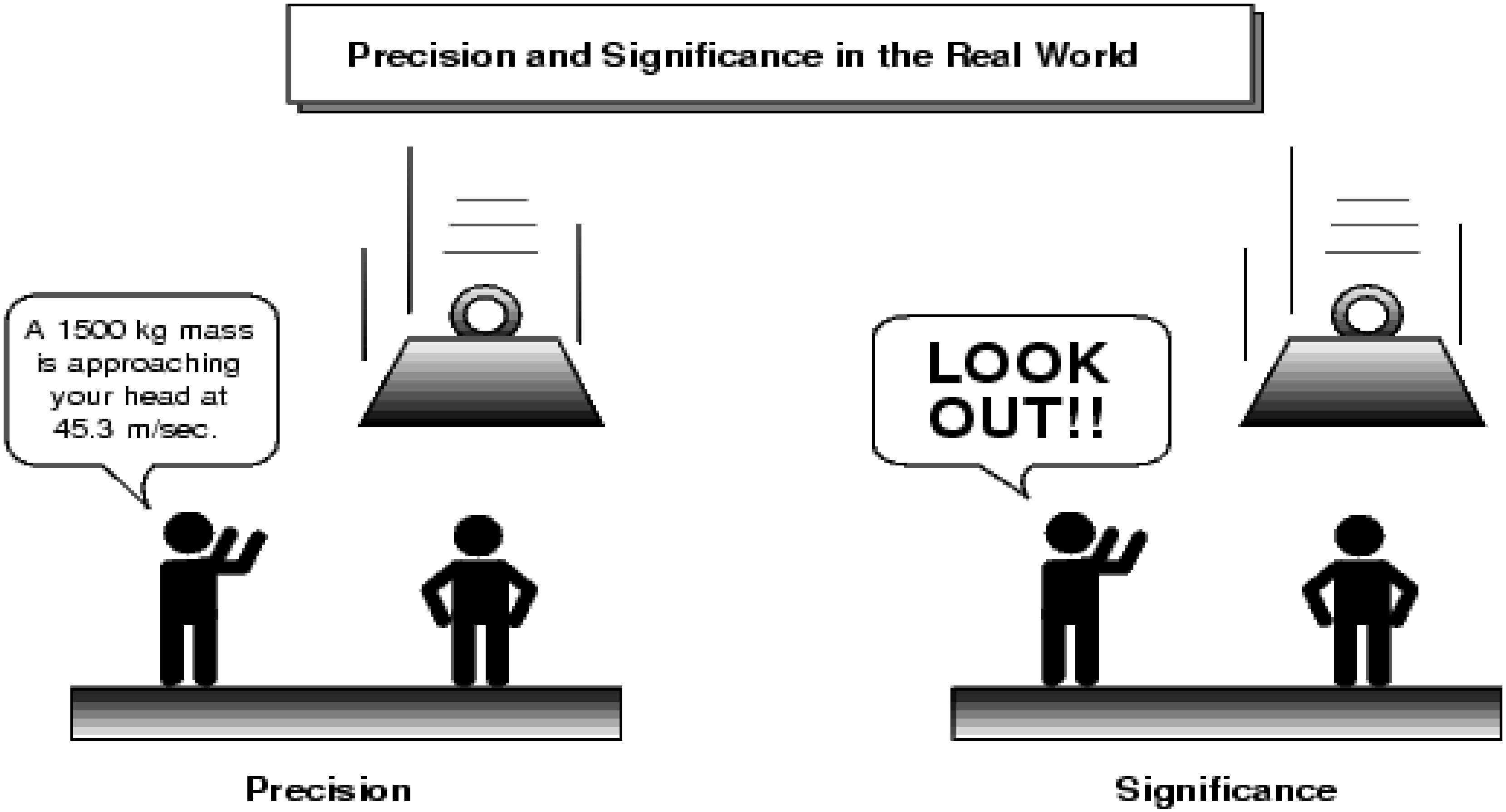
ARBITER:
Fuzzy Logic Controller

Elec 422 Group A:
Kevin Duh, Vernon Evans, Chris Flesher, David Suksumrit

AMD-Rice VLSI Design Contest
Dec 5, 2002

Precision vs. Significance

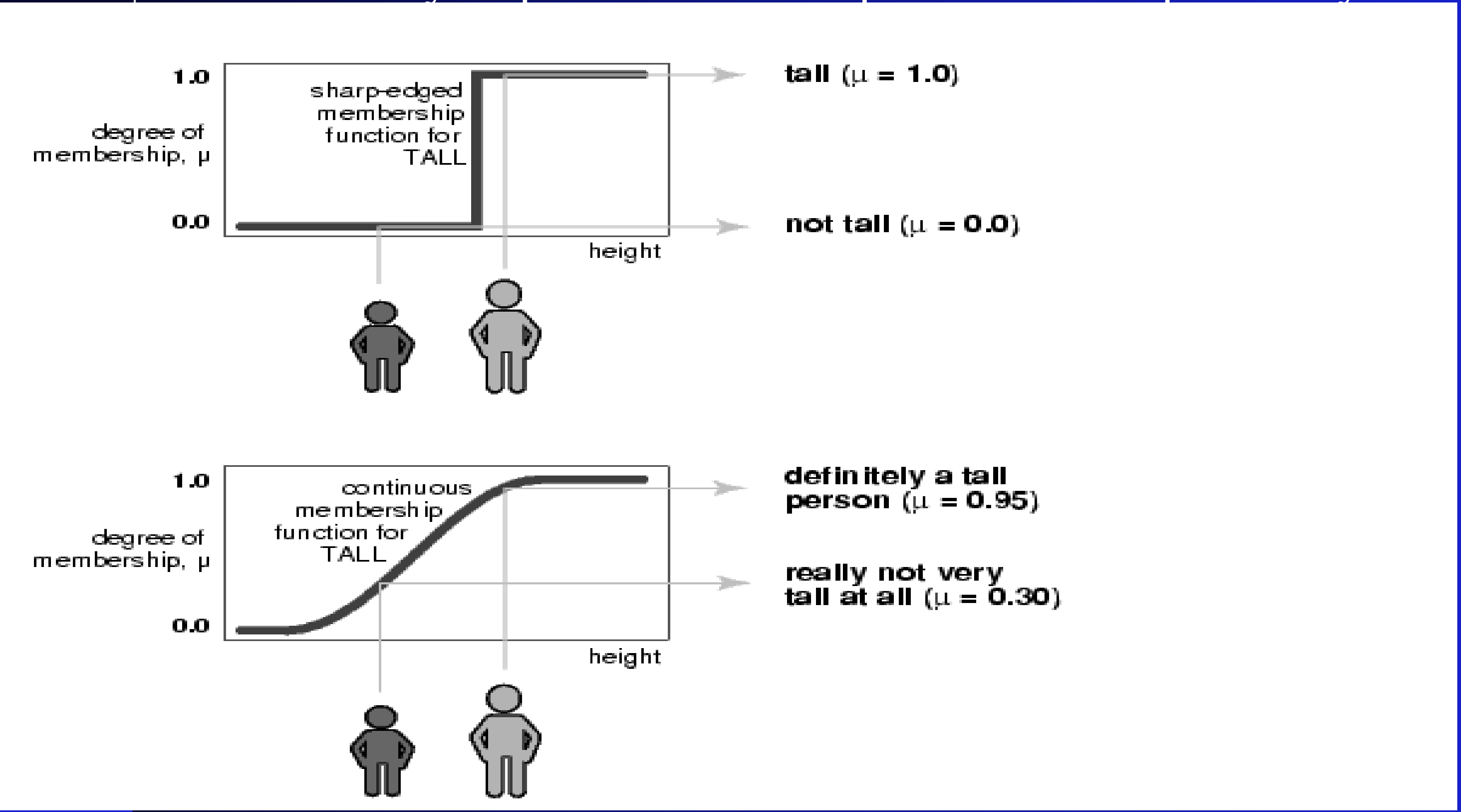


Conventional Logic

Fuzzy Logic / Human Logic

Basics: Membership Function

Words are inherently imprecise. This imprecision is captured by MF



Basics: IF-THEN Rules

- Example
  - “If falling object looks BIG, then yell ‘Watch Out!’ LOUDLY”
  - “If falling object looks SMALL, then tell him ‘Watch Out!’ CALMLY”
- BIG, LOUDLY, SMALL, CALMLY are defined by membership functions
- Enables fuzzy chip to make decision like humans!

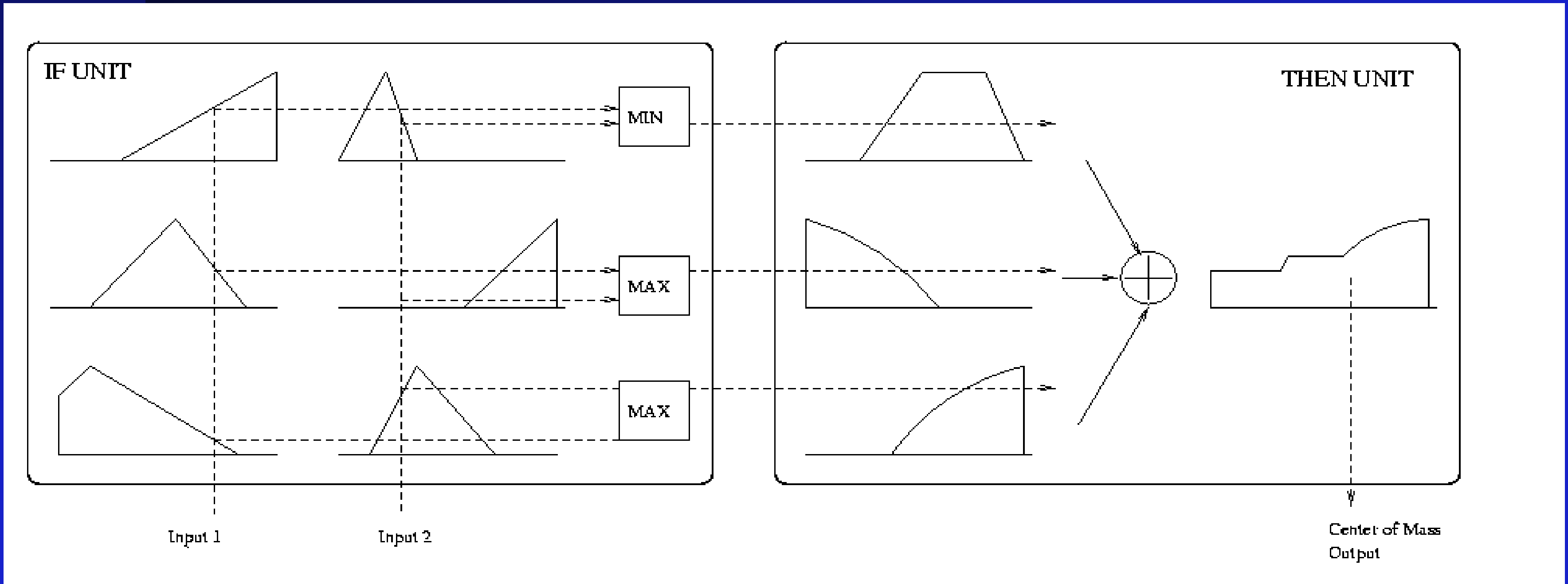
Why VLSI Implementation?

- Speed
  - Needed for Real-time applications
- Scalability
  - Parallel processing of fuzzy rules

- Our Design Goal:
  - General-Purpose VLSI Fuzzy Controller
  - Flexible & Fast (best of both worlds)
  - Analagous to DSP Chip

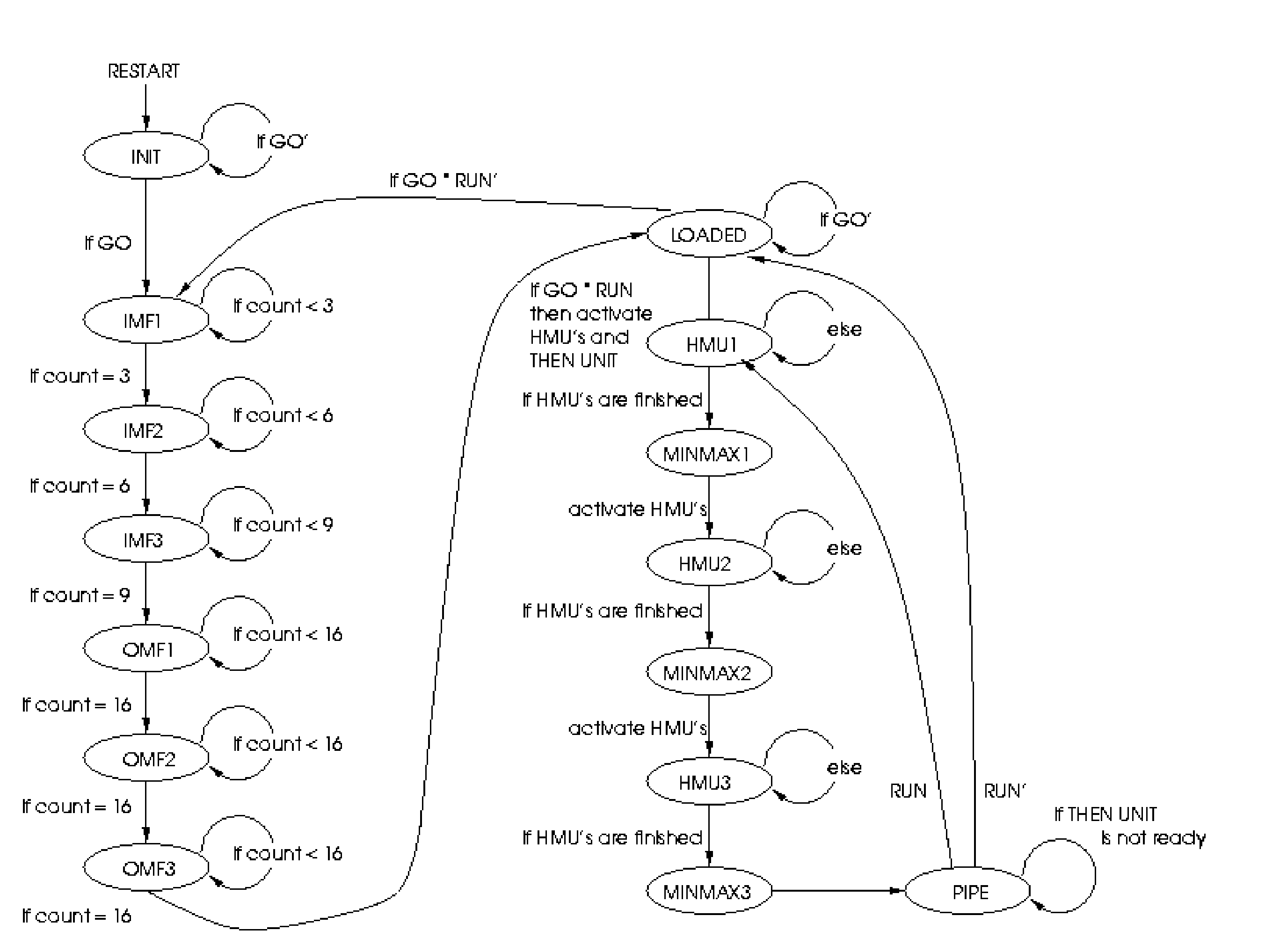
Functional Description

- 2 Input, 1 Output, 3 Fuzzy Rule
- Loadable Membership Functions (MF)
- 2-Stage Pipeline



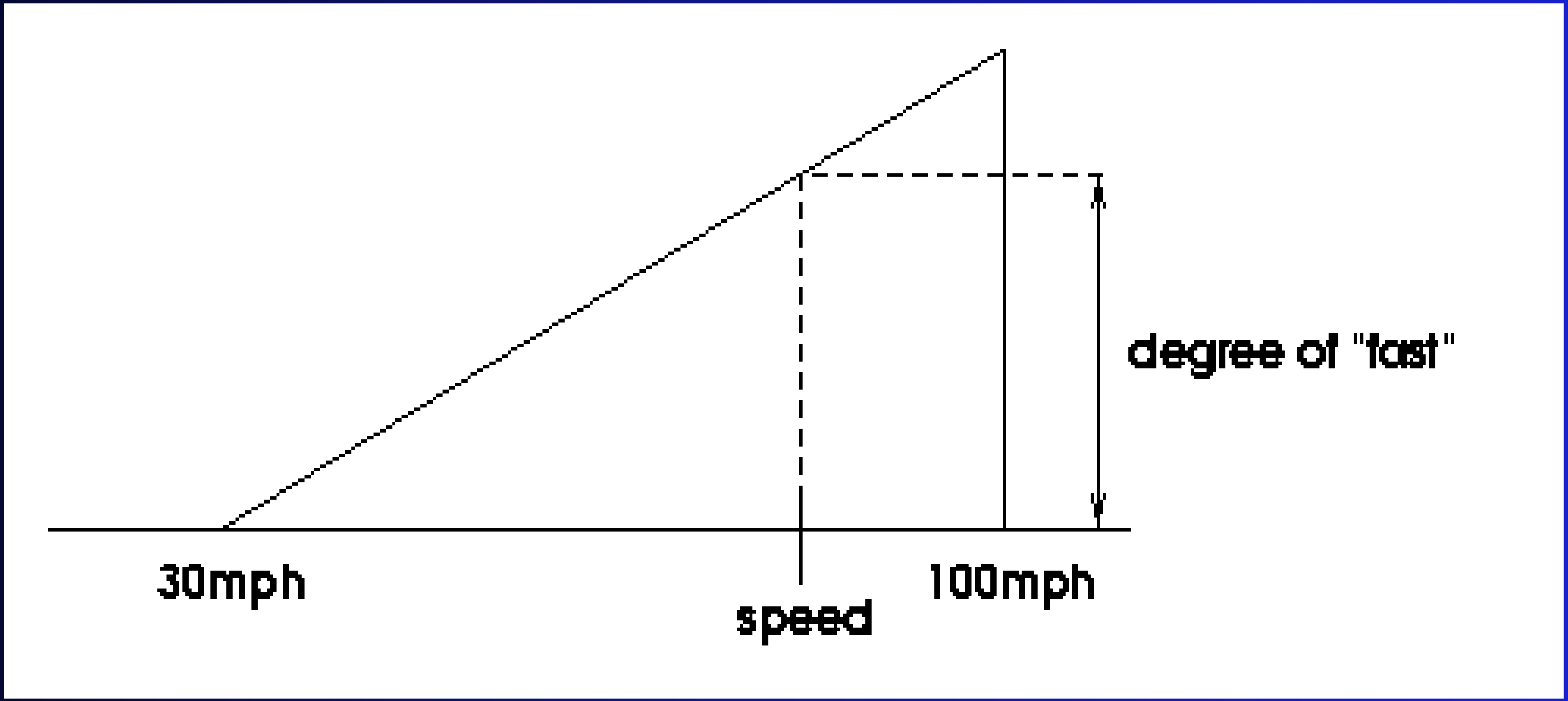
Main PLA

Controls loading, IF & THEN PLA's, pipeline



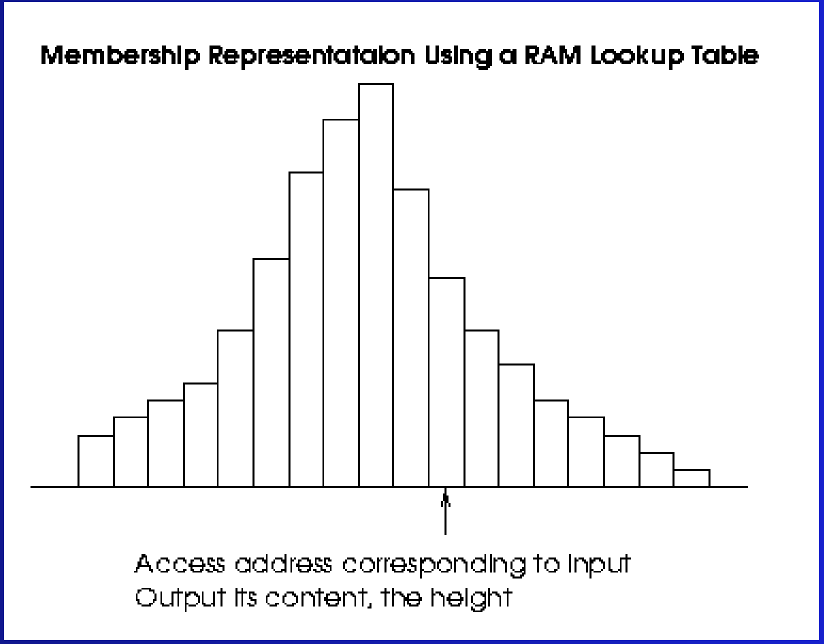
IF Unit: Purpose

- Evaluates height of intercept (“degree of truth”) for each IF statement



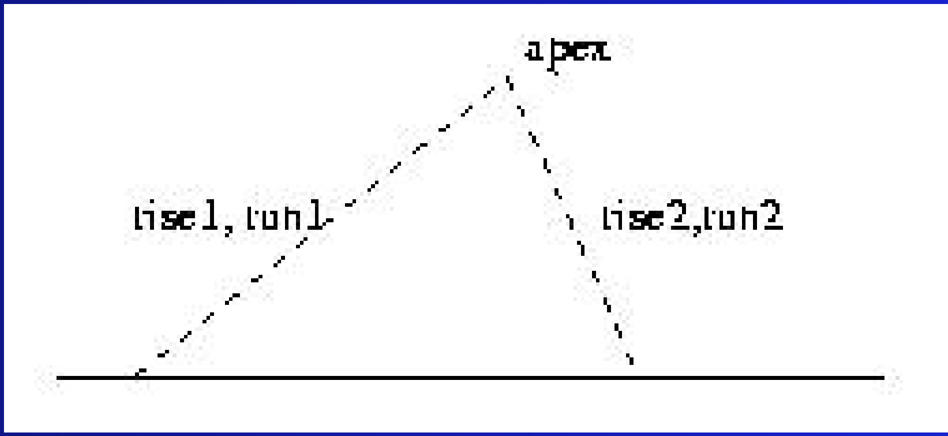
Challenge: How to represent membership function?

- Problem: Space vs. Flexibility
- Possible solution: Lookup Table
  - Pros:
    - Flexible expression of function
    - Fast access
  - Cons:
    - Takes too much space
    - Zero values waste space
    - Not challenging



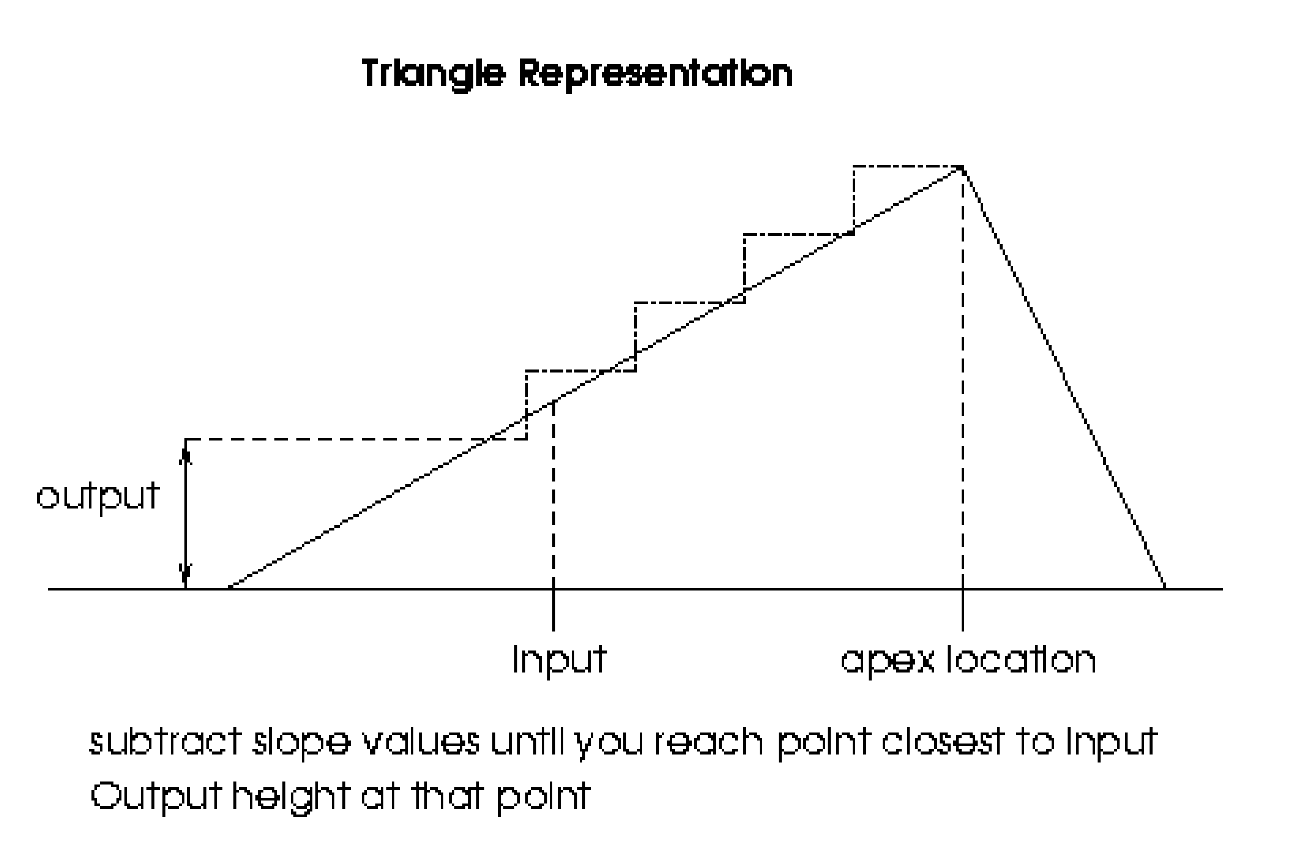
Solution: Point-Slope MF

- Our solution:
  - Represent geometric shape with slopes & point
  - Cons:
    - Math hardware required
    - Slower, variable-time calculation
  - Pros:
    - Much less space
    - Represent most MF shapes



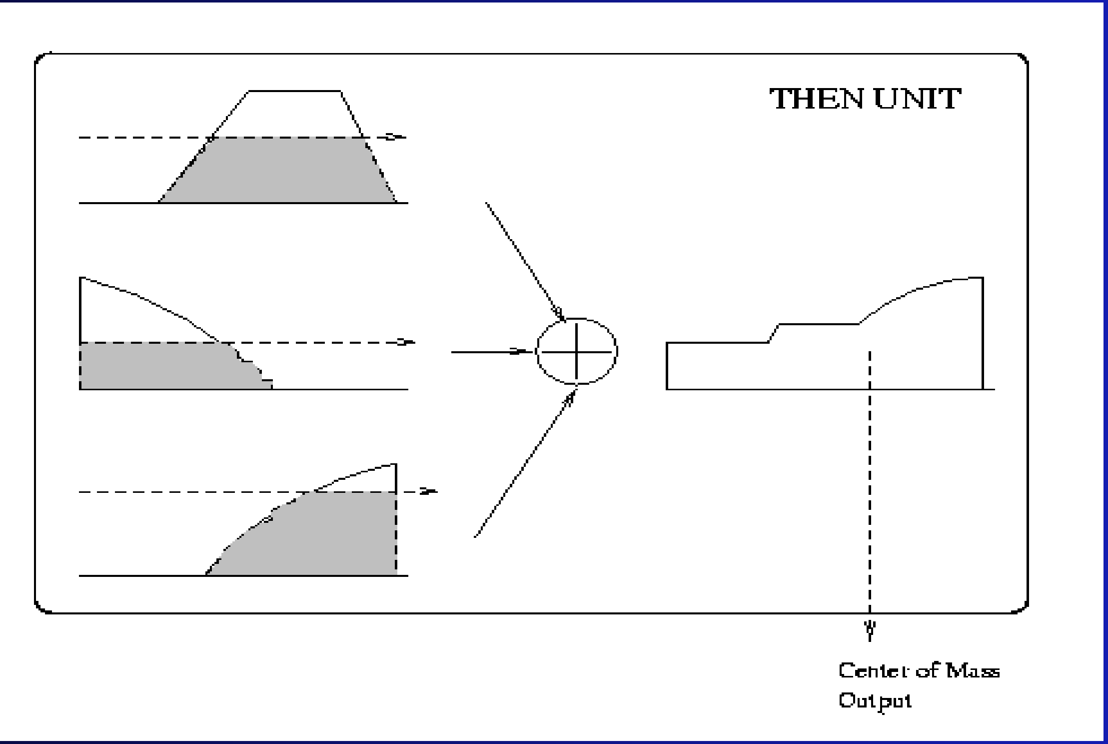
Algorithm for Finding Intercept

- Begin at apex, iterate subtractions until x
  Result is y (height/degree of truth)



THEN Unit: Purpose

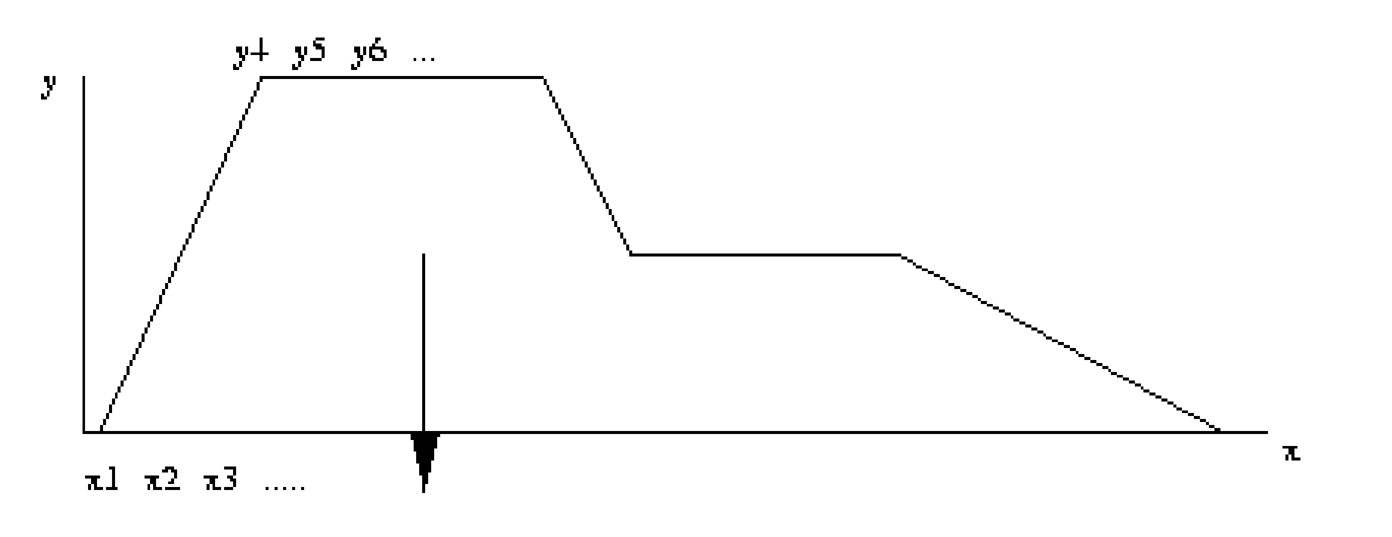
- Find the areas under the “cut” value for each THEN statement and Aggregate into a big MF
- Find Center of Mass for big MF -> final answer!



Challenge: Center of Mass

- Problem: complicated Center of Mass equation -

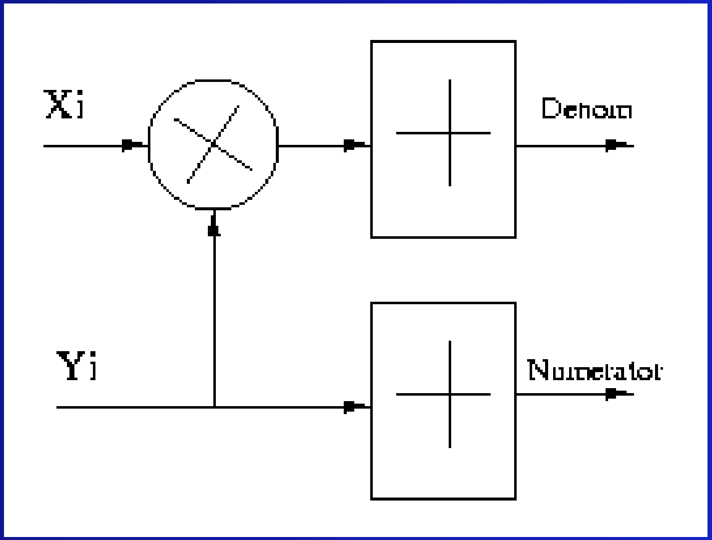
$$COM = \sum_{i=1}^{16} x_i y_i \Big/ \sum_{i=1}^{16} y_i$$



Possible Solution

- Possible Solution: Direct Implementation

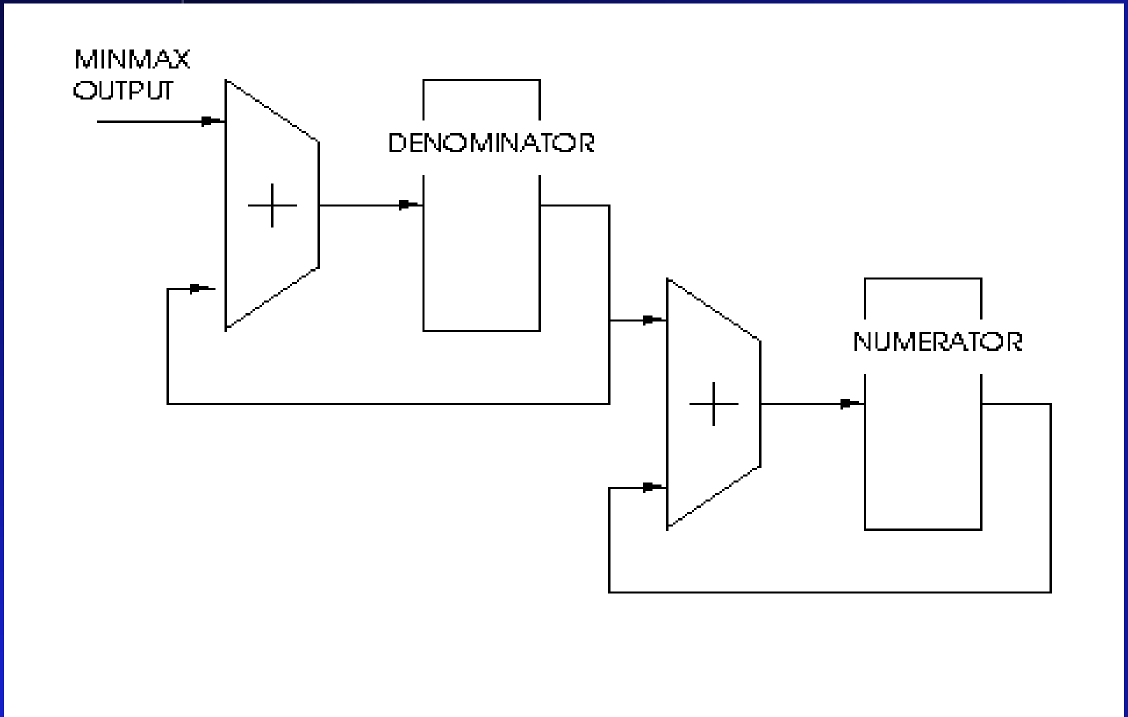
$$COM = \sum_{i=1}^{16} x_i y_i \Big/ \sum_{i=1}^{16} y_i$$



- Too much hardware!
- Too slow (multiplication)

Our solution: DoubleLoop Adder

- Calculate Num & Den simultaneously
- No multiplier needed



$$\frac{num}{den} = \sum_{i=1}^{16} x_i y_i \Big/ \sum_{i=1}^{16} y_i$$

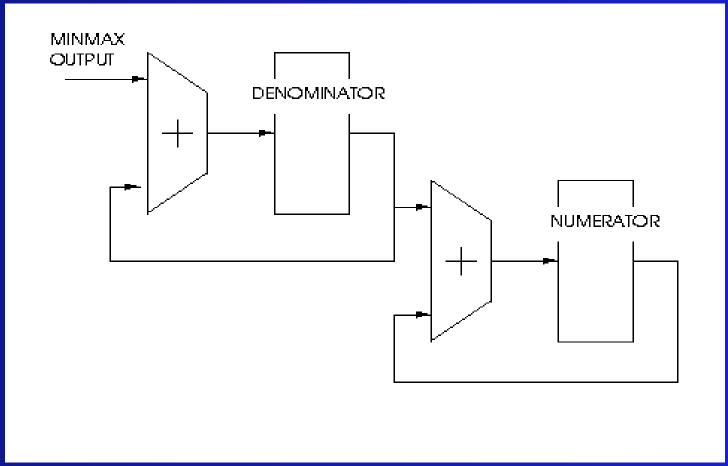
t	den	num
0	y_{16}	0
1	$y_{16}+y_{15}$	y_{16}
2	$y_{16}+y_{15}+y_{14}$	$y_{16}+(y_{16}+y_{15})$
3	…	$y_{16}+(y_{16}+y_{15})+(y_{16}+y_{15}+y_{14})$

More on DoubleLoop Adder

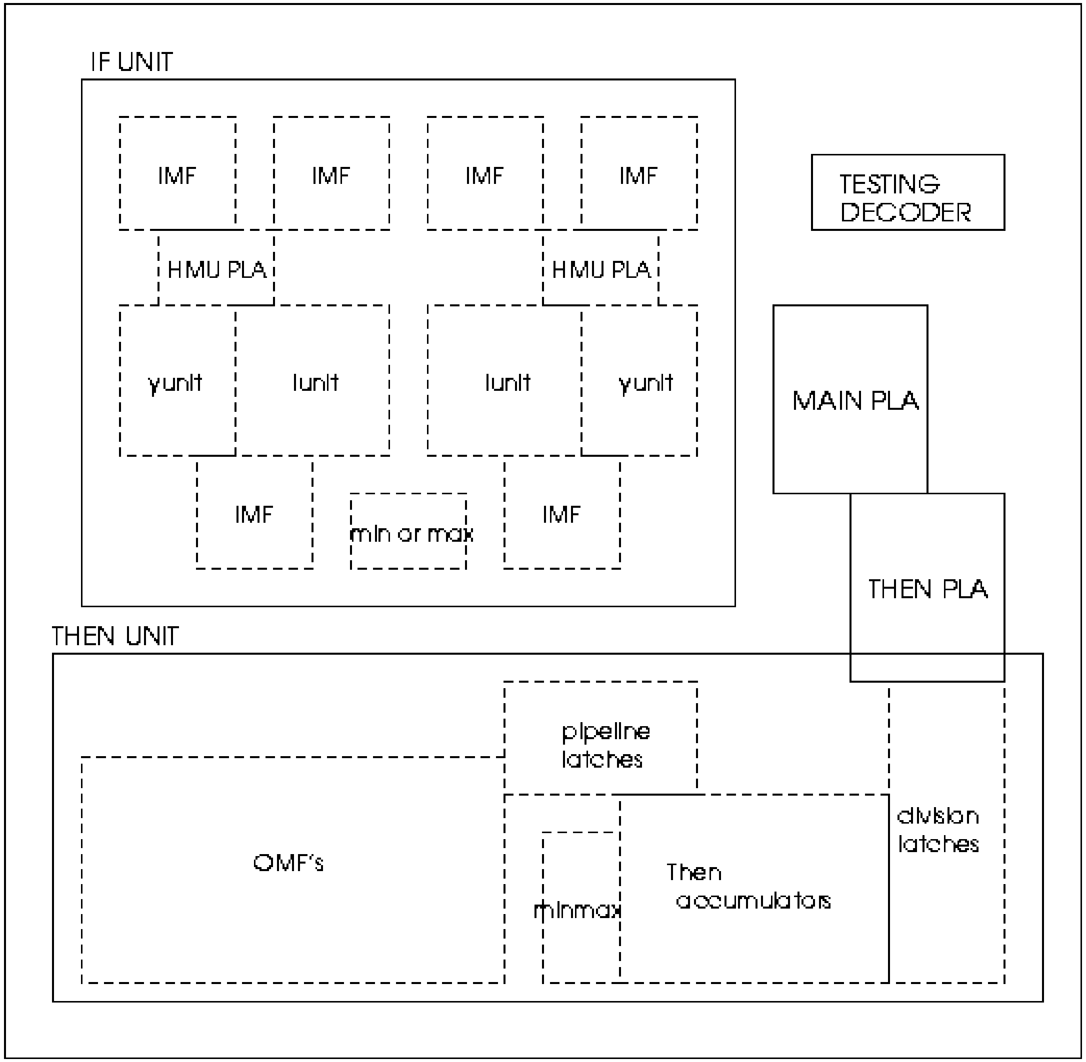
- Proof:

$$
\begin{aligned}
num &= 16y_{16} + 15y_{15} + 14y_{14} + \ldots + 1y_1 \\
&= y_{16} + (y_{16} + y_{15}) + (y_{16} + y_{15} + y_{14}) + \ldots + (y_{16} + y_{15} + y_{14} + \ldots y_1) \\
&= y_{16} + (y_{16} + y_{15}) + (y_{16} + y_{15} + y_{14}) + \ldots + den
\end{aligned}
$$

- Pros:
  - Fast: Only 17 cycles
  - Minimize hardware: no multipliers needed
- Division:
  - Re-use hardware!

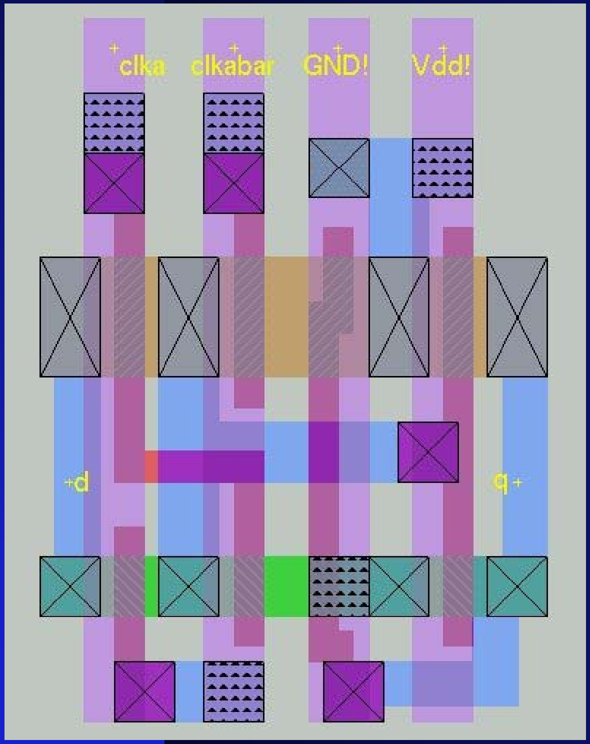


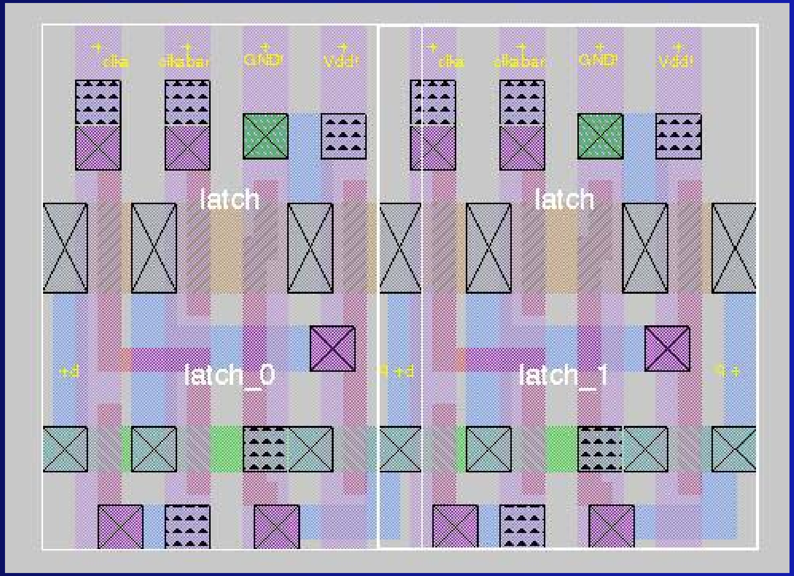
System Floorplan

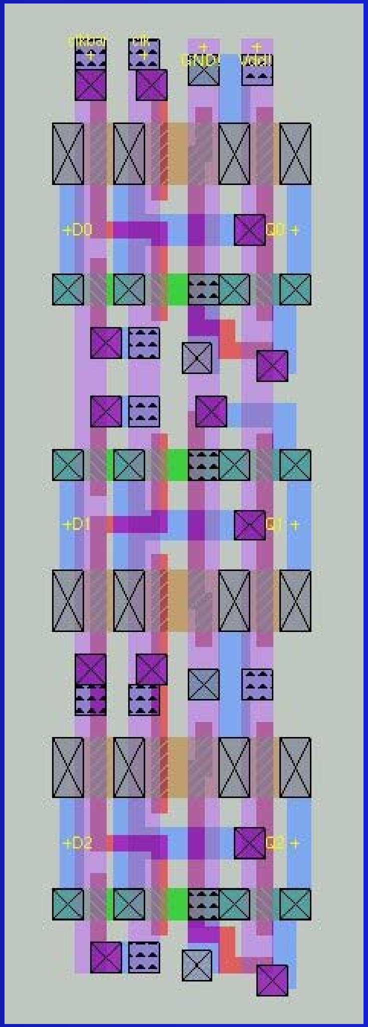


Standard Cell Layout: LATCH

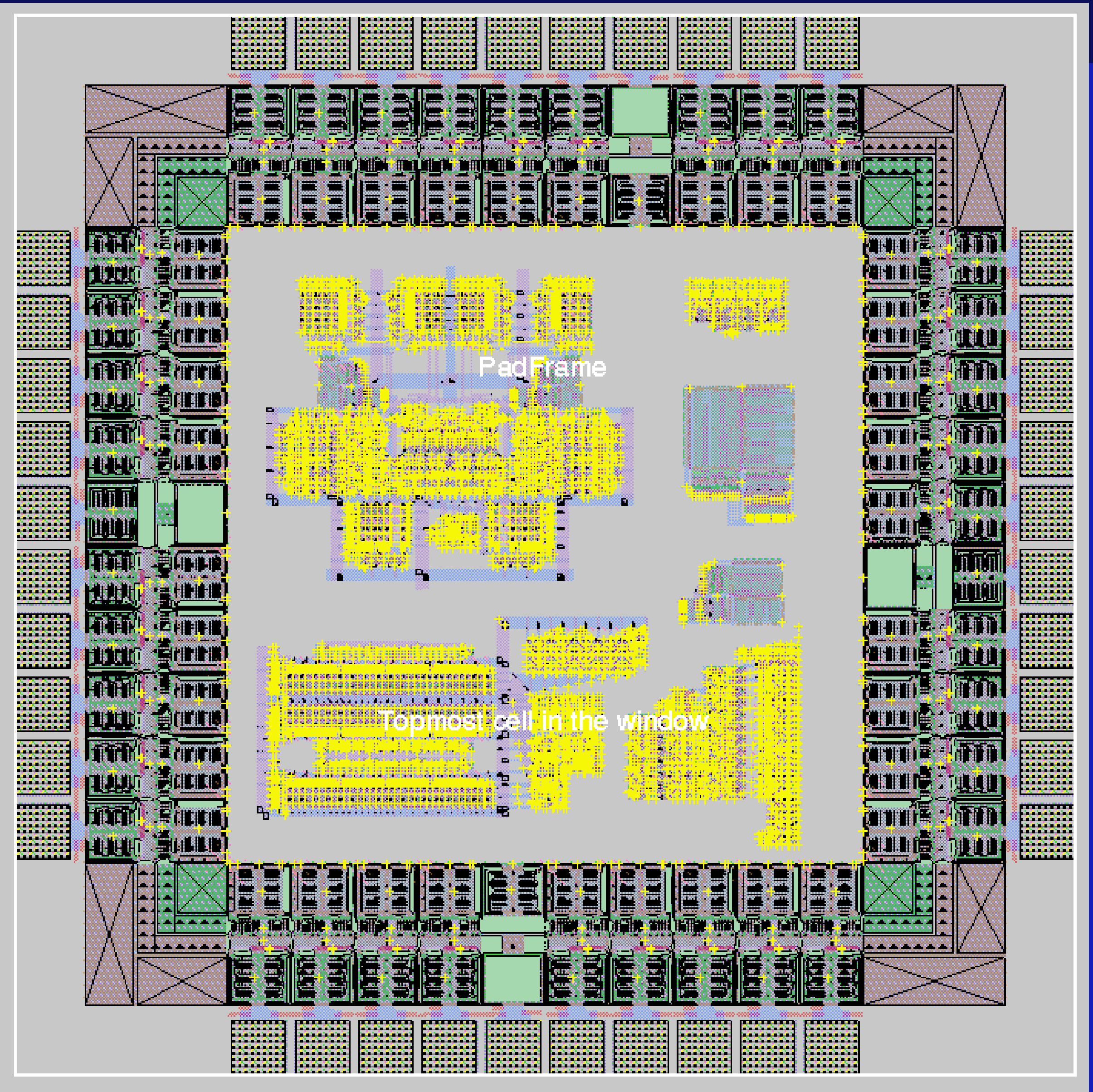
- Compact
- Scalable in any direction







Full Layout & Status

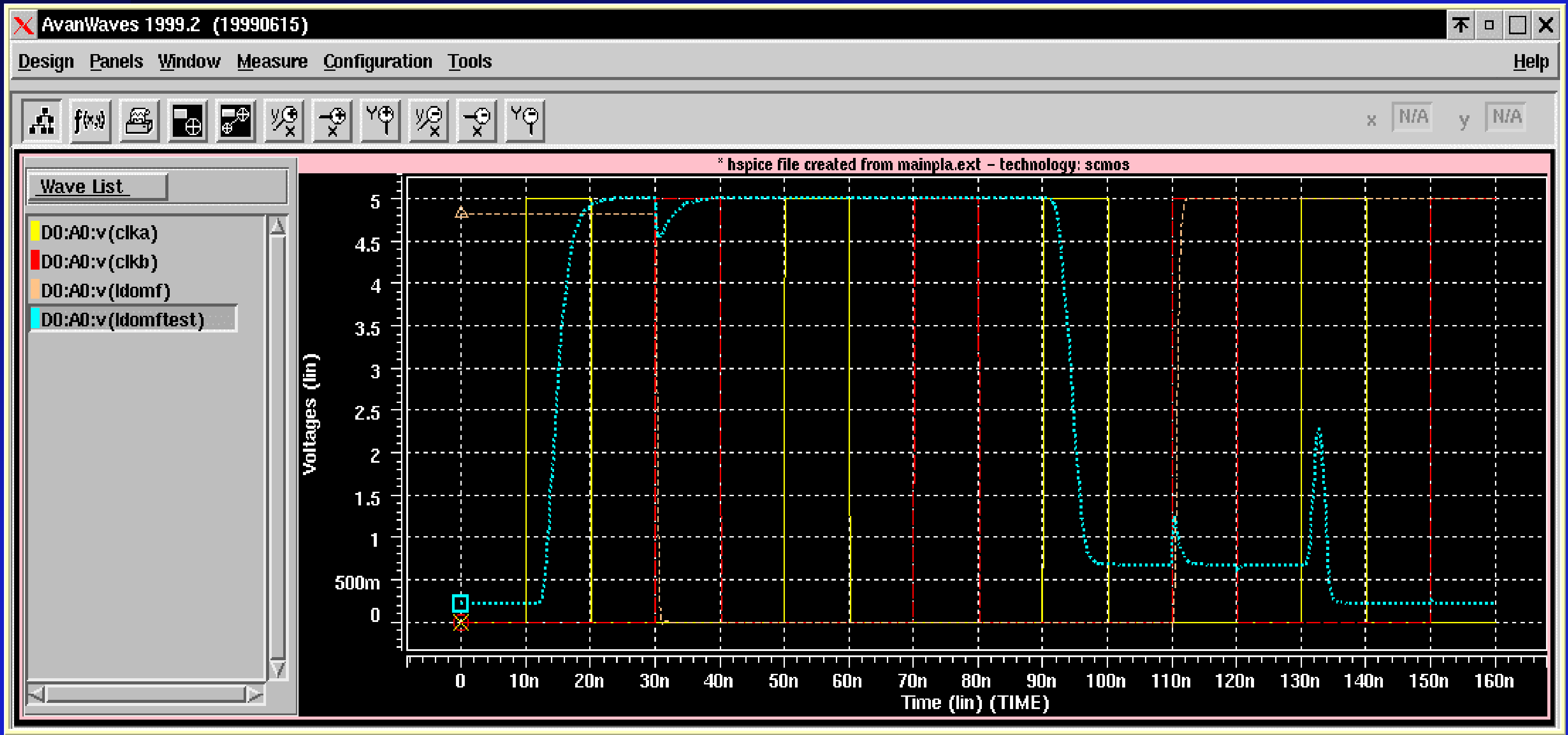


Design for Test

- Decoder
  - 15 mutually control signals
  - Watch 105 signals total, 7 at a time
  - Asynchronous
- Matlab verification
  - Simulate test vector solutions

Timing Analysis

- Main PLA: 15.72ns -> clock freq: 63MHz
- 11 bit Carry-Select Adder: 14.74ns



Conclusion

- We have:
  - Demonstrated a fuzzy controller that’s both FAST and FLEXIBLE
- Applications:
  - Expert system:
    - FuzzyMD
    - Data Mining
  - Real-time:
    - robot control
    - image processing
    - environment control