

Asynchronous Computing

3. Mutual Exclusion: Who's first?

Marly Roncken and Ivan Sutherland

Asynchronous Research Center (ARC)
Maseeh College of Engineering and Computer Science
Portland State University
July 2018

2018:is29v7 slide 1

Outline

- Quantize a continuous variable
- Time is our continuous variable
- Which (of two) happened first?
- Exactly the same time?
- Mutual Exclusion circuit (Arbiter) will decide, but may take time

July, 2018

Slide 2

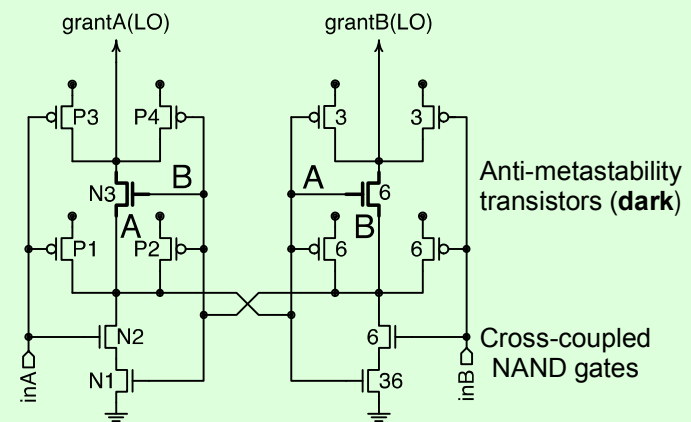
Mutual exclusion

- Two events at “same” time
 - > which choice doesn't matter
 - > but choice must be clean
- Flip-flop can hang metastable
 - > exit is Poisson distributed
 - > may take a long time, but rarely will
- Asynchronous system can wait

July, 2018

Slide 3

Mutual exclusion (Seitz)



July, 2018

Slide 4

Continental divide



July, 2018

Slide 5

Continental divide



July, 2018

Slide 6

Continental divide



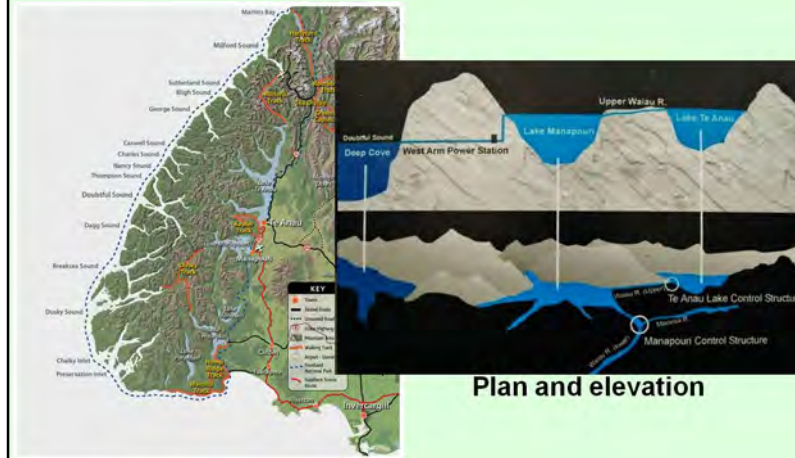
View east

View west

July, 2018

Slide 7

Moving the Continental divide



Plan and elevation

July, 2018

Slide 8

Manapouri powerplant, NZ



July, 2018

Slide 9

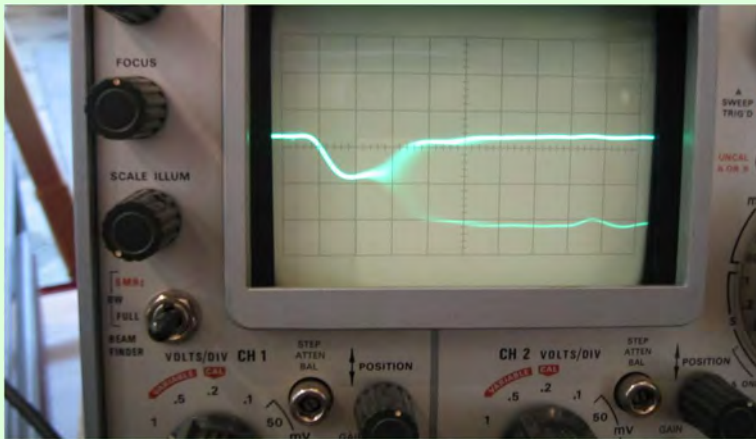
Metastability demonstration



July, 2018

Slide 10

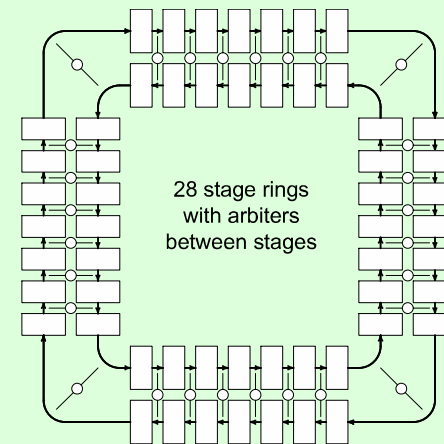
Scope trace



July, 2018

Slide 11

Zeke to test arbiters (1998)

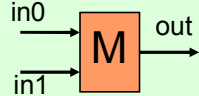


July, 2018

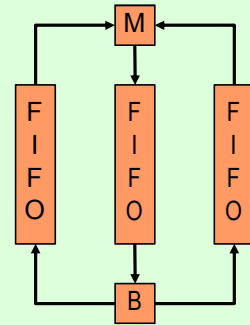
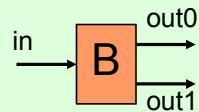
Slide 12

Infinity test (2008)

Demand Merge



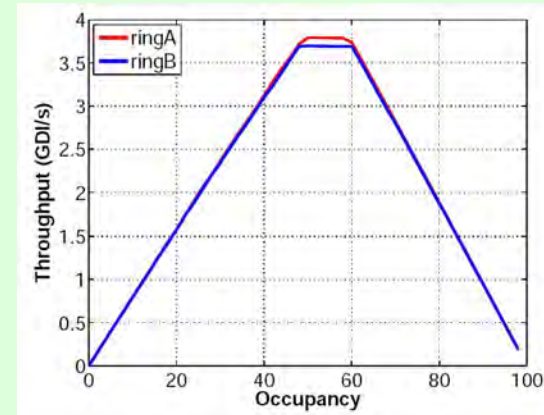
Data-directed Branch



July, 2018

Slide 13

Infinity: Throughput vs Occupancy



July, 2018

Slide 14

Stopping self-timed systems

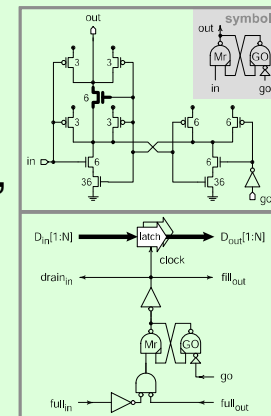
- Clean stop requires arbitration
 - > Stop in mid action OR
 - > Finish the action, then stop
- Without arbitration, runt pulses
 - > Give chance of data error or
 - > Loss of whole data item

July, 2018

Slide 15

MrGO

- Half an arbiter
- To stop cleanly
- A “proper stopper”
- *Shall I stop now or complete this action?*



July, 2018

Slide 16