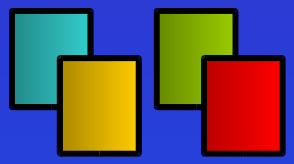

RCIM Presentation

Introduction to Asynchronous Circuits and Systems

Kristofer Perta

April 02 / 2004

University of Windsor – Computer and Electrical Engineering Dept.

Presentation Outline

Section 1 - **Introduction**

Section 2 - **Asynchronous Circuits and Systems**

Section 3 - **Designing Asynchronous Systems**

Section 4 - **References**

Section 1 - Introduction

1.1.1 - **Brief Introduction**

1.2.1 - **Asynchronous Circuits and Systems (ACAS)**

1.2.2 - **VLSI Design Issues and ACAS Advantages**

1.2.3 - **Recent Developments in ACAS**

1.1.1 - Brief Introduction

To investigate asynchronous building blocks and protocols.

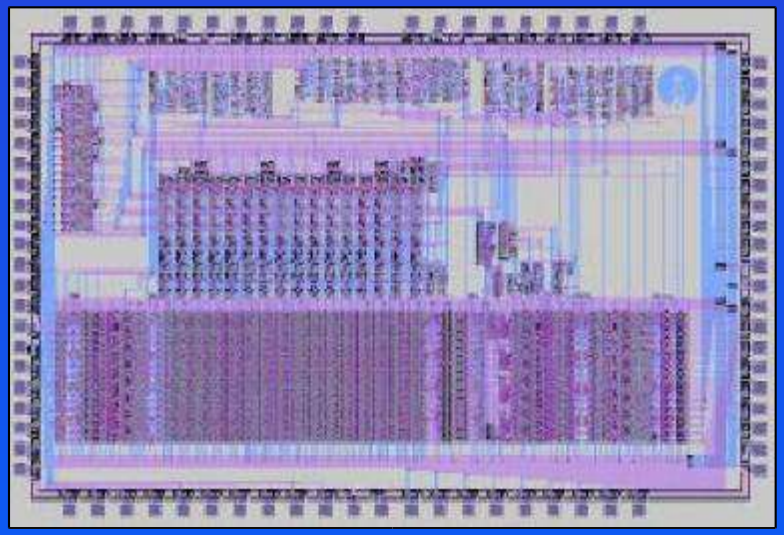

Figure 1 - World's first Asynchronous Microprocessor developed by Caltech in 1989

1.2.1 - Asynchronous Circuits and Systems (ACAS)

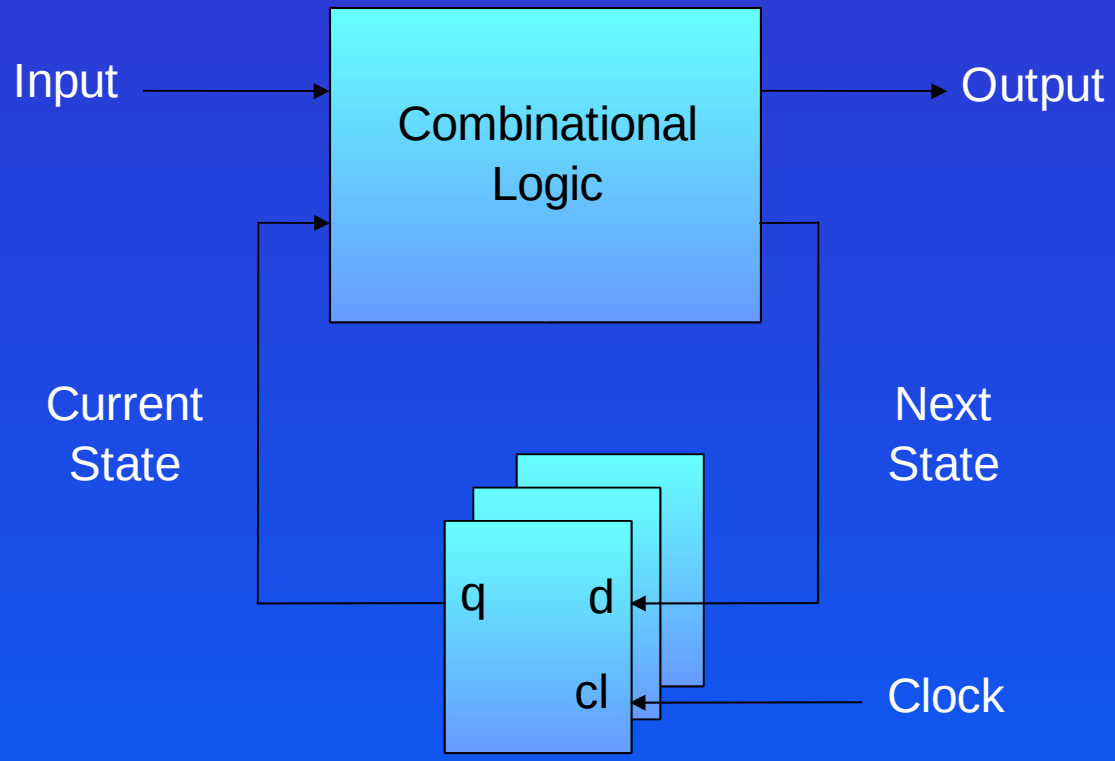

Figure 2 – Synchronous Circuit

1.2.1 - Asynchronous Circuits and Systems (ACAS)

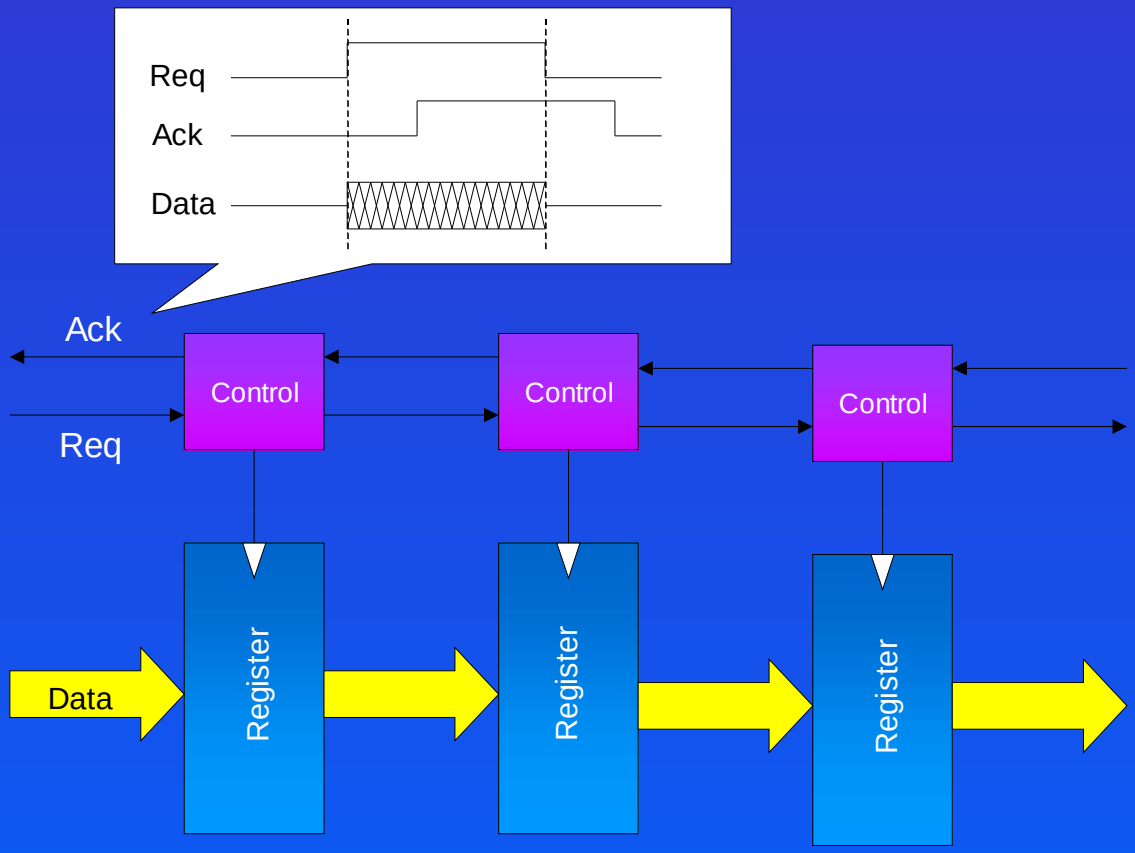

Figure 3 – Asynchronous Circuit

1.2.2 - VLSI Design Issues and ACAS Advantages

VLSI Design Issues

Lowering power consumption

Addressing clock skew issues

Decreasing noise

Increasing performance

ETC,ETC.....

ACAS Advantages

Elimination of Clock Skew

Average Case Performance

Adaptivity to Processing and Environmental Variations

Component Modularity and Reuse

Lower System Power Requirements

Reduced Noise

1.2.3 - Recent Developments in ACAS

Use of asynchronous circuits in the UltraSPARC IIIi synchronous processor at Sun Microsystems

Brackenbury et al. use of asynchronous techniques with VLSI implementations of communication systems, specifically the Viterbi decoder

Section 2 - Asynchronous Circuits and Systems

2.1.1 - **Introduction**

2.2.1 - **Bundled Data or Single Rail Protocols**

2.2.2 - **4 Phase Bundled Data Protocol**

2.2.3 - **2 Phase Bundled Data Protocol**

2.3.1 - **Dual Rail Protocols or 1-of-2 Protocol**

2.3.2 - **4 Phase Dual Rail or 1-of-2 RTZ Protocol**

2.3.3 - **2 Phase Dual Rail or 1-of-2 NRTZ Protocol**

2.4.1 - **Discussion On Protocol Choice**

2.5.1 - **The Muller C-Element**

2.5.2 - **The Muller Pipeline**

2.1.1 - Introduction

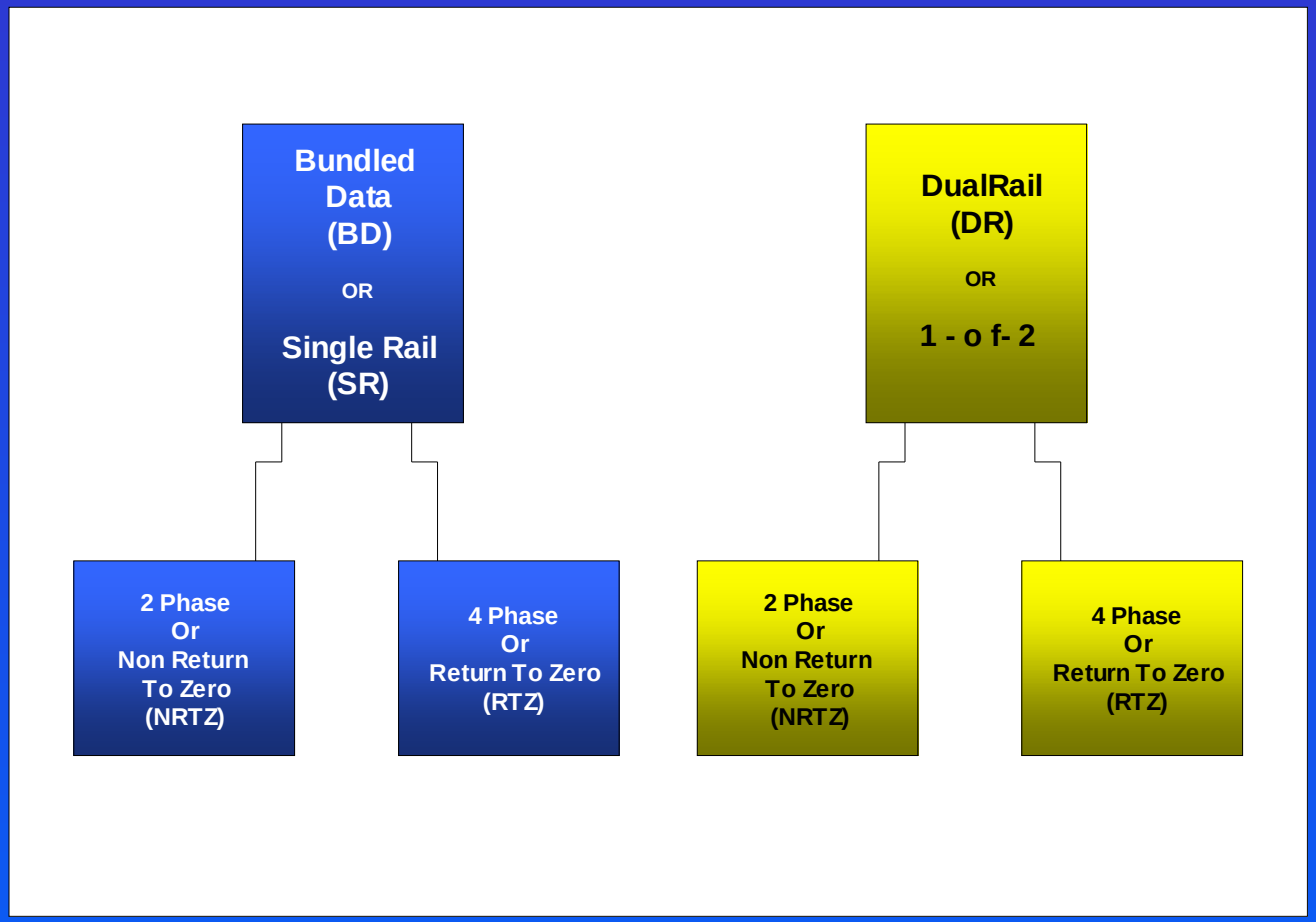

Figure 4 – Protocols

2.2.1 - Bundled Data or Single Rail Protocols

Bundled Data (BD) and Single Rail (SR) refers to the separate single (SR) request and acknowledge wires that are bundled (BD) together with the data signals

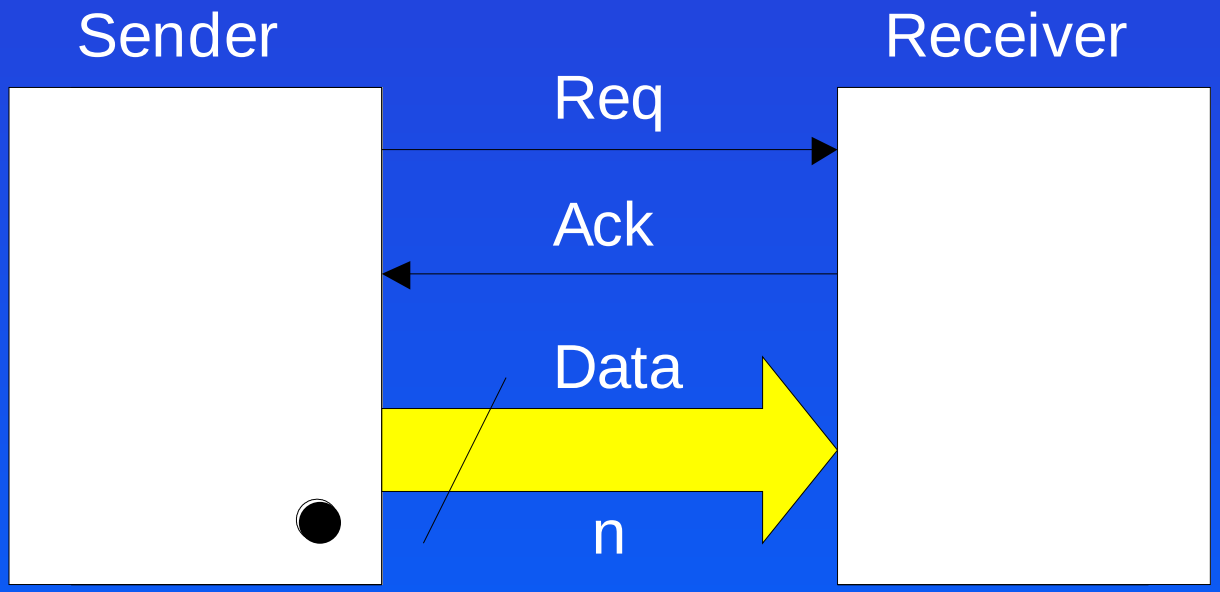

Figure 5 – Bundle Data Channel

2.2.2 - 4 Phase Bundled Data Protocol

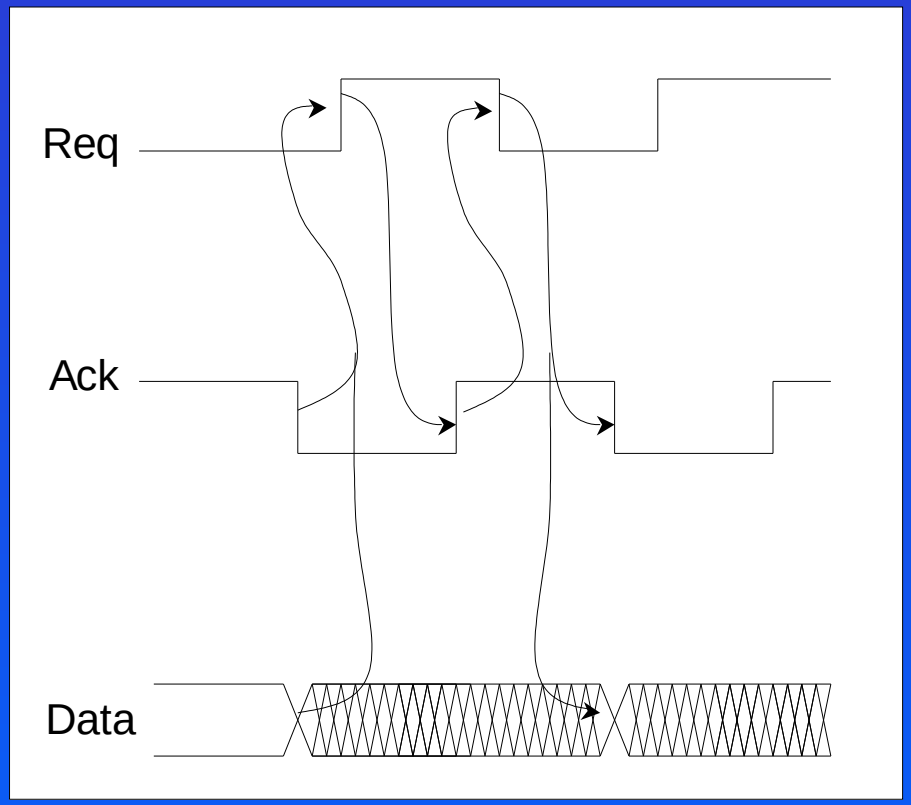

Figure 6 – 4-Phase Bundled Data Protocol

2.2.3 - 2 Phase Bundled Data Protocol

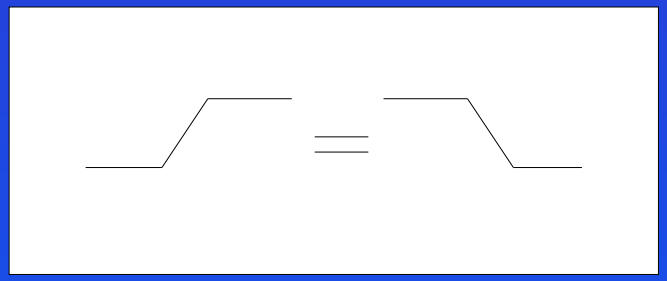

Figure 7 – Transition Signalling Paradigm

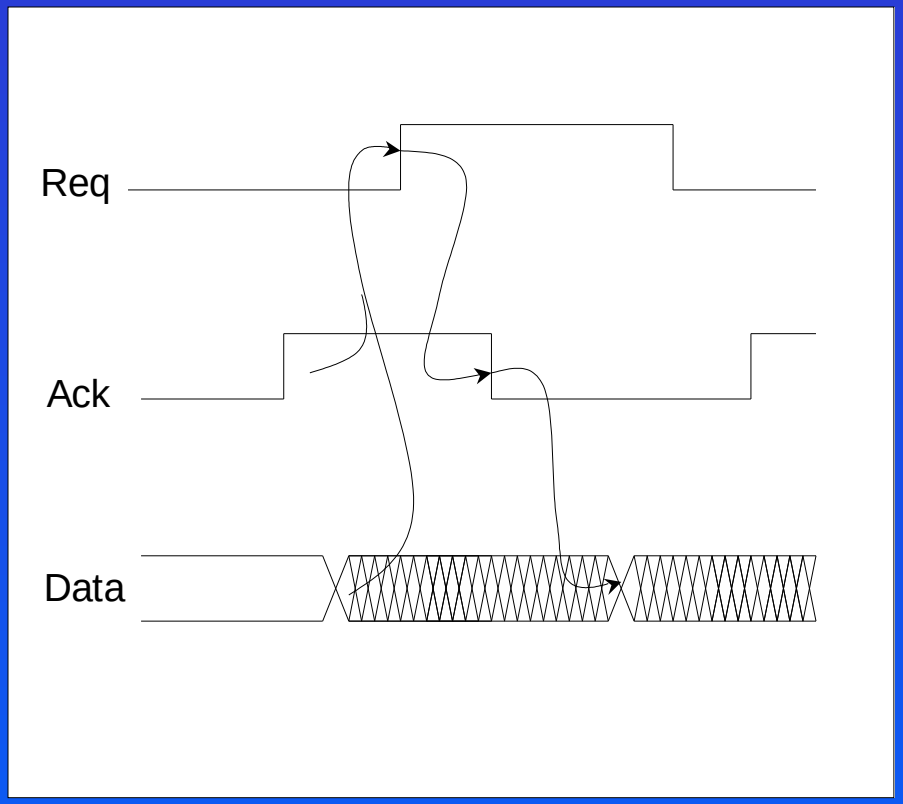

Figure 8 – 2-Phase Bundled Data Protocol

2.3.1 - Dual Rail Protocols or 1-of-2 Protocol

Dual Rail (DR) refers to the protocol's use of 2 (DR) wires to encode 1 bit of data information (1-of-2)

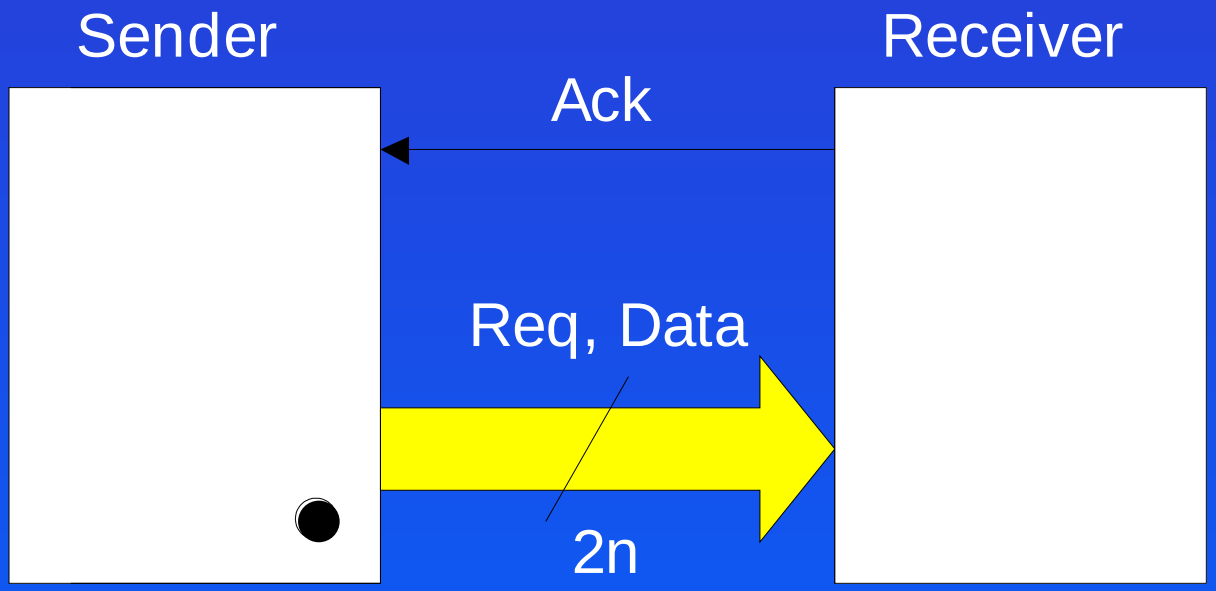

Figure 9 – 4 Phase Dual Rail Channel

2.3.2 - 4 Phase Dual Rail or 1-of-2 RTZ Protocol

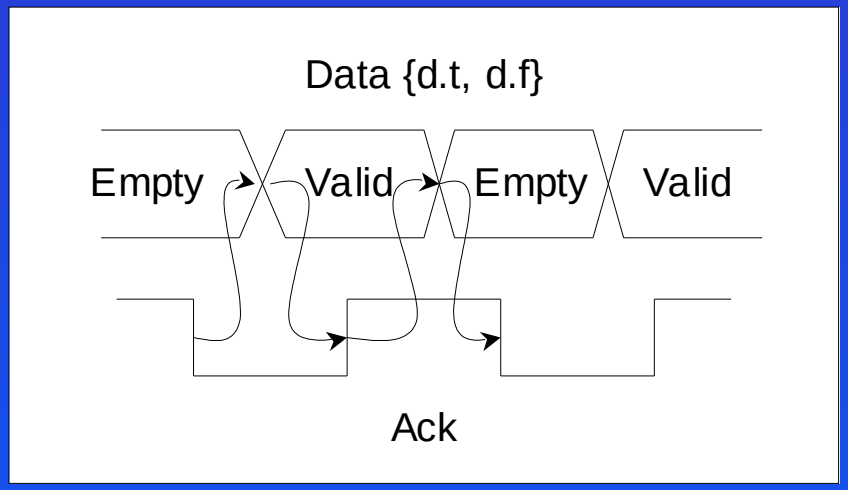

Figure 10 - 4-Phase Dual Rail Protocol

For n=1	d.t	d.f
Empty	0	0
Valid “0”	0	1
Valid “1”	1	0
Not Used	1	1

Table 1 - 1 bit Channel Encoding Chart

2.3.3 - 2 Phase Dual Rail or 1-of-2 NRTZ Protocol

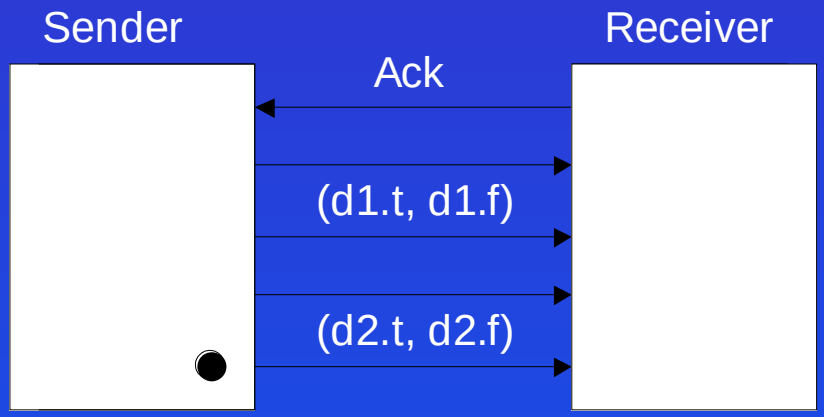

Figure 11 - 2-Phase Dual Rail Channel

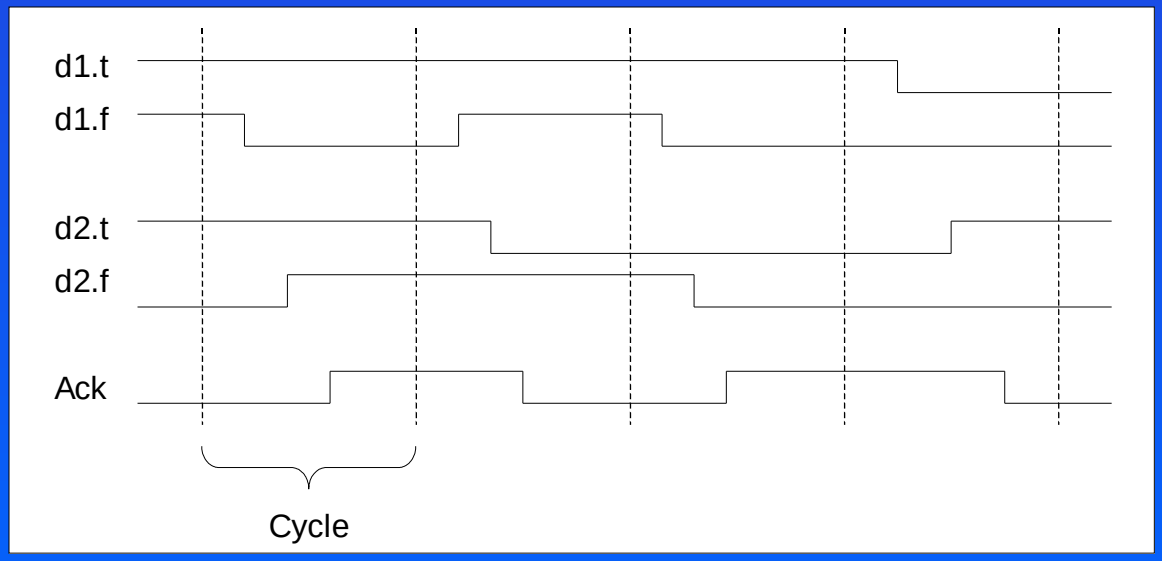

Figure 12 - 2-Phase Dual Rail Protocol

2.4.1 - Discussion On Protocol Choice

D. W. Lloyd et al. have presented a comparison on asynchronous design styles

	Area (wires/bit)	Energy (transitions/bit)
2PBDP	1	1/2 (average)
2PDRP	2	1
4PDRP	2	1
1-of-4 (RTZ)	2	1
1-of-4 (NRTZ)	2	1/2 (average)

Table 2 – Comparison of Protocols

2.5.1 - The Muller C-Element

David Muller invented the Muller C-Element in 1959

The Muller pipeline is the backbone for handshaking circuitry

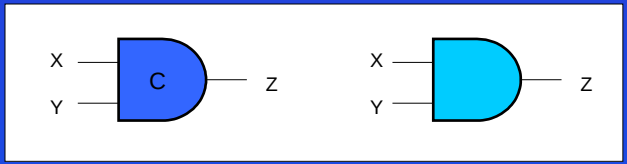

Figure 13 – The C-Element and OR Element schematic, respectively

C-Element Truth Table		
X	**Y**	**Z**
0	0	0
0	1	Retain previous z value
1	0	Retain previous z value
	1	1

Table 3 - C- Element Truth Table

OR-Element Truth Table		
X	**Y**	**Z**
0	0	0
0	1	1
1	0	1
1	1	1

Table 4 - OR Element Truth Table

2.5.2 - The Muller Pipeline

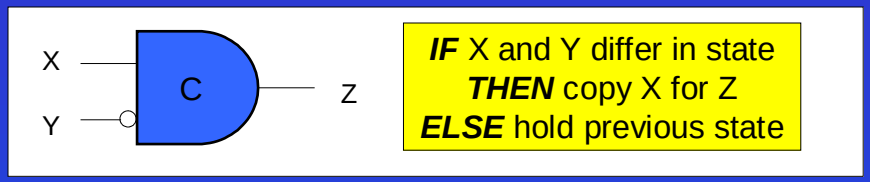

Figure 14 – Behaviour of C-Element with Inverter

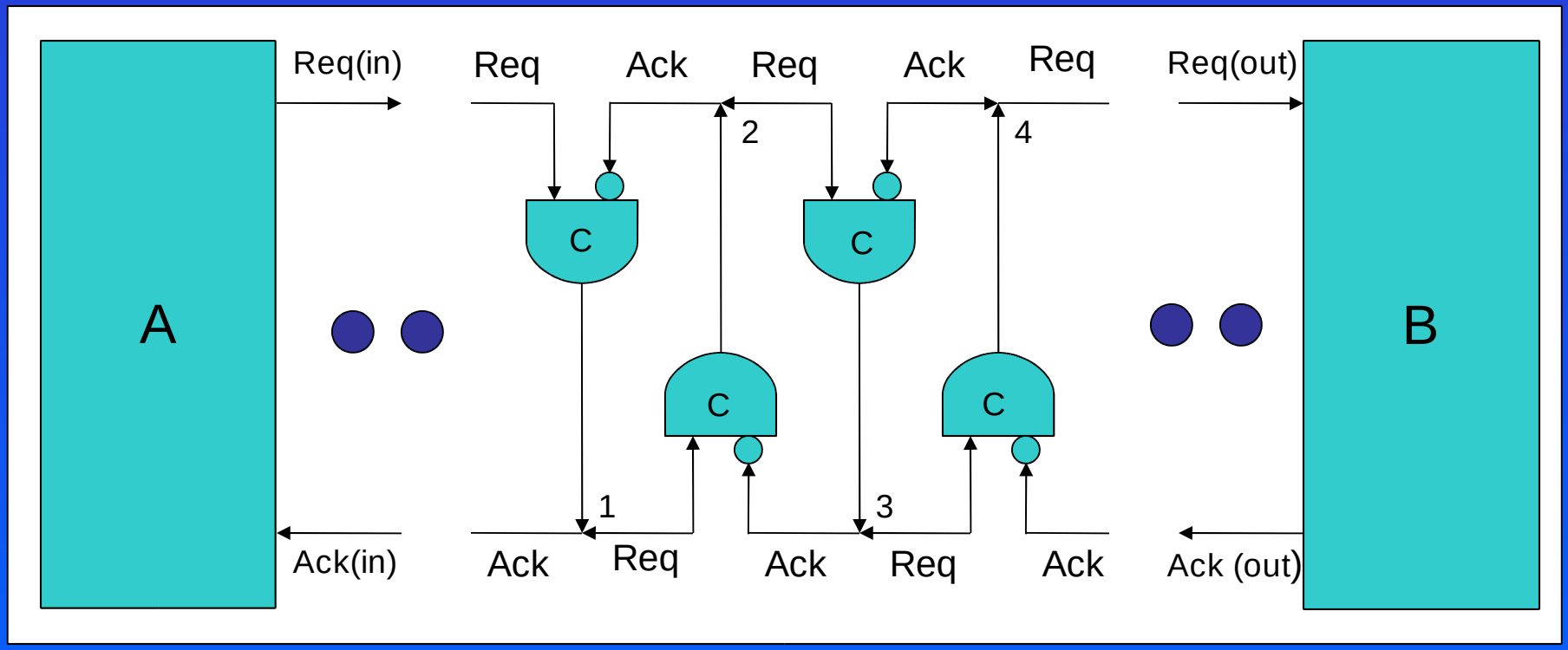

Figure 15 – The Muller Pipeline

Section 3 - Designing Asynchronous Systems

3.1.1 - **Possible Design Issues**

3.2.1 - **Balsa (Asynchronous Hardware Language)**

3.2.2 - **Balsa Design Flow**

3.2.3 - **Balsa Buffer Example**

3.3.1 - **Conclusion**

3.1.1 - Possible Design Issues

Asynchronous circuits, at a hardware level, are complex creatures that have not been addressed by the industry standard Electronic Design and Automation (EDA) tools and companies.

“EDA tools are lacking”, explains Ian Sutherland, vice president and fellow at Sun Microsystems Laboratories. EDA tools are intended for synchronous system design.

VHDL and Verilog, industry standard hardware description languages, lack the needed “concurrency and platform for handshaking channels” that are required by asynchronous systems.

3.2.1 - BALSA (Asynchronous Hardware Language)

A recent free tool from the University of Manchester, one of the academic leaders in asynchronous design, is 'BALSA'.

Balsa is both a “framework for synthesis of asynchronous hardware systems and a language for describing such systems”.

Balsa used the adopted approach of proprietary languages like Tangram of “syntax-directed compilation into communication handshaking components”.

This means that there is direct “one-to-one” mapping between language and direct handshaking circuits produced.

3.2.2 - BALSA Design Flow

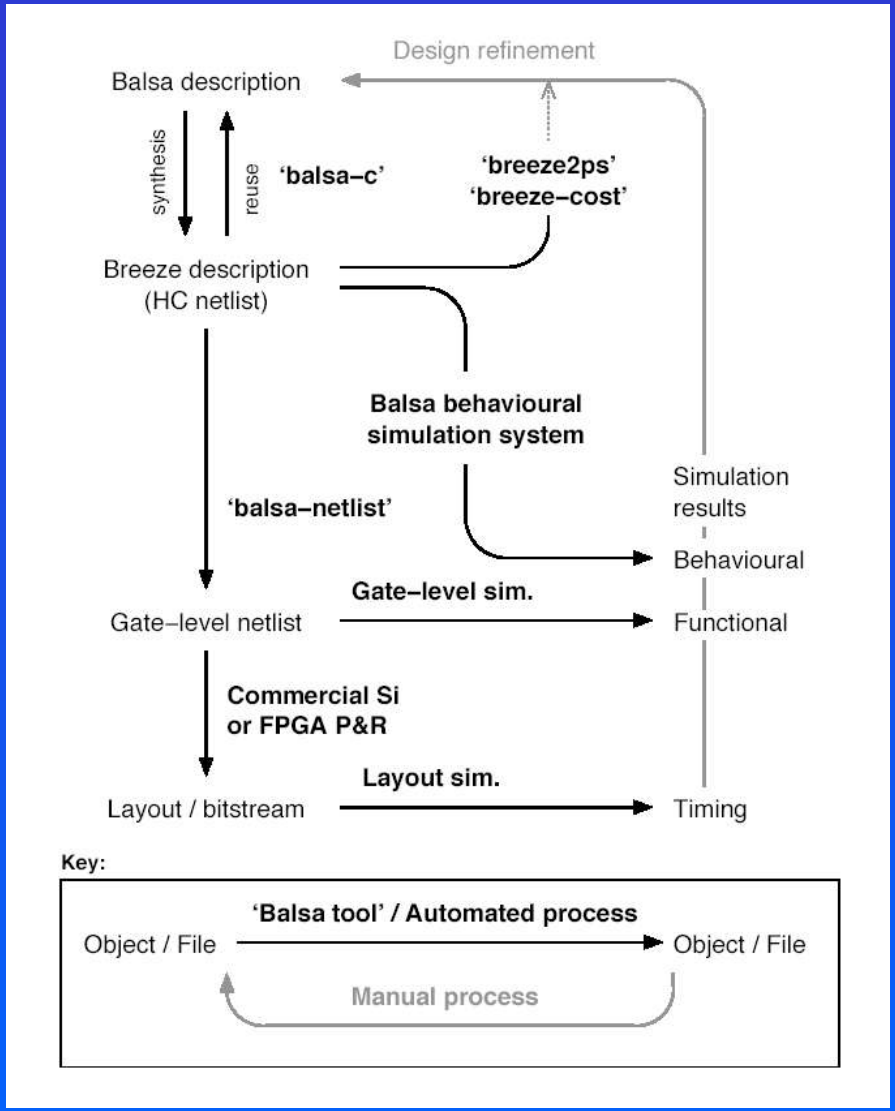

Figure 16 – Asynchronous Digital Design Flow

3.2.3 - Balsa Buffer Example

```
import [balsa.types.basic]
procedure bufferloop (input i : bit; output o : bit)
is
variable x : bit

begin
        loop
                i -> x -- Input communication
                ; -- Sequence operator
                o <- x -- Output communication
        end
end
```

3.2.3 - Balsa Buffer Example

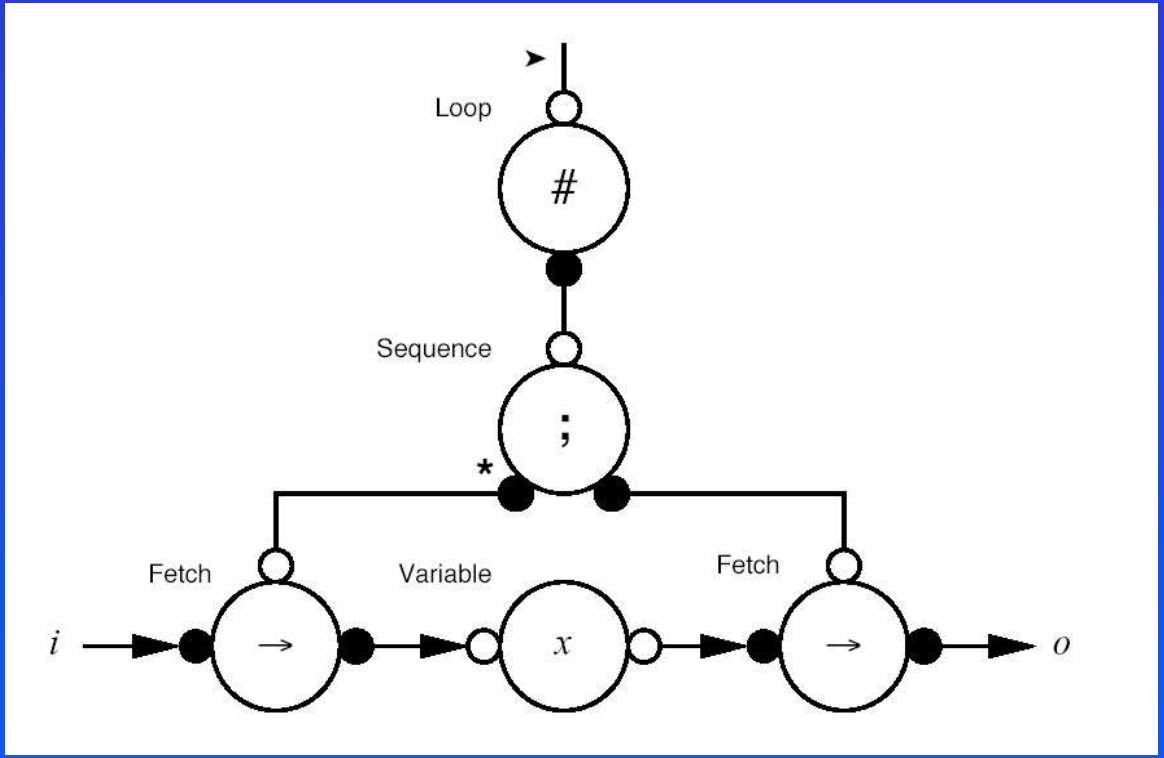

Figure 17 – Handshake Circuit For a Single Place Buffer

3.2.3 - Balsa Buffer Example

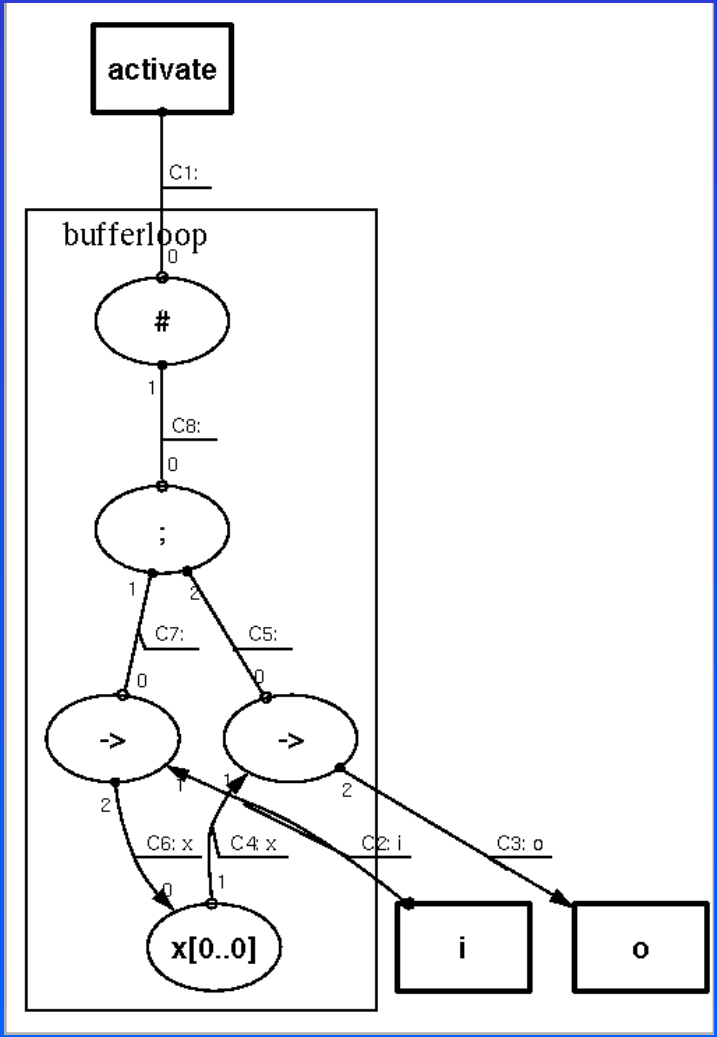

Figure 18 – Generated Handshake Circuit For a Single Place Buffer

3.3.1 - Conclusion

Asynchronous design is not a direct mapping of a Synchronous Design, this depends on the type of design one is doing (analog layout or digital design flow)

Asynchronous design does have the possibility of being a feasible alternative to the 'norm', which is synchronous

The combination of asynchronous design with synchronous design is a slow, yet emerging field of study

Section 4 - References

[1] http://www.async.caltech.edu/cam.html

[2] C.H. Van Berkel, M.B. Josephs, and S.M. Nowick. “Scanning theTechnology: Applications of Asynchronous Circuits”, Proceedings ofthe IEEE, Volume 87, Issue 2, Feb. 1999, pages 223-233

[3] Edited by Jens Sparsø, Steve Furber. “Principles of Asynchronous Circuit Design – A System Perspective”, Kluwer Academic Publishers, 2001.

[4] http://www.sciam.com/article.cfm?articleID=00013F47-37CF-1D2A-97CA809EC588EEDF

[5] L. E. M. Brackenbury, M. Cumpstey, S.B. Furber, P.A. Riocreuz, “Applying Asynchronous Techniques to a Viterbi Decoder Design”, IEEE, 2001.

[6] P.A. Riocreux, L.E.M. Brackenbury, M. Cumpstey, S.B. Furber ,”Low-Power Self-Timed Viterbi Decoder”, Seventh International Symposium on Asynchronous Circuits and Systems, March 2001

Section 4 - References

[7] K. E. Tepe, “Iterative Decoding Techniques for Correlated Rayleigh Fading and Diversity Channels”, Communication, Information and Voice Processing Report Series, Report TR-2001-1, University of Lund - Information Technology Department and Rensselaer Polytechnic Institute - Electrical, Computer and System Engineering Department February 2001.

[8] I. Sutherland, “Micropipelines: Turing Award Lecture”, Communications of the ACM, June 1989. Vol 32, #6, pp. 720-738

[9] D. Muller and W. Bartky, “A Theory of Asynchronous Circuits”, Proceedings of an International Symposium on the Theory of Switching, p.p. 204-243, April 1959

[10] R. B. Wells, ”Applied Coding and Information Theory for Engineers”, Prentice Hall, 1999

[11] S. Shahzad Shah, S. Yaqub, F.l Suleman, "Self-Correcting Codes Conquer Noise, Part One: Viterbi Codecs", EDN Magazine, 2001 http://www.reed-electronics.com/ednmag/contents/images/75255.pdf

[12] R. Goering. “Keynoter Sees Asynchronous Future For Digital Designs”, EE Times, December 4, 2002. http://www.eetimes.com/story/OEG20021204S0018