Technische Hochschule **Rosenheim**
Technical University of Applied Sciences

Prof. Dr. Florian Künzner
Technical University of Applied Sciences Rosenheim, Computer Science

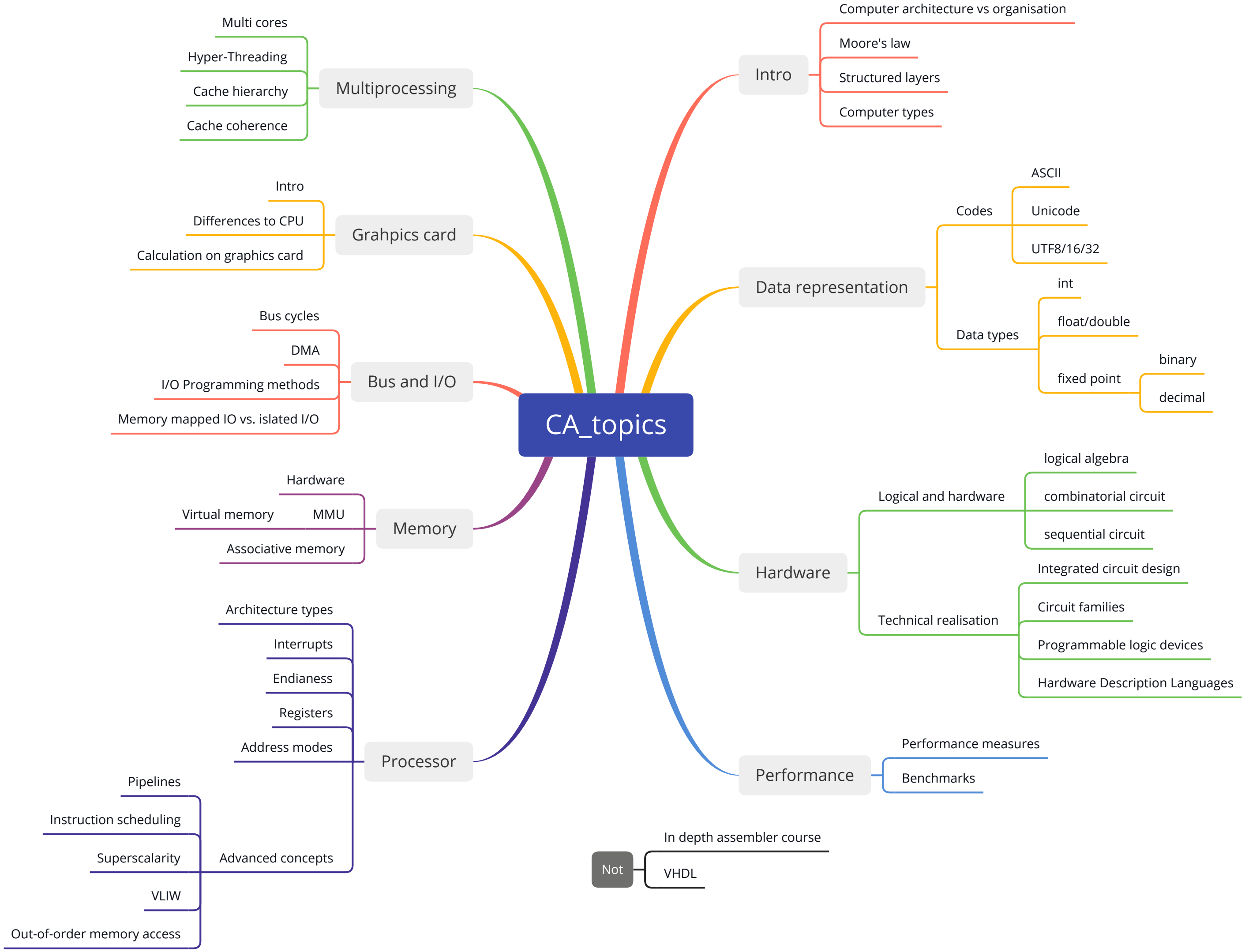
CA 3 – Hardware performance

The lecture is based on the work and the documents of Prof. Dr. Theodor Tempelmeier

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

Goal



- CA_topics
  - Intro
    - Computer architecture vs organisation
    - Moore's law
    - Structured layers
    - Computer types
  - Data representation
    - Codes
      - ASCII
      - Unicode
      - UTF8/16/32
    - Data types
      - int
      - float/double
      - fixed point
        - binary
        - decimal
  - Hardware
    - Logical and hardware
      - logical algebra
      - combinatorial circuit
      - sequential circuit
    - Technical realisation
      - Integrated circuit design
      - Circuit families
      - Programmable logic devices
      - Hardware Description Languages
  - Performance
    - Performance measures
    - Benchmarks
  - Multiprocessing
    - Multi cores
    - Hyper-Threading
    - Cache hierarchy
    - Cache coherence
  - Grahpics card
    - Intro
    - Differences to CPU
    - Calculation on graphics card
  - Bus and I/O
    - Bus cycles
    - DMA
    - I/O Programming methods
    - Memory mapped IO vs. islated I/O
  - Memory
    - Hardware
    - MMU
      - Virtual memory
    - Associative memory
  - Processor
    - Architecture types
    - Interrupts
    - Endianess
    - Registers
    - Address modes
    - Advanced concepts
      - Pipelines
      - Instruction scheduling
      - Superscalarity
      - VLIW
      - Out-of-order memory access
- Not
  - In depth assembler course
  - VHDL

CAMPUS Rosenheim
Computer Science

Goal

CA::Hardware performance

- Performance measures
- Benchmarking
- Popular benchmark suites
- Tools: CPU, GPU, Memory, IO, Network

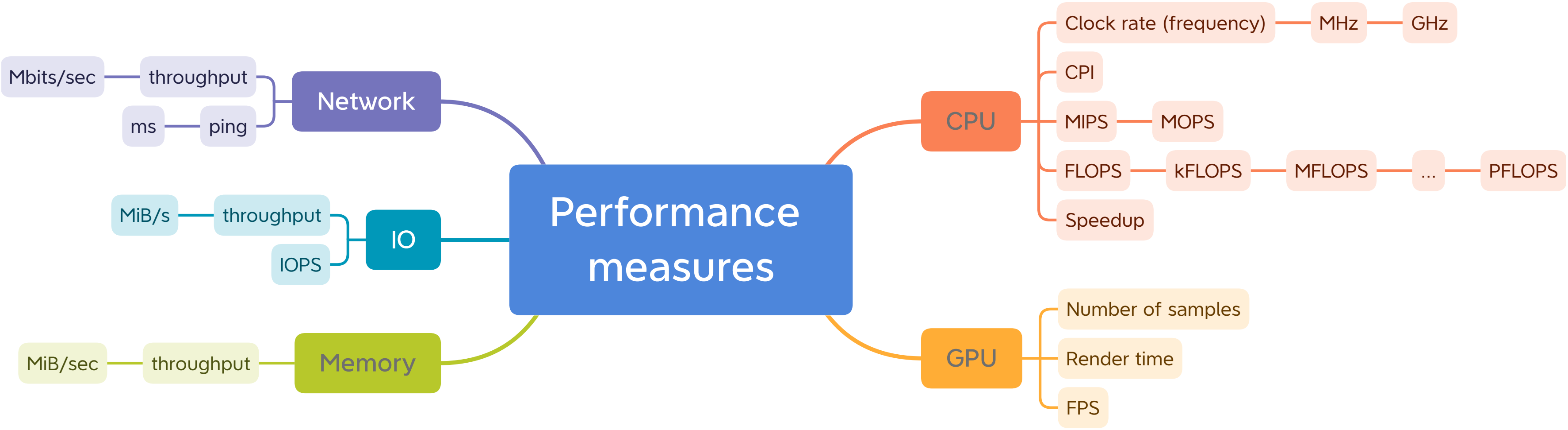
Performance of a computer: hardware

What are the most important performance related hardware parts of a computer?

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

Performance measures



Clock frequency

The CPU clock rate (or frequency) is measured in Hz – usually in MHz, or GHz.

`CR (Hz)`: **Clock rate (frequency) (Hz):** (number of pulses per second)

$$\texttt{CR (Hz)} = \frac{1}{\texttt{t\_CC (s)}} \tag{1}$$

`t_CC (s)`: **Clock cycle (seconds):** (amount of time between two pulses)

$$\texttt{t\_CC (s)} = \frac{1}{\texttt{CR (Hz)}} \tag{2}$$

Cycles per instruction (CPI)

The mean number of elapsed cycles per instruction.

`CPI`: **Cycles per instruction (CPI):**

$$\mathtt{CPI} = \frac{\sum_{i=1}^{n} I_{C_i} \times CC_i}{I_C} \tag{3}$$

- I_{C_i}: # Number of instructions of instruction type i
- CC_i: # Number of clock cycles CC for instructions of type i
- I_C : # Total number of instructions for a program

Attention: Only compare computers with same architecture (RISC/CISC)!

Instructions per second

`MIPS`: **Million instructions per second:**

$$\texttt{MIPS} = \frac{CR}{\texttt{CPI} \times 10^6} \tag{4}$$

Problem: MIPS is not comparable between CISC/RISC architectures.

Therefore:
`MOPS`: **Million operations per second:**

- CISC: $1\ \texttt{instruction} \geq n\ \texttt{operations}$
- RISC: $1\ \texttt{instruction} \approx 1\ \texttt{operation}$

Floating point operations per second

FLOPS: Floating point operations per second:

$$\mathtt{FLOPS} = \frac{\mathtt{FLOPs}}{\mathtt{s}} \tag{5}$$

- FLOPs: Number of floating point operations (counted)
- s: Second
- FLOPS -> kFLOPS (10^3), MFLOPS (10^6), GFLOPS (10^9), TFLOPS (10^{12}), PFLOPS (10^{15})

Attention: single-precision (`float`) vs double-precision `double`.

More details: https://en.wikipedia.org/wiki/FLOPS
Performance development: Top 500 https://www.top500.org/statistics/perfdevel/

Speedup

The speedup is the relative performance of two systems (or programs) processing the same problem.

S: **speedup – relative performance:**

$$S = \frac{t_a}{t_b} \tag{6}$$

- t_a runtime of system (or program) a
- t_b runtime of system (or program) b

Input/output operations per second (IOPS)

Performance measurement for input/output: used to characterise HDD, SSD, and SAN.

`IOPS`: **Input/output operations per second:**

$$\texttt{IOPS} = (\texttt{MBps}/\texttt{Block Size}) \times 1024 \tag{7}$$

Attention: Don't use IOPS for direct comparison of IO hardware. Instead, use the real measured throughput: e.g. MiB/s.

More details: `https://en.wikipedia.org/wiki/IOPS`

CAMPUS Rosenheim
Computer Science

Benchmark

Benchmarking is the practice of comparing performance metrics.

A **benchmark** runs (multiple) predefined reference programs and collects performance metrics (e.g. MIPS, FLOPS, ...)

A **benchmark suite** is a collection of multiple predefined references programs of a benchmark.

Common benchmark suites

Benchmark suite	Introduced	Details
Whetstone	1972	Measures in *mega whetstone instructions per second* (MWIPS). A synthetic benchmark including floating point operations.
Dhrystone	1984	Measures general-purpose ("integer") performance
Linpack	1976	Solves systems of linear equations (FLOPS)
SPEC-Benchmark	1988	Standard performance evaluation corporation: supports various benchmarks: CPUs (integer, float), server, GPU, ...

Often used in literature and for high performance computers (HPC), but not popular for private use.

Popular benchmark suites

- **Geekbench** Browser -> **Geekbench 5**
  Cross-Platform benchmark that measures systems performance: CPU (single core/multi core)
- **PassMark** (CPU-Mark) -> **PerformanceTest**
  Cross-Platform benchmark for CPU, Disk, 2D/3D Graphics, Networking, Memory, GPU compute, ...
- **OpenBenchmarking.org** -> **Phoronix Test Suite**
  Cross-Platform open source benchmark for CPU, GPU, OpenGL, Disks, ...
- **Cinebench**
  Cross-Platform (Windows, macOS) benchmark for CPU (single core/multi core)

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

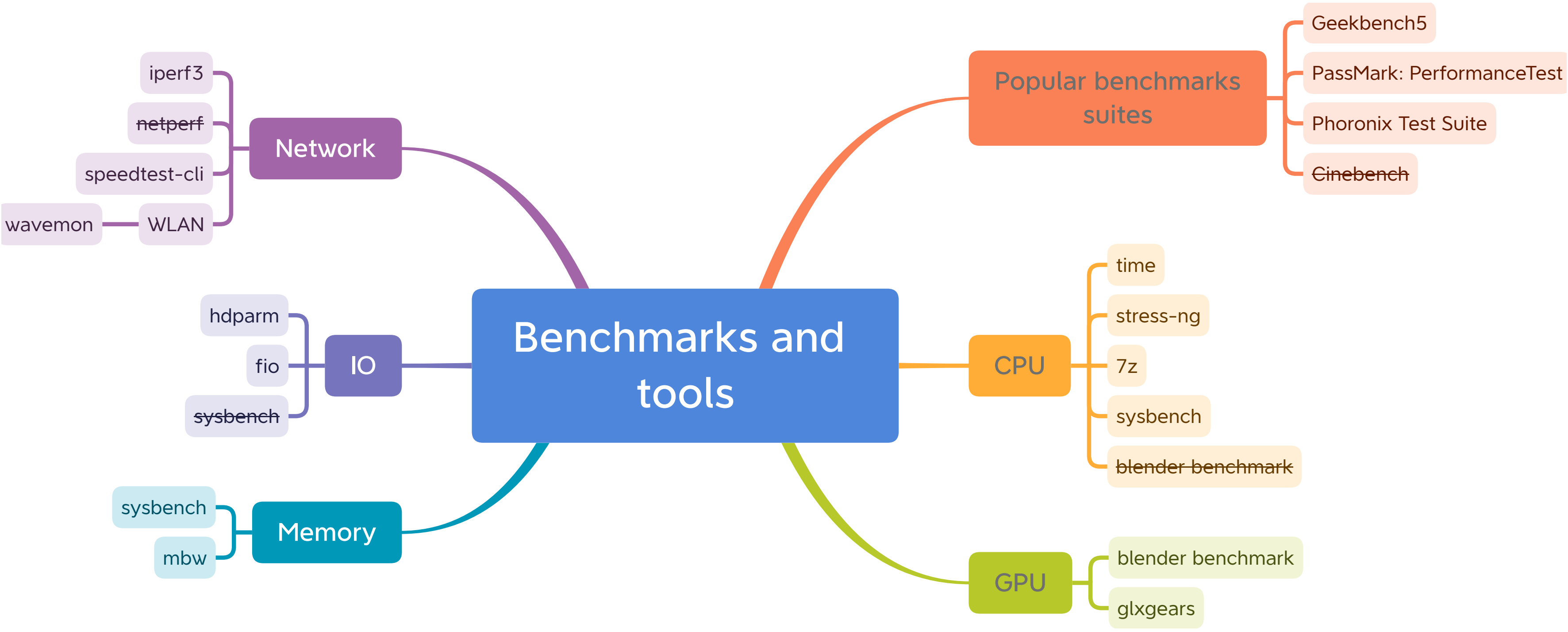
Benchmarking on microcontrollers

- Dhrystone (see above on slide 13: outdated)
- **CoreMark** Scores -> **CoreMark**
  Benchmarks microcontroller in CoreMarks and CoreMarks/MHz.

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

(Linux) tools overview



Benchmarks and tools

- Popular benchmarks suites
  - Geekbench5
  - PassMark: PerformanceTest
  - Phoronix Test Suite
  - ~~Cinebench~~
- CPU
  - time
  - stress-ng
  - 7z
  - sysbench
  - ~~blender benchmark~~
- GPU
  - blender benchmark
  - glxgears
- Network
  - iperf3
  - ~~netperf~~
  - speedtest-cli
  - WLAN
    - wavemon
- IO
  - hdparm
  - fio
  - ~~sysbench~~
- Memory
  - sysbench
  - mbw

Hints

- Don't trust one measurement (**repeat** it **several times**)
- Don't trust one tool, try to **verify** the measurement **with different tools**
- **Use** a **load free system** for the measurements

Who measures, measures crap!

CAMPUS Rosenheim
Computer Science

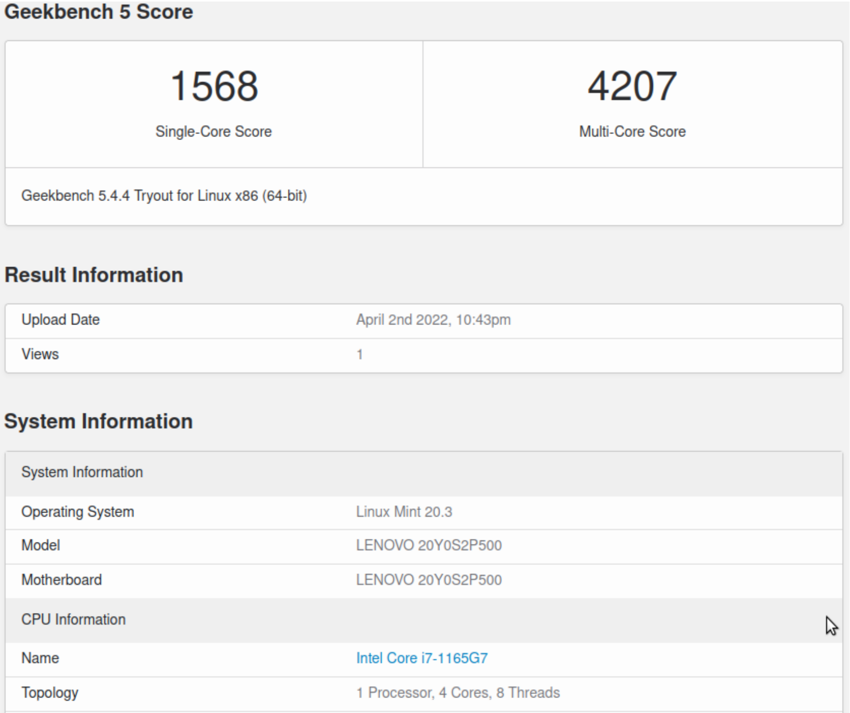
Technische Hochschule Rosenheim
Technical University of Applied Sciences

Geekbench5

Example usage:

```
#start benchmark
./geekbench5
```

Example output:



Geekbench 5 Score

1568	4207
Single-Core Score	Multi-Core Score

Geekbench 5.4.4 Tryout for Linux x86 (64-bit)

Result Information

Upload Date	April 2nd 2022, 10:43pm
Views	1

System Information

System Information	
Operating System	Linux Mint 20.3
Model	LENOVO 20Y0S2P500
Motherboard	LENOVO 20Y0S2P500
CPU Information	
Name	Intel Core i7-1165G7
Topology	1 Processor, 4 Cores, 8 Threads

Geekbench result

PassMark PerformanceTest

Example usage:

```
#start benchmark
./pt_linux_x64
```

Example output:

```
...
Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz (x86_64)
4 cores @ 4500 MHz  |  15.6 GiB RAM
Number of Processes: 8  |  Test Iterations: 1  |  Test Duration: Medium
--------------------------------------------------------------------------
CPU Mark:                          10049
  Integer Math                     32169 Million Operations/s
  Floating Point Math              19656 Million Operations/s
  Prime Numbers                    29.2 Million Primes/s
  Sorting                          18981 Thousand Strings/s
  Encryption                       4321 MB/s
  Compression                      143594 KB/s
  CPU Single Threaded              2740 Million Operations/s
  Physics                          664 Frames/s
  Extended Instructions (SSE)      6296 Million Matrices/s

Memory Mark:                       2744
  Database Operations              4159 Thousand Operations/s
  Memory Read Cached               30655 MB/s
  Memory Read Uncached             16957 MB/s
  Memory Write                     13017 MB/s
  Available RAM                    4385 Megabytes
  Memory Latency                   31 Nanoseconds
  Memory Threaded                  29175 MB/s
--------------------------------------------------------------------------
```

CAMPUS Rosenheim
Computer Science

Phoronix Test Suite

Example usage:

```
#list all tests
phoronix-test-suite list-all-tests

#benchmark (download and start test)
phoronix-test-suite benchmark pts/compilebench
```

Example output:

```
...
Compile Bench 0.6:
    pts/compilebench-1.0.3 [Test: Compile]
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 13 Minutes [22:43 CEST]
        Started Run 1 @ 22:30:53
        Started Run 2 @ 22:31:46
        Started Run 3 @ 22:32:33

    Test: Compile:
        728.62
        741.34
        747.39

    Average: 739.12 MB/s
    Deviation: 1.30%

    Comparison of 8,480 OpenBenchmarking.org samples since 20 September 2011; median result: 629 MB/s. Box plot o
    [|-------################!####*#################---------*-------------------------------*-*-----*-*---|
                                ^ This Result (55th Percentile): 739
    ...
```

time - measure runtime of a process

Run programs and summarize system resource usage

Example usage:

```
#measure time of program stress-ng
time stress-ng --cpu 2 -t 2
```

Example output:

```
real     0m2,035s
user     0m3,998s
sys      0m0,020s
```

- **real**: Elapsed time (from start to end)
- **user**: Amount of CPU time spent in user space
- **sys**: Amount of CPU time spent in kernel space

CAMPUS Rosenheim
Computer Science

Technische
Hochschule
Rosenheim
Technical University of Applied Sciences

stress-ng – cpu benchmark

A tool to load and stress a computer system

Example usage:

```
#enable usage of perf counters
sudo sh -c 'echo 1 >/proc/sys/kernel/perf_event_paranoid'

#single thread measurement (matrixprod) for 10 seconds
stress-ng --cpu 1 --cpu-method matrixprod --metrics-brief -t 10 --perf --times --tz
```

Example output:

```
stress-ng: info:  [34080] dispatching hogs: 1 cpu
stress-ng: info:  [34080] successful run completed in 10.00s
stress-ng: info:  [34080] stressor       bogo ops real time  usr time  sys time   bogo ops/s   bogo ops/s
stress-ng: info:  [34080]                          (secs)    (secs)    (secs)   (real time) (usr+sys time)
stress-ng: info:  [34080] cpu                3539     10.00      9.83      0.14       353.82       354.96
stress-ng: info:  [34080] cpu:
stress-ng: info:  [34080]             43.870.377.365 CPU Cycles                      4,39 B/sec
stress-ng: info:  [34080]             63.174.471.709 Instructions                    6,32 B/sec (1,440 instr. per
...
stress-ng: info:  [34080] cpu:
stress-ng: info:  [34080]               acpitz   29,80 C (302,95 K)
stress-ng: info:  [34080]               acpitz   28,80 C (301,95 K)
stress-ng: info:  [34080]         x86_pkg_temp   40,20 C (313,35 K)
stress-ng: info:  [34080] for a 10,00s run time:
stress-ng: info:  [34080]      80,03s available CPU time
stress-ng: info:  [34080]       9,83s user time   ( 12,28%)
stress-ng: info:  [34080]       0,14s system time (  0,17%)
stress-ng: info:  [34080]       9,97s total time  ( 12,46%)
stress-ng: info:  [34080] load average: 1,35 1,07 0,98
```

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

7z – benchmark feature

A file archiver with high compression ratio format

Example usage:

```
#benchmark on 1 core
7z b -mmt1

#benchmark on all cores
7z b
```

Example output:

```
...
Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz (906E9)
CPU Freq: - - - - - - - - -

RAM size:   15966 MB,  # CPU hardware threads:   8
RAM usage:    435 MB,  # Benchmark threads:      1

                       Compressing  |                  Decompressing
Dict     Speed Usage    R/U Rating  |      Speed Usage    R/U Rating
         KiB/s     %   MIPS   MIPS  |      KiB/s     %   MIPS   MIPS

22:       5216   100   5094   5075  |      53131   100   4536   4536
23:       4872   100   4965   4965  |      52547   100   4548   4548
24:       4707   100   5061   5061  |      51996   100   4565   4565
25:       4497   100   5135   5135  |      50957   100   4536   4536
----------------------------------  | ------------------------------
Avr:             100   5064   5059  |              100   4546   4546
Tot:             100   4805   4803
```

sysbench – cpu benchmark

Multi-threaded benchmark tool for database systems

Example usage:

```
#measure single core performance
sysbench cpu run

#measure multi core performance (4 threads)
sysbench cpu run --threads=4
```

Example output:

```
CPU speed:
    events per second:  5756.06

General statistics:
    total time:                          10.0002s
    total number of events:              57568

...

Threads fairness:
    events (avg/stddev):           14392.0000/25.41
    execution time (avg/stddev):   9.9974/0.00
```

blender - benchmark 3D rendering

The Blender Open Data Benchmark launcher command line interface

Example usage:

```
#interactive mode
./benchmark-launcher-cli
```

Example output:

```
? Choose a Blender version: 3.1.0
> Will render scenes: monster, junkshop, classroom
? No files need to be downloaded, continue? Yes
? Choose a device: NVIDIA GeForce GTX 1070
? Start benchmarking? Yes
Warming up monster
Benchmarking monster
100 / 100 [--------------------------------------------------------------------------------------------] 100.00%
Warming up junkshop
Benchmarking junkshop
107 / 100 [--------------------------------------------------------------------------------------------] 107.00%
Warming up classroom
Benchmarking classroom
103 / 100 [--------------------------------------------------------------------------------------------] 103.00%
Benchmark complete:
monster: 272.452127 samples per minute
junkshop: 181.386735 samples per minute
classroom: 144.055940 samples per minute
```

glxgears - measure OpenGL performance

“gears” demo for GLX

Example usage:

```
#measure frame rate
glxgears
```

Example output:

```
303 frames in 5.0 seconds = 60.436 FPS
300 frames in 5.0 seconds = 59.947 FPS
300 frames in 5.0 seconds = 59.955 FPS
...
```

sysbench – memory benchmark

Multi-threaded benchmark tool for database systems

Example usage:

```
#measure memory bandwidth (1 thread)
sysbench memory run

#measure memory bandwidth (4 threads)
sysbench memory run --threads=4
```

Example output:

```
...
Total operations: 76154496 (7614562.32 per second)

74369.62 MiB transferred (7436.10 MiB/sec)

...

Threads fairness:
    events (avg/stddev):           76154496.0000/0.00
    execution time (avg/stddev):   4.8623/0.00
```

mbw - Memory BandWidth benchmark

Example usage:

```
#measure memory bandwidth (with 1024MiB data)
mbw -n 2 1024
```

Example output:

```
0       Method: MEMCPY  Elapsed: 0.12472    MiB: 1024.00000    Copy: 8210.589 MiB/s
1       Method: MEMCPY  Elapsed: 0.12635    MiB: 1024.00000    Copy: 8104.536 MiB/s
AVG     Method: MEMCPY  Elapsed: 0.12553    MiB: 1024.00000    Copy: 8157.218 MiB/s
0       Method: DUMB    Elapsed: 0.08119    MiB: 1024.00000    Copy: 12613.012 MiB/s
1       Method: DUMB    Elapsed: 0.08092    MiB: 1024.00000    Copy: 12655.256 MiB/s
AVG     Method: DUMB    Elapsed: 0.08105    MiB: 1024.00000    Copy: 12634.098 MiB/s
0       Method: MCBLOCK Elapsed: 0.07854    MiB: 1024.00000    Copy: 13037.942 MiB/s
1       Method: MCBLOCK Elapsed: 0.07797    MiB: 1024.00000    Copy: 13132.751 MiB/s
AVG     Method: MCBLOCK Elapsed: 0.07826    MiB: 1024.00000    Copy: 13085.175 MiB/s
```

hdparm - IO read performance

get/set SATA/IDE device parameters

Example usage:

```
#list available block devices
lsblk

#measure read io performance
sudo hdparm -tT /dev/sda1

#measure read io performance (without buffers)
sudo hdparm -tT --direct  /dev/sdb1
```

Example output:

```
/dev/sdb1:
Timing cached reads:   35680 MB in  1.99 seconds = 17954.51 MB/sec
Timing buffered disk reads: 668 MB in  3.00 seconds = 222.37 MB/sec
```

fio - flexible I/O tester

Example usage:

```
#measure read io performance
fio -rw=read --size=1G -name=test

#measure read io performance (without buffers)
fio -direct=1 -rw=read --size=1G -name=test

#measure write io performance
fio -direct=1 -rw=write --size=1G -name=test
```

Example output:

```
test: (g=0): rw=read, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=psync, iodepth=1
fio-3.16
Starting 1 process
Jobs: 1 (f=1): [R(1)][100.0%][r=173MiB/s][r=44.2k IOPS][eta 00m:00s]
test: (groupid=0, jobs=1): err= 0: pid=18314: Sat Apr  2 19:50:51 2022
read: IOPS=43.3k, BW=169MiB/s (177MB/s)(1024MiB/6056msec)

...

Run status group 0 (all jobs):
READ: bw=169MiB/s (177MB/s), 169MiB/s-169MiB/s (177MB/s-177MB/s), io=1024MiB (1074MB), run=6056-6056msec

Disk stats (read/write):
sdb: ios=255234/0, merge=0/0, ticks=4978/0, in_queue=100, util=98.40%
```

iperf3 - perform network throughput tests

Example usage:

```
#server
iperf3 -s

#client: measure network throughput
iperf3 -c 192.168.1.2
```

Example output:

```
Connecting to host 192.168.1.2, port 5201
[  5] local 192.168.1.2 port 47618 connected to 192.168.1.2 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.00   sec   113 MBytes   946 Mbits/sec    0    361 KBytes
[  5]   1.00-2.00   sec   111 MBytes   933 Mbits/sec    0    361 KBytes
[  5]   2.00-3.00   sec   111 MBytes   935 Mbits/sec    0    361 KBytes
[  5]   3.00-4.00   sec   111 MBytes   933 Mbits/sec    0    382 KBytes
[  5]   4.00-5.00   sec   111 MBytes   930 Mbits/sec    0    382 KBytes
[  5]   5.00-6.00   sec   111 MBytes   935 Mbits/sec    0    382 KBytes
[  5]   6.00-7.00   sec   111 MBytes   935 Mbits/sec    0    382 KBytes
[  5]   7.00-8.00   sec   111 MBytes   934 Mbits/sec    0    403 KBytes
[  5]   8.00-9.00   sec   111 MBytes   932 Mbits/sec    0    403 KBytes
[  5]   9.00-10.00  sec   111 MBytes   930 Mbits/sec    0    403 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-10.00  sec  1.09 GBytes   934 Mbits/sec    0             sender
[  5]   0.00-10.01  sec  1.09 GBytes   932 Mbits/sec                  receiver
```

speedtest-cli

Command line interface for testing internet bandwidth using speedtest.net

Example usage:

```
#benchmark internet bandwidth and ping
speedtest-cli
```

Example output:

```
Retrieving speedtest.net configuration...
Testing from Deutsche Telekom AG (79.251.1.2)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by ProSiebenSat.1 Tech Solutions (Unterföhring) [69.19 km]: 8.48 ms
Testing download speed................................................................................
Download: 266.06 Mbit/s
Testing upload speed..............................................................................................
Upload: 97.26 Mbit/s
```

CAMPUS Rosenheim
Computer Science

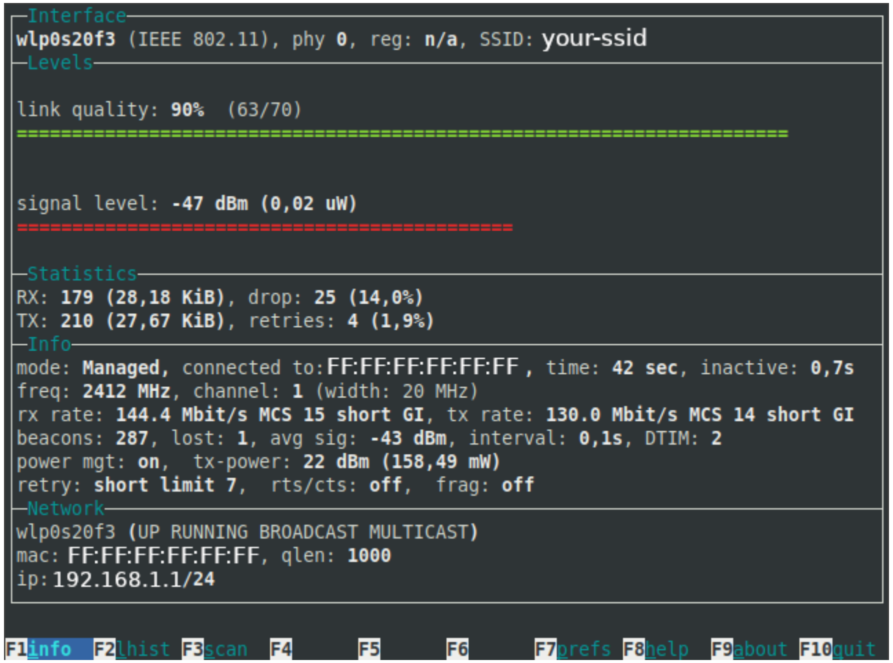
Technische Hochschule Rosenheim
Technical University of Applied Sciences

wavemon - a wireless network monitor

Example usage:

```
#monitor wlan quality and benchmark read/write speed
wavemon
```

Example output:



```
Interface
wlp0s20f3 (IEEE 802.11), phy 0, reg: n/a, SSID: your-ssid
Levels

link quality: 90%  (63/70)
======================================================================


signal level: -47 dBm (0,02 uW)
=============================================

Statistics
RX: 179 (28,18 KiB), drop: 25 (14,0%)
TX: 210 (27,67 KiB), retries: 4 (1,9%)
Info
mode: Managed, connected to:FF:FF:FF:FF:FF:FF , time: 42 sec, inactive: 0,7s
freq: 2412 MHz, channel: 1 (width: 20 MHz)
rx rate: 144.4 Mbit/s MCS 15 short GI, tx rate: 130.0 Mbit/s MCS 14 short GI
beacons: 287, lost: 1, avg sig: -43 dBm, interval: 0,1s, DTIM: 2
power mgt: on,  tx-power: 22 dBm (158,49 mW)
retry: short limit 7,  rts/cts: off,  frag: off
Network
wlp0s20f3 (UP RUNNING BROADCAST MULTICAST)
mac: FF:FF:FF:FF:FF:FF, qlen: 1000
ip:192.168.1.1/24


F1info   F2lhist  F3scan   F4       F5       F6       F7prefs  F8help   F9about  F10quit
```

CAMPUS Rosenheim
Computer Science

Technische Hochschule Rosenheim
Technical University of Applied Sciences

Summary and outlook

Summary

- Performance measures
- Benchmarking
- Popular benchmark suites
- Tools: CPU, GPU, Memory, IO, Network

Outlook

- Processor architecture