October 27, 2003

- Coding Guidelines
  - How to format code and other rules to follow for this course
- Coding Efficiency
  - Tutorial Manuals
    - Handel-C Code Optimization
    - Handel-C Advanced Optimization

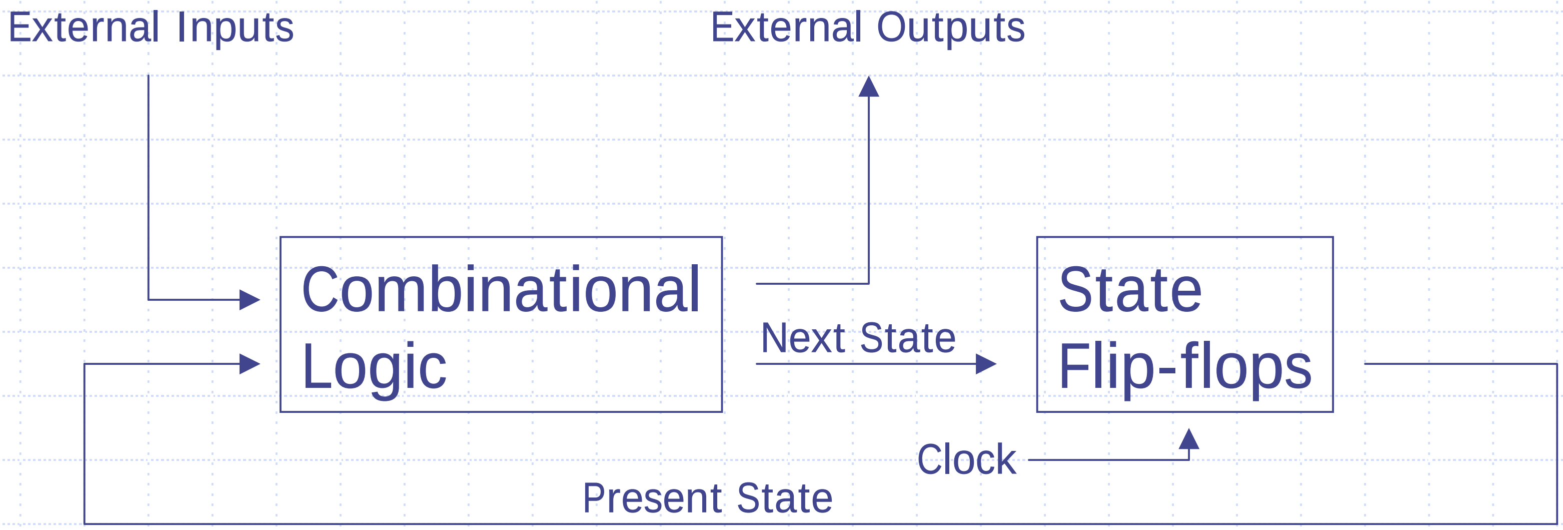
Coding Efficiency

- Software
  - Decide what to compute
  - Then compute it
- Hardware
  - Compute in parallel
  - Select output

Efficiency Issues

- Engineering Tradeoffs
  - Number of gates
  - Number of clock cycles
  - Clock speed
  - Power dissipation
- Software Design
  - Execution time
  - Memory usage

Finite State Machine (FSM)



Maximum clock speed is determined by propagation delays in the Combinational Logic.

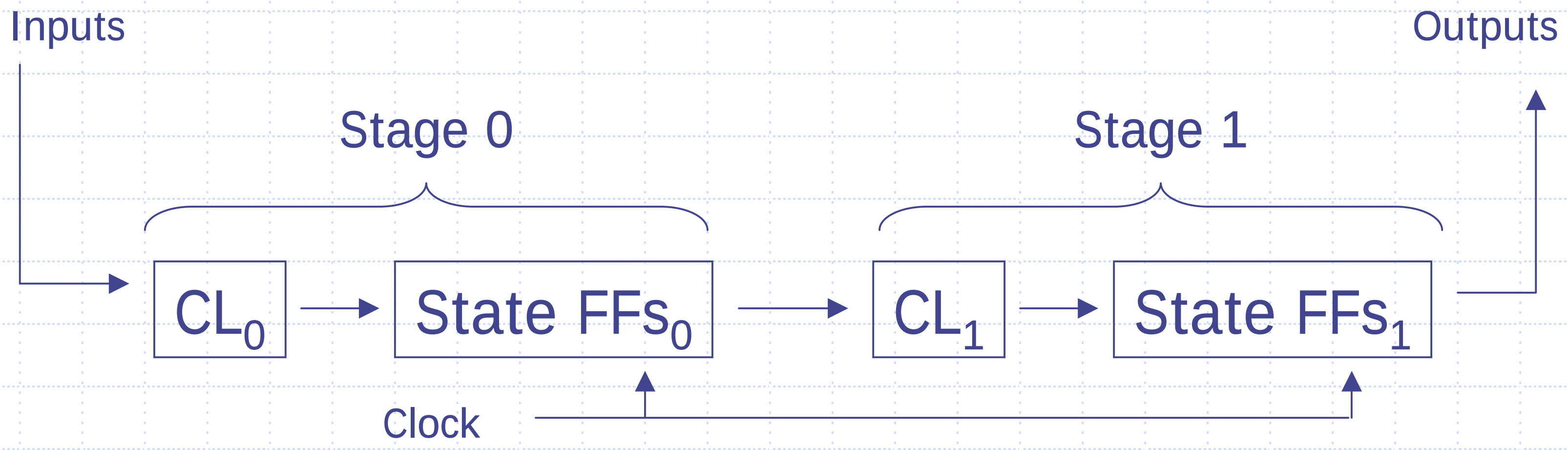
Functions *vs.* Macros

- One copy of the circuit for a function
  - Plus multiplexers and registers for passing parameters and returning result
  - Only one execution at a time
    - Consider an array of functions to overcome this
- Macros and inline functions
  - Are expanded for each reference
  - Passing parameters *vs.* constant args

Pipeline Design

- Add extra circuits to increase clock speed
  - Works for repetitive computations
    - Signal processing
    - CPU fetch-execute cycle
  - Delay to prime the pipeline
  - Generate new result on every clock
  - Faster clock

Pipeline Design



Propagation delays in CL_0 and CL_1 are less than delays in a single combined block of combinational logic, allowing for a faster clock speed.

Parallel Multiplier

- See Handel-C Reference Manual
  - p. 146 and 229
  - But this version generates full products for arbitrary operand widths
- Simulation Walkthrough
  - pipeline.zip – (Downloadable DK Workspace)
  - Compare summaries in Evaluation dirs
  - Compare build.log files