

UC Berkeley Computer Science

Subject: An Outline for the Review of Infinity
Date: May 13, 2007
From: Ivan Sutherland
UCIES #2007-is46

References:

UCIES #2007-is46 Infinity: A Proposed Test Chip, Ivan Sutherland, 18 April 2007
A more detailed version of this memo, for internal Sun use only, is called SML# 2007-0224. I have expunged a few references to internal Sun documents from this version.

PURPOSE

This memo offers a schedule for the May 15 design review of Infinity.

I plan four sections to the review:

OBJECTIVE — CIRCUITS — TO DO — WORRIES

A separate page of this memo outlines each section.

This document is a product of a collaboration between Sun Microsystems and the University of California at Berkeley.. The ideas contained herein are freely available for any academic purpose.

SECTION 1 – Why and What – 20 minutes

PURPOSE: INFINITY IS A WARM UP FOR A FLEET CHIP

- 1) Test the 6/4 GasP family
- 2) Learn TSMC 90 design rules
- 3) Debug assembly software
- 4) Functional test – speed and power data

THE PROPOSED ARTIFACT

UCIES 46: Infinity: A Proposed Test Chip

- 1) ~150 stages of half a dozen types with 52 data bits
- 2) Tests branch and merge stages
- 3) Builds confidence in 6/4 GasP, the design style, and tools
- 4) Interesting behavior
- 5) About 1 mm by 1 mm core size

DESIGN STYLE

- 1) Stage types hand crafted
- 2) Java code for placing and wiring stages
- 3) Stages designed to permit automatic wiring

SECTION 2 – The circuits – 40 minutes

THE LATCHES

- 1) Pass gate input and feedback
- 2) Active HI control for scan chain
Active LO control for data latches
- 3) Versions with one and two inputs
- 4) 136 lambda bit pitch – also fits the sensing scan chain.

THE ARBITER

- 1) Combines three-input NAND with arbitration in a tall stack
- 2) Used in Merge and Cross stages and in the proper stopper.

SIX – FOUR GasP

A tutorial with pictures illustrates this part

There are eight stage types:

Plain, Strong, Switch, Branch, Merge, Cross, Fill, Drain

Switch and Branch have two outputs. Fill and Drain form the “proper stopper.” Merge, Cross, Fill and Drain all have arbiters. Merge is by demand.

INFINITY INTENDED TO TEST BRANCH AND MERGE

STATUS

Stage types are DRC and NCC clean. Layout code places stages and finds the routing channels.

SECTION 3 – TO DO – 20 minutes

CIRCUITS

Finish the counter.

SIMULATION

Very a little SPICE simulation has been done.

latches arbiter counter ring of GasP

No logic simulation has been done.

Test code remains to write.

LAYOUT

Finish the channel router for core assembly.

Metal 4 and above are for power wiring – automatic stuff from Russell?

Integration with the scan chain controller.

Pad ring.

Final assembly of core in pad ring.

TESTING

PC mounting board

Test software

SECTION 4 – My worries – 20 minutes

CIRCUITS

Keepers in the arbiter come primarily from the predecessor. In the case of sink limited operation this may permit improper behavior.

There may be a logic flaw – e.g. use of the scan chain control wires.

Will different wire length for latches and state wires cause problems.

TIMING

Address bit [1] controls the branch direction

The “T” bit selects whether or not to latch 37 data bits.

ARCHITECTURE

What can we learn if it's only partly functional?

What should we include to make testing easier?

Should we plan another experiment with a very simple ring?

Should we try to test the Cross stage as well? How?

OMISSIONS

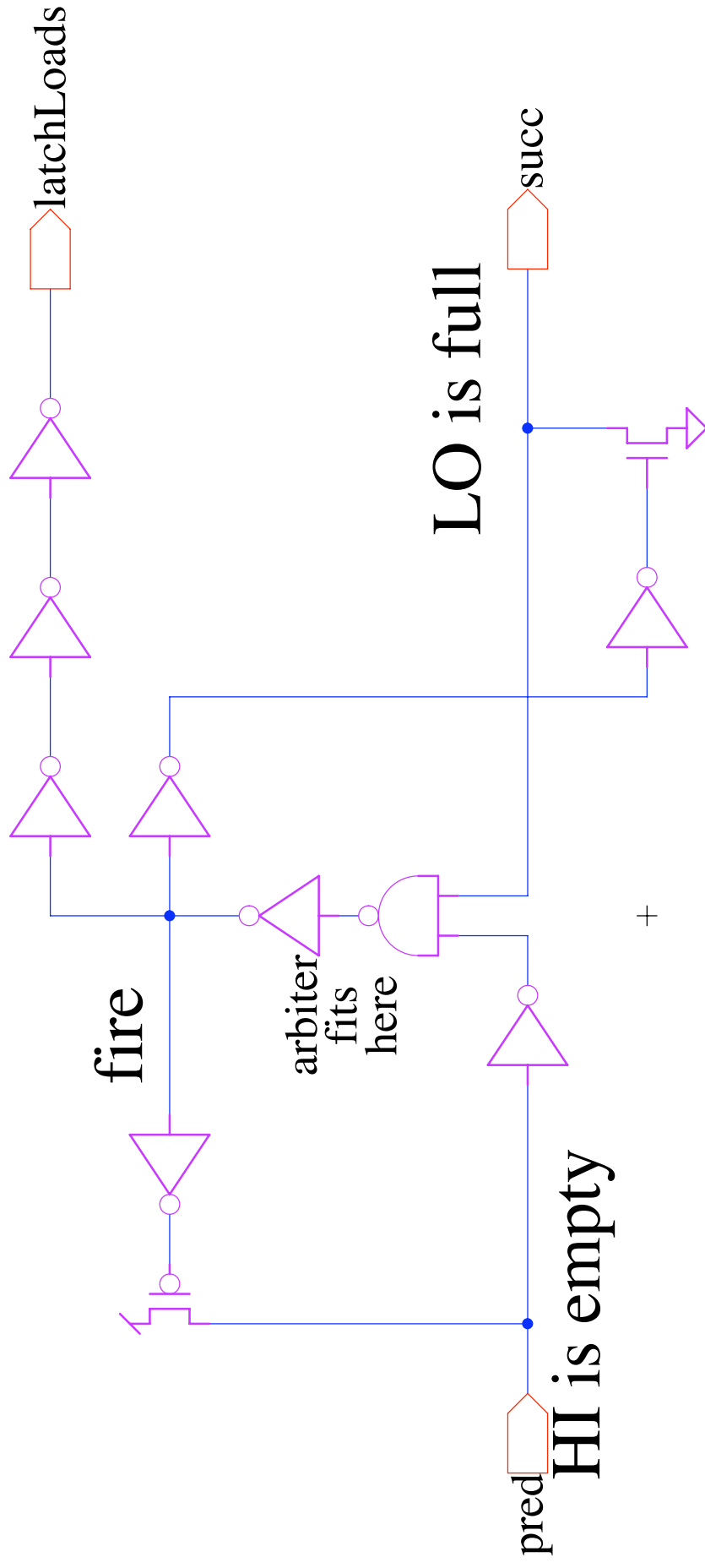
What should we put in to learn more?

Samplers? Voltage probes? Frequency output signals?

Data path sensing – e.g. output from some data bit as it passes by.

basicGasP-A

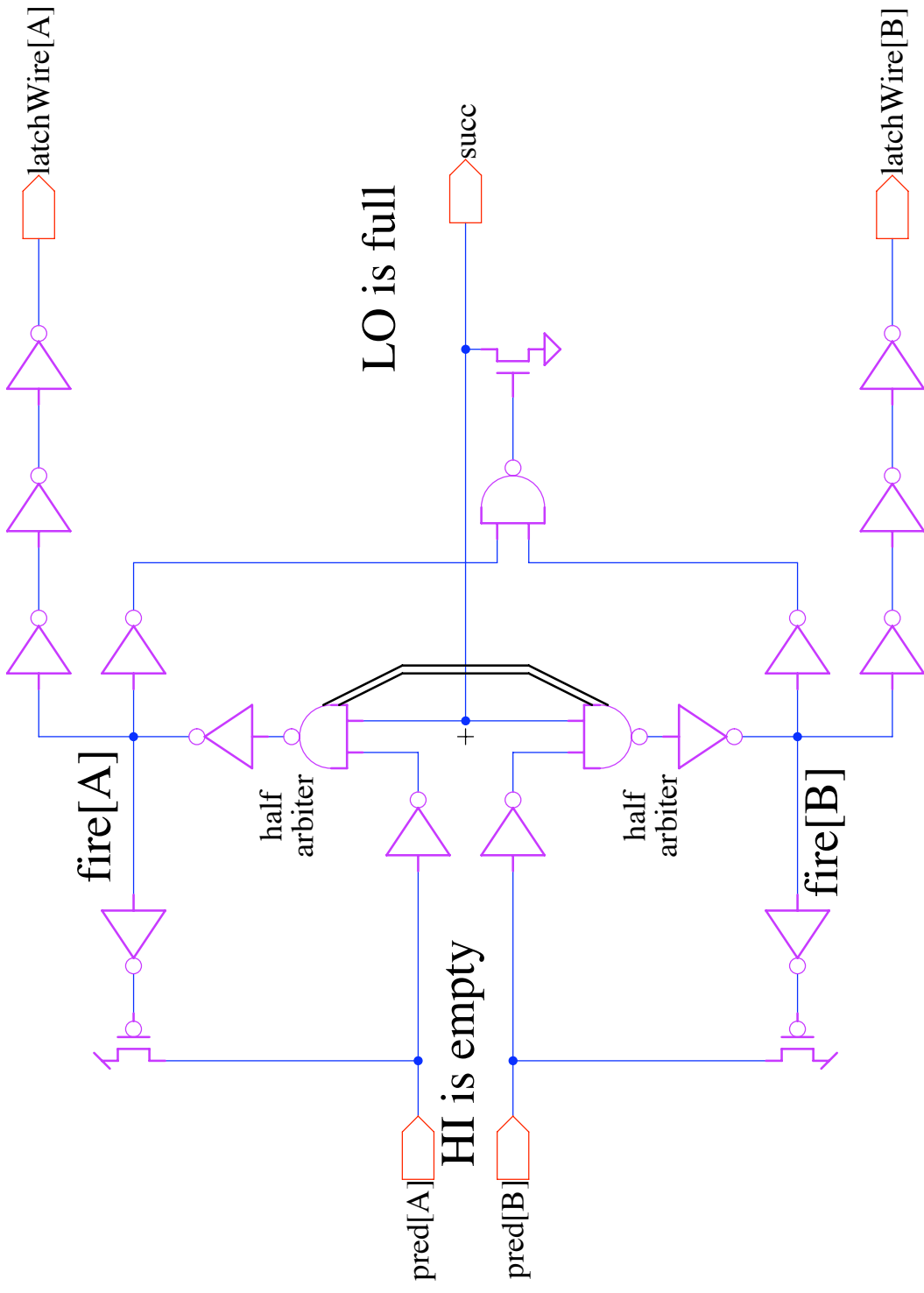
ies 6 May 2007



basicGasP-B

ies 7 May 2007

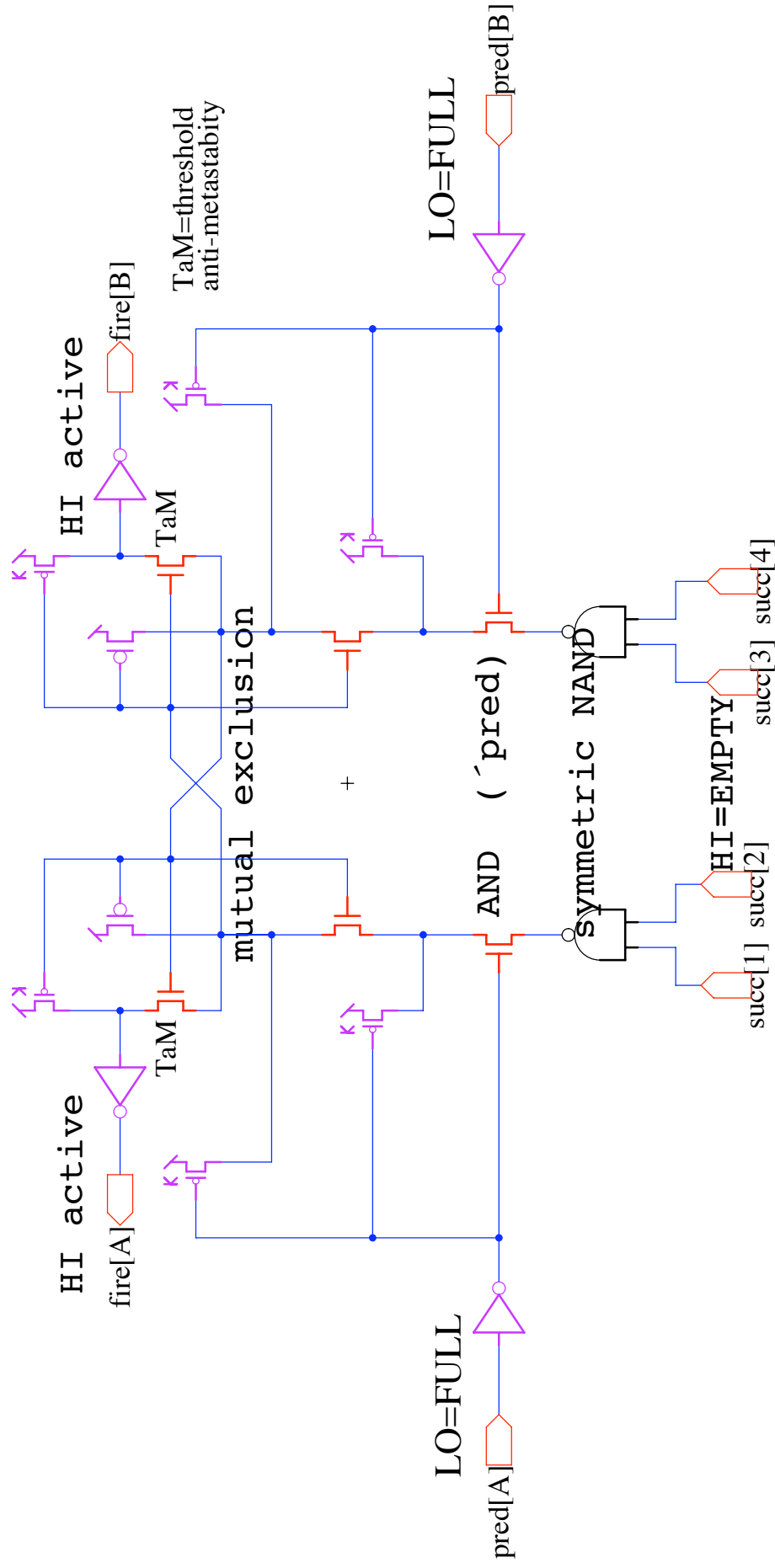
arbiter for two predecessors



basicGasP-C

ies 10 May 2007

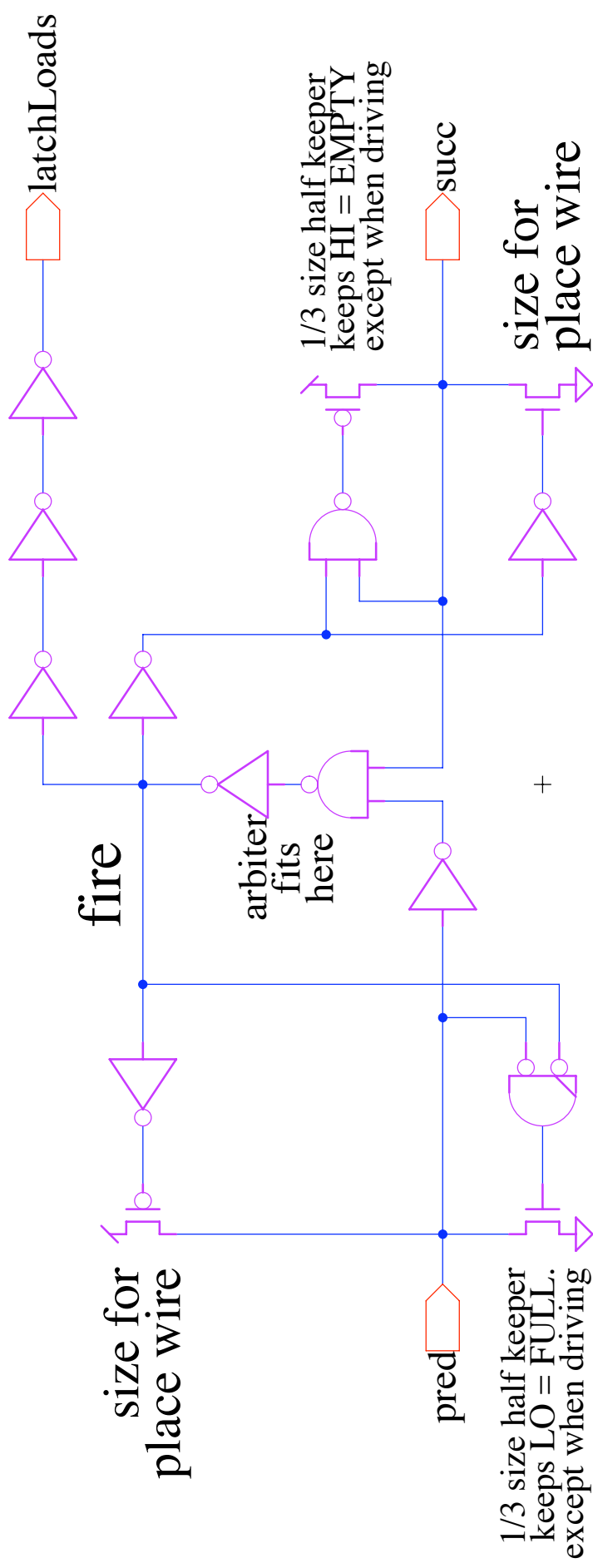
the arbiter



basicGasP-D

ies 6 May 2007

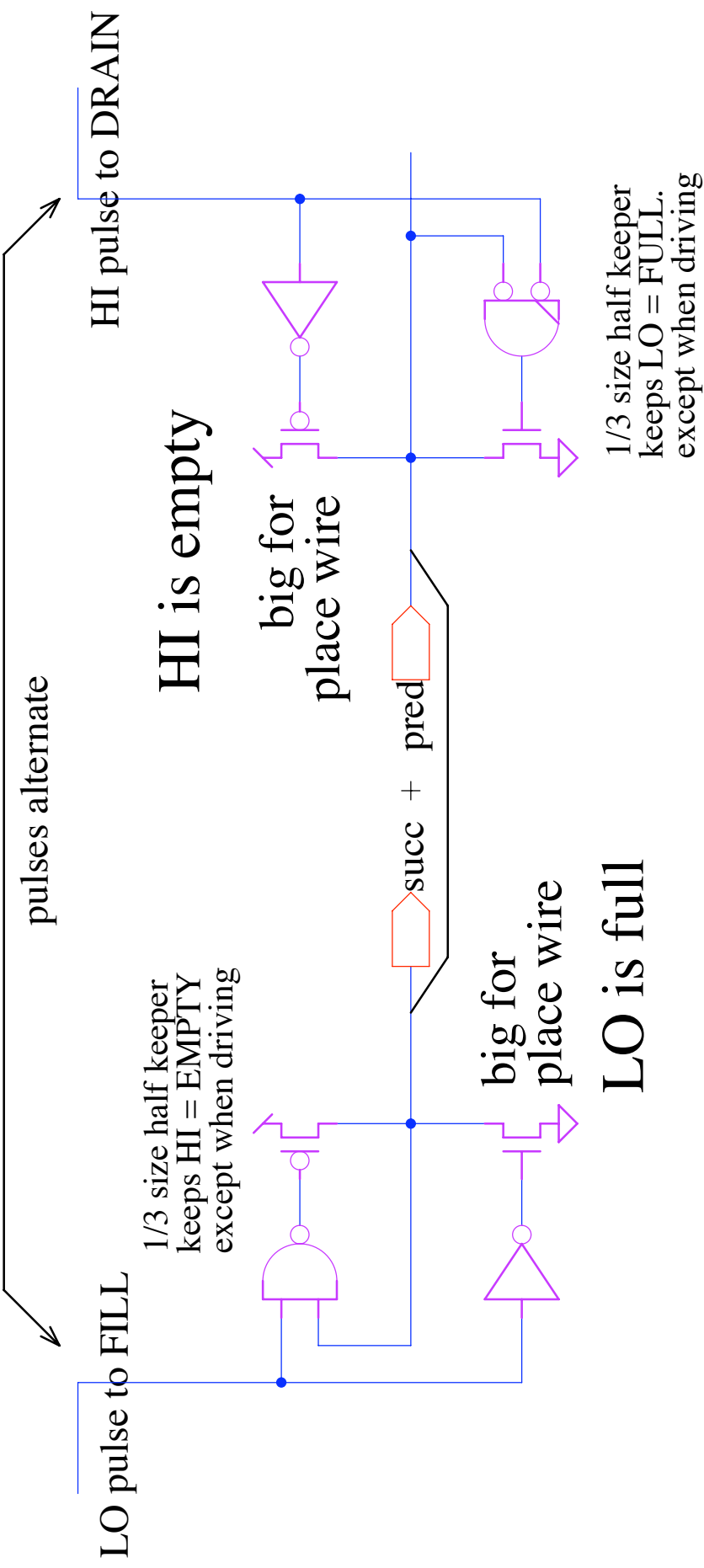
with half keepers shown



basicGasP-E

ies 6 May 2007

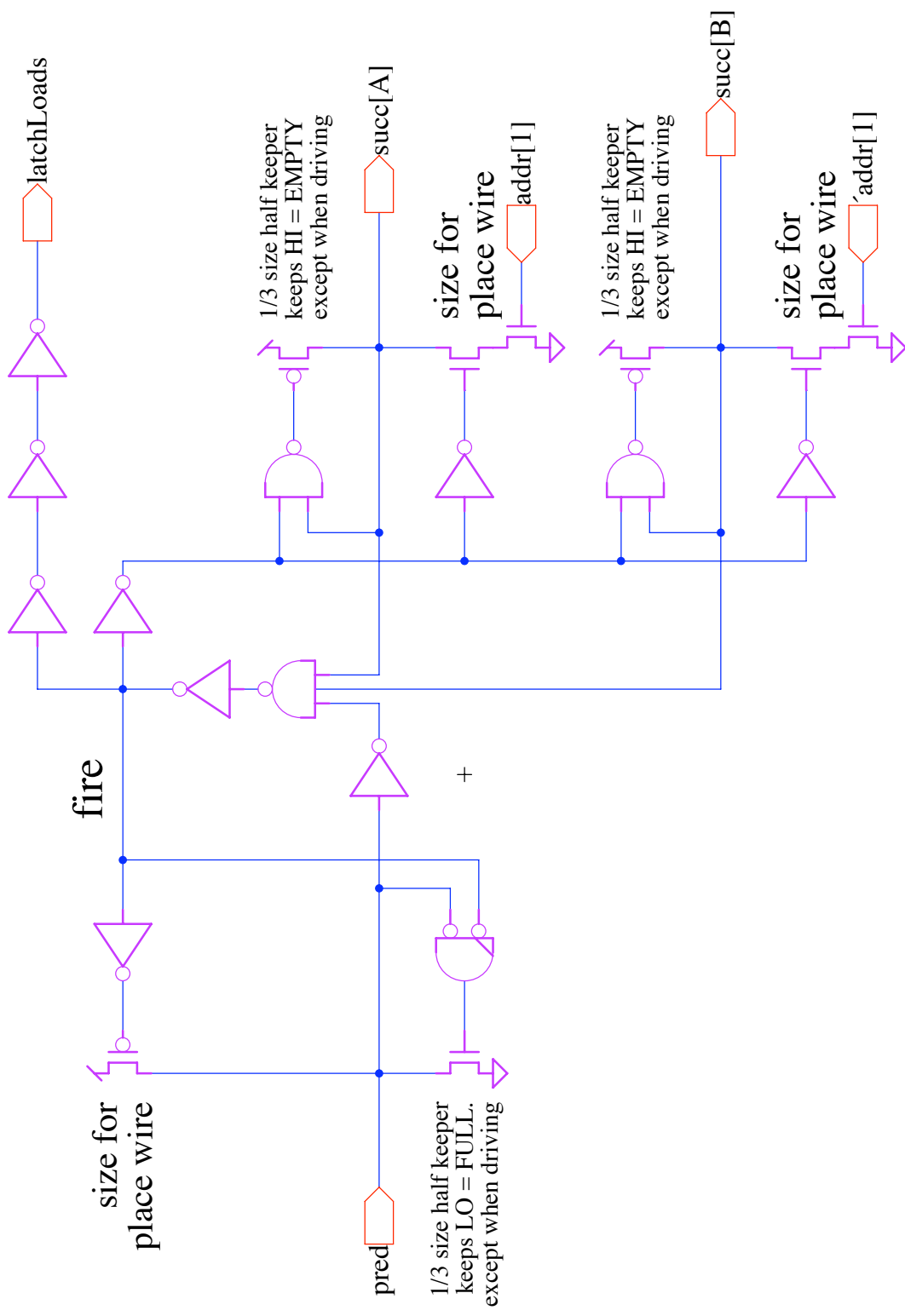
half keepers for two stages



basicGasP-F

ies 6 May 2007

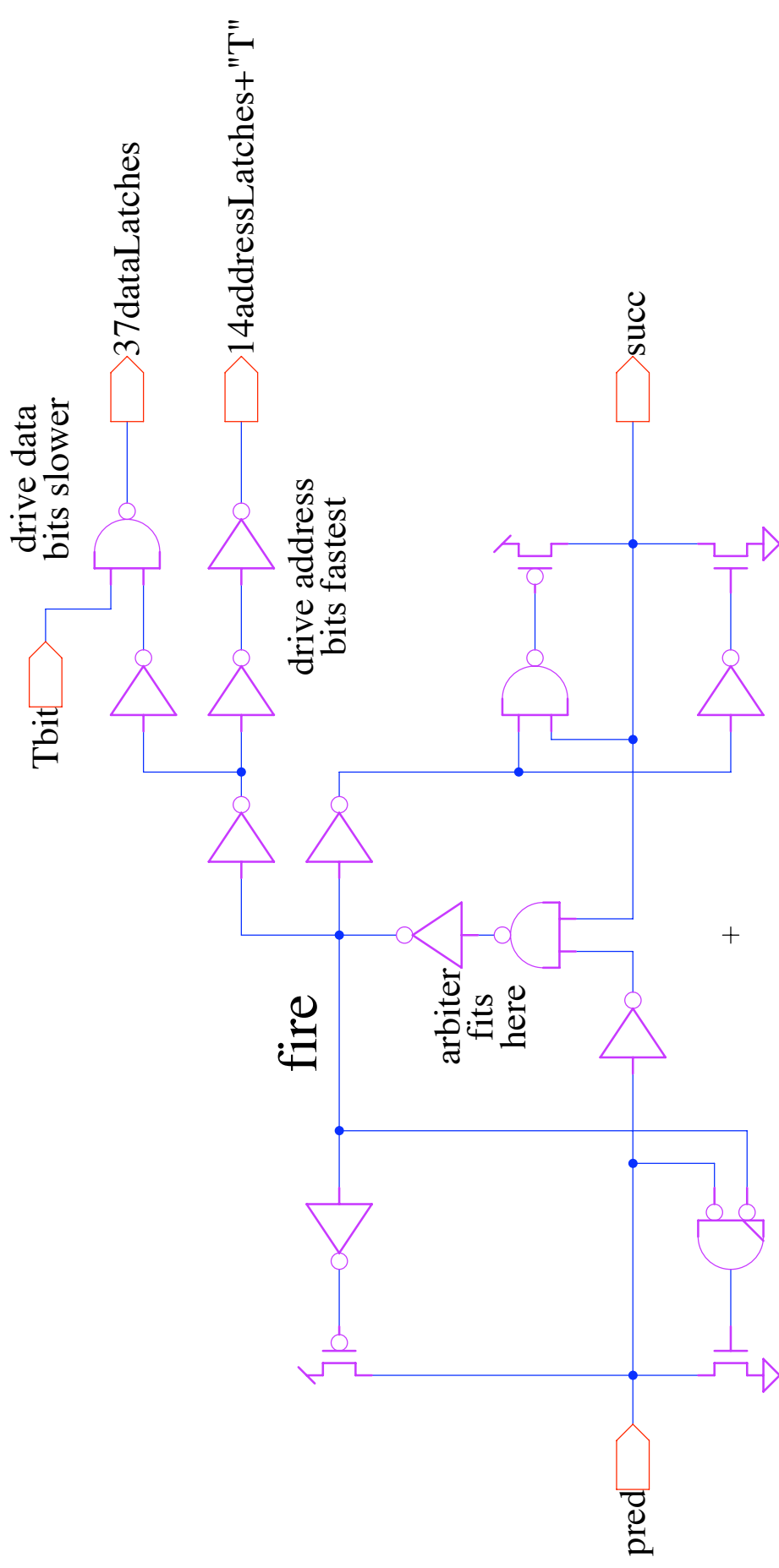
selection for two outputs



basicGasP-G

ies 6 May 2007

separate latch drivers



basicGasP-H

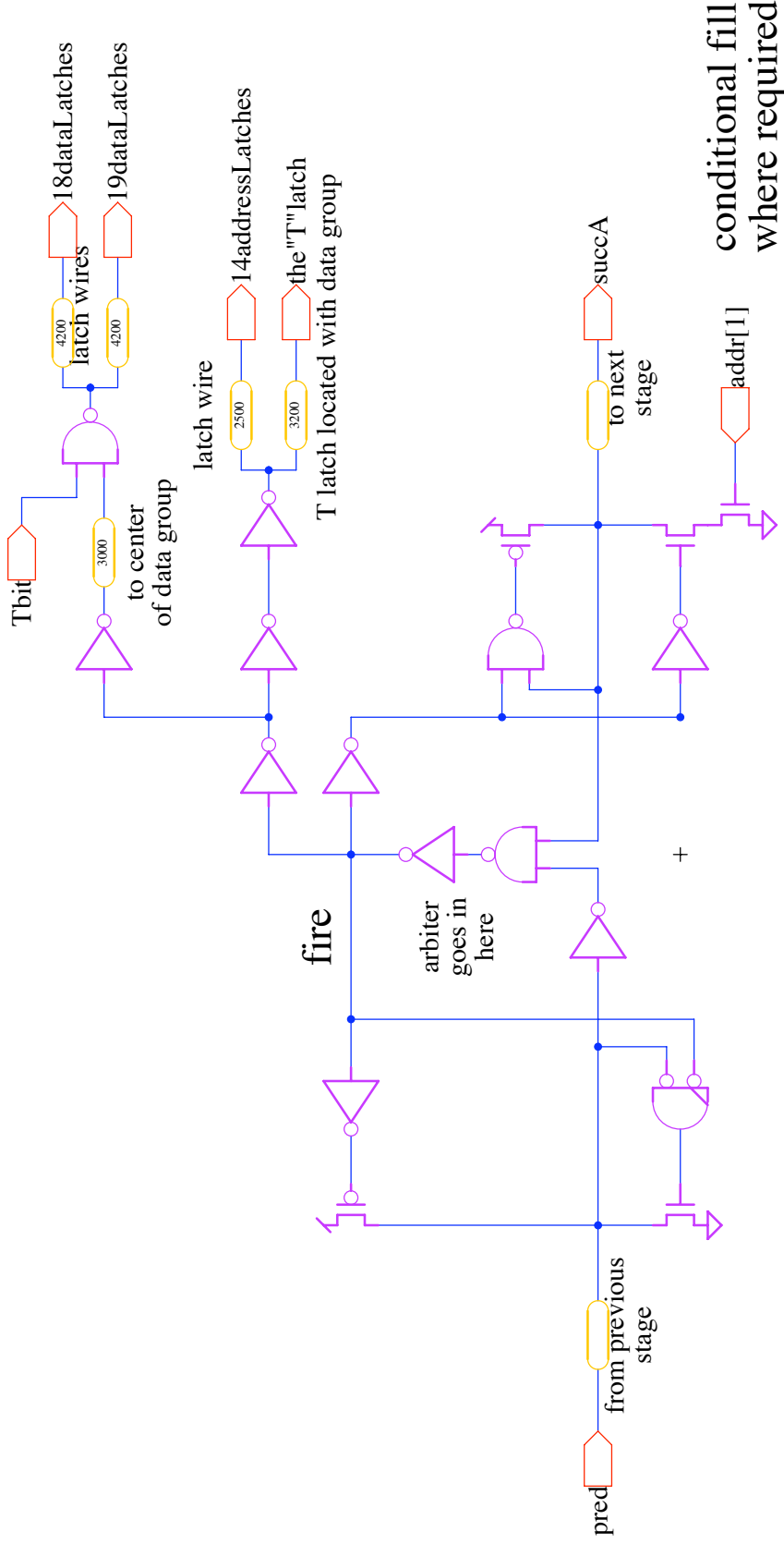
ies 6 May 2007

separate latch drivers
multiple successors

ddd=driver

14 latches	GasP	18 data latches	ddd 19 T ddd	18 data latches
------------	------	-----------------	-----------------	-----------------

layout, about to scale



ddd=driver

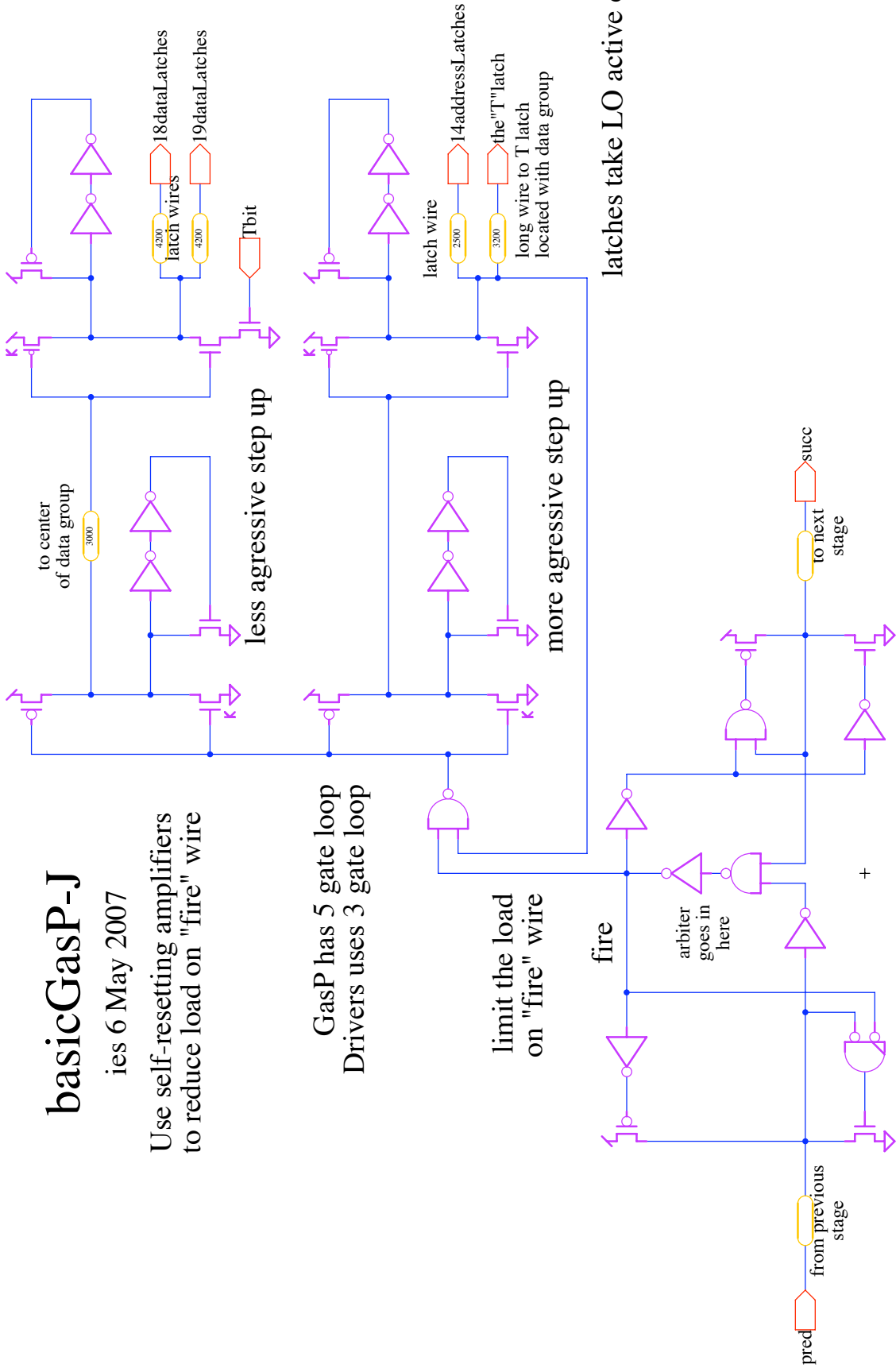
14 latches	GasP	18 data latches	ddd 19 T ddd	18 data latches
------------	------	-----------------	-----------------	-----------------

layout, about to scale

basicGasP-J

ies 6 May 2007

Use self-resetting amplifiers to reduce load on "fire" wire



GasP has 5 gate loop
Drivers uses 3 gate loop

limit the load
on "fire" wire

fire

arbiter
goes in
here

pred
from previous
stage

to next
stage

succ

to center
of data group

3000

less aggressive step up

more aggressive step up

latch wire
2500
14addressLatches
3200
the "T" latch
long wire to T latch
located with data group

latches take LO active drive

basicGasP-K

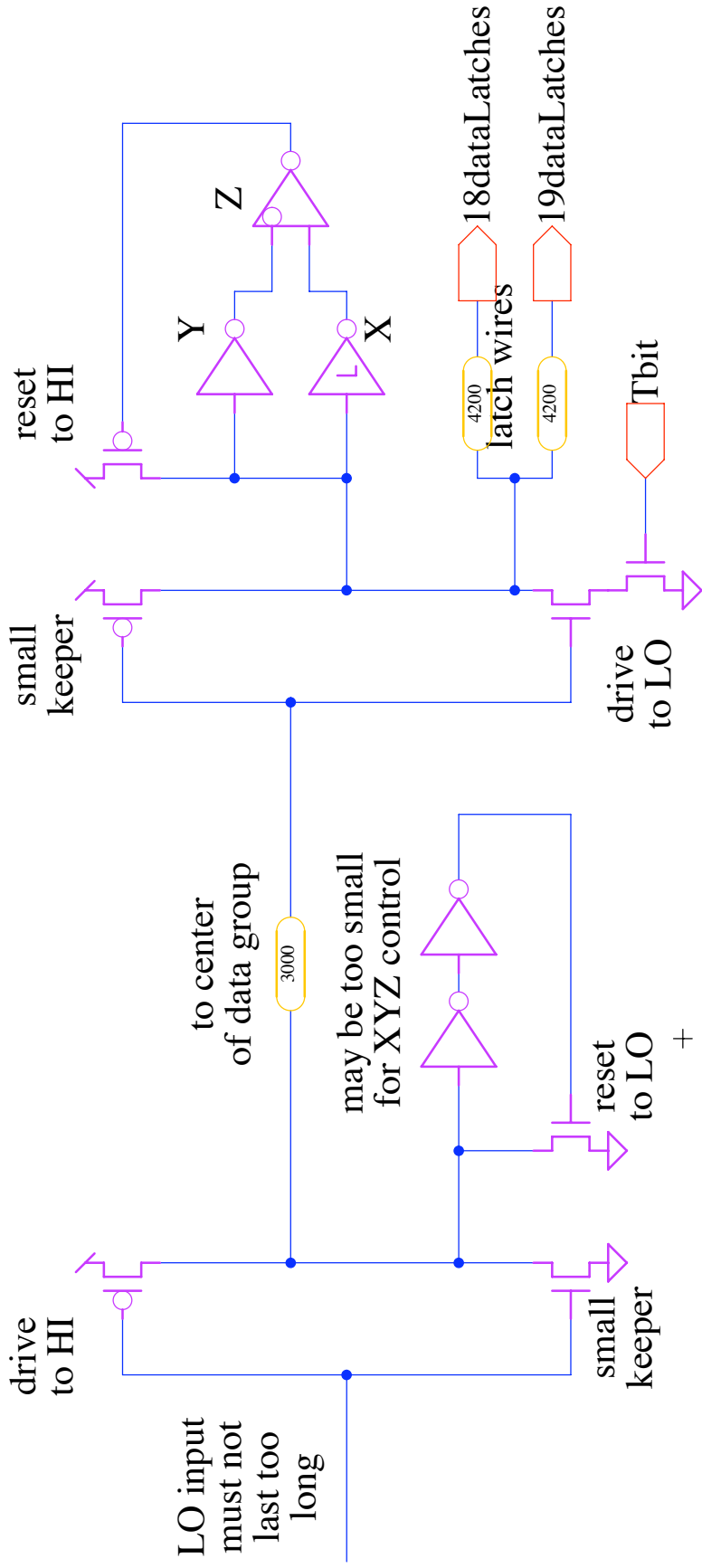
ies 6 May 2007

a closer look at self reset

X acts when "LO enough"
 Y controls length of reset
 X,Z size sets pulse width

this amp mostly
drives wire

this amp drives
latches and wire



basicGasP-L

ies 6 May 2007

an even closer look at self reset

