

## A Novel Approach to Implement A High Speed CMOS Parallel Counter Using Pipeline Partitioning

## Gunda Manasa<sup>1</sup>& T.Vilasini<sup>2</sup>

<sup>1</sup>PG Scholar, Dept of ECE, Sahasra College of Engineering For Women, Warangal, Telangana <sup>2</sup>Associate Professor, Dept of ECE, Sahasra College of Engineering For Women, Warangal, Telangana

## Abstract

A high-speed wide-range parallel counter that achieves high operating frequencies through a novel pipeline partitioning methodology (a counting path and state lookahead path) is proposed, and can be implemented using only three simple repeated CMOS-logic module types: an initial module generates anticipated counting states for higher significant bit modules through the state look-ahead path, simple D-type flip-flops, and 2-bit counters. The state lookahead path prepares the counting path's next counter state prior to the clock edge such that the clock edge triggers all modules simultaneously, thus concurrently updating the count state with a uniform delay at all counting path modules/stages with respect to the clock edge. The structure is scalable to arbitrary N-bit counter's delay is comprised of the initial module access time (a simple 2-bit counting stage), one three-input AND-gate delay, and a D-type flip-flop setup hold time. Thus the proposed counter can be implemented without AND gate and hence speed can be increased. The design can be implemented with Modelsim simulator. The parallel counter can give a maximum operating speed of 2GHz for 8-bit counter. Finally, the area of a sample 8-bit counter is 78 125 µm2 (510 transistors) and power consumption is 13.89Mw at 2GHz.

Keywords – High performance Counter design; Parallel counter design; Pipeline counter design

## **I.INTRODUCTION**

Counters are widely considered as essential building blocks for a variety of circuit operations such as programmable frequency dividers [1], shifters. code generators, memory select management, and various arithmetic operations. Since many applications are comprised of these fundamental operations, much research focuses on efficient counter architecture design. Counter methodologies architecture design explore tradeoffs between operating frequency, power consumption, area requirements and target application specialization. Early design methodologies improved counter operating frequency by partitioning large counters into multiple smaller counting modules, such that modules of higher significance (containing higher significant bits) were enabled when all bits in all modules of lower significance (containing lower significant bits) saturate. Initializations and propagation delays such as register load time, AND logic chain decoding [2], and the half incremented component delays in half adders operating frequency. dictated Subsequent methodologies improved counter operating frequency using half adders in the parallel counting modules that enabled carry signals generated at counting modules of lower significance to serve as the count enable for counting modules of higher significance, essentially implementing a carry chain from modules of lower significance to modules of higher significance. The carry chain cascaded synchronously through intermediate D-type flipflops (DFFs). The maximum operating frequency was limited by the half adder module delay, DFF access time, and the detector logic delay. Since the module outputs did not directly represent count



International Journal of Research (IJR) e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015 Available at http://internationaljournalofresearch.org

state, the detector logic further decoded the module outputs to the output count state value. enhancements improved Further operating frequency using multiple parallel counting modules separated by DFFs in a pipelined structure. The counting modules were composed of an incrementer that was based on a carry-ripple adder with one input hardcoded to "1". In this design, counting modules of higher significance contained more cascaded carry-ripple adders than counting modules of lower significance. Each counting module's count enable signal was the logical AND of the carry signals from all the previous counting modules (all counting modules of lower significance), thus pre scaling clocked modules of higher significance using a low frequency signal derived from modules of lower significance. Due to this pre scaling architecture, the maximum operating frequency was limited by the incrementer, DFF access time, and the AND gate delay. The AND gate delay could potentially be large for large sized counters due to large fan-in and fan-out parasitic components. Design modifications enhanced AND gate delay, and operating frequency, subsequently bv redistributing the AND gates to a smaller fan-in and fan-out layout separated by latches [4]. However, the drawback of this redistribution was increased count latency (number of clock cycles required before the output of the first count value). In addition, due to the design structure, this counter architecture inherited an irregular VLSI layout structure and resulted in a large area overhead.

#### **II.PARALLEL COUNTER ARCHITECTURE**

Fig. 1 shows the functional block diagram of the 8-bit parallel counter architecture. It consists of the state look-ahead path and the counting path. The counter is partitioned into uniform 2-bit synchronous up counting modules. Next state transitions in counting modules of higher significance are enabled on the clock cycle preceding the state transition using stimulus from the state look-ahead path. Therefore, all counting

modules concurrently transition to their next states at the rising clock edge. The counting path controls counting operations and the state lookahead path anticipates future states and thus prepares the counting path for these future states. There are three module types, module-1, module-2, and module-3 S, where S=1, 2, 3, etc. and represents the position of module-3 used to construct both paths.

1) **Counting Path:** Fig. 2 shows the hardware schematic of Module-1. It is a parallel synchronous binary 2-bit counter, which is responsible for low-order bit counting and generating future states for all module-3 S's in the counting path by pipelining the enable for these future states through the state look-ahead path. Module-1 and module-3 are exclusive to the counting path and each module represents two counter bits. In the counting path, each module-3 is preceded by an associated module-2. The output of module-1 is Q1Q0 and QEN1 connects to the module-2's DIN input.

Module-2 is a conventional positive edge triggered DFF and is present in both paths. In the counting path, it act as a pipeline between the module-1 and module-3 1 and subsequent module-3S. In state look-ahead logic module -2 placement operating counter frequency increases bv eliminating the lengthy AND-gate rippling and large AND gate fan-in and fan-out present in large width parallel counters [3]. Instead of the modules of higher significance are enabled by the module-3S and state look-ahead logic. Then the coupling of module-2 with module-3 1introduces an extra cycle delay before module-3 1 is enabled. Thus the module-2s in the counting path provide a 1-cycle look-ahead mechanism for triggering the module-3S's, and enabling the module-2s to maintain a constant delay for all stages.

**Module-3S's** serve two main purposes. Their first purpose is to generate all counter bits associated with their ordered position and the second purpose is to enable future states in module-3S's in [8]



International Journal of Research (IJR) e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015 Available at http://internationaljournalofresearch.org

conjunction with stimulus from the state lookahead path. Fig. 3 shows the hardware schematic of module-3S. It is a parallel binary 2-bit counter whose count is enabled by INS. INS connects to the Q output of the preceding module-2. It also provides one-cycle look ahead mechanism.



Fig. 1 Block diagram of 8-bit parallel counter



Fig.2 Hardware schematic of module-1



Fig. 3 Hardware schematic of module-3S

2) State Look-Ahead Path: The state look-ahead logic operation avoids the use of an overhead delay detector circuit that decodes the low order modules to generate the enable signals for higher order modules, and enables all modules to be triggered concurrently on the clock edge, thus avoiding delay and rippling. The state look-ahead logic is principally equivalent to the one-cycle look-ahead mechanism in the counting path. To enabling the next state's high order bits depends on early overflow pipelining across clock cycles through the module-2S in the state look-ahead path.

Fig. 4 shows a generalized counter topology for an N-bit counter state look-ahead path details. Module-2s in the state look-ahead logic are responsible for propagating the early overflow detection to the appropriate module-3S. Early overflow is initiated by the module-1 through the left-most column of decoders state-2, state-3, etc. Each module-2S early overflow pipelining chain is preceded by a small logic block state-X, where X denotes the number of clock cycles that the early overflow pipelining must carry through. Each State-X block consists of simple two input AND logic that decodes the module-1's output. Fig. 1. shows the internal logic for State-2 and State-3 respectively, and whose outputs QB1 and QC1 are connected to the module-2s DIN input, thus starting the early overflow pipelining exactly X clock cycles before the overflow must be detected to enable counting in a module-3S.



**International Journal of Research (IJR)** 

e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015 Available at http://internationaljournalofresearch.org



Fig. 4 Generalized N-bit counter showing state look-ahead path details

The state look-ahead path operates similarly to a carry look-ahead adder in that it decodes the loworder count states and carries this decoding over several clock cycles in order to trigger high-order count states. The state look-ahead logic is principally equivalent to the one-cycle look-ahead mechanism in the counting path. For example, in a 4-bit counter constructed of two 2-bit counting modules, the counting path's module-2 decodes the low- order state Q1Q0=10 and carries this decoding across one clock cycle and enables Q3Q2=01 at module-3 1on the next rising clock edge. This operation is equivalent to decoding Q1Q0=11 [7] and enabling Q3Q2=01 on the next immediate rising clock edge. The state look-ahead logic expands this principle to an X cycle lookahead mechanism. For example In a traditional 6bit ripple counter constructed of three 2-bit counting modules [6], the enabling of bits Q5Q4 happens only after decoding the overflow at Q1Q0 to enable Q3Q2 and decoding the overflow at Q3Q2 to enable Q5Q4. However, combining the one cycle look-ahead mechanism in the counting path for Q3Q2=10 and a two-cycle look ahead mechanism for Q1Q0=01 from can enable Q5Q4 Q1Q0=01 is pipelined across one cycle, thus enabling Q5Q4 at the next rising clock edge (further details will be discussed in section II-C). Thus, enabling the next state's high order bits depends on early overflow pipelining across clock cycles through the module-2s in the state lookahead path. This state look-ahead logic organization and operation avoids the use of an overhead delay detector circuit that decodes the low order modules to generate the enable signals for higher order modules, and enables all modules to be triggered concurrently on the clock edge, thus avoiding rippling and long frequency delay

#### **III.PROPOSED PARALLEL COUNTER ARCHITECTURE**

The proposed high speed parallel counter consists of two sections- counting section and state anticipation module. The counting section consists of three different modules. They are BCM, SCM1, and SCM2. The module BCM represents the Basic Counting Module. SCM1 and SCM2 represent the first and second Subsequent Counting Modules respectively.

The basic module *BCM* is a parallel synchronous 3-bit up counter using JK flip-flops . The module *BCM* is responsible for the three low-order bit counting and these three LSBs generate future states for counting modules SCM1 and SCM2 in the counting section. SCM1 is a two bit counting module and SCM2 is a three bit counting module. Similar to module BCM, JK flip-flops are used to realize the circuits of modules SCM1 and SCM2.

The State Anticipation Module(SAM) consists of three D flip-flops [5], three 3-input AND gates and two inverters. It decodes the count states of basic counting module BCM. This decoding is carried over two clock cycles through two DFFs to trigger the second subsequent module, SCM2.

#### **IV. RESULTS AND DISCUSSIONS:**

In order to illustrate our counter's parallel counting ability,Fig.5.1 depicts the simulation waveforms for the top-level design outputs (counter value Q7Q6Q5Q4Q3Q2Q1Q0) for several count iterations. We synthesized the HDL using Xilinx for an Spartan3 FPGA device running



**International Journal of Research (IJR)** e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015

Available at http://internationaljournalofresearch.org

at 250 MHz.The vertical axis shows traced logic values, while the horizontal time scale is represented in nanoseconds. All signal traces reflect the block diagram in Fig.5.1 and follow precise counter timing, for example Q0=CLKIN/2,Q1=CLKIN/4, and Q7=CLKIN/256.



Fig 5.1 Simulation Results of Parallel Counter

## **RTL Schematic**

The RTL (Register Transfer Logic) can be viewed as black box after synthesize of design is made. It shows the inputs and outputs of the system. By doubleclicking on the diagram we can see gates, flip-flops and MUX.

topmodule				
<u>clk</u>		<u>q0</u>		
		q2		
		q3		
		q4		
res		q7		
topmodule				

Figure 5.2 Schematic with Basic Inputs and Outputs



#### Fig.5.3 BLOCKS OF RTL SCHEMATIC

Using Xilinx ISE 8.1i synthesis is done and analyzed which is shown in Fig. 7. The synthesis report shows the number of flip-flops and IOBs (input-output blocks) used is very less. Also gives the parameters enhanced like power dissipation, delay and gate count for design.

Device Utilization Summary						
Logic Utilization	Used	Available	Utilization	Note(s)		
Number of Slice Flip Flops	23	4,896	1%			
Number of 4 input LUTs	8	4,896	1%			
Number of occupied Slices	17	2,448	1%			
Number of Slices containing only related logic	17	17	100%			
Number of Slices containing unrelated logic	0	17	0%			
Total Number of 4 input LUTs	9	4,896	1%			
Number used as logic	8					
Number used as a route-thru	1					
Number of bonded IOBs	10	92	10%			
Number of BUFGMUXs	1	24	4%			
Average Fanout of Non-Clock Nets	3.21					

# Fig. 5.4 Synthesis report of 8-bit parallel counter

#### V. CONCLUSION

In this paper, the counter design logic is comprised of only 2-bit counting modules and three-input AND gates. The counter structure's main features are a pipelined paradigm and state look-ahead path logic whose interpolation activates all modules concurrently at the system's clock edge, thus providing all counter state values at the exact same time without rippling affects. In addition, this



structure avoids using a long chain detector circuit typically required for large counter widths. An initial m-bit counting module pre-scales the counter size and this initial module is responsible for generating all early overflow states for modules of higher significance. In addition, this structure uses a regular VLSI topology, which is attractive for continued technology scaling due to repeated module types (module-2S and module-3S) forming a pattern paradigm and no increase in fan-in or fan-out as the counter width increases, resulting in a uniform frequency delay that is attractive for parallel designs. Consequently, the counter frequency is greatly improved by reducing the gate count on all timing paths to two gates using advanced circuit design techniques However, extra precautions must be considered during synthesis or layout implementations in order to align all modules in vertical columns with the system clock.

#### **VI. REFERENCES**

[1] S. Abdel-Hafeez, S. Harb, and W. Eisenstadt, "High speed digital CMOS divide- by-N frequency divider," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), 2008, pp. 592–595.

[2] M. Alioto, R. Mita, and G. Palumbo, "Design of high-speed power-efficientMOS current-mode logic frequency dividers," IEEE Trans. CircuitsSyst. II, Expr. Briefs, vol. 53, no. 11, pp. 1165–1169, Nov. 2006. [3] Altera Corp., Santa Clara, CA, "FLEX8000, field programmable gate array logic device," 2008.

[4] J. Ousterhout, Berkeley, 1980, "Berkley magic layout tools,"

[5] B. Chang, J. Park, and W. Kim, "A 1.2 GHz CMOS dual-modulus prescalar using new dynamic D-type flip-flops," IEEE J. Solid-State Circuits, vol. 31, no. 5, pp. 749–752, May 1996.

[6] M. Ercegovac and T. Lang, "Binary counters with counting period of one half adder independent of Counter size," IEEE Trans. Circuits Syst., vol. 36, no. 6, pp. 924–926, Jun. 1989.

[7] M. D. Ercegovac and T. Lang, Digital Arithmetic. San Mateo, CA:Mogan Kaufmann, 2004.

[8] N. Homma, J. Sakiyama, T. Wakamatsu, T. Aoki, and T. Higuchi, "A systematic approach for analyzing fast addition algorithms using counter tree diagrams," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), 2004, pp. V-197–V-200.

[9] B. Hoppe, C. Kroh, H. Meuth, and M. Stohr, "A 440MHz16 bit counter in CMOS standard