

Quick History of Asynchronous Design

Univ of Illinois: Muller
Illiac II
initial mathematical theory
C-element

Sutherland/Sun
Micropipelines

Philips/Manchester: Furber
Bundled Data/Micropipelines

Conceptual foundation:
Boolean Logic + C-element + timing
Delay sensitivity:

Boolean combination circuit with Matched delay line

Handshake Solutions
Async Arm,
8051,
design tools

Silistix
interconnect

Sun
GP Processor

Cal Tech: Martin
Quasi Delay Insensitive (QDI)

Conceptual foundation:

Production rule primitives with hysteresis + Boolean logic + C element + timing

Delay sensitivity:

Isochronic fork

Fulcrum
ehernet router

Achronix
Async FPGA

Invocation Model

Theseus Research: Fant
NULL Convention Logic (NCL)

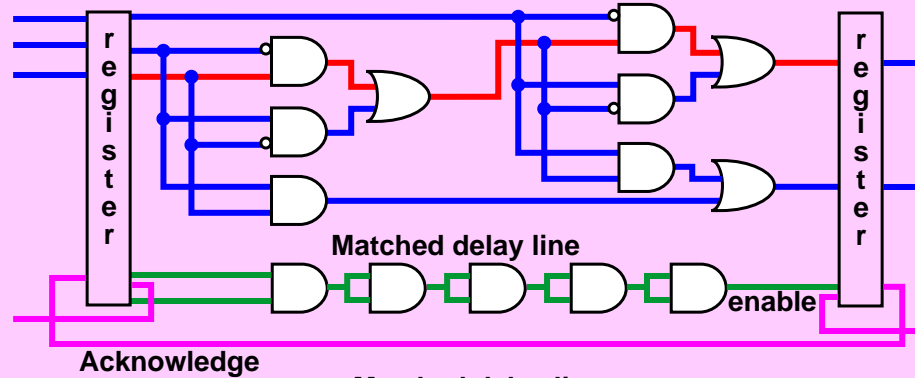
Conceptual foundation:
Complete and coherent logic
Delay sensitivity:
Orphan path

Theseus Logic/
Camgian
Mixed signal

Wave Semiconductor
Flow Graph Machine

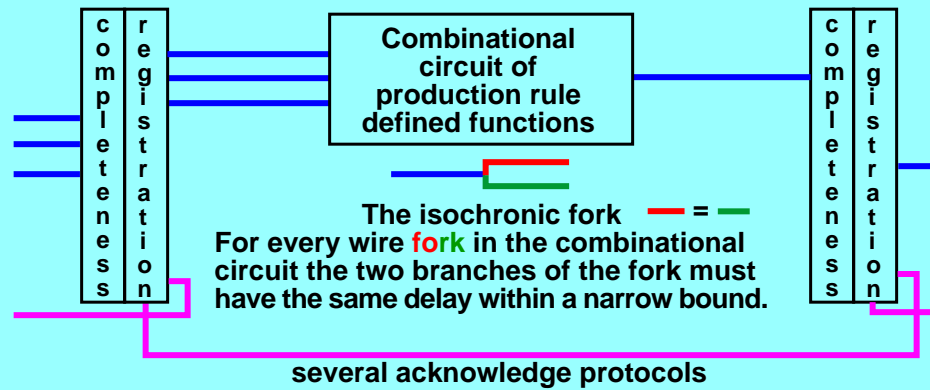
Quick View of Style and Delay Sensitivity

Micropipelines/
Bundled Data



Matched delay line — $\text{red} < \text{green}$
Delay line is strictly greater than the delay of the **critical path** through the circuit for all conditions of voltage and temperature. Timing closure can be more difficult than for clocked circuits.

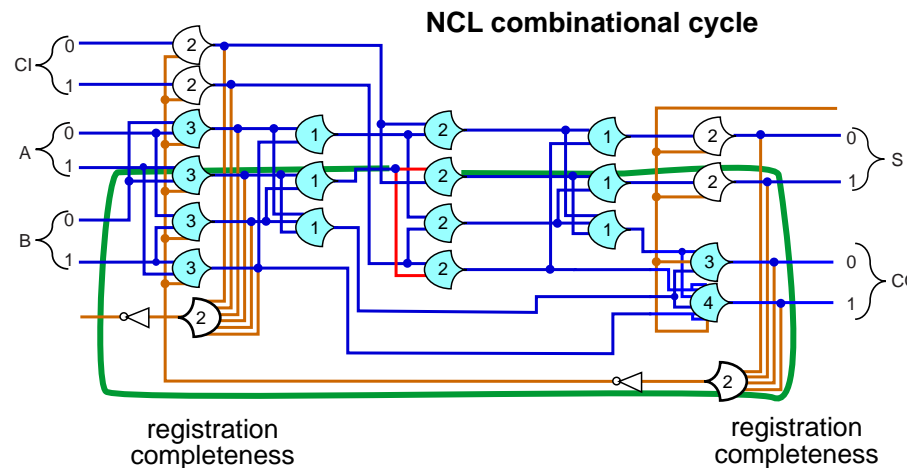
Quasi Delay Insensitive
(QDI)



The isochronic fork — $\text{red} = \text{green}$
 For every wire **fork** in the combinational circuit the two branches of the fork must have the same delay within a narrow bound.

several acknowledge protocols

NULL Convention Logic



The orphan
 For every wire fork in the combinational circuit the delay of each **branch** must be less than the **path closing the cycle**.

Quick Comparison of Approaches

Micropipelines/ Bundled Data

A system is designed in terms of Boolean logic and C-Elements.

Coordination is with handshake protocols and matched delay lines.

There are critical timing issues to resolve that are much more complex than clock design issues.

Quasi Delay Insensitive (QDI)

A system is designed in terms of custom production rules, Boolean logic and C-Elements.

Coordination is with several flavors of handshake protocols mixed together.

The critical timing issue is the isochronoc fork which complicates combinational circuit design.

NULL Convention Logic

Every aspect of a system coordination, control, data transformation and data flow is in terms of NCL. No other form of expression is needed.

Coordination is with a single simple handshake protocol.

The timing issue is the orphan path which is easily managed and is non-critical.

NULL Convention Logic

A Complete and coherent logic:

NCL is a complete and coherent logic, sufficient in itself, fully logically determined in its conceptual foundations.

Homogeneous expression:

System expressed solely in terms of NCL logical relationships.

System composed in terms of coupled cycles.

Other Asynchronous Methods

Ad hoc methodologies:

A collection of ad hoc methods attempting to make Boolean logic do something it is intrinsically incapable of doing.

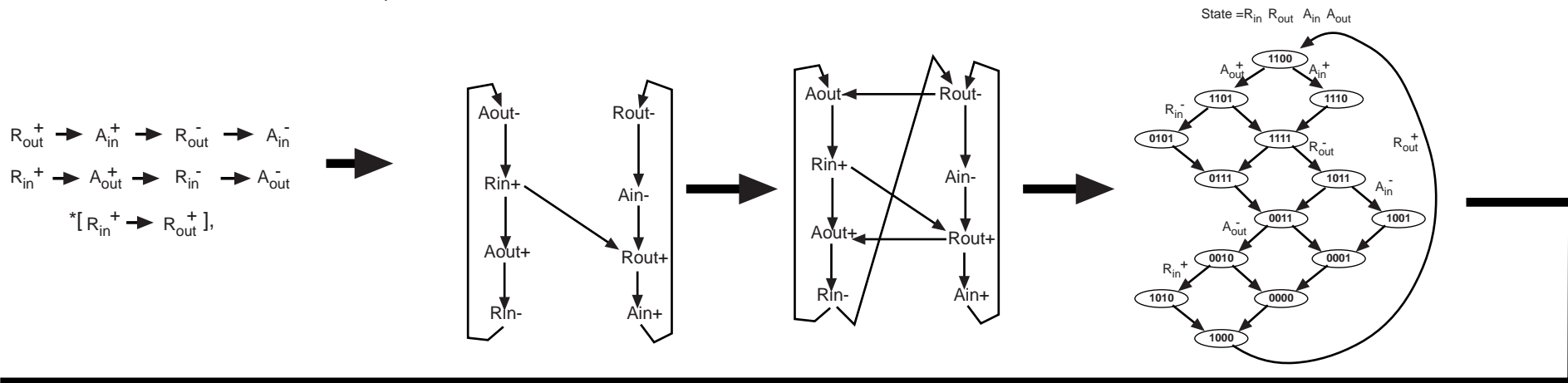
Heterogeneous expression:

System expressed in terms of multiple interrelated forms of expression: Boolean logical relationships, C-element relationships, timing relationships.

System composed in terms of combinational circuits, registers and handshake protocols.

DCVSL Synthesis (QDI)

One Bit, Dual Rail Two Place Buffer

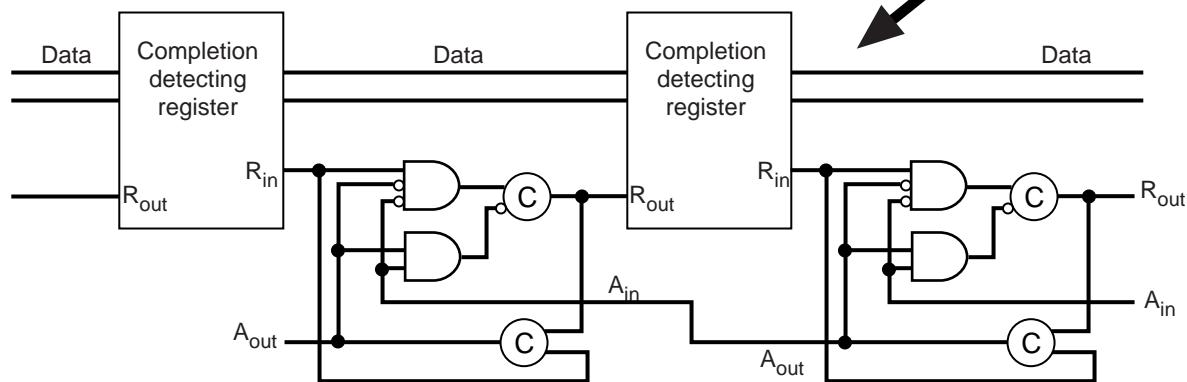
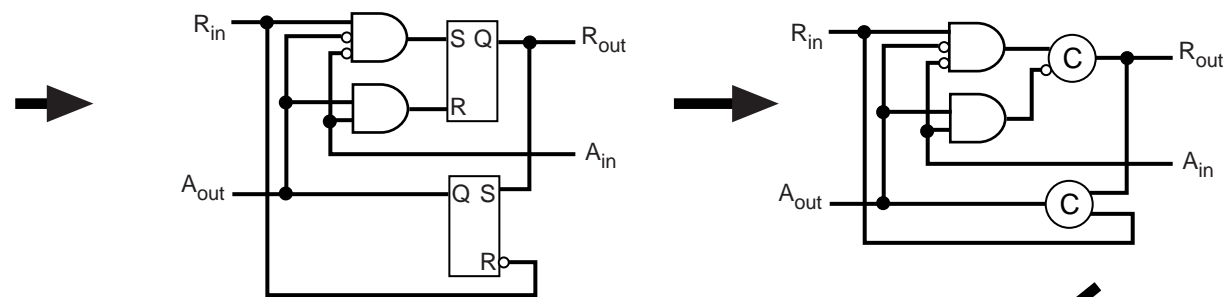


R_{out}	A_{in} A_{out}	R_{in} R_{out}	00	01	11	10
		00	0	x	1	1
		01	0	1	1	0
		11	0	0	0	0
		10	0	x	1	0

$$R_{out} = A_{out} A_{in} + R_{out} A_{out} + R_{out} A_{in}$$

A_{out}	A_{in} A_{out}	R_{in} R_{out}	00	01	11	10
		00	0	x	1	0
		01	0	1	1	1
		11	0	1	1	1
		10	0	x	1	0

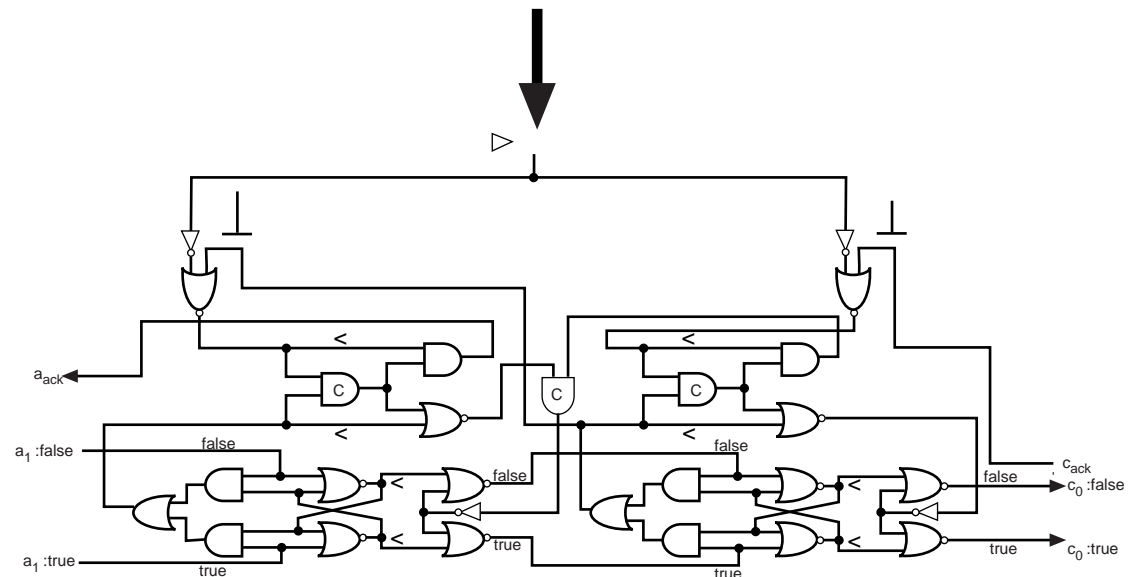
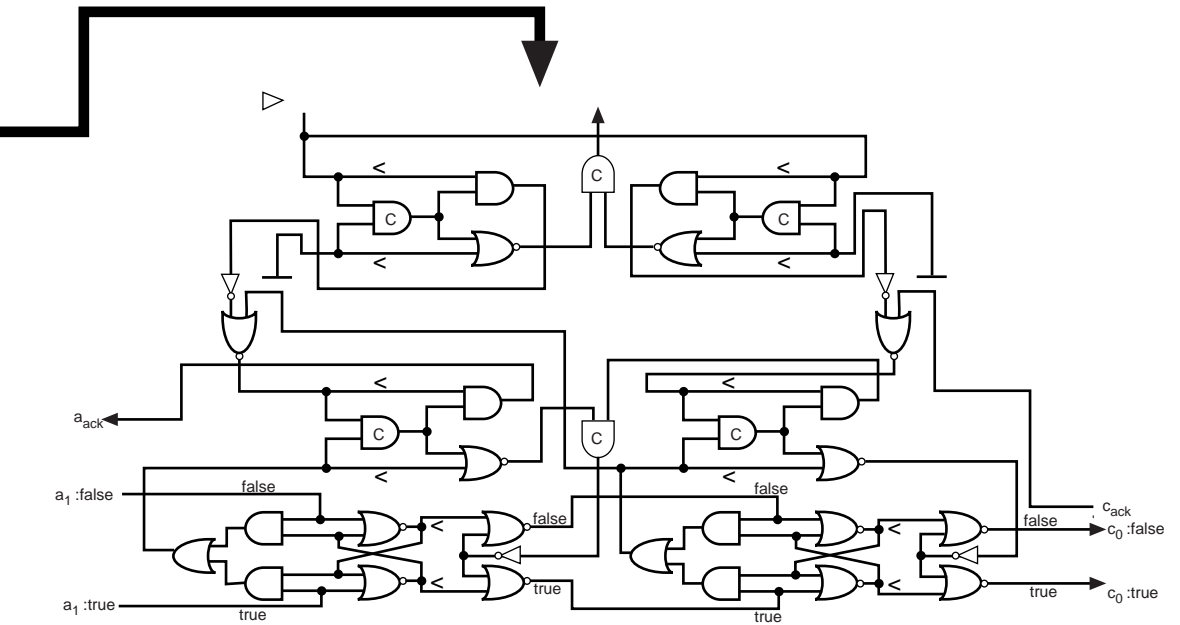
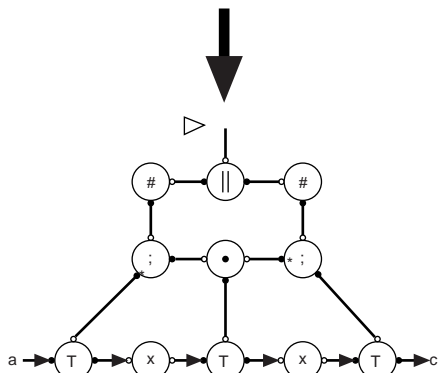
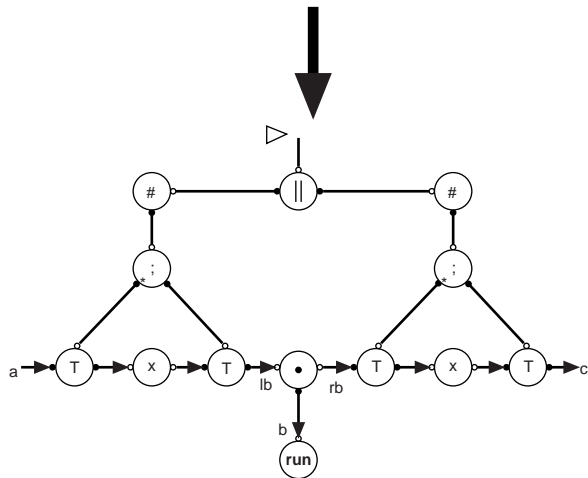
$$A_{out} = R_{out} + R_{in} A_{out}$$



Tangram Synthesis

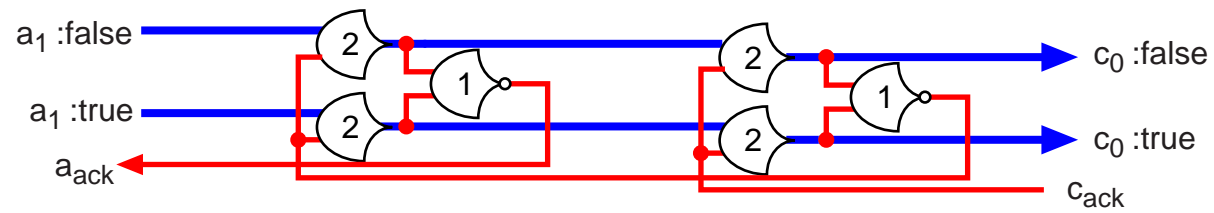
One Bit, Dual Rail Two Place Buffer

$(a?W \& c!W) \bullet [b : \text{chan } W \mid (\text{BUF}_1(a,b) \parallel \text{BUF}_1(b,c))]$

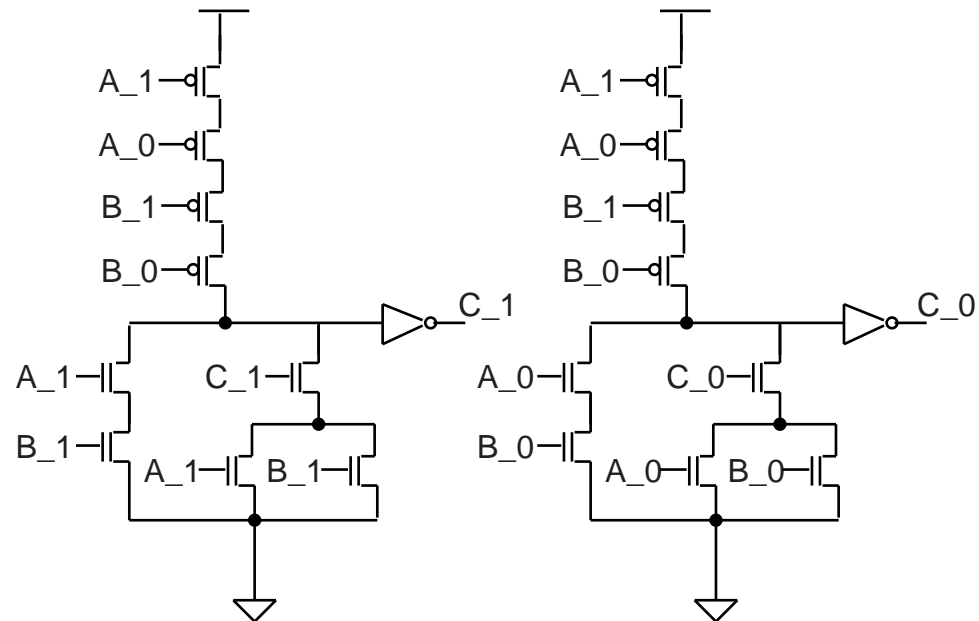
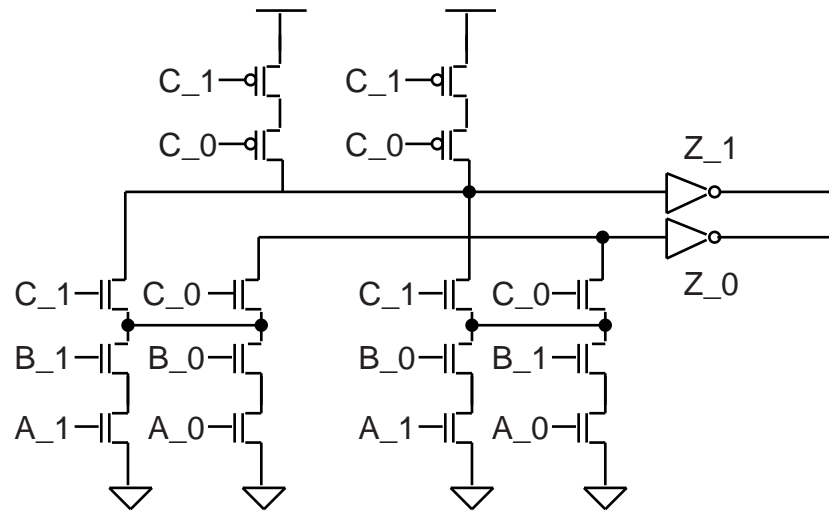


NCL

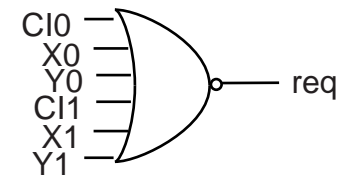
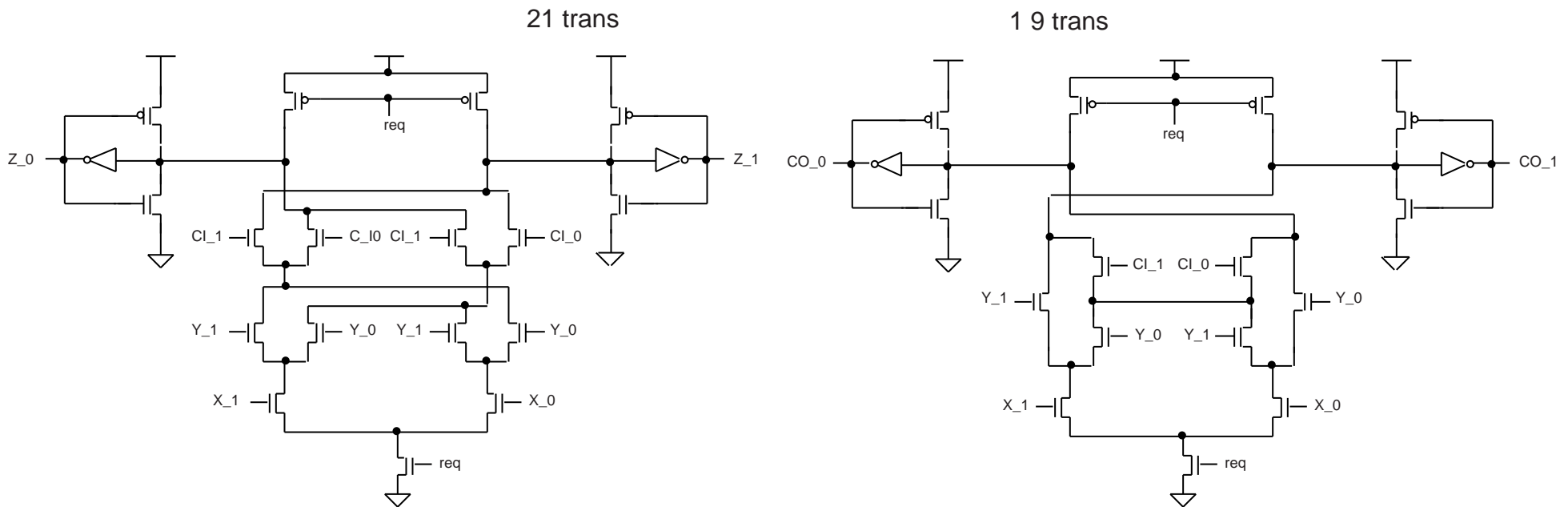
One Bit, Dual Rail Two Place Buffer



Martin full adder 42 trans



DCVSL circuit full-adder sum and carry



12 trans

Req completeness might be taken from previous completion

Invocation Model: General Concurrency

Invocation Model

Distributed and concurrent in its foundations



NULL Conception Logic

A complete and coherent fully determined logic sufficient in itself



Cycle

A unit of composition that integrates combinational transformation, flow coordination, memory and liveness.



Pipeline

spontaneously flowing data



Pipeline processors

structure of locally coordinated processor pipeline segments.



Programmable Array of pipeline processors

exploits all the parallel and pipeline concurrency of individual programs as well as the parallel concurrency of multiple independent programs

QDI: Improvised Concurrency

Sequential Model Communicating sequential processes

Boolean Logic Boolean logic enhanced with the C-element, STGs, STDs, semi-modularity

Register, handshake, combinational circuit A heterogeneous mix of composition units.

Register Transfer mix of spontaneous and explicitly controlled data flow

Asynchronous sequential processors structure of locally and globally controlled registers and processing stages.

Programmable Sequential processor performs strictly one operation at a time in sequence.

Sequential Model: Strict Sequentiality

Sequential Model Centralized and sequential in its foundations

Boolean Logic Requires the assistance of time expression

Register, clock,
combinational circuit A heterogeneous mix of composition units.

Register Transfer explicitly controlled data flow

Sequential processors structure of globally controlled registers and processing stages.

Programmable
Sequential processor performs strictly one operation at a time in sequence.