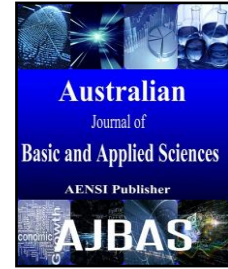




ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com



Design and Characterization of a Completely Pipelined Null Convention Logic Floating Point Multiplier

¹Anitha Juliette Albert and ²Seshasayanan Ramachandran

¹ Research Scholar, Centre for Research, Department of Electronics and Communication Engineering, Faculty of Information and Communication, Anna University, Chennai, Tamilnadu, India, 600025

² Associate Professor, Department of Electronics and Communication Engineering, Faculty of Information and Communication, College of Engineering, Anna University, Chennai, Tamilnadu, India, 600025

ARTICLE INFO

Article history:

Received 10 March 2015

Received in revised form 20

March 2015

Accepted 25 March 2015

Available online 10 April 2015

Keywords:

Null Convention Logic, Asynchronous, Delay insensitive, Floating point multiplier, pipelined, VLSI, low power

ABSTRACT

This paper presents the design and characterization of a completely pipelined Null Convention Logic based IEEE 32 bit single precision floating point multiplier using full word and bit wise completion strategies. Performance metrics such as speed, area and power of the Null convention logic pipelined floating point multiplier, obtained from Xilinx simulation and Cadence are compared with the NCL non-pipelined and synchronous counterparts. The proposed multiplier offers 55.9% increase in throughput when compared with the non pipelined architecture and presents 62% reduction in power when compared with its equivalent synchronous architecture. Moreover, the proposed multiplier pipelined using bit wise completion strategy offered 22% increase in throughput when compared with the same pipelined using full word completion strategy. Rounding support has not been included to enhance precision. Consequently, the proposed pipelined multiplier can be deployed in asynchronous NCL based digital signal processing applications that demand low power, high speed and high precision. It can be used as a primary library component in the design of NCL based floating point Multiply and Accumulate unit, Floating point Arithmetic Logic Unit and Floating point digital signal processors, that will gain significance in near future, in the field of asynchronous VLSI.

© 2015 AENSI Publisher All rights reserved.

To Cite This Article: Anitha Juliette Albert and Seshasayanan Ramachandran., Design and Characterization of a Completely Pipelined Null Convention Logic Floating Point Multiplier. *Aust. J. Basic & Appl. Sci.*, 9(15): 7-13, 2015

INTRODUCTION

Clocked circuits have dominated semiconductor industry for the past two decades. Excessive clock skew, clock noise and larger power dissipation of clocked circuits have led the way to the asynchronous world of very large scale integration (VLSI). NULL convention logic (NCL) is an asynchronous paradigm that requires less power, generates less noise, radiates less EMI and allows reusability of components compared to the synchronous counterparts (Smith, 2009). High dynamic range applications such as Fast Fourier Transforms require an efficient hardware to support floating point data. We have designed an IEEE 754 single precision floating point multiplier (FPM) in NCL architecture. This multiplier can be used as one of the key components of NCL integrated digital signal processors, a promising future of NCL design. The designed NCL FPM is compared with the synchronous FPM. The performance attributes of

NCL FPM are analyzed in terms of power, average delay and area.

Delay insensitivity, hysteresis and input completeness are the distinct advantages of NCL circuits. Delay insensitivity specifies that the circuit operates correctly regardless of when the circuit inputs are available (Smith, 2009). Delay insensitivity is achieved through dual rail or quad rail logic (Smith, 2009). A dual rail signal D consists of two wires, D0 and D1, whose values are from the set {DATA0, DATA1, NULL} as illustrated in Table 1. DATA0 and DATA1 represent Boolean logic levels 0 and 1 respectively. NULL represents empty set, a state when DATA is not available (Smith, 2009). The two rails are mutually exclusive emphasizing that they cannot be asserted simultaneously. If assigned, it is called an illegal state.

Threshold NCL gates with hysteresis state holding capability are constructed to realize the NCL circuits (Smith, 2009). A basic threshold gate, specified as TH_mn gate in figure 1 has n inputs and 1 output. At least m of the inputs must be asserted

Corresponding Author: Anitha Juliette Albert, Research Scholar, Centre for Research, Department of Electronics and Communication Engineering, Faculty of Information and Communication, Anna University, Chennai, Tamilnadu, India, 600025

before the output will become asserted. Hysteresis is enforced by the fact that after the output is asserted, all inputs must be deasserted before the output becomes deasserted. Input completeness illustrates that all outputs must not transit from NULL to DATA or DATA to NULL until all inputs have transited from NULL to DATA or DATA to NULL (Smith, 2009).

The NCL modules, designed using threshold gates, are sandwiched between the delay insensitive (DI) registers to realize a DI, input complete NCL system. The flow of DATA and NULL wavefronts is controlled by the request and acknowledge signals, k_i and k_o as shown in figure 2.

The input wavefronts NULL and DATA are controlled by the handshaking signals and completion detection circuitry. Two adjacent register stages interact through their request and acknowledge signals, k_i and k_o , respectively. The handshaking signals ensure that the two DATA wavefronts are always separated by a NULL wavefront. The acknowledge signals are combined in the Completion detection circuitry to produce the request signals to the previous register stage (Smith, 2009). NCL registration is realized through cascaded

arrangements of single-bit dual-rail registers. These registers consist of th_{22} gates that pass a DATA value at the input only when k_i is request for data (rfd) (i.e., logic 1) and likewise pass NULL only when k_i is request for null (rfn) (i.e., logic 0). They also contain a NOR gate to generate k_o , which is rfn when the register output is DATA and rfd when the register output is NULL. The registers are reset to NULL, since all th_{22} gates are reset to logic 0. An N-bit register stage, comprised of N single-bit dual-rail NCL registers, requires N completion signals, one for each bit. The NCL completion component uses these k_o lines to detect complete DATA and NULL sets at the output of every register stage and request the next NULL and DATA set, respectively. In full-word completion, the single-bit output of the completion component is connected to all k_i lines of the previous register stage (Smith, 2009).

Literature:

Existing synchronous fpm:

Synchronous FPM, utilizes IEEE 754 single precision binary format, to represent floating point numbers as shown in Figure 3.

Table 1: Dual-rail signal representation.

	DATA0	DATA1	NULL	ILLEGAL
D^0	1	0	0	1
D^1	0	1	0	1

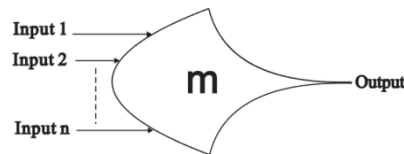


Fig. 1: THmn threshold gate.

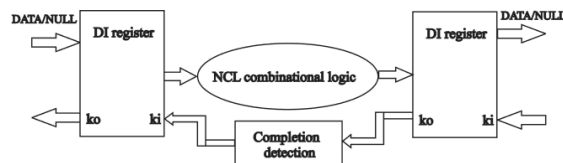


Fig. 2: NCL architecture.



Fig. 3: IEEE single precision floating point format.

The format consists of a sign bit (S), an 8 bit exponent (E) and a 23 bit mantissa (M). An extra bit is added to the MSB of the mantissa to form the significand. If the exponent ranges between 0 and 255, and there is a 1 in the MSB of the significand, the result is said to be normalized (Mohamed Al-Ashrafy et.al., 2011)

The real number is represented by the equation (1)

$$Z = (-1)^S * 2^{(E-Bias)} * (1.M) \quad (1)$$

where $m = m_{22}2^{-1} + m_{21}2^{-2} + \dots + m_12^{-21} + m_02^{-22}$
 and $Bias = 127$

$m_{22}, m_{21} \dots m_1, m_0$ represents the 23 mantissa bits.

Multiplication of two numbers in floating point format is done by (i) Addition of the exponent of the two numbers then subtraction of the bias from their result, (ii) multiplication of the significand of the two

numbers, and (iii) calculation of the sign by XORing the sign bit of the two numbers(Mohamed Al-Ashrafy et.al., 2011). In order to represent the multiplication result as a normalized number there should be 1 in the MSB of the result (leading

one).The result is normalized to obtain 1 at the MSB of the results' significand. The architecture presented in Figure 4 does not include rounding support to suit high precision applications.

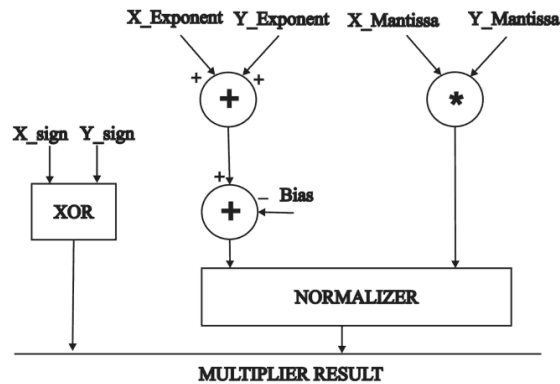


Fig. 4: Existing Synchronous floating point multiplier.

Non-Pipelined Ncl Fpm:

Non pipelined NCL FPM has been designed and demonstrated by Anitha Juliette and Seshasayanan(2015). The architecture of the non pipelined NCL FPM is shown in Figure 5. The fundamental components of the NCL FPM were designed using NCL combinational logic design

concepts. NCL XOR gate, NCL ripple carry adder, NCL ripple borrow subtractor, NCL mantissa multiplier, NCL exponent incremter and NCL intermediate product shifter are the NCL components that were needed to realize the NCL FPM. The components were integrated and sandwiched between the DI registers.

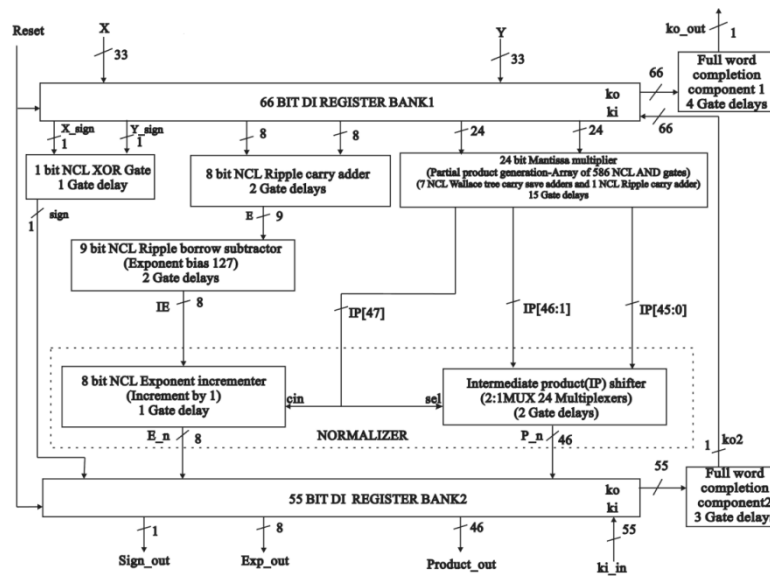


Fig. 5: Existing nonpipelined NCL floating point multiplier.

Existing Pipelined Ncl Architectures:

International Technology Roadmap for Semiconductors (ITRS) has predicted that semiconductor industry will have a smooth transition from synchronous to asynchronous designs, thereby, establishing the advantages of asynchronous designs such as low power, increased circuit robustness and decreased clock related issues. Researchers have proved that the dynamic power consumption of NCL designs is comparatively much smaller than the

conventional CMOS styles. However, the decrease in power is obtained only at the decrease of speed. Hence, techniques such as pipelining bit wise completion and NULL cycle reduction (NCR) with embedded registration have been used to increase the throughput of the design as shown in Table II. The existing NCL multiplier architectures support only non-fractional and fixed point fractional multiplication. The NCL multiplier proposed in (Anitha Juliette and Seshasayanan, 2015) performs

floating point multiplication. The architecture presents a predominant decrease in power when

compared with its synchronous version, at the expense of decrease in speed.

Table II: Comparison of existing pipelined and nonpipelined NCL designs.

NCL Circuits	Average delay (T_{DD}) in ns		Technique adopted for throughput optimization
	Non-pipelined	Pipelined	
Self Timed Multipliers (Bandapati.et.al.,2003)	9.21	3.84	Pipelined using full word completion
Multiply and Accumulate Unit(Smith, 2005)	12.7	8.6	NCR and pipelining using bit wise completion
NULL convention Arithmetic logic units (Bandapati and Smith, 2007)	7.75	5.89	NCR with embedded registration and pipelining using full word completion
8 X 8 2 ² complement Booth multiplier(Christina and Smith, 2010)(Sankar.et.,2007)	10	5.348	Pipelining using bit wise completion

Hence, we propose a pipelined NCL architecture that will perform floating point multiplication at a much higher speed than its non-pipelined counterpart. Higher throughput is achieved by pipelining the NCL floating point multiplier at component boundaries, using full word and bit wise completion strategies. Consequently, this paper focuses on the design and development of a completely pipelined, low power, comparatively high speed and high precision NCL floating point multiplier.

Design And Implementation:

The design of multipliers was performed at the gate level using NCL design methodology. We used

NCL VHDL Library, with delays based on physical level simulations of static gates using TSMC's 1.8V 0.18 μ m CMOS technology, retrieved from http://www.ndsu.edu/pubweb/~scotsmit/CCLI_async.html. The complete structural model of the proposed pipelined multiplier was verified for functional performance using an exhaustive VHDL test bench and the average delay time T_{DD} was obtained. The design was synthesized on to TSMC 1.8V 180 nm process technology libraries using Cadence and compared with existing synchronous FPM in terms of speed, power and area. The design flow utilized for the development and characterization of the completely pipelined floating point architecture is shown in Figure 6.

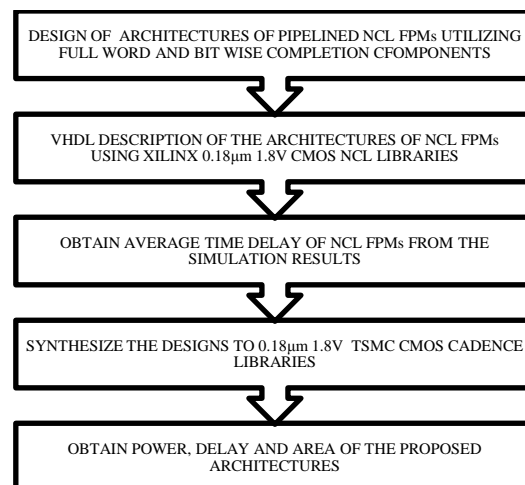


Fig. 6: Design flow.

To realize the proposed multiplier, the sub modules of the design such as NCL based XOR gate, 8 bit exponent adder, 9 bit exponent subtractor, 24 bit mantissa multiplier and normalizer were designed individually. It was ensured that the submodules are delay insensitive, input complete and possess hysteresis state holding capability (Smith, 2009). NCL systems can be optimized for speed by partitioning the combinational circuitry and inserting NCL registers and corresponding completion components (Smith, 2009). In order to preserve delay

insensitivity, NCL circuits should be pipelined at component boundaries alone (Smith, 2009). The combinational data path can be pipelined at any level of granularity. Since, the objective was to design a high speed NCL FPM, the data path was pipelined at the lowest level of granularity despite the additional DI registers used. Hence, a completely pipelined NCL FPM was realized by interleaving the mantissa multiplier to fine grain NCL architecture. The resulting structure was 10 levels pipelined NCL FPM as shown in Figure 10.. Full word completion and bit

wise completion detection components (Smith, 2009) are used to detect the completion of wavefronts in each stage. In full word completion component, the completion signal from register_i is sent back to all the bits of register_{i-1}. In bit wise completion component, the completion signal for bit b in register_i is sent back

to the bits in register_{i-1}, that took part in the calculation of bit b. The gate delay of the bitwise completion component is lesser than the full word completion component, thereby maximizing throughput of the design.

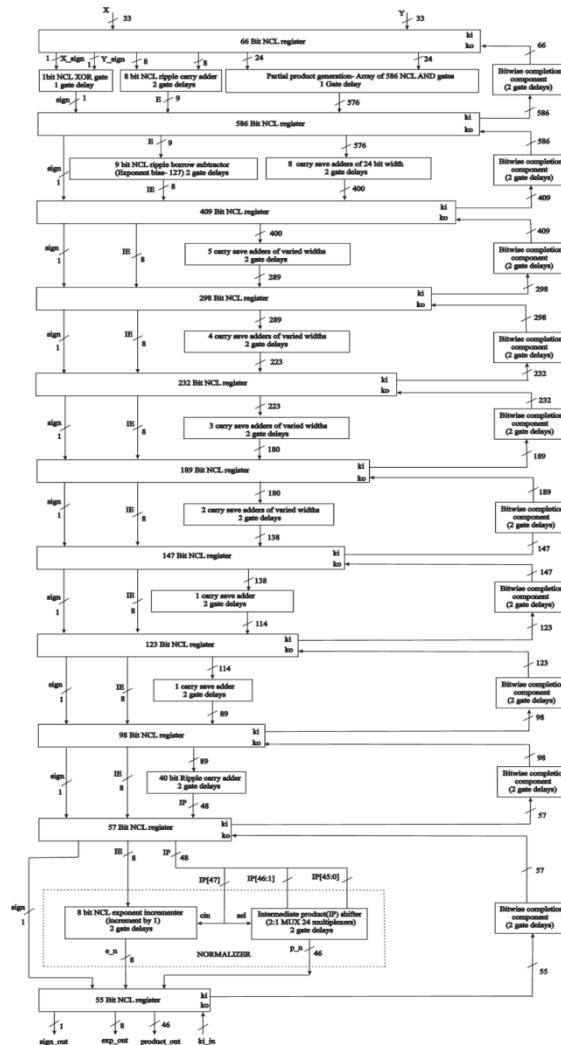


Fig. 7: Completely pipelined NCL floating point multiplier.

RESULTS AND DISCUSSION

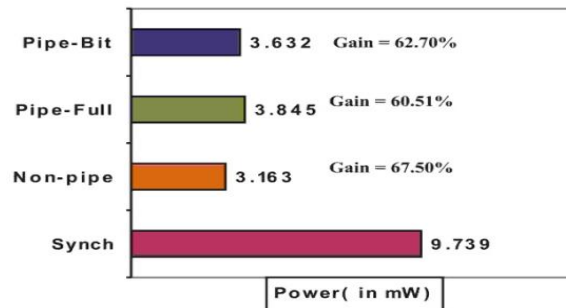
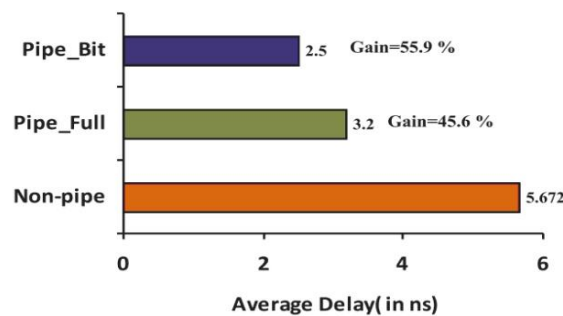
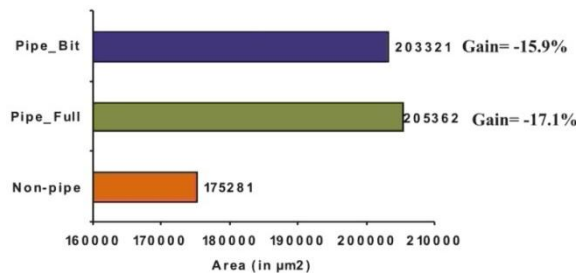
The VHDL description of the structural model of the pipelined (full word, bit wise) NCL FPMs were designed using gate delays based on physical-level simulations, with TSMC 1.8V 0.18µm static CMOS technology libraries. It was simulated on Xilinx ISE simulator using an exhaustive VHDL testbench that generates 233 x 233 possible input test vector combinations. At the simulation level, TDD is obtained as the arithmetic mean of DATA/NULL cycle times corresponding to all possible pairs of input operands. Full word and bit wise completion component pipelined NCL FPMs yielded TDD of 4.5 ns and 3.3 ns respectively. The proposed pipelined multipliers were synthesized to TSMC 1.8V 180 nm

process technology libraries. The pipelined NCL FPMs are compared with the non-pipelined NCL FPM and synchronous FPM in terms of power, speed and area, using the synthesis results obtained from cadence. The results are summarized in Table 3.

The throughput of the completely pipelined NCL FPM is realized by the stage with the largest T_{DD} (Zhou and Smith, 2012). The first stage of the pipelined NCL full word and bitwise completion NCL FPMs, comprising of NCL XOR gate, exponent adder and partial product generator contributed to the maximum T_{DD} of 3.2 ns and 2.5 ns respectively. The comparison of results in terms of power, throughput and area are depicted in Figures 8, 9 and 10 respectively.

Table III: Comparison of existing and proposed floating point multipliers.

Design	Completion strategy	Computation time $T_{DD}(ns)$	Power(mW)	Area(μm^2)
Synchronous FPM	-	0.39	9.739	67642
Non pipelined NCL FPM	Full word	5.672	3.163	175281
Pipelined NCL FPM	Full word	3.2	3.845	205362
Pipelined NCL FPM	Bit wise	2.5	3.632	203321

**Fig. 8:** Power analysis of existing and proposed floating point multipliers.**Fig. 9:** Throughput analysis of existing and proposed floating point multipliers.**Fig. 10:** Area analysis of existing and proposed floating point multipliers.

The proposed full word and bitwise pipelined FPMs have offered 45.6% and 55.9% increase in speed when compared to its non-pipelined version of NCL FPM as shown in Figure 9. Due to the utilization of more DI registers, the power consumption of the full word and bit wise completion pipelined NCL FPMs have increased by 18% and 15% respectively, when compared with the non-pipelined NCL FPM. However, when compared with the synchronous FPM, all versions of NCL FPM show a significant decrease in power as shown in Figure 8. The pipelined architecture presents 17.1% and 15.9% increase in area for the full word and bit wise completion strategies, when compared with

non-pipelined NCL FPM. Smith and Jia Di (Smith, 2009) have clearly stated that NCL based systems produce a significant decrease in power at the expense of increase in area, which is approximately 1.5 – 2 times as much as the equivalent synchronous systems. The disadvantage of the proposed multiplier is that the number of threshold gates required to realise NCL FPM components and DI registers contribute to the increase in area. However when realising SoC's, DSP processor cores which will include NCL FPM as one of the components, will generally require less than half of the entire chip area(Anitha Juliette and Seshasayanan, 2015). The remaining chip area will be occupied by flash, cache,

memory and peripherals which are the same for both synchronous and NCL designs. Hence the increase in area is comparatively less significant when compared to other advantages such as low power, elimination of clock related issues and lower electromagnetic interference (Smith, 2009).

Conclusion:

This paper has focused on developing a completely pipelined NCL floating point multiplier, by exploiting the advantages of pipelining, consequently leading to an improvement in the throughput. The proposed pipelined NCL floating point multiplier has exhibited a significant increase in throughput, decrease in power and high precision. Semiconductor industry is witnessing a rapid growth in design of asynchronous chips. The proposed multiplier can be used as a key library component in the design of NCL based floating point DSP processors.

REFERENCES

Anitha Juliette Albert, Seshasayanan Ramachandran, 2015. "Null Convention floating point multiplier". The Scientific World journal [in press].

Christina Smith and S.C. Smith, 2010. "Comparison of NULL Convention booth2 multipliers", International Conference on Computer Design- CDES, 3-9.

MohamedAl-Ashrafy, Ashraf Salem, Wagdy Anis, 2011. "An efficient implementation of floating point multiplier", *SIEPCPC*, pp: 1-5.

Sankar, R., V. Kadiyala, R. Bonam, S. Kumar, S. Mohan, F. Kacani, W.K. Al-Assadi and S.C. Smith, 2007. "Implementation of Static and Semi-Static Versions of a Bit-Wise Pipelined Dual-Rail NCL 2s Complement Multiplier", *Region 5 Technical Conference IEEE*, 228-233.

Satish, K., Bandapati, Scott C. Smith, 2007. "Design and characterization of NULL convention arithmetic logic units", *Microelectronic Engineering*, 84: 280-287.

Satish, K., Bandapati, Scott C. Smith and Minsu Choi, 2003. "Design and characterization of Null convention self-timed multipliers", *IEEE Design and Test of Computers: Special Issue on Clockless VLSI Design*, 30/6: 26-36.

Scott C. Smith and Jia Di, 2009. "Designing asynchronous circuits using NULL convention logic". *Synthesis Lectures on Digital circuits and systems*, vol.4,no.1,pp. 1-96

Smith, S.C., 2005. "Development of a large word-width, high-speed asynchronous multiply and accumulate unit". *Integration, the VLSI journal*, 39: 12-28.

Dr.ScottC.Smith:http://www.ndsu.edu/pubweb/~scotsmit/CCLI_async.html.

Zhou, L. and S.C. Smith, 2012. "Accurate Throughput Derivation of Pipelined NULL

Convention Logic Asynchronous Circuits ". *International Conference on Computer Design*, 51-54.