

What is the dataflow execution model?

- A program is a graph:
 - nodes are computational blocks
 - Nodes synchronize data.
 - only all inputs are valid is an output produced
 - data flows along edges
- Like an FPGA: pieces (computational components), and inputs

What are the variants of DF models?

- Static dataflow
- k-Static dataflow
 - downside: badly written programs deadlock and produce incorrect outputs
 - upside: finite resources
- Dynamic dataflow
 - upside: easy to code loops
 - downside: infinite resources
- Demands-driven dataflow
 - upside: know everything you have to execute
 - downside: not everything code-able

What is “memory” in a DF computer?

- M-structures
 - alternate between empty and full
 - write: set full
 - read: go empty
- I-structures
 - empty, or premaritally full: i.e. write-once
 - write: set full
 - read: get copy

What is a side-effect?

What/Why is a token store?

What are the two fundamental limits of DF?

- Lack of locality / Context switching
 - Partitioning and it's about efficiency per-die unit area
- Scheduling
 - k-loop bounding
- Does not match imperative languages

Where is DF computing used?

What are the motivations behind WaveScalar?