A Design of Synchronous Binary Counters Implemented Using Variable Clock for Real Time Applications

Chukka Prasanna[1], Dr. Kolisetty Rama Devi[2]

M. Tech Student[1], Assistant Professor[2]

Department of ECE, UCEK (A), JNTUK, Kakinada, India

ABSTRACT

In this work, a new fast structure for synchronous binary counting, which has a minimal counting period for practical counters. In many applications, a synchronous binary counter is necessary to be quick and handle a wide bit-width. However, most of the previous counters are associated with a limited counting rate due to large fan-outs and long carry chains, especially when the counter size is not small. In this paper, we first adopt a 1-bit Johnson counter to reduce the overall hardware complexity, and then duplicate the 1-bit Johnson counter to decrease the propagation delay caused by large fan-outs. Implementation results show that the proposed design can be realized with a small number of flip-flops, which is almost linear to the counter size thereby neglecting minimum delays and considering only the maximum delay path of the sub counters implemented in the proposed design. The effectiveness of the proposed method is synthesized and simulated using Xilinx ISE14.7.

Keywords : Backward Carry Propagation, Binary Counter, Constant-Time Counter, Pre-Scaled Counter.

I. INTRODUCTION

A counter is a basic component that can be found in a wide range of applications, including measuring systems, analog-to-digital converters, frequency divisions, phase-locked-loop frequency synthesizers, and so on. Due to recent advances in the applications, it is commonly required to implement a fast, wide counter supporting a constant counting rate doesn't depends upon the counter size. Moreover, the counting rate and the size conflict with each other, because the carry propagation from a low-order bit to a high-order bit becomes longer as the counter size gets larger.

The ripple carry chain in the traditional binary counter was replaced with a carry-look-ahead circuit in order to achieve a significant speedup. In addition, a hierarchical Manchester carry chain was used for carry propagation, and a state-look-ahead topology was used to break the carry chain by adding D F/F's, avoiding the rippling. In the carry chain was constructed with employing a tree structure. However, regarding a counter as a combination of an adder and a state register is not effective in achieving

a constant clock period, since the lower bound of the adder delay is not constant. There have been other efforts to speed up the counter by improving the F/F. For example, high-speed synchronous counters were developed by using the F/F based on the true single-phase clock.

If fast synchronous counting is only required instead of the binary sequence, a counter associated with a constant clock period can be achieved by employing a state generator. For instance, a pipelined carry propagation chain was presented by taking systolic structures, but it doubles the number of F/Fs required as well as the overall hardware complexity. To accomplish both constant delay and binary sequence, another carry propagation method called backward carry propagation was presented. It exploits the characteristics of a binary sequence that the more significant bits become high earlier than the less significant bits. This approach can be applied to achieve a constant-delay counter since the carry propagation is only determined by the least significant bit (LSB). However, the LSB has to drive all F/Fs of the counter, leading to a large fan-out problem. In other words, the number of input ports connected to the LSB exceeds the maximum value that can be derived by the LSB. In addition, another synchronous binary counter based on pre-scaling was presented.

A wide counter is partitioned into sub-blocks. The high-order block is enabled by a pre-scaled enable (PEN) signal generated from the low-order block, and the clock period of a pre-scaled counter is determined by the least significant block. However, there are still issues related to the large fan-out and the wide distribution of a PEN signal that is necessary to drive a large number of the write enable inputs of the F/Fs in the next block. The huge fan-out is in fact the critical issue to be solved in realizing a fast binary counter. As the counter size increases, the fan-out issue becomes more severe, leading to the longer propagation delay.

In this paper, we present a binary synchronous counter that operates with a constant delay for practical counter sizes ranging up to 128 bits. In the proposed counters, the large fan-out issue is mitigated by duplicating the one-bit Johnson counter and by applying the backward carry propagation method to get rid of the additional delay induced by the ripple carry propagation. The suggested counters achieves the highest counting rate, and the counting rate is determined only by the least-significant 1-bit counter regardless of the counter size. The following is a breakdown of the structure of this paper: Section 2 represents the process that has already been accomplished. Section 3 covers the proposed work. The simulation results for the proposed work is presented in Section 4. Section 5 deals with the conclusion.

II. EARLIER WORK

The n-MOD ripple counter can count 2n states, and then the counter resets to its initial value. Toggle mode is utilized with the flip flops. Just one flip flop receives the external clock signal. This flip flop's result is used as a clock signal for the succeeding flip flop. The flip flop through which the exterior clock pulse is transferred acts as the LSB in the counting series. Unlike an asynchronous counter, a synchronous counter uses a single global clock to drive every flip flop, resulting in concurrent output changes. But one benefit of a synchronous counter over an asynchronous counter is that it can function at a greater frequency since there is no cumulative delay so that each flip flop receives having similar clock. The block diagram of synchronous counter is as shown in fig.1
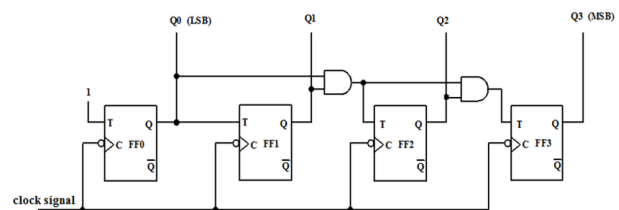

Fig.1: synchronous counter

Ring counters frequently use the Shift resister. The ring counter is virtually equivalent to the shift counter. The only difference is that the last flip-flop's

output is coupled to the first flip-flop's input in every ring counter, whereas it is utilized as an outlet in a shift resister. Except for that, everything else is the same. The number of states in the Ring counter equals the number of flip-flops utilized.

Switch-tail ring counter, wandering ring counter, and Johnson counter are all examples of twisted ring counters. It repeats a sequence of 1s, succeeded by 0s around the ring by connecting the counterpart of the final shift register's outcome to the first register's source. Figure 2 is a block diagram of the Johnson counter.
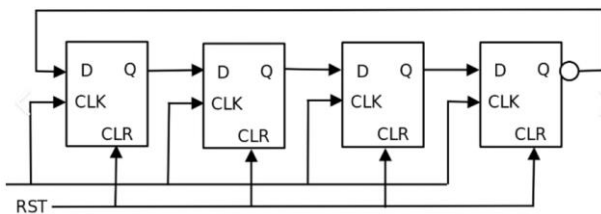

Fig.2: Johnson counter

Johnson counters are synchronous which sometimes referred as a creeping counter. To make an n-bit Johnson counter, the output of the last flip flop is connected to the input of the first flip flop. A popular type of shift register counter is this one. The feedback of the outcome to its own source gives it shape. A ring with a reversal is the Johnson counter. Johnson counters are sometimes referred to as creeping counters, twisted ring counters, walking counters, mobile counters, as well as switch tail counters. In an n-bit Johnson counter, the total number of used and unused states is: Number of used states=2n. Number of unused states=$2n - 2*n$

The Johnson counter has almost the same number of flip flops as that of the ring counter, but it would count double the number of states. This may be done with either D or JK flip flops. A Johnson ring counter is used to count input in a continuous loop. A self-decoding technology is the Johnson counter. Johnson's drawbacks are as follows: A binary series is not used by the Johnson counter. More states are unutilized than are used, according to the Johnson

counter. Flip flops only require half of the total number of timing signals. Any timing sequence can be built into it.

The Johnson counter is a synchronous decade counter or divider. In hardware logic circuits, it is utilized to create powerful Finite State Machines. Examples include ASIC and FPGA design. A 1200 phase shift three-phase square wave generator is being generated using a three-stage Johnson counter. The frequency of the clock signal is separated by varying its feedback.

A synchronous binary counter can be used to produce a reliable binary output. The ripple carry counter is the probably the easiest synchronous counter, that connects the carry-out of a one-bit adder to the carry-in of the next step. The carry pulse is continuously rippled into the following stage, giving rise to the term ripple carry chain. The carry chain's long carry propagation is the primary major bottleneck in the performance of a synchronous counter. Several approaches for fast adders have been devised, and some of them have been adapted to fast counters. In order to acquire a large speed boost, the classic binary counter's ripple carry chain was replaced with a carry-look-ahead circuit. Furthermore, for carry propagation, a hierarchical Manchester carry chain was employed, as well as a state-look-ahead architecture is being used to disrupt the carry chain by inserting D F/ F's, eliminating ripples. Using a tree - like structure, the carry sequence was built. However, because the lower bound of the adder delay is not constant, considering a counter as a combination of an adder as well as a state register is ineffective in establishing a constant clock period. Other attempts to improve the F/F have been made to speed up the counter. High-speed synchronous counters, for example, have been constructed utilizing the F/F based on a real single-phase clock. A state generator can be used to create a counter with a constant clock period instead of a binary sequence if only quick synchronous counting is required. By using systolic structures, for example, a pipelined carry propagation chain was developed, although it

doubles the quantity of F/Fs needed and increases overall hardware overhead. A linear-feedback shift register (LFSR) would be another way to build a state generator, so that requires a lot of extra circuitry to translate the state order to binary or rather make the number of states a powers of 2. Others carry propagation approach termed backward carry propagation was offered to achieve both constant delay & binary sequence.

Combining backward carry propagation with carry look-ahead and standard carry propagation. Phases A, B, and C are used to divide the counter. To minimize the maximum logic depth, the outputs of the bits in Phase A are connected to the carry look-ahead circuit in Reference 1 as shown later. As a result, the toggle signal for high order bits is created by connecting these bits to the input of each AND gate. Backward carry propagation is used in section B, and it operates like this: When Wb2 is set to high, the following clock pulse will cause b3 to switch. We initially combine b2 and bl in an AND gate because they are high before b0. This is paired with b0 to produce the b3 flip signal. The time sensitive AND gate that generates the toggle output has a low fan-in here, indicating high-speed capabilities. A backward ripple chain for each bit is needed in Phase B equipment. Because this is expensive, we suggest a new Phase C with a typical low-cost ripple carry chain to the right of bit. The high-speed data from phases A and B, unfortunately, must be injected. There's really early and late data, just like there is earlier. Phase C's ripple chain forms initially, after which phase B's bits get high, and ultimately phase A's bits rise. We apply the backward carry propagating concept once more, combining the data from the ripple carry chain with the outputs of phase B's rightmost chain firstly. This is then coupled with the phase A bit values to form a toggle output for every bit in phase C. Phase C will use 2 cascaded AND gates to build a tiny backward carry propagation sequence for every bit. The data transmission duration in a simple counter's ripple carry sequence is restricted to Z..,. T, wherein T

denotes the counter's clock duration. It exploits the characteristics of a binary sequence that the more significant bits become high earlier than the less significant bits. This approach can be applied to achieve a constant-delay counter since the carry propagation is only determined by the least significant bit (LSB). However, the LSB has to drive all F/Fs of the counter, leading to a large fan-out problem. In other words, the number of input ports connected to the LSB exceeds the maximum value that can be derived by the LSB. In addition, another synchronous binary counter based on pre-scaling was presented.

III. PROPOSED WORK

The proposed N-bit counter is illustrated in Fig.3 below. For the sake of simplicity, let we assume that $n = \lceil \log 2\ N \rceil$ and $m = \lceil (N - n)/L \rceil$, where L is the maximum fan-out to be determined by conducting simulations. An N-bit counter is partitioned into three different sub counters in order to take advantage of pre scaling, and m 1-bit Johnson counters are employed to generate m PEN signals to be used for the last sub counter. The Johnson counter is initialized to 0, and the PEN signal is generated to enable the counting of the next sub counter when the Johnson counter undergoes a state change from 0 to 1.
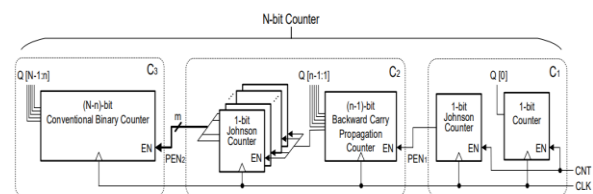

Fig.3: proposed N-bit counter

The backward carry propagation presented is an important concept in implementing a fast synchronous counter. What makes it work is the fact that a more significant bit of the counter becomes high earlier than the least significant bit due to the characteristics of the binary number system. Instead of a single chain used in the conventional binary counter, each counter bit has a separate AND chain

connected in the backward direction, as depicted in Fig. below. In a carry chain, the early arriving signals are evaluated in advance before the lately available signals arrive. The LSB's quickly changing pulse must be coupled to the carry chain's last AND gate, which means that the very last AND gate determines the critical path latency of backward carry propagation. As a result, the propagation delay is mostly dictated by the last AND gate's latency and a T F/F. As shown in Fig.4 below, however, the LSB, Q [0] in this figure, is connected to all the AND chains, having a large fan-out. In other words, the load of the LSB is too large to drive fast, making the critical path delay dependent on the fan-out.
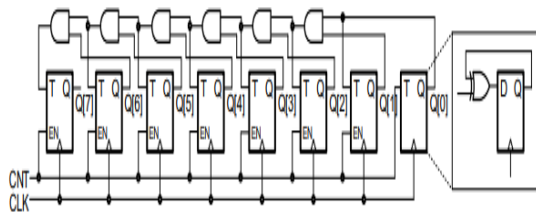

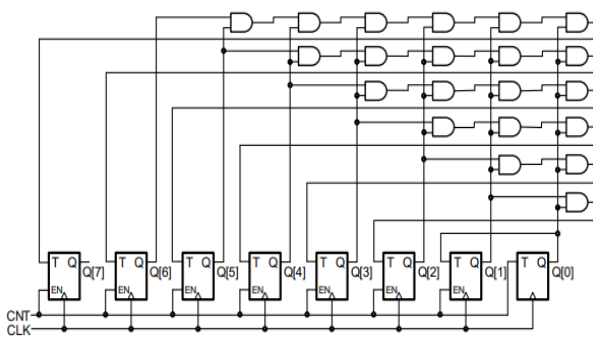Fig.4: Conventional 8-bit synchronous binary counter with a ripple carry chain

Fig.5: 8-bit synchronous binary counter designed with backward carry propagation.

A. Counter Block plays the role of a sequence generator that counts from 00...000 to 11…111. A counter is in general composed of a register part that stores the present state and a combinational incrementer that computes the next value. The counting rate is mainly limited by the computation time of the incrementer. The delay of the incrementer can be mitigated by pre-scaling. In the proposed counter architecture, an N-bit counter is realized by partitioning it into three sub counters, C1, C2, and C3, as shown in Fig.5 above.

Sub counter C1 is a 1-bit counter that toggles between 0 and 1 every clock. Sub counter C2 is an $(n-1)$-bit counter that works based on the backward carry propagation, and the last sub counter C3 is an $(N-n)$-bit conventional binary counter. The basic principle of the partitioned counter is to pre scale the high-order block by considering the low-order block. An N-bit counter is divided into 3 sub counters such that the propagation delay of the $(N-n)$-bit synchronous ripple carry binary counter C3, which consists of $(N-n-1)$ AND gates, is smaller than the period of PEN2 generated in C2. As the period of PEN2 is $2n$ clock cycles, the carry propagation in C3 can be stabilized before the next PEN2 arrives from C2. And sub counter C2 is an $(n-1)$-bit backward carry propagation counter, and enabled by the 1-bit counter C1. Observing that the delay of the long carry chain is reduced to only one AND gate by employing the backward carry propagation, we can guarantee that the carry propagation of C2 is shorter than the period of PEN1 generated in C1. As a result, we have only 3 sub counters, and the partitioning process is not recursively applied to the sub counters.

The carry propagation delay of C2 is just the summed delay of an AND gate and an XOR gate plus the loading delay of a D F/F. Note that the fan-out effect of the first bit is little enough to be negligible because the size $(n-1)$ is quite small. If the minimum clock period is set by considering the setup time of a D F/F additionally, the carry propagation delay of C2 is always faster than the period of PEN1 generated in C1, which is 2 clock cycles. As a result, the clock period is indeed determined by the least significant sub counter C1.

B. Pre scaled Enable Signal Generation In the pre scaled counter, the PEN signal should be synchronous with the clock and its delay caused by the fan-out be negligible. The typical method to generate the PEN is to use a ring or twisted-tail counter. The ring counter

connects the output of the last F/F to the input of the first one, making a circular structure. When the n-bit ring counter reaches 2 n–1 value, the PEN signal becomes 1.

Similarly, the n-bit twisted-tail counter or the Johnson counter, in which the inverted output of the last F/F is connected to the input of the first one, activates the PEN signal when the count value becomes 2 n–1. They can operate at a high frequency, as there is no combinational circuit between adjacent F/Fs, allowing the PEN to be synchronous with the clock. However, the approach is not efficient, as it needs N F/Fs to traverse N states, increasing the hardware complexity. Moreover, the PEN signal needs to drive all the F/Fs in the next partition, leading to a high fan-out and increasing the propagation delay and thus decreasing the overall counting speed.

For example, a 64-bit counter can be made of a 1-bit sub counter C1, a 5-bit sub counter C2, a 58-bit sub counter C3, a 2-bit ring counter generating PEN1, and a 64-bit ring counter making PEN2. The fan-out of PEN1 is small enough to be ignored. However, as PEN2 drives 58 enable ports in C3, the delay caused by the high fan-out should be considered in the design. As will be found in the simulation and implementation results later, the propagation delay of PEN2 induces the critical path, preventing the counter from having a minimal clock period. To deal with the fan-out issue, a 2 n -bit ring counter is replaced with an 1-bit Johnson counter as illustrated in Fig.6 below, where a 5-bit backward carry propagation counter and a PEN generator are exemplified for N = 64, n = 6 and m = 4. The 1-bit Johnson counter changes its state when enabled after being initialized to 0 at the beginning. Our goal is to make PEN2 have a pulse every 2 n cycles, 64 cycles in this example.
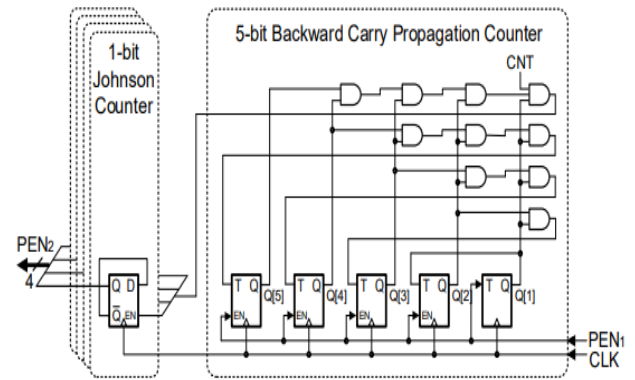

Fig.6: Pre scaled enable signal generation with redundant 1-bit Johnson counters

For the purpose, the enable signal should be high at the (2 n – 2)th and (2 n – 1)th cycles, the 62nd and 63rd cycles in the example, in order to make PEN2 being 1 at the (2 n – 1)th cycle, or 63rd cycle in the example. Such a signal can be generated by exploiting the backward carry propagation method depicted in Fig. 6. The AND operation of Q[5], Q[4], Q[3] and Q[2] can be realized by employing backward AND chains. The Q[5]&Q[4]&Q[3]&Q[2] signal becomes high when Q[2] undergoes a transition from low to high and lasts for four cycles.

The late arriving signal Q[1] is connected to the last AND gate to make the output of the AND chain high for two cycles. The enable signal is equivalent to the result of & Q[5:1], and the computation takes only one AND gate as &Q[5:2] is already computed in advance thanks to the backward carry propagation. The enable signal is high at the 62nd and 63rd cycles, and repeats periodically every 64 cycles. The PEN2 is inverted one clock cycle after the enable signal is asserted.

Consequently, PEN2 becomes high once every 64 cycles. In other words, PEN2 is equivalent to & Q[5:0] that is computed with the least 6 bits of the counter. The 1-bit Johnson counter can be redundantly duplicated to cope with large fan-out nodes. The number of redundant Johnson counters, m, is determined by evaluating m = ⌈$(N – n)/L$⌉, where L is the number of input ports that can be driven by a F/F. For N ranging from 8 to 128 bits, m is usually less

than 8. As the redundant F/Fs are 8 at maximum, the additional complexity caused by the redundancy is a small portion of the whole counter.

The maximum fan-out of a Johnson counter is set to 16 by conducting intensive simulations in order not to degrade the counting rate. All the Johnson counters are driven by the same signal and thus generate m identical PEN2 signals. Each PEN2 is distributed evenly to drive up to L F/Fs of the next sub counter C3.

PROGRAMMABLE CLOCK DIVIDER

In all other existing methods analog circuitry is used as voltage-controlled oscillators for dividing input frequency. The major demands of dynamic range requirement and speed constraints in phase detection system adaptive line selection based fully reconfigurable all digital frequency divider is proposed. The input sample rate driven rate-controller asserts delay line consists of a combination of inverters and gate chain. Based on unit delay for each logic cell chains are interconnected in order to meet the clock skew design specification. Therefore, a fine delay control scheme is employed.

· PLL is used for synchronous clock generation system to drive counter at variable rate.

· By using this model single source clock can be used for multi rate clock domains.

· Reconfigurable clock divider can able to divide source clock into any integer multiples.

Reconfigurable clock divider

Here, the rate of delay buffer chain depth transition is moderated according to the incoming clock rate and each unit in proposed phase estimation system run at its own speed. Input samples with all types of sampling rate are accommodated and its efficiency through concurrent data propagation process is validated using appropriate clock down sampling and synchronizing events.
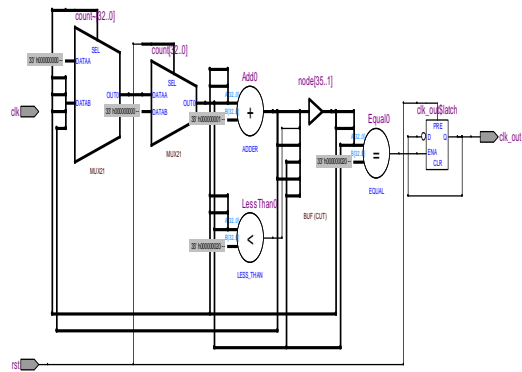

Fig.7 Clock divider unit path delay report

The optimal operating speed of buffer state transition can be determined based on incoming clock rate of the projected samples and adaptively optimization need to be configured using variable delay lines for the different clock-rate of input. As shown in fig.7 it can be observed that the synchronizer model leads to optimal critical path reduction during clock dividing and proved to be efficient in terms of throughput performance. In case of conventional PLL VCO for phase synchronization complex in nature but phase matching is proposed model is performed using reconfigurable delay lines and directly synthesized as buffers.
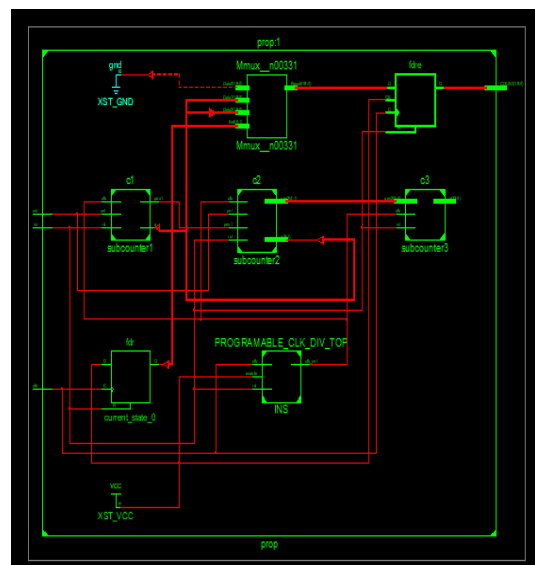
IV. EXPERIMENTAL RESULTS

Fig.8: RTL schematic

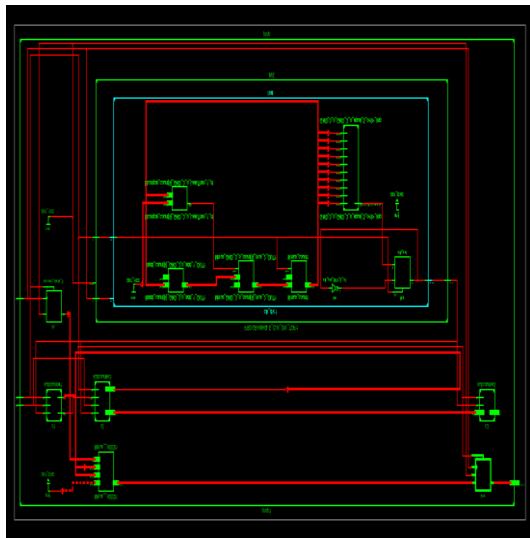Figure 8 shows RTL schematic of Proposed binary counter.

Fig.9: Technology schematic

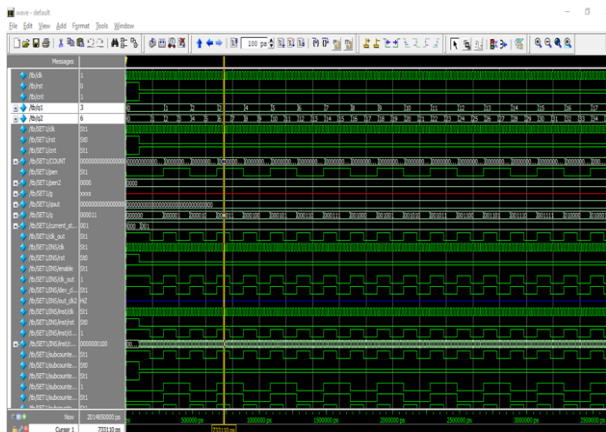Figure 9 shows Technology schematic of Proposed binary counter.

Fig.10: Simulation results

Figure 10 shows the simulation results for the proposed synchronous binary counter.

V. CONCLUSION

In this paper, we have proposed configurable synchronous binary counter architecture which can dynamically modified into any counter sizes. We implemented a variable rate constant synchronous binary counter using reconfigurable integer point clock divider. In this a variable clock frequency can be implemented, so that we can adjust the counting speed of the design. The proposed counter design also used variable rate clock divider which change the counter speed in accordance with different application requirements. The proposed counter are realized with a small number of flip-flops, which is a little higher than the counter size, which is almost independent of the counter size.

VI. REFERENCES

[1]. M.R. Stan, A.F. Tenca, M.D. Ercegovac, Long and fast up/down counters, Computers IEEE Transactions on, vol. 47, no. 7, pp. 722-735, 1998.

[2]. J. -. Yuan, Efficient CMOS counter circuits, in Electronics Letters, vol. 24, no. 21, pp. 1311-1313, 13 Oct. 1988, doi: 10.1049/el:19880891.

[3]. A. Ajane, P. M. Furth, E. E. Johnson and R. L. Subramanyam, Comparison of binary and LFSR counters and efficient LFSR decoding algorithm, 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), Seoul, 2011, pp. 1-4, doi: 10.1109/MWSCAS.2011.6026392.

[4]. S. Abdel-Hafeez, S. M. Harb and W. R. Eisenstadt, High speed digital CMOS divide-by-N frequency divider, 2008 IEEE International Symposium on Circuits and Systems, Seattle, WA, 2008, pp. 592-595, doi: 10.1109/ISCAS.2008.4541487.

[5]. M. Kondo and T. Watnaba, Synchronous Counter, U.S. Patent no. 5,526,393, June 1996.

[6]. P. R. Thota and A. K. Mal, A high speed counter for analog-to-digital converters, 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, 2016, pp. 1-5, doi: 10.1109/MicroCom.2016.7522592.

[7]. D. R. Lutz and D. N. Jayasimha, Programmable modulo-K counters, in IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 43, no. 11, pp. 939-941, Nov. 1996, doi: 10.1109/81.542285.

[8]. K. Z. Pekmestzi and N. Thanasouras, Systolic frequency dividers/counters, in IEEE Transactions on Circuits and Systems II: Analog

and Digital Signal Processing, vol. 41, no. 11, pp. 775-776, Nov. 1994, doi: 10.1109/82.331551.

[9]. D. Morrison, D. Delic, M. R. Yuce and J. Redouté, Multistage Linear Feedback Shift Register Counters With Reduced Decoding Logic in 130-nm CMOS for Large-Scale Array Applications, in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 27, no. 1, pp. 103-115, Jan. 2019, doi: 10.1109/TVLSI.2018.2872021.

[10]. P. Larsson and J. Yuan, Novel carry propagation in high-speed synchronous counters and dividers,in Electronics Letters, vol. 29, no. 16, pp. 1457-1458, 5 Aug. 1993, doi: 10.1049/el:19930975.

Cite this article as :