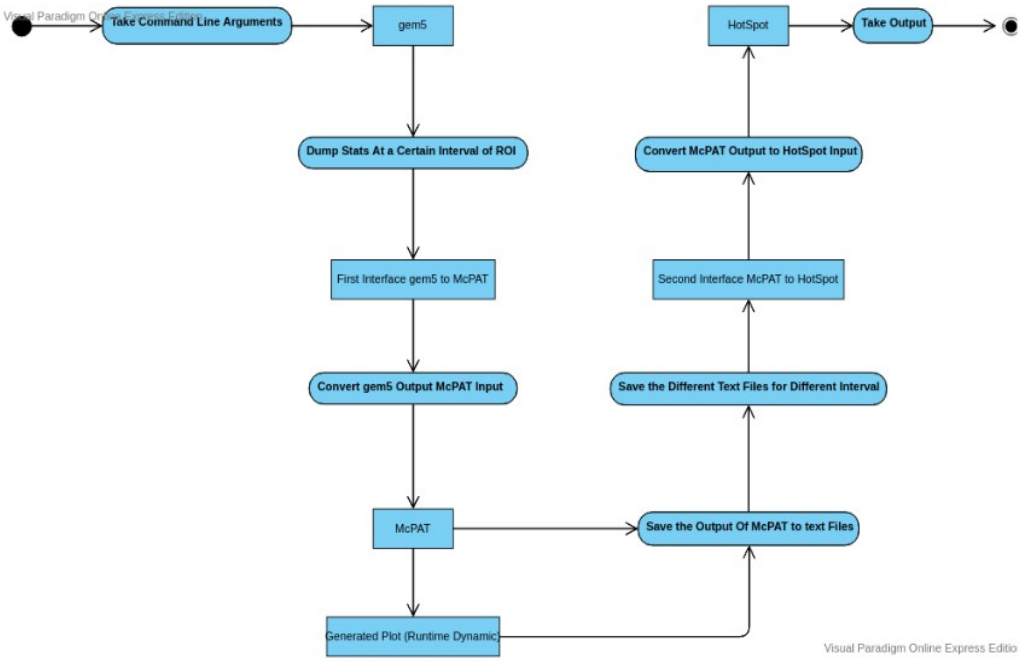
8.4) The Two Main Interface of Automation:-

There is two main interface of the Automation

- First One is in between gem5 and McPAT.
    - The Gem5 Output is Converted into the Input of the McPAT and Fed into the McPAT
- The Second One is in between the McPAT and HotSpot
    - The Output of the McPAT is Converted to the Input of the HotSpot and Fed into the Hotspot.
- This Two Interface is Joined together to Integrate this Three tools for Automation.

- Automation will work by taking into this two interface consideration.

- The Pictorial View is Given Below.

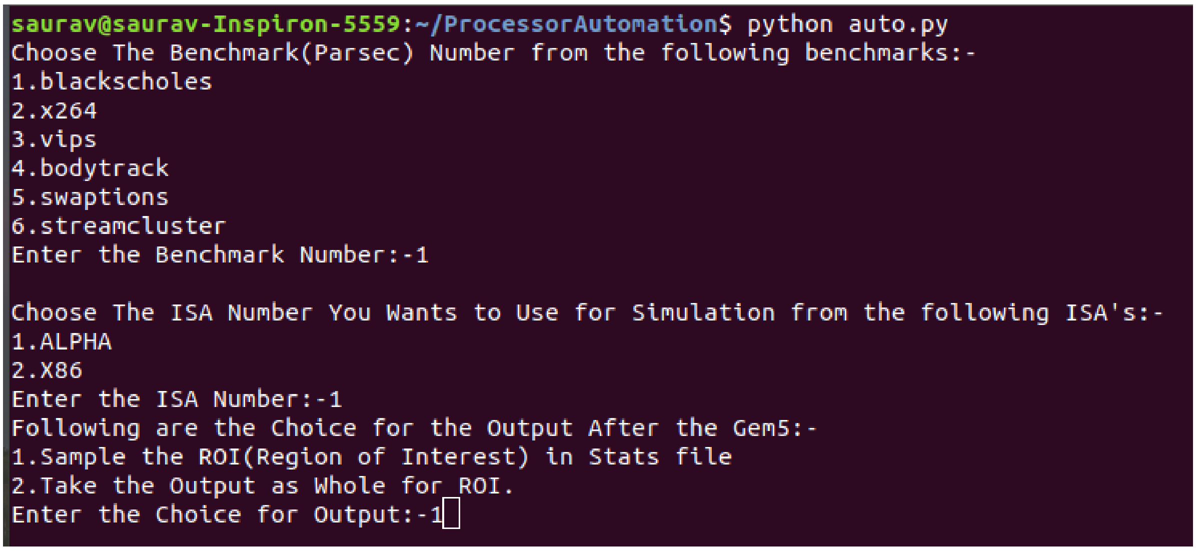

Take Command Line Arguments
gem5
Dump Stats At a Certain Interval of ROI
First Interface gem5 to McPAT
Convert gem5 Output McPAT Input
McPAT
Generated Plot (Runtime Dynamic)
Save the Output Of McPAT to text Files
Save the Different Text Files for Different Interval
Second Interface McPAT to HotSpot
Convert McPAT Output to HotSpot Input
HotSpot
Take Output
Visual Paradigm Online Express Editio

This is How Automation Works

9) A Demo Run:-

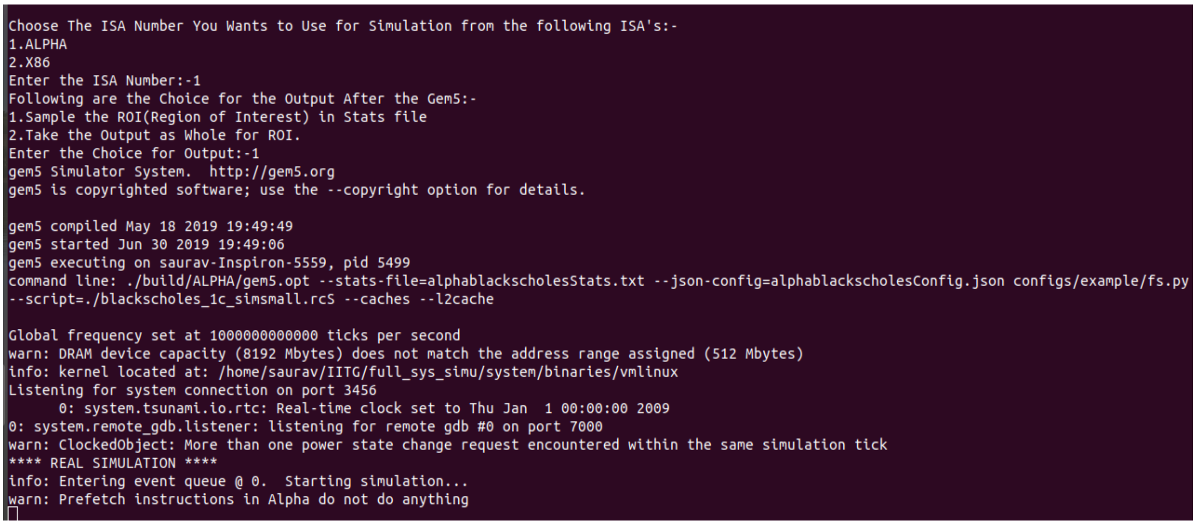
This Section is all about a demo run of the Automation Script.

- Go to the ProcessorAutomation Directory in the Terminal where the **auto.py** Script is there.
- Run the Script in Terminal by Following Command.
  - python auto.py
- Now Choose the Program, ISA and Output type for ROI in the terminal.
- This May Look Like Below
  - Screenshot 8.1



```
saurav@saurav-Inspiron-5559:~/ProcessorAutomation$ python auto.py
Choose The Benchmark(Parsec) Number from the following benchmarks:-
1.blackscholes
2.x264
3.vips
4.bodytrack
5.swaptions
6.streamcluster
Enter the Benchmark Number:-1

Choose The ISA Number You Wants to Use for Simulation from the following ISA's:-
1.ALPHA
2.X86
Enter the ISA Number:-1
Following are the Choice for the Output After the Gem5:-
1.Sample the ROI(Region of Interest) in Stats file
2.Take the Output as Whole for ROI.
Enter the Choice for Output:-1
```

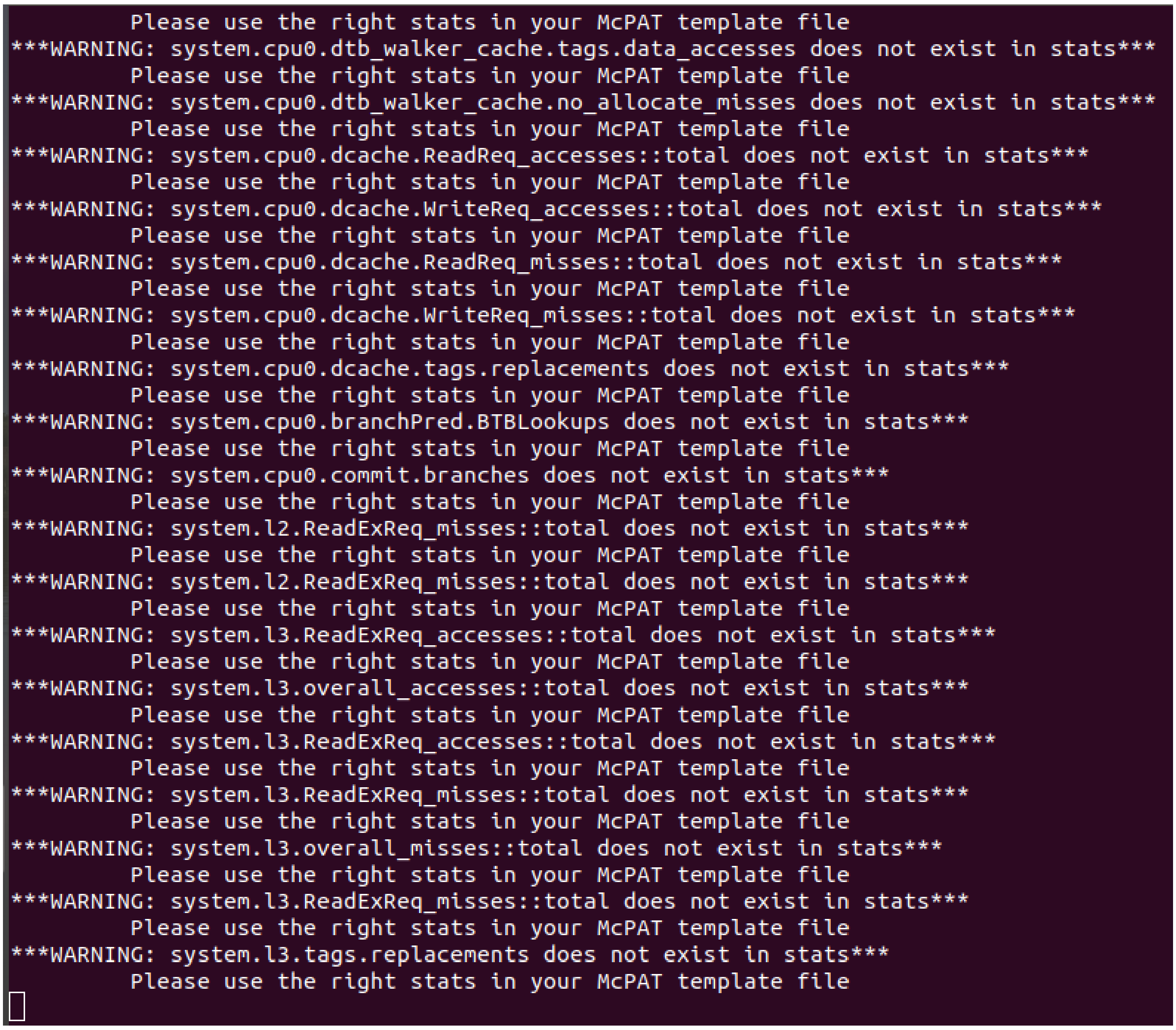
- After Choosing the Choices The Simulation will start and it may look like Following
  - Screenshot 8.2



```
Choose The ISA Number You Wants to Use for Simulation from the following ISA's:-
1.ALPHA
2.X86
Enter the ISA Number:-1
Following are the Choice for the Output After the Gem5:-
1.Sample the ROI(Region of Interest) in Stats file
2.Take the Output as Whole for ROI.
Enter the Choice for Output:-1
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled May 18 2019 19:49:49
gem5 started Jun 30 2019 19:49:06
gem5 executing on saurav-Inspiron-5559, pid 5499
command line: ./build/ALPHA/gem5.opt --stats-file=alphablackscholesStats.txt --json-config=alphablackscholesConfig.json configs/example/fs.py
--script=./blackscholes_1c_simsmall.rcS --caches --l2cache

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
info: kernel located at: /home/saurav/IITG/full_sys_simu/system/binaries/vmlinux
Listening for system connection on port 3456
      0: system.tsunami.io.rtc: Real-time clock set to Thu Jan  1 00:00:00 2009
0: system.remote_gdb.listener: listening for remote gdb #0 on port 7000
warn: ClockedObject: More than one power state change request encountered within the same simulation tick
**** REAL SIMULATION ****
info: Entering event queue @ 0.  Starting simulation...
warn: Prefetch instructions in Alpha do not do anything
```

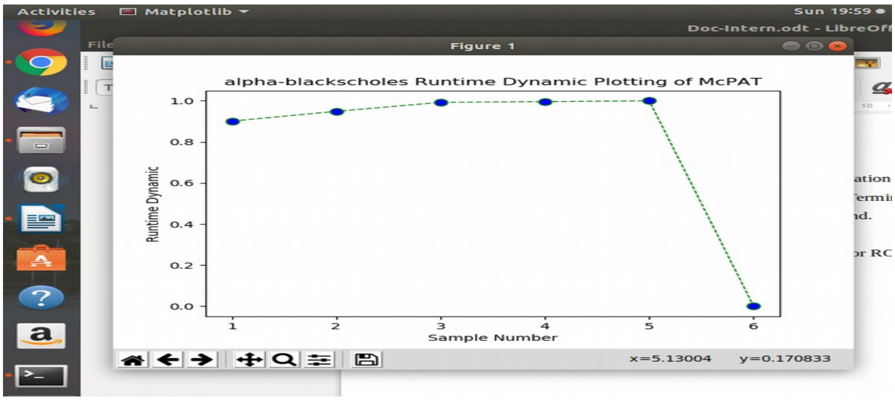
- The First Phase of Simulation is of gem5 Simulation.

- After gem5 Simulation is Complete automatically the Parsing of gem5 Output to the McPAT input will start for different sampled or Whole ROI of stats of gem5.
- The Termianl will look like as follows
  - Screenshot 8.3

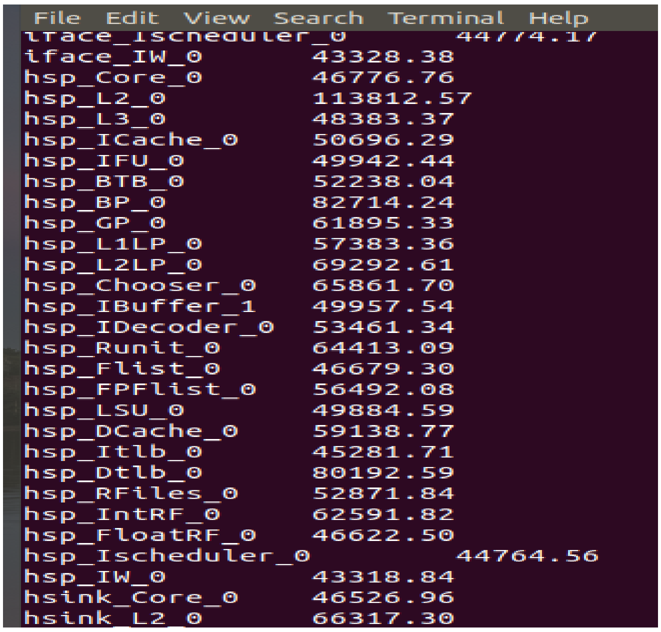


```
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dtb_walker_cache.tags.data_accesses does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dtb_walker_cache.no_allocate_misses does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dcache.ReadReq_accesses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dcache.WriteReq_accesses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dcache.ReadReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dcache.WriteReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.dcache.tags.replacements does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.branchPred.BTBLookups does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.cpu0.commit.branches does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l2.ReadExReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l2.ReadExReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.ReadExReq_accesses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.overall_accesses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.ReadExReq_accesses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.ReadExReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.overall_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.ReadExReq_misses::total does not exist in stats***
         Please use the right stats in your McPAT template file
***WARNING: system.l3.tags.replacements does not exist in stats***
         Please use the right stats in your McPAT template file
```

- After Successfully Completion of the Parsing the McPAT will run Automatically and will dump the McPAT Data into some text file in the following loacation for alpha-blackscholes **gem5/mcpat/Output/alpha-blackscholes.**
- After the McPAT running Completes for all the Sample Now Autoatically a Graph will pop up in PC(a similar PDF version is saved in the above directory).
  - Screenshot 8.4

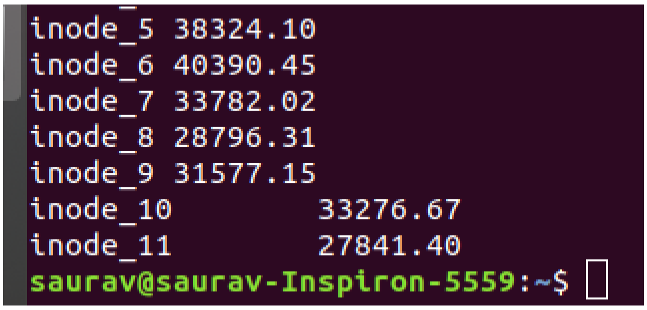


- You need to Camcel this Graph after getting the view for the Continuation of the Automation.
- Then Automatically the ptrace files will be generated for hotspot input by McPAT to hotspot Parser and will be saved in **IITG/hotspot/alpha-blackscholes** (for alpha-blackscholes program)
- Automaticlly the Hotspot will run and Terminal Will Look like this
  - Screenshot 8.5



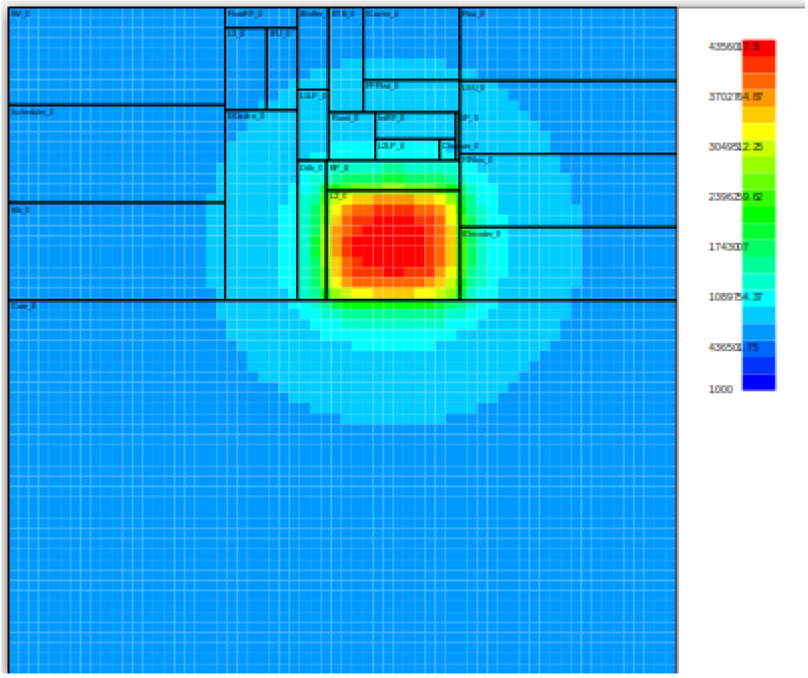
```
iface_Ischeduler_0              44774.17
iface_IW_0              43328.38
hsp_Core_0              46776.76
hsp_L2_0                113812.57
hsp_L3_0                48383.37
hsp_ICache_0            50696.29
hsp_IFU_0               49942.44
hsp_BTB_0               52238.04
hsp_BP_0                82714.24
hsp_GP_0                61895.33
hsp_L1LP_0              57383.36
hsp_L2LP_0              69292.61
hsp_Chooser_0           65861.70
hsp_IBuffer_1           49957.54
hsp_IDecoder_0          53461.34
hsp_Runit_0             64413.09
hsp_Flist_0             46679.30
hsp_FPFlist_0           56492.08
hsp_LSU_0               49884.59
hsp_DCache_0            59138.77
hsp_Itlb_0              45281.71
hsp_Dtlb_0              80192.59
hsp_RFiles_0            52871.84
hsp_IntRF_0             62591.82
hsp_FloatRF_0           46622.50
hsp_Ischeduler_0                44764.56
hsp_IW_0                43318.84
hsink_Core_0            46526.96
hsink_L2_0              66317.30
```

- And After the Completion of the Automation the terminal Will Look Like this
  - Screenshot 8.6



```
inode_5 38324.10
inode_6 40390.45
inode_7 33782.02
inode_8 28796.31
inode_9 31577.15
inode_10        33276.67
inode_11        27841.40
saurav@saurav-Inspiron-5559:~$ 
```

- You will also get a Pictorial view of Hotspot Output as Follows:-
  - Screenshot 8.7 (Whole ROI Hotspot Output for alpha-blackscholes)



- The Output Files after different simulation during the automation you will get in following loaction
  - For Gem5:- **gem5/m5out**
  - For McPAT:- **gem5/mcpat/Output/ISA-programName**
  - For Hotspot:- **IITG/hotspot/ISA-programName**

10) **References:-**

- PARSEC Report for Studying about PARSEC Benchmark
    - https://parsec.cs.princeton.edu/doc/parsec-report.pdf
- Gem5 Website for Studying about the Gem5 in details.
    - http://gem5.org/Main_Page
- McPAT Website for Studying the McPAT
    - https://www.hpl.hp.com/research/mcpat/
- HotSpot Website for Studying about Hotspot
    - http://lava.cs.virginia.edu/HotSpot/
- Template File and Some Functions of Daya Khudia’s gem5 to McPAT parser
    - https://bitbucket.org/dskhudia/gem5tomcpat/src/master/
- Daniel Palomino’s Basic Functions for McPAT to Hotspot Parser
    - https://github.com/danielpalomino/mcpat-hotspot-parser
- Documentations of
    - McPAT
        - https://github.com/HewlettPackard/mcpat/tree/master/Documents
    - Hotspot
        - http://lava.cs.virginia.edu/HotSpot/documentation.htm
- Youtube For Some Video Reference

11) Abbreviations Used:-

- PARSEC:- Princeton Application Repository for Shared-Memory Computers.
- ROI:- Region of Interest
- McPAT:- Multicore Power, Area, and Timing
- ISA:- Instruction Set Architecture
- PC:- Personal Computer
- STATS File:- Statistics File
- CMP:- Chip Multi-Processor

Thank You