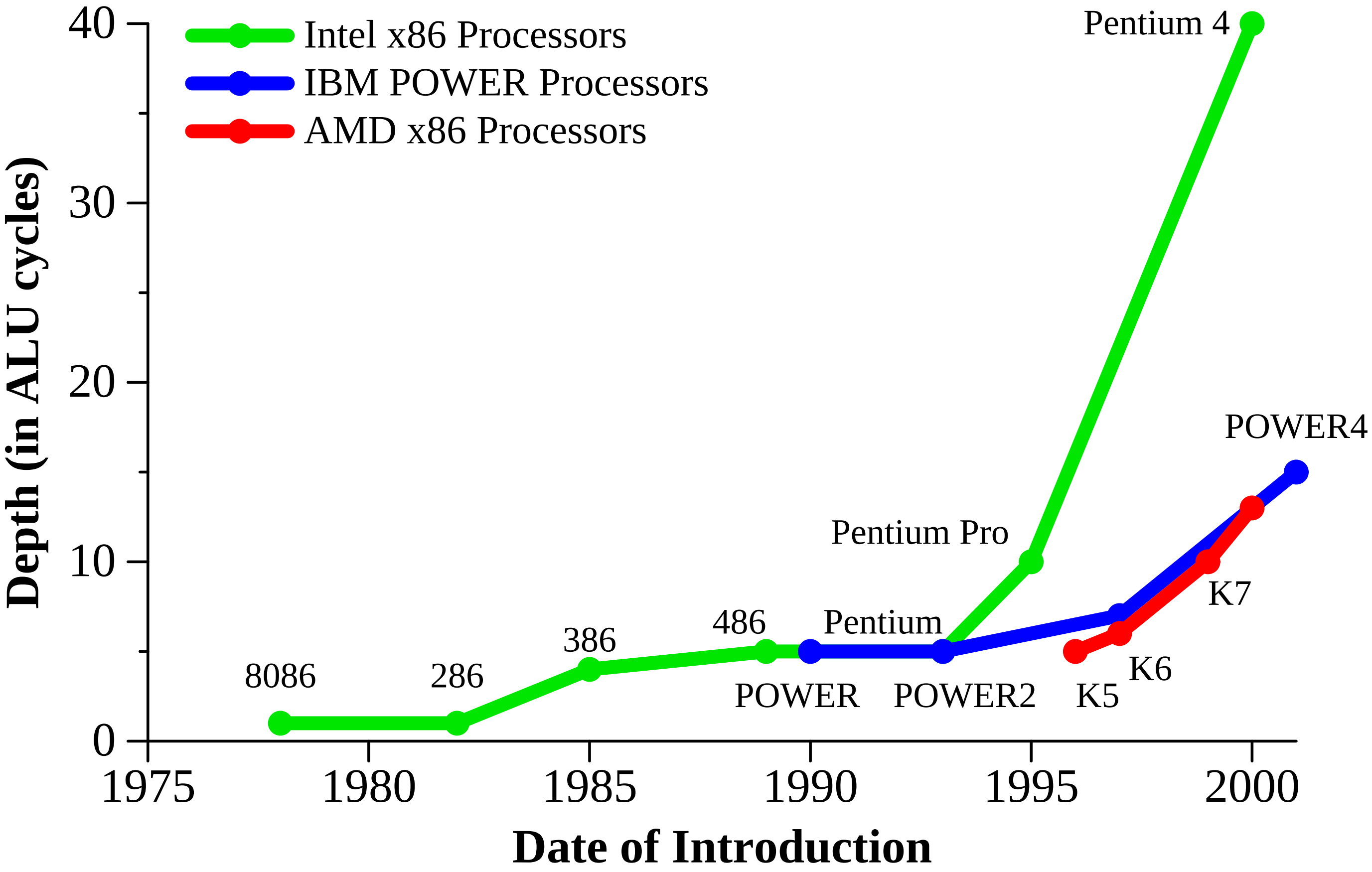
Select-Free Instruction Scheduling Logic

Mary D. Brown † Jared Stark ‡ Yale N. Patt †

†Dept. of Electrical and Computer Engineering
The University of Texas at Austin
{mbrown,patt}@ece.utexas.edu

‡Microprocessor Research
Intel Labs
(formerly MRL)
jared.w.stark@intel.com

Integer Pipeline Depths



Critical Loops

Critical loops determine the minimum time to execute a chain of dependent instructions.

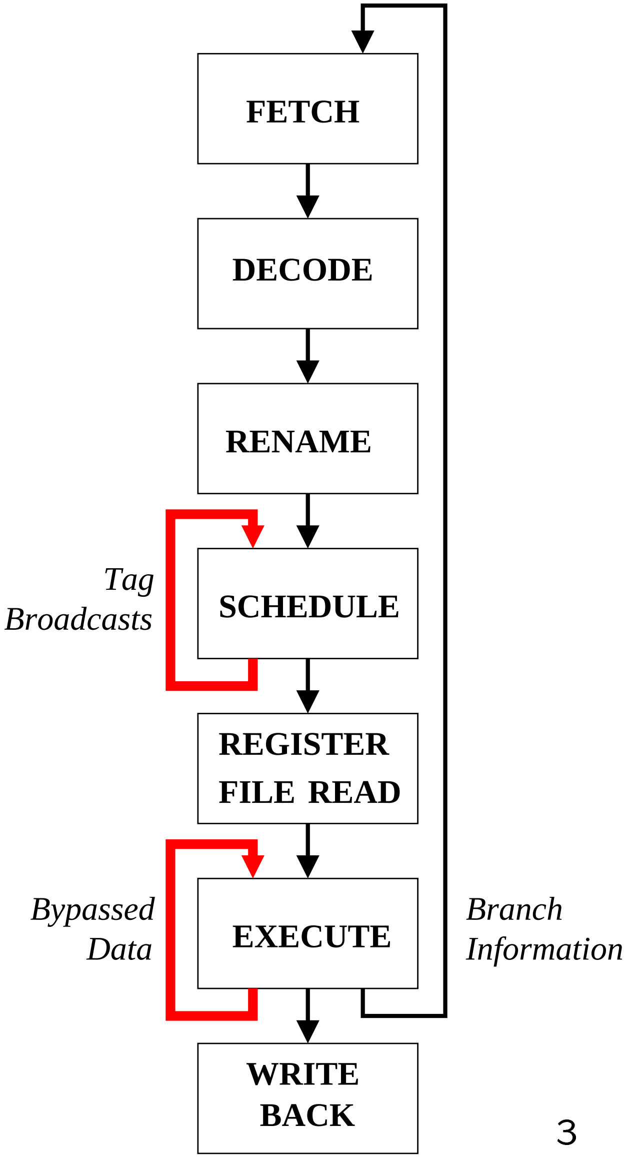
They cannot be pipelined without a significant impact on (single-thread) IPC.

Two examples:

- Execute and Bypass Loop
- Scheduling Loop

  Impact on IPC of pipelining the scheduling loop

  over 2 cycles: 10%
  over 3 cycles: 18%



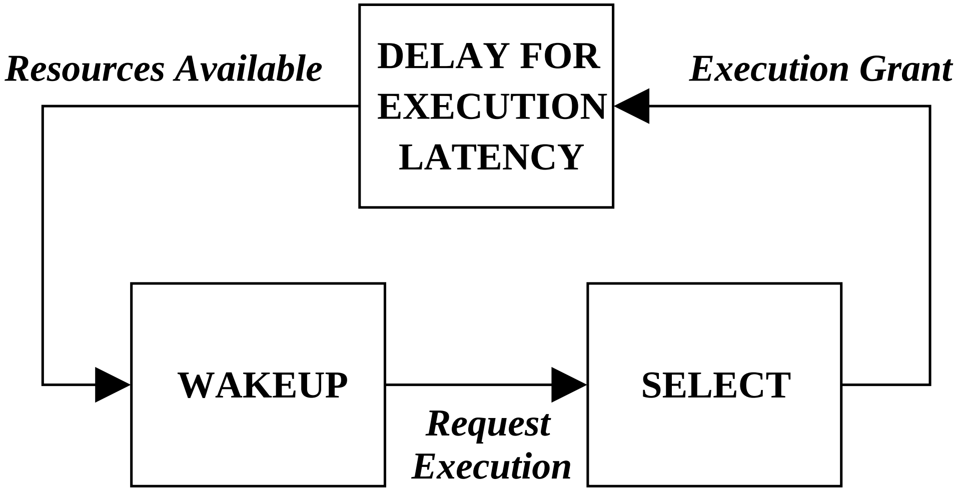
Outline

1. Background
   1.1 Baseline scheduling logic and execution core
   1.2 Pipelining the Baseline scheduling logic
2. Select-Free Scheduling Logic
   2.1 Description and Terminology
   2.2 Performance Results
3. Predict Another Wakeup
   3.1 Description and example
   3.2 Performance Results
4. Previous Work
5. Conclusion

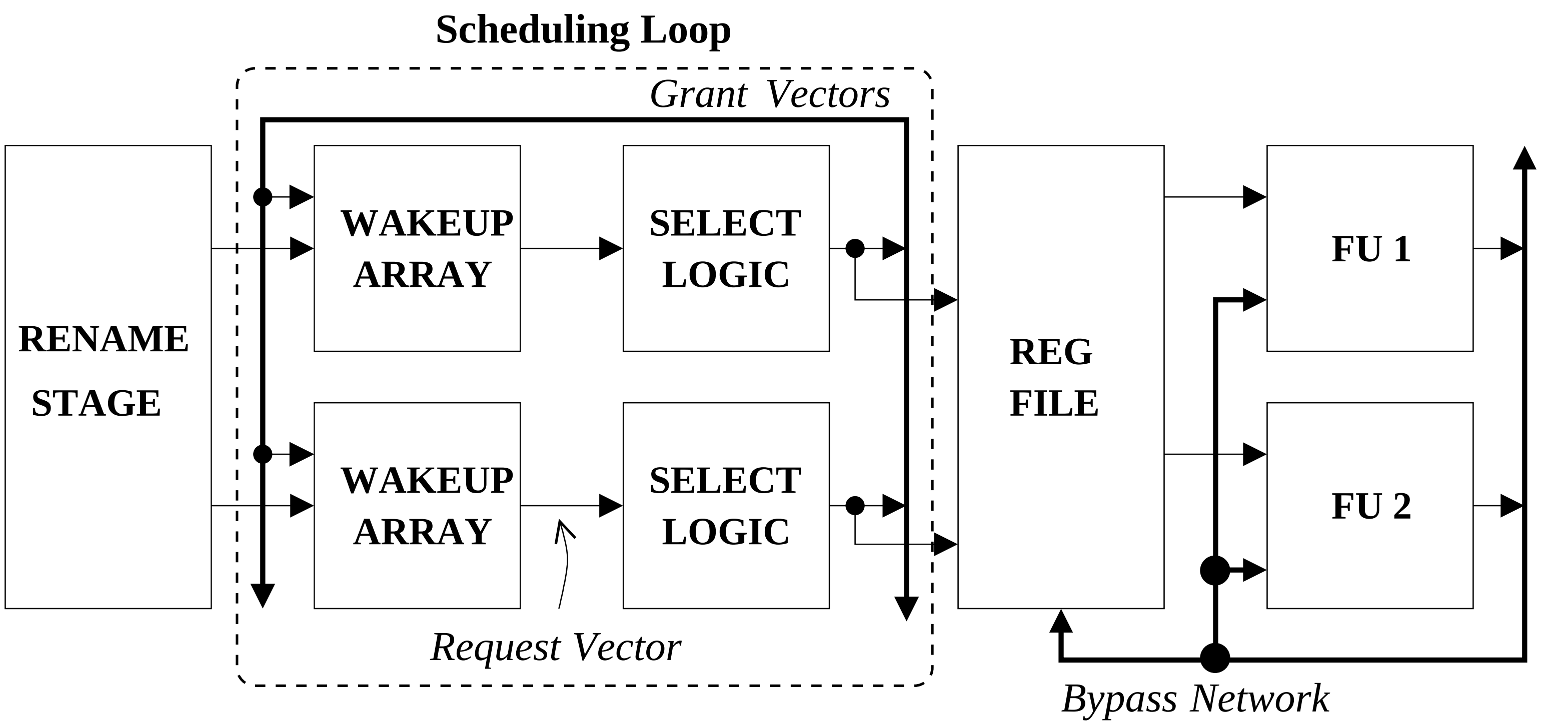
Instruction Scheduling

Three operations:

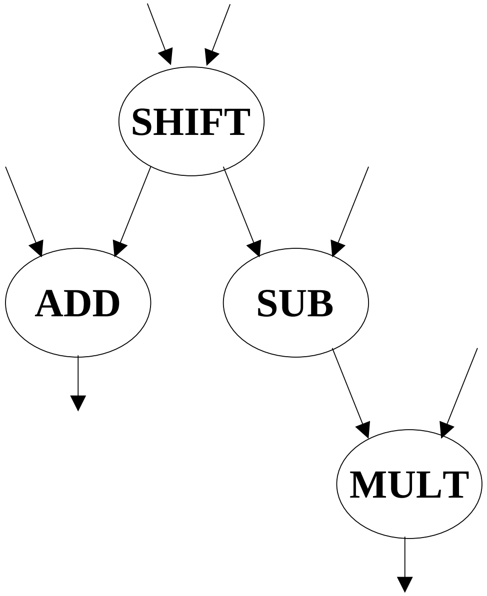
- Determine which instructions are ready to be scheduled (**Wakeup**)
- Pick instructions for execution (**Select**)
- Notify instructions when resources are available



Baseline Execution Core



The Wakeup Array

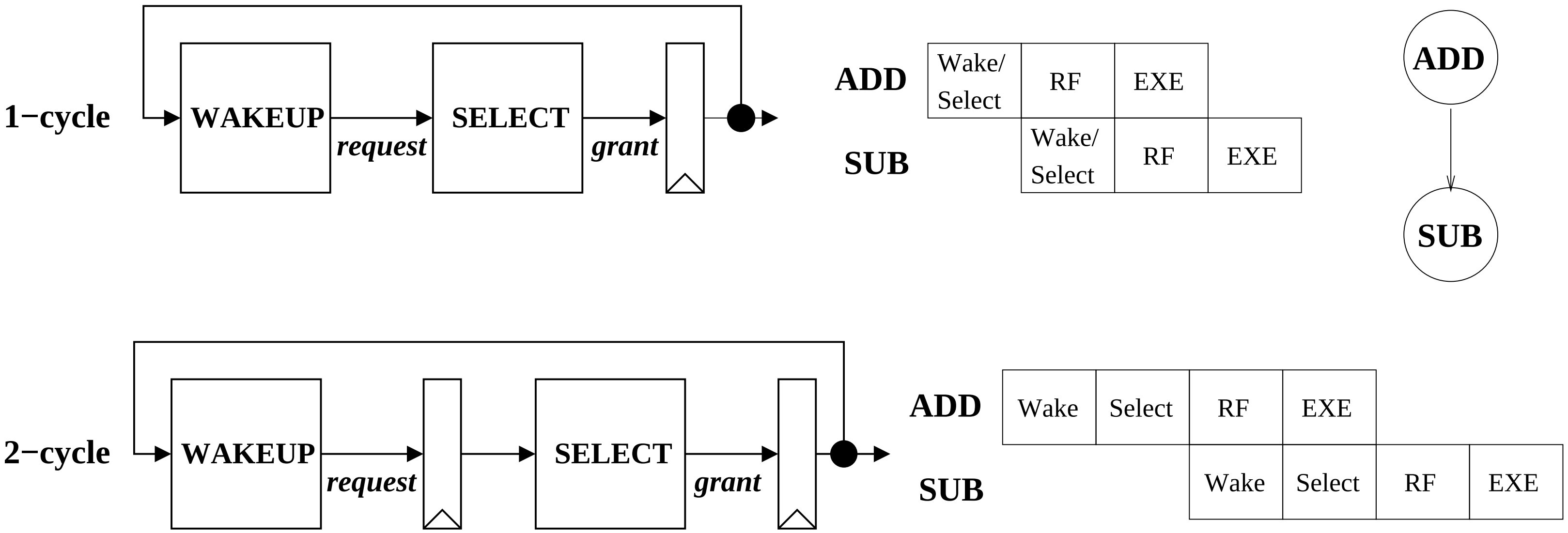


RESOURCE AVAILABLE line for the instruction in ...

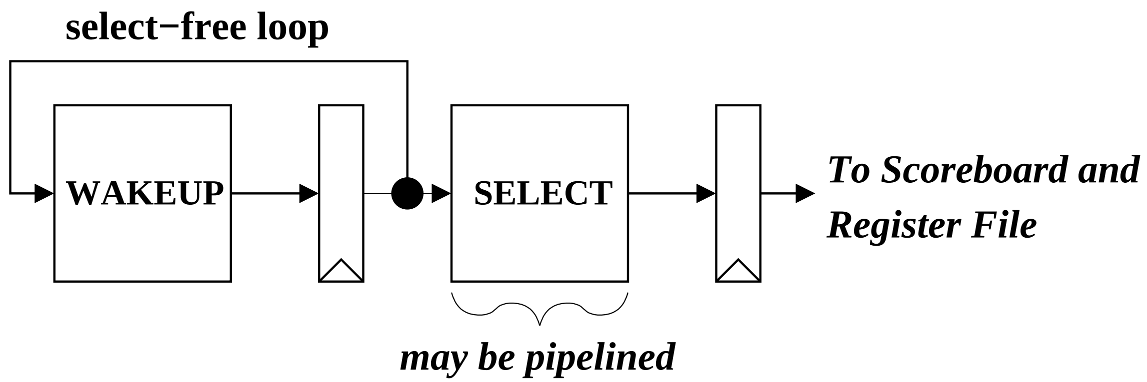
		ENTRY 1	*ENTRY 2*	*ENTRY 3*	*ENTRY 4*		*ENTRY N*
(SHIFT)	**ENTRY 1**					• • •	
(ADD)	**ENTRY 2**	**1**				• • •	
(SUB)	**ENTRY 3**	**1**				• • •	
(MULT)	**ENTRY 4**			**1**		• • •	

'1' indicates a set bit of a Resource Vector.

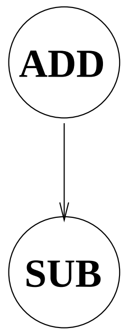
1-Cycle and 2-cycle Scheduling Loops



Select-Free Scheduling Loop



ADD	Wake	Select	RF	EXE	
SUB		Wake	Select	RF	EXE



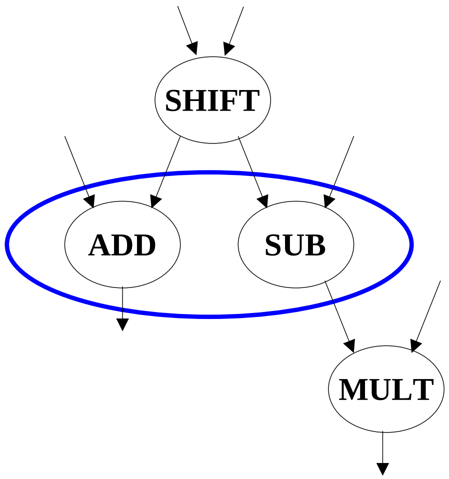
Select logic accounts for more than half of the scheduling latency.

(Palacharla, Jouppi, and Smith, ISCA-24)

Collisions

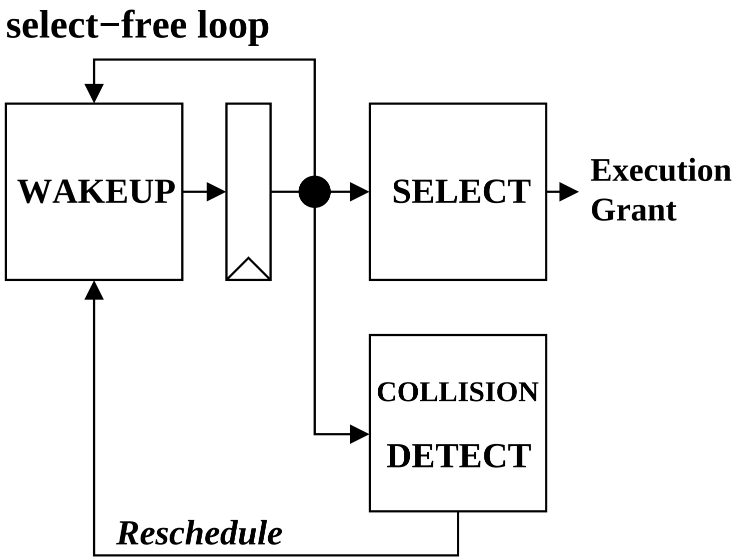
Definitions

- A *collision* occurs when more than one instruction wake up at a time.
- A *collision victim* is an instruction that is involved in a collision and is not selected.



Detection

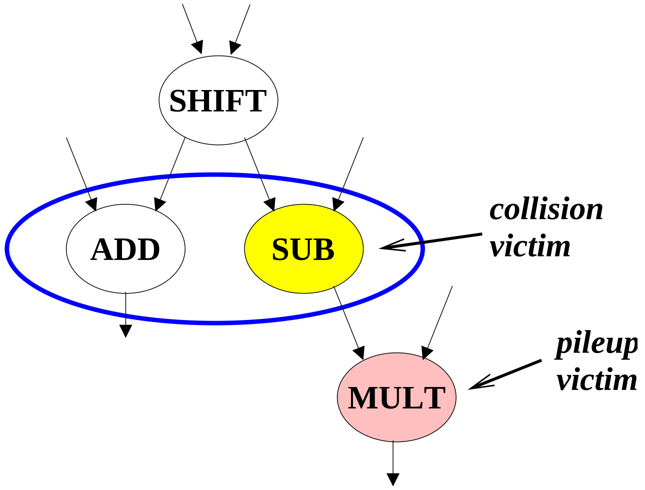
- Detected using additional circuit in parallel with select



Pileups

Definitions

- A *pileup* occurs when instructions wake up early because their source operands were collision victims.
- A *pileup victim* is an instruction that is involved in a pileup and is selected.



Detection

- Detected using scoreboard in parallel with register file access
- Scoreboard records correctly scheduled instructions.
- Operation:
  - Selected instructions read scoreboard entries for their source operands.
  - All sources correctly scheduled?

    Yes: update scoreboard, continue execution

    No: pileup victim, reschedule

Machine Assumptions

- 256-entry instruction window
- 128-entry scheduling window (partitioned)
- 8 functional units (4 single-cycle, 4 multi-cycle + branches)

Note that 1-cycle scheduling loops are only needed for scheduling 1-cycle instructions that produce results.

Collision and Pileup Victims

Three scheduler configurations:

- 8 pick-1 schedulers, each have 16 wakeup array entries
- 4 pick-2 schedulers, each have 32 wakeup array entries
- 2 pick-4 schedulers, each have 64 wakeup array entries

	8 Pick-1 Schedulers	4 Pick-2 Schedulers	2 Pick-4 Schedulers
% of instructions that are collision victims	10.6%	7.3%	4.4%
% of instructions that are pileup victims	11.0%	7.2%	3.7%

Machine Models

Three machines:

- **Baseline:** 2-cycle, pipelined conventional scheduling loop.
- **Select-Free:**

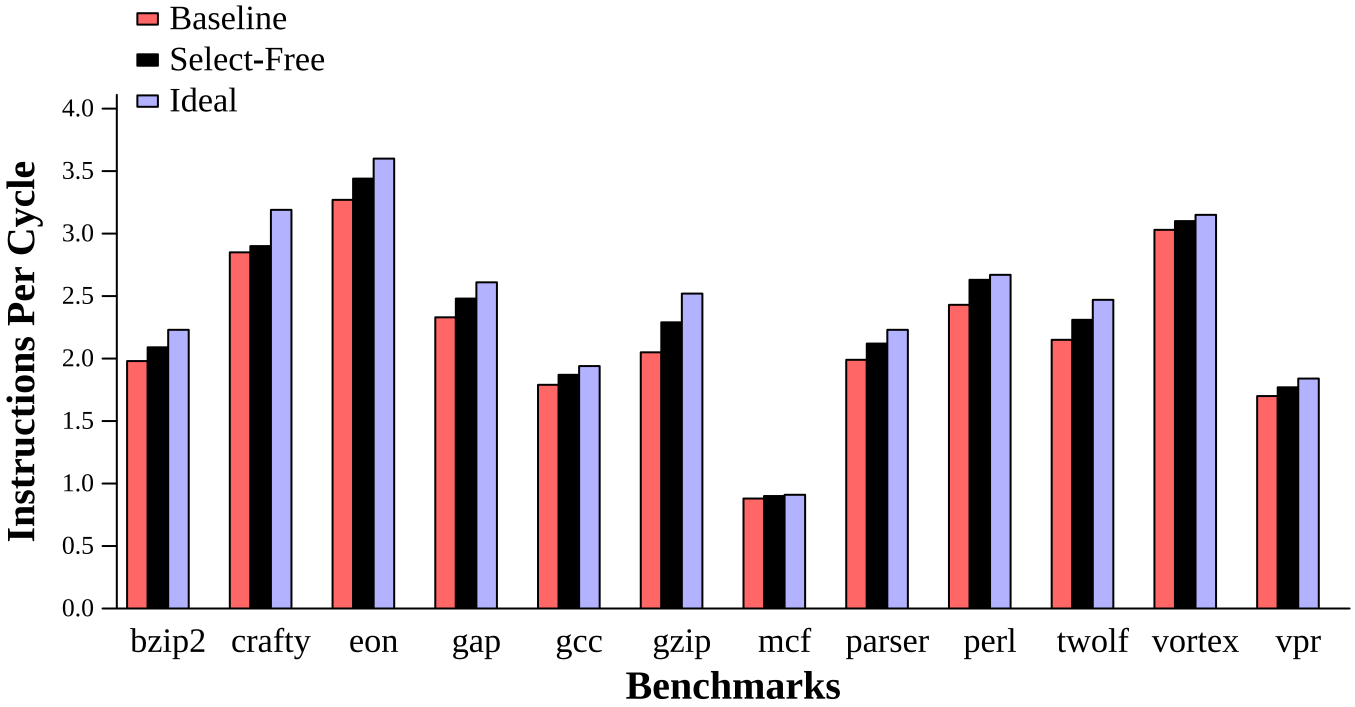
  Schedulers for single-cycle functional units have select-free loops.

  Schedulers for multi-cycle functional units have 2-cycle loops.
- **Ideal:** 1-cycle scheduling loop.

All machines have the same pipeline depth.

Baseline and Select-Free may be clocked at a higher frequency.

IPC using 8 Pick-1 Schedulers

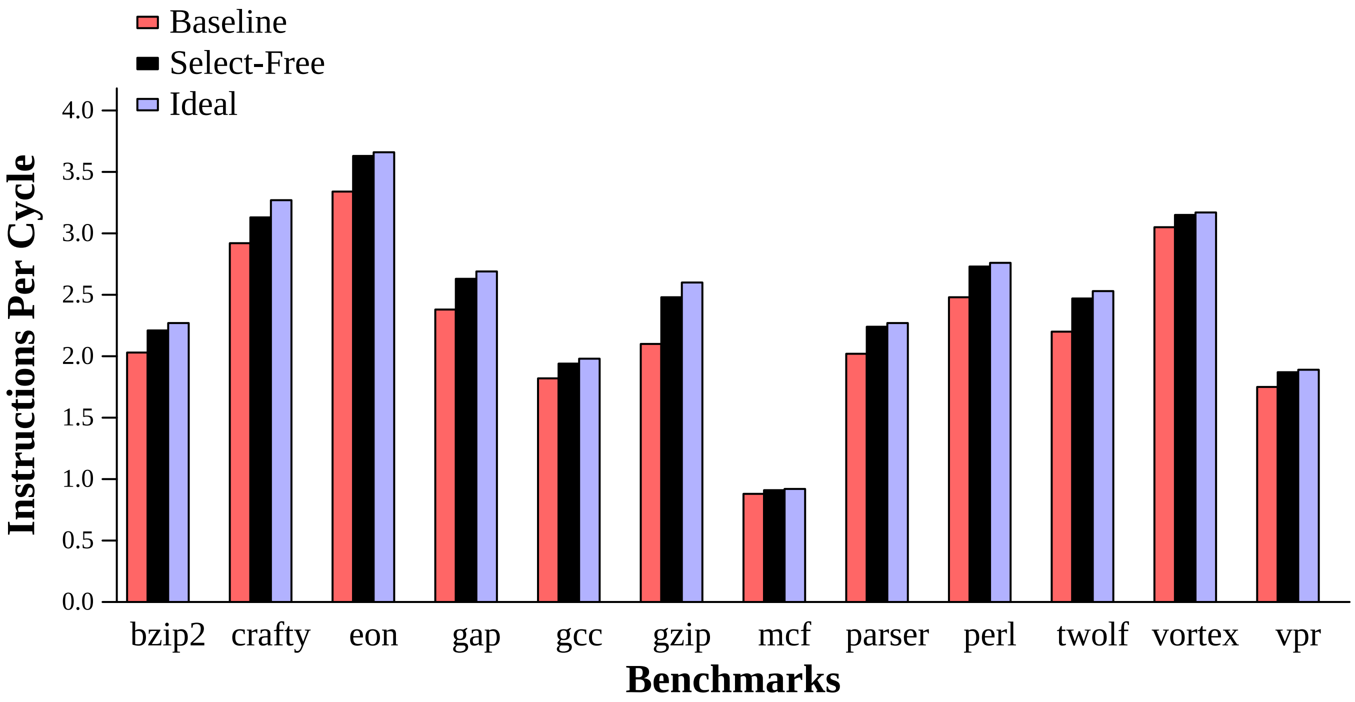


IPC of Ideal is 10% above Baseline

IPC of Select-Free is 6% above Baseline

IPC of Select-Free is 4% from Ideal

IPC using 4 Pick-2 Schedulers

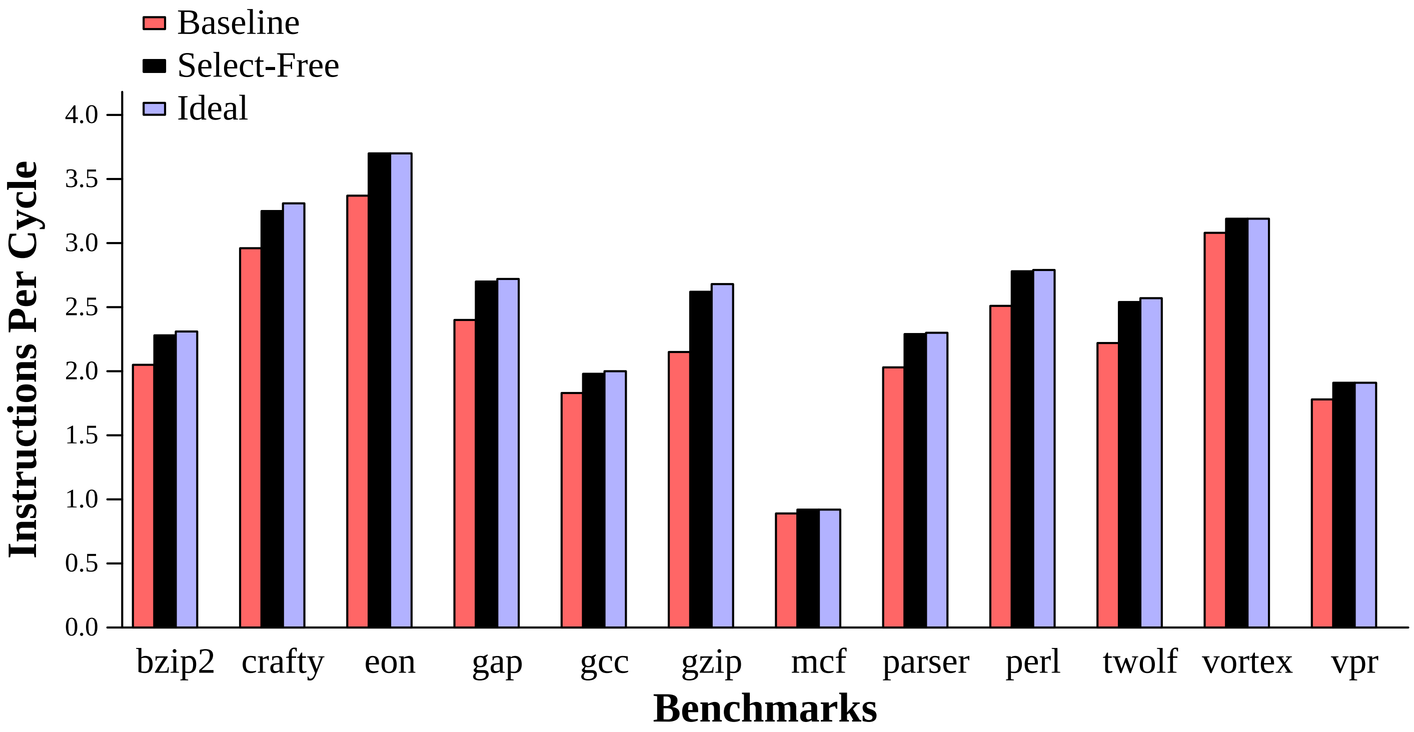


IPC of Ideal is 10% above Baseline

IPC of Select-Free is 8% above Baseline

IPC of Select-Free within 2% of Ideal

IPC using 2 Pick-4 Schedulers



IPC of Ideal is 10.2% above Baseline

IPC of Select-Free is 9.5% above Baseline

IPC of Select-Free is 0.7% from Ideal

Preventing Collisions using Predict Another Wakeup (PAW)

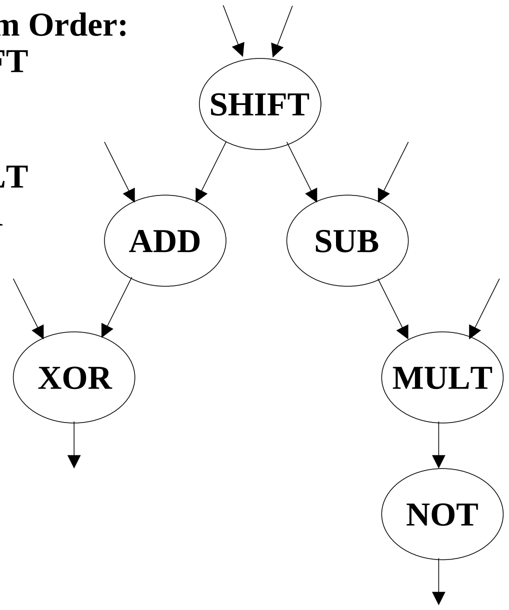
Idea: Instructions don't wake up in the same cycle that an older instruction in the same wakeup array might wake up.

An instruction delays its wakeup by 1 cycle if a source operand for an older instruction becomes available.

- Each instruction has an additional resource vector: the PAW vector.
- The PAW vector indicates what sources are needed by older instructions in the same wakeup array.
- Instructions don't wakeup when there is a PAW vector match.

Example Wakeup Array with PAW Vectors

Program Order:
1: SHIFT
2: ADD
3: SUB
4: MULT
5: XOR
6: NOT



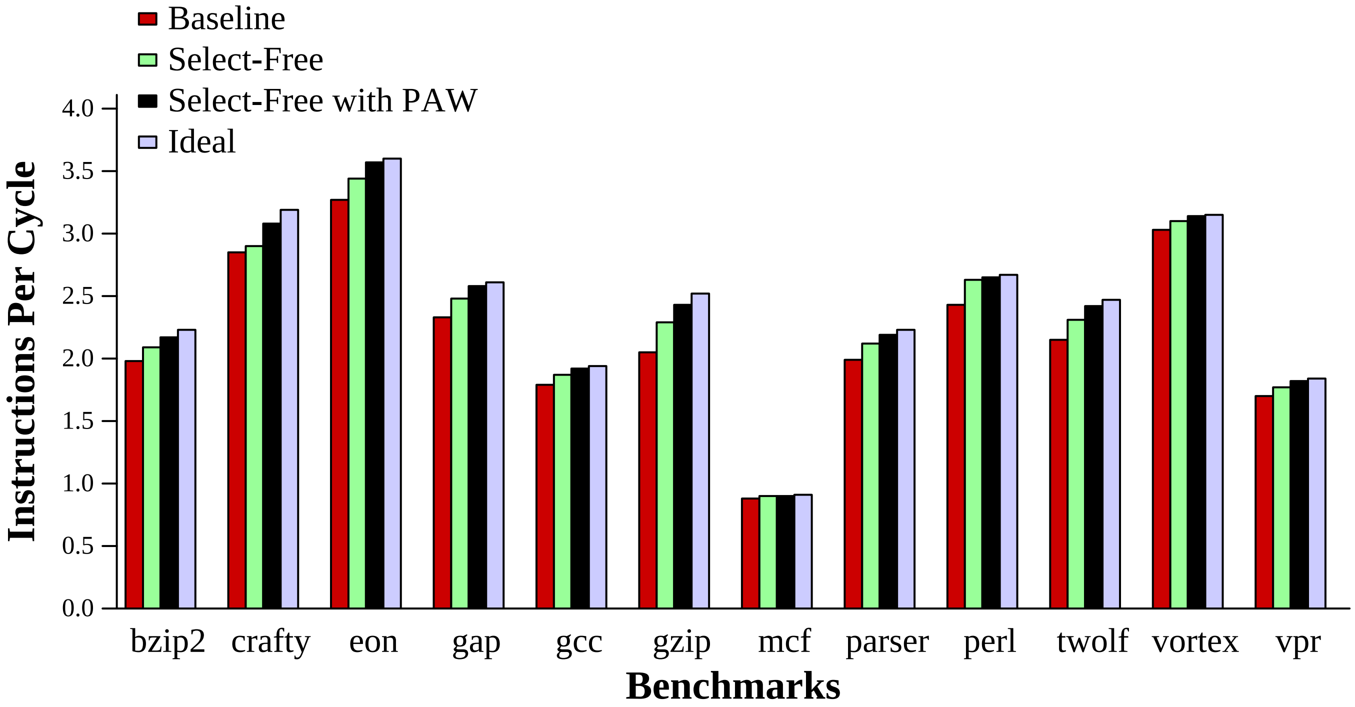
RESOURCE AVAILABLE line for the instruction in ...

		ENTRY 1	*ENTRY 2*	*ENTRY 3*	*ENTRY 4*	*ENTRY 5*	*ENTRY 6*		*ENTRY N*
(SHIFT)	**ENTRY 1**							• • •	
(ADD)	**ENTRY 2**	1						• • •	
(SUB)	**ENTRY 3**	1						• • •	
(MULT)	**ENTRY 4**			1				• • •	
(XOR)	**ENTRY 5**		1					• • •	
(NOT)	**ENTRY 6**				1			• • •	

'1' indicates a set bit of a Resource Vector.

Shading indicates a set bit of a PAW vector.

IPC using Pick-1 Schedulers



IPC of Ideal is 10% above Baseline

IPC of Select-Free is 6% above Baseline, 4% from Ideal

IPC of Select-Free + PAW is 8% above Baseline, 1.6% from Ideal

Comparison to Our Previous Work

Pipelined Scheduling Logic with Speculative Wakeup

- Instructions wake up based on *grandparents'* tags rather than just *parents'* tags.
- Doesn't "break" the critical loop, just stretches it over two cycles.

Speculative Wakeup may be used in conjunction with Select-Free Scheduling Logic for even deeper pipelining.

Conclusions

- The scheduling operation is a critical loop in conventional processors.
- Select-Free Scheduling logic breaks this loop, allowing higher clock frequency.

  Before: scheduling loop was wakeup + select
  Now: scheduling loop is just wakeup

- IPC with Select-Free Schedulers is 8% to 9% above IPC with 2-cycle, pipelined schedulers.
- IPC with Select-Free Schedulers is within 1% to 2% of a machine with Ideal schedulers.