Teaching perf to show processor hazards

Madhavan Srinivasan
maddy@linux.vnet.ibm.com
Linux Technology Centre - IBM
Agenda

• Processor Pipeline
• Pipeline issues/Hazard
• IBM Processor Sampling support
• Perf API - arch neutral interface
• Perf tool – options and enhancements
• Screenshot
Instruction cycle

- Processing instruction includes these steps
  - Fetch opcode
  - Decode stage
  - register/memory fetch based on opcode type
  - Execute instruction
  - write-back/store the result
Instruction cycle

- Most modern microprocessors employ complex instruction execution (pipelined superscalar)
  - Multiple instruction in parallel
    - more execution units
    - Execution unit divided in different stages
  - Speculation/OOOO Execution
  - Multiple different pipelines/Sub-pipelines
Example: IBM Power9

Performance (Instruction per Cycles)

- Increases Stall cycles
- Reduce workload performance
- Lowers Instruction per cycles
  - Hazard
Hazards

• Prevent the next instruction in the instruction stream from being executing during its designated clock cycle
  • Performance hit
• Classes of Hazards:
  • Structural
    • part of the processor's hardware is needed by two or more instructions at the same time
  • Control
    • conditional branches interfere with instruction fetches in a pipeline
  • Data
    • instructions that exhibit data dependence modify data in different stages of a pipeline
    • Read after Write (RAW) hazards, also known as true dependences
    • Write after Write (WAW) hazards, also known as output dependences
    • Write after Read (WAR) hazards, also known as antidependences
IBM Power Processor Sampling Support

• Why sample
  • Identification of hotspots in code/data and performance-sensitive areas
• Mark (sample) an instruction and collection details
  • 64bit register records details about marked instruction during its lifetime in pipeline
• Power ISA provides three special purpose registers
  • Sampled Instruction Address Register (SIAR)
  • Sampled Data Address Register (SDAR)
  • Sample Instruction Event Register (SIER)

• Source: https://www-355.ibm.com/systems/power/openpower/tgcmDocumentRepository.xhtml?aliasId=POWER9_Sforza#
Why perf?

• Performance tool used by eco-system
• Provides access to Performance Monitoring Unit (PMU)
  • Allows to closer look at hardware behaviour
• Capability to generate reports out of data collected
• It is fast, lightweight and precise
Why add hazard information in perf

- perf today support exporting memory sampling information
  - PERF_SAMPLE_DATA_SRC and PERF_SAMPLE_WEIGHT
- Based on hardware support, it expose
  - Instruction class (load, store ....)
  - where the data came from (memory hierarchy, hit, hitm, miss)
  - how long did it took for the reload (time in cycles)
  - Data translation (TLB), snoop
Challenges extending -- `perf_mem_data_src`

- `perf_mem_data_src` intended for memory sampling
- Not enough bits to expose
  - pipeline stage
  - hazard reason
  - Stall reason
  - Other instruction class
Approach to export hazard data via *perf*

- Struct to collect hazard data
- Sampling type/format
- Tool option to notify hazard data collection
- New reporting mode to present the hazard data
- Optional new built-in tool (wrapper for *perf* record)
  - Capture and present Hazard data – Usability
  - Similar to "*perf mem"
Hazard data – perf screenshot
perf_pipeline_haz_data

• Pipeline Stages as u32
  • Arch can decide how many
  • Bit mask or Value as index
• Hazard and stall reasons as separate fields
  • Cleaner implementation
  • Multiple hazard representation
• Instruction cache hierarchy
• Processor version
  • tool side to post process
**perf_pipeline_haz_data** – struct to collect hazard data

- Added new perf sample type/format
  - `PERF_SAMPLE_PIPELINE_HAZ`
- Proposed to be part of `include/uapi/linux/perf_event.h`
- Macros could be part of arch folder (ex.. arch/powerpc/include/uapi/asm/perf_pipeline_haz.h)
perf tool -- Enhancements for hazard capture

• New perf tool option
  • User to indicate hazard data capture
  • Proposing "-H" as option
  • Needed to enable attr_sample_type

Raw event "r401e0" used here is "PM_MRK_INST_CMPL" which enables IBM Power processor sampling support to capture hazard/stall data
perf tool – Enhancements for hazard capture

• Support functions to present raw hazard structure data
  • Perf report "-D" support

Screenshot show one PERF_RECORD_SAMPLE data output from "perf report -D" command. Presents all the elements of *perf_pipeline_haz_data* struct
**hazard-info** – perf report enhancement

- New "--hazard-info" mode
- Support new –sort types
- Focused on hazard data presentation
Legal Statement

• This work represents the view of the authors and does not necessarily represent the view of the employers (IBM Corporation).
• IBM and IBM (Logo) are trademarks or registered trademarks of International Business Machines in United States and/or other countries.
• Linux is a registered trademark of Linus Torvalds.
• Other company, product and service names may be trademarks or service marks of others.
Thank you