

The purpose of a standard is to -

- Bless the past**
- Arbitrate the present**
- Avoid the future**

Givens -

- **Performance matters**
- **Fast is limited by Moore's Law**
- **Many is limited by algorithm**
- **The speed of light is constant**

Present High Performance -

- 10's to 1000's of FPUs per chip
- 10's to 100's of cycles across chip
- Dynamic logic; no clock
- Pure dataflow internal model
- No precise exceptions
- Some have no program at all
- No notion of global time

754 assumes global time -

- **Modes, rounding and otherwise**
- **Flags**
- **Traps**
- **Control flow determinism**

Dynamic modes (rounding etc.) -

- **VERY expensive to come to a meaningful consistent global state**
- **VERY expensive to be able to come to a consistent global state, whether you actually ever do or not**
- **No problem with static (known to compiler) modes**

Traps -

- **Imprecise non-resuming traps are easy and potentially useful**
- **Precise, resuming traps are hopeless or VERY expensive if taken**
- **Precise, resuming traps are hopeless or VERY expensive even if never taken**

Control Flow Determinism -

- No notion of global time means that hardware computes some things in the future – this is called speculation
- Sometimes the future doesn't happen
- Waiting for the future to get here is VERY expensive
- Discarding futures that didn't happen must be cheap

Proposal – Modes:

- **Retain modes**
- **Explicitly assert that the mode of an operation must be determinable by an inspection of the program according to its language rules (i.e. static binding) and must not require its execution**

Proposal – Flags:

- **Retain flags**
- **State the flag propagation rules**
- **State that inspection points shall be individually specified in the program text**
- **Explicitly assert that the operations which determine the value of the flags at any inspection point must be determinable by an inspection of the program according to its language rules (i.e. static binding) and must not require its execution**

Proposal – Speculation:

- **Explicitly state that arbitrary speculation of operations is permitted, but not across inspection points present in the text of the program**
- **Explicitly state that the speculation state of asynchronous parts of the program (including threads, and signal and exception handlers of all sorts) is undefined**

Ignore these issues, and 754R will -

- Bless the past**
- Be flouted in the present**
- Be a footnote in the future**