Delay and Bypass: Ready and Criticality Aware Instruction Scheduling in Out-of-Order Processors

Mehdi Alipour, Stefanos Kaxiras, David Black-Schaffer, Rakesh Kumar

You will learn

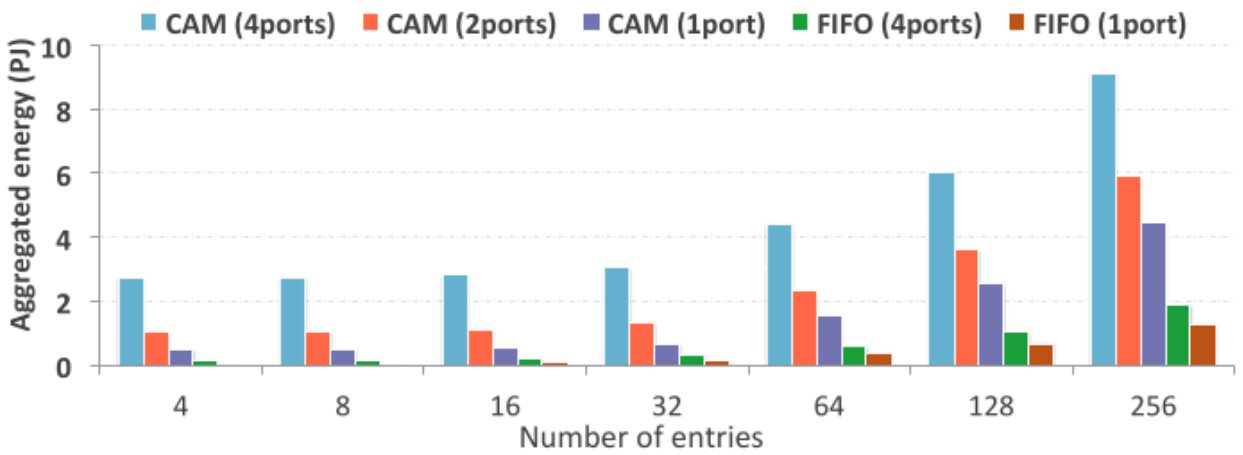
- How instruction scheduling works
- What affects the energy efficiency of the scheduler
- How the energy efficiency can be improved

Instruction Scheduling

- Achieve high performance in out of order execution
- Use instruction queue
  - Width: Instructions issued per cycle
  - Depth: Window of schedulable instructions

Energy usage of instruction queue

- Most complex and power-intensive core component (18-40%)
  - “Wake up” instructions when operands are ready
  - Select instruction for execution based on priority heuristics
- Power consumption of the IQ grows dramatically with depth and width

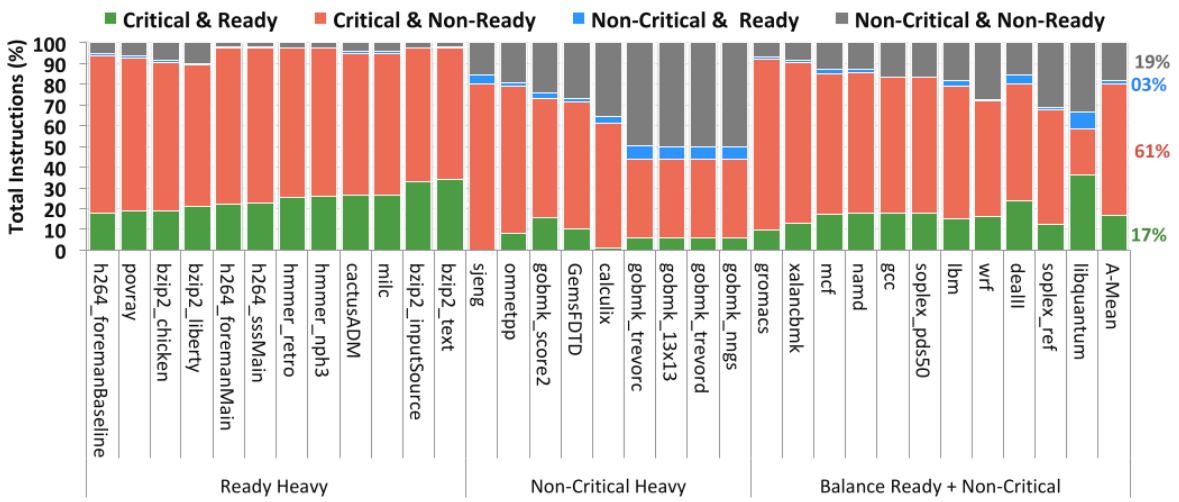


Energy efficient instruction scheduling methods

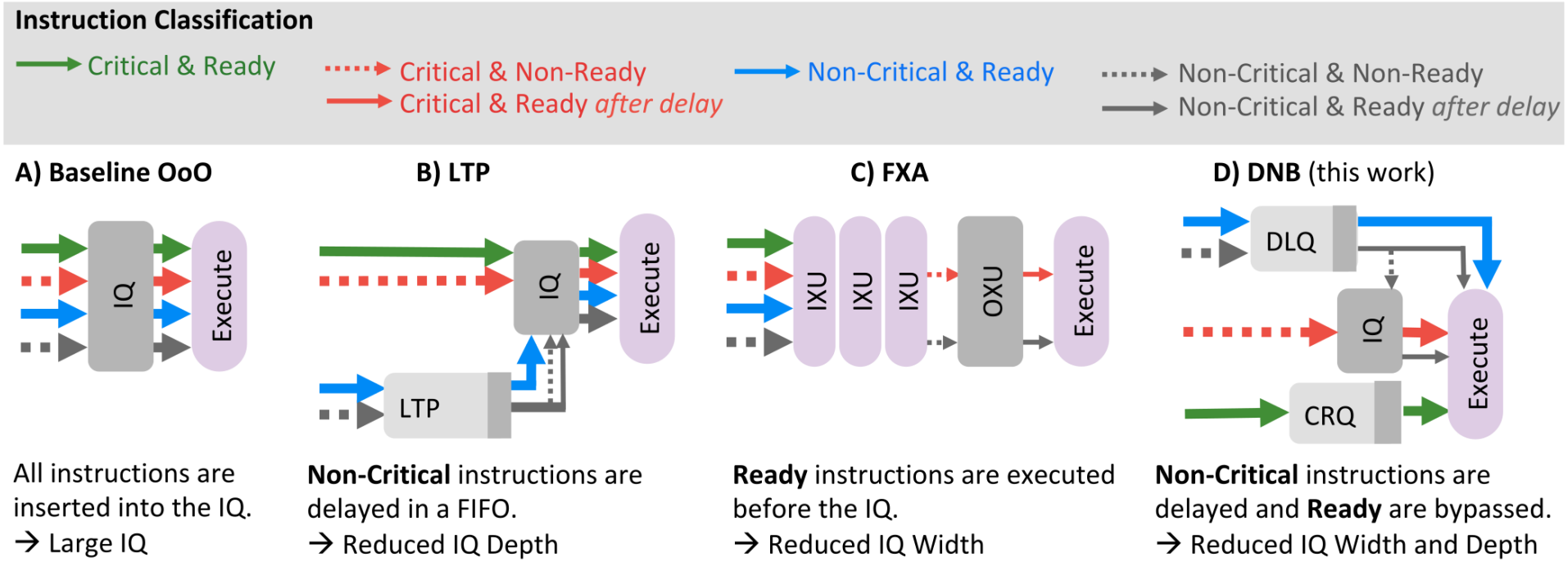
- Long Term Parking (LTP)
  - 91% performance with 74% energy usage
- Front-end Execution Architecture (FXA)
  - 89% performance with 53% energy usage
- **New: Delay and Bypass (DNB)**
  - 95% performance with 33% energy usage

Instruction classification

- Critical: Hurt performance if delayed
- Ready: All operands available



Instruction scheduling methods



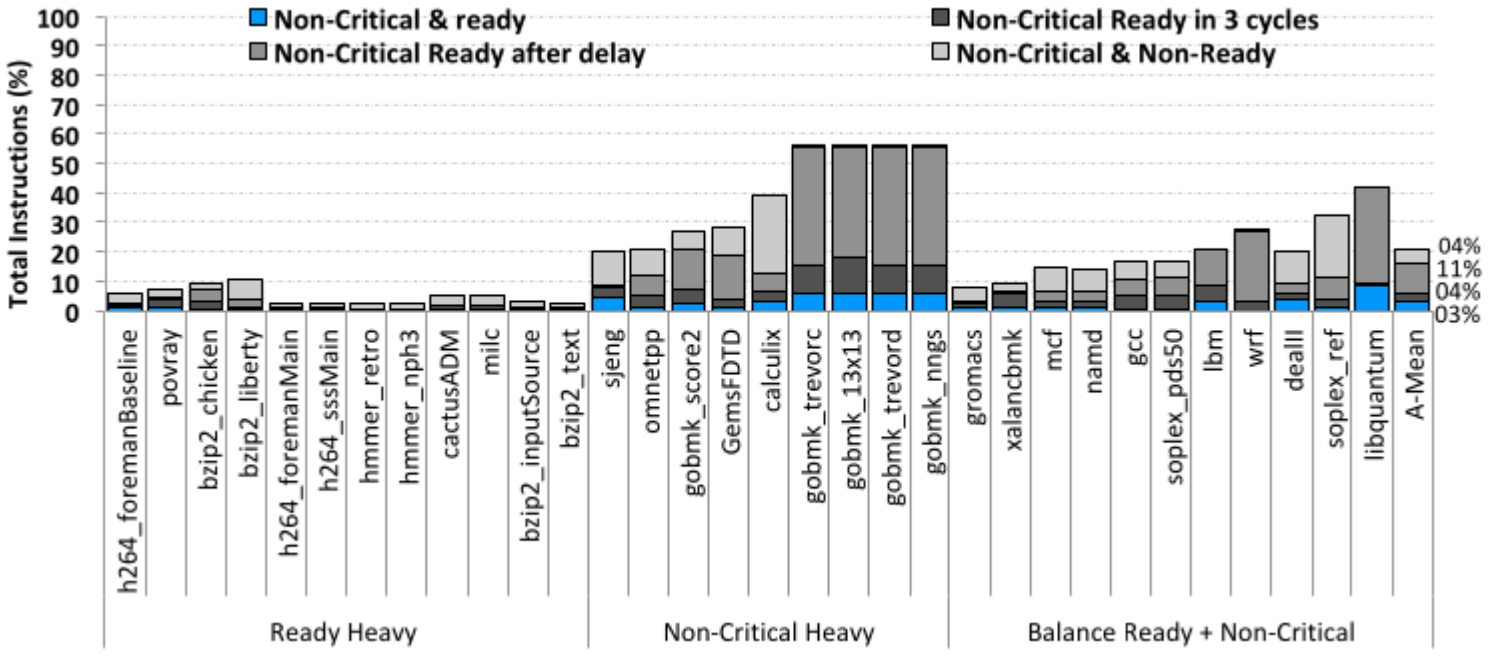
All instructions are inserted into the IQ.
→ Large IQ

Non-Critical instructions are delayed in a FIFO.
→ Reduced IQ Depth

Ready instructions are executed before the IQ.
→ Reduced IQ Width

Non-Critical instructions are delayed and **Ready** are bypassed.
→ Reduced IQ Width and Depth

Readiness of non-critical instructions

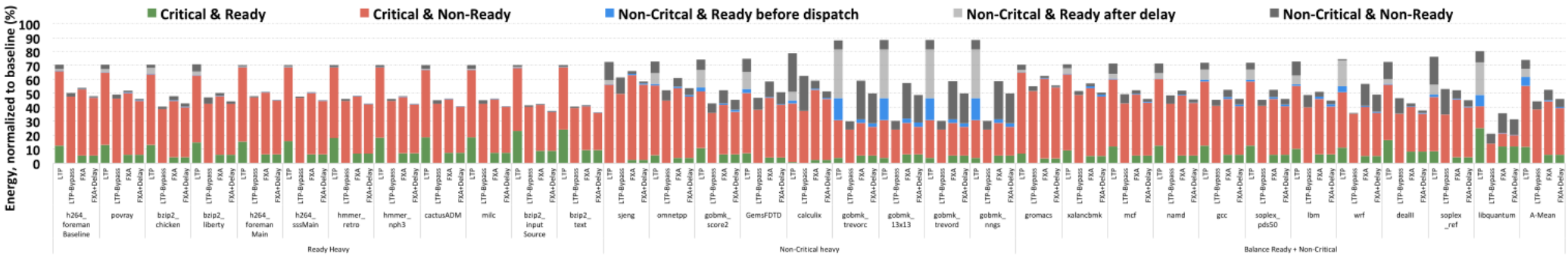


Method comparison

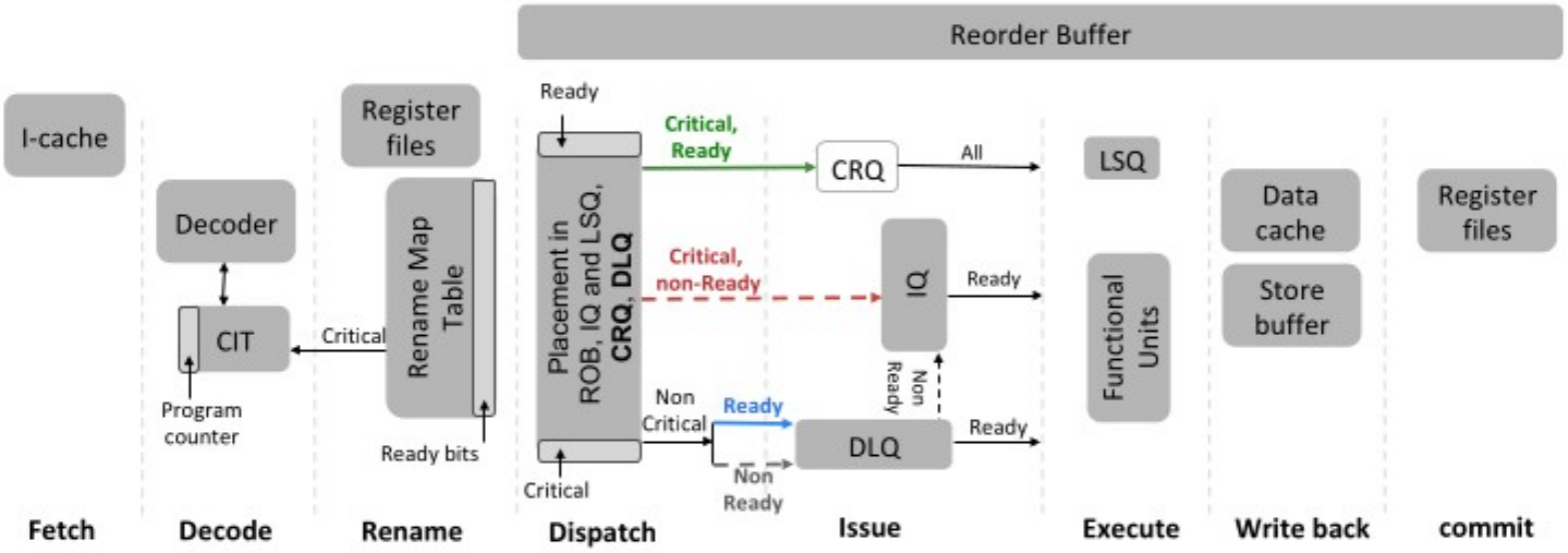
Design	Technique	Awareness			Instruction Issue						Results	
		C	R	R after delay	R-C	R-NC	NR-C NR after delay	NR-C *R after delay*	NR-NC NR after delay	NR-NC *R after delay*	Performance	Scheduling Energy
OoO					**IQ**	**IQ**	**IQ**	**IQ**	**IQ**	**IQ**	100%	100%
LTP	Park (delay)	✓			**IQ**	**Park ➜ IQ**	**IQ**	**IQ**	**Park ➜ IQ**	**Park ➜ IQ**	91%	74%
FXA	Filter (bypass)		✓	3 cycles	Filter	Filter	**Filter ➜ IQ**	Filter (2% in 3c)	**Filter ➜ IQ**	Filter (4% in 3c)	89%	53%
DNB	Delay & Bypass	✓	✓	✓	Bypass	Delay ➜ Bypass	**IQ**	**IQ**	**Delay ➜ IQ**	Delay ➜ Bypass	**95%**	**34%**

Naive extension of FXA and DNB

- FXA + Delay
  - Energy usage reduced from 53% to 46%
- LTP + Bypass
  - Energy usage reduced from 74% to 44%



DNB


Reorder Buffer
I-cache
Decoder
CIT
Program counter
Register files
Rename Map Table
Critical
Ready bits
Ready
Placement in ROB, IQ and LSQ, CRQ, DLQ
Critical
Critical, Ready
CRQ
All
Critical, non-Ready
IQ
Ready
Non Ready
Non Critical
Ready
DLQ
Ready
Non Ready
LSQ
Functional Units
Data cache
Store buffer
Register files
Fetch
Decode
Rename
Dispatch
Issue
Execute
Write back
commit

Evaluation Methodology

THE HASWELL-LIKE [24] BASELINE MICROARCHITECTURAL PARAMETERS.

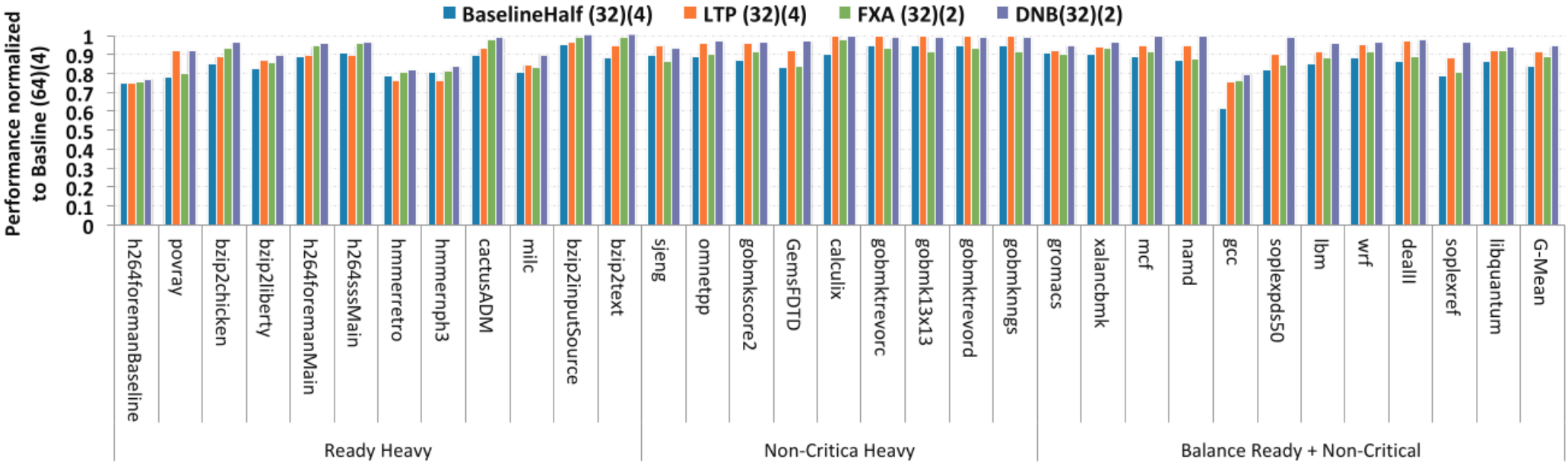
Freq, ISA	3.4 GHz, x86-64
L1i/d	32KiB, 8-way, 4clk
L2	256KiB, 8-way, 12clk
L3	1MiB, 8-way, 36clk
DRAM	200clk
Branch Predictor	Two level, front end penalty 10clk
ROB/IQ/RF(Int,FP)/LQ/SQ	192/60/(130,130),72/42
Prefetcher	enabled
Technology/VDD/temp	22nm itrs-hp/0.8/360K

Evaluation Methodology

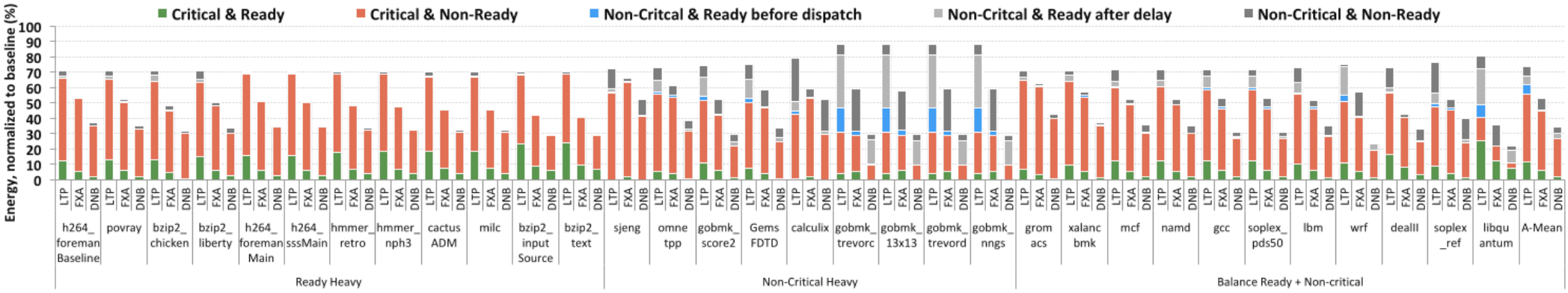
SCHEDULING RESOURCE CONFIGURATIONS.

Design	IQ Depth/Width	Other Scheduler	Issue	RF ports
Baseline	64/4		4	12
LTP [3]	32/4	FIFO 128/4	4	12
FXA [18]	32/2	3-stage pipeline	5	16
DNB	32/2	FIFOs 32/2, 128/2	4	12

Performance results



Energy results



Performance loss