

Two Way Clock Scheme In Pipeline To Minimize The Clock Skew

GJCST Computing Classification
C.1.3, C.1.1

¹ N. Suresh Kumar, ²D.V.Rama KotiReddy, ³B. Pramod Kumar, ⁴A. Chaitanya Prabhu

Abstract- In most of the digital systems the clock skew decreases the performance of the digital systems in terms of providing good sensitivity or to maintain data synchronization. In the conventional pipeline system it is facing problems due to improper synchronization of clock pulses. This is a universal problem in all the digital systems mostly called jitter or skew. The propagation of information in the digital systems mainly controlled on the basis of clock pulses. In most of the digital systems the clock skew decreases the performance of the digital systems. Here a new system is implemented in the path of the clock to reduce the clock skew.

Keywords- Clock skew, Pipeline, Microcontroller.

I. INTRODUCTION

In conventional Pipeline technology the clock signal arrive at the first register take the same time as the data takes time to arrive the first register. But the clock signal

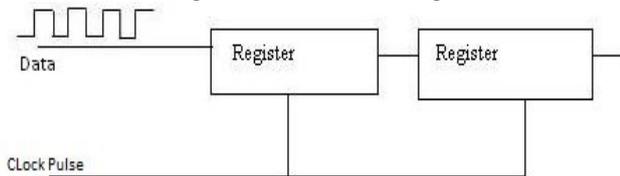


Figure 1 Conventional pipeline System

may take more time than the data to reach the second stage of the pipeline. So there may be a chance of data overlapping at the second stage of the pipeline. But the conventional pipeline system is facing problems due to improper synchronization of clock pulses. This is a universal problem in all the digital systems mostly called jitter or skew. In conventional pipeline systems the clock signal is derived as [2]

$$T_{clk_{conv}} \geq D_{max} + D_r + T_s + \Delta_{clk} \quad (1)$$

Where D_{max} is the maximum propagation delay D_r is clock-to-output delay of the pipeline register T_s , T_h is pipeline register setup and hold time. In the present work a new system is proposed in the path of the clock to remove or reduce the clock skew. There are already few methods effectively working on clock skew such as wave-pipelining [5] and Mesynchronous pipeline [1] methods. The idea of wave-pipelining [5] was originally introduced by Cotton [7], who named it *maximum rate pipelining*. Cotton observed that

¹ N. Suresh Kumar, ²D.V.Rama KotiReddy, ³B. Pramod Kumar, ⁴A. Chaitanya Prabhu

¹ Department of IT, GITAM University, Visakhapatnam, India

² College of Engineering, Andhra University, Visakhapatnam, India

³ Venkateswara college of Engineering, Srikakulam, India

⁴ Sri Sivani Institute of Technology, Srikakulam, India

Inskgitam2009@gmail.com

the rate at which logic can propagate through the circuit depends not on the longest path delay but on the difference between the longest and the shortest path delays. As a result, several computation —waves,” i.e., logic signals related to different clock cycles, can propagate through the logic simultaneously. The system clocking must be such that the output data is clocked *after* the latest data has arrived at the outputs and *before* the earliest data from the *next* clock cycle arrives at the outputs. Critical speed-limiting factors in wave-pipelining [5] are the uncontrolled clock-skew, the sampling time of registers, and the worst case transition time at the logic outputs. While the minimization of these factors has been a major challenge in the design of conventional high-speed pipelined systems as well, the equalization of path delays comes as a new challenge for the design of wave-pipelined systems. Different clock signal paths can have different delays for a variety of reasons [8]. Differences in delays of any active buffers within the clock distribution network may cause un-synchronization of data and clock in wave pipeline method.

Smaller clock periods are achieved in wave pipelining [2][5] by reducing the maximum propagation delay (D_{max}) by splitting the stages into number of stages. The clock signal is derived in the wave pipelining is

$$T_{clk_w} \geq (D_{max} - D_{min}) + t_h + T_s + 2\Delta_{clk} \quad (2)$$

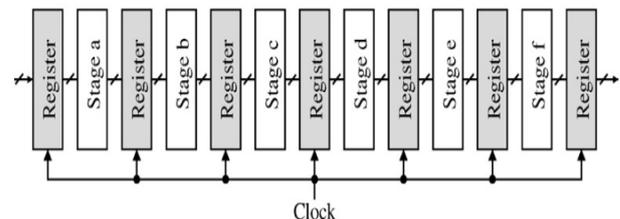


Figure 2 Wave Pipelining

And further the propagation delay in is reduced and the clock synchronization is controlled by introducing a delay element in the path of clock signal of Mesynchronous pipelining [1]. This delay will be equal to the delay created by the pulse passed from one stage to other stage of the pipeline. The system is clocked such that a pipeline stage is operating on more than one data wave simultaneously. At any given time, multiple waves can be present in a stage and the waves are separated based on physical properties of internal nodes in the logic stage. The clock signal is derived in the Mesynchronous pipelining is

$$T_{clk_m} \geq (D_{max(j)} - D_{min(j)}) + t_h + T_s + 2\Delta_{clk} \quad (3)$$

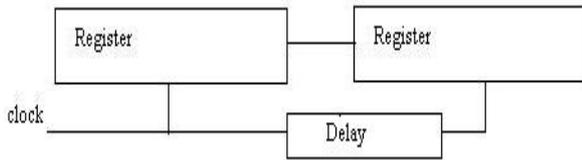


Figure 3 Mesynchronous pipelining

From equation 1, 2, 3 it implies that

$$T_{clk_m} \leq T_{clk_w} \leq T_{clk_{conv}} \quad (4)$$

It derives that mesochronous pipelining delivers an improved performance compared to conventional pipeline scheme.

II. ENHANCED METHOD

In the present work an external pipeline is developed using external registers and the working of these registers are controlled and observed by a microcontroller. The present system performance is almost similar to mesynchronous pipeline performance which will produce,

$$T_{2way\ clk_p} \leq T_{clk_w} \leq T_{clk_{conv}}$$

When a single clock pulse is applied to manage the data transmission through the registers in the pipeline. But it will create a clock skew in the pipeline which will decrease the data speed from one stage to other stage. The pulses from the encoder are fed into the first register when clock pulse is applied to the first stage of the pipeline. The pulse will be passed to the next stage after applying the clock pulse to the next stage. The clock pulse path is directly given to the registers where the encoder pulses are passes from one stage to another stage through flip flops of the registers. This may create a problem of overlapping of pulses in the first stage before it enters into the next stage.

In the present work a new method is proposed to introducing clock pulses alternatively to the parallel pipeline stages. Here two pipelines alternatively work to carry the data pulses. The two pipelines are selected alternatively by activating the clock pulses alternatively. The clock pulses are supplied by port pins of microcontroller instead of external clock generator. The port pins alternatively supply the clock pulses to the pipelines. The alternative supply of clock pulse will create a delay in the data propagation. This will synchronize the data propagation. In the register hold time of the first pipeline the register set time will be enabling in the second pipeline. Mean while the data will be output through the first register. While the register holds time of second pipeline the third clock pulse will be given to first pipeline. Now the next pulse from the encoder enters into first stage of first pipeline. And the out from the first stage of first pipeline will enter into second stage of first pipeline. So the encoder pulses enter into pipeline alternatively by changing the clock pulses from first pipeline to second pipeline vice versa. The clock signal is derived in the present system pipeline method as,

$$T_{2way\ clk_p} \geq (D_{max(j)} - D_{min(j)}) + t_h + T_s + 2\Delta_{clk}$$

The system is clocked such that pipeline stages are operating on more than one data pulse simultaneously. The present system at any given time, multiple data pulses can be present in a stage similar to the mesynchronous pipelining.

III. HARDWARE

In the present work two pipelines are individually connected and controlled through the port pins of microcontroller. Each pipeline has integrated with two stages. Each stage is connected with a single pipeline. These pipeline stages are alternatively controlled by p1.0 and p1.1 as shown in figure5.

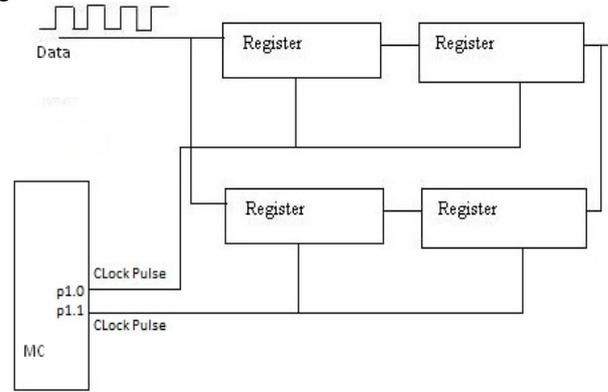


Figure 4 A New method to control parallel pipeline

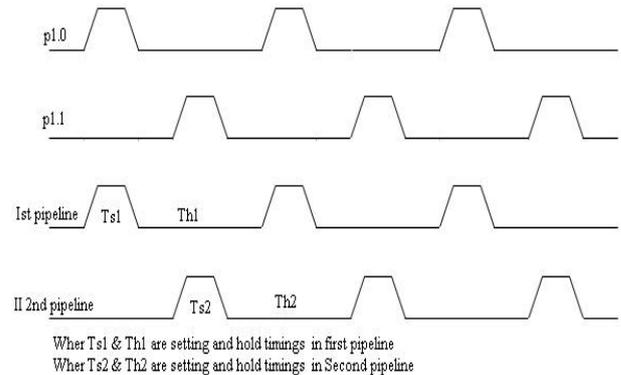


Figure 5 Data entering into first and second pipelines at alternate clock pulses applied through p1.0 and p1.1

IV. SOFTWARE ROUTINES

- Step1-Send high and low signals alternatively through parallel port to enable clock input to the pipelines
- Step2-Enable P1.0 to enable the first pipeline
- Step3-Disable P1.0 and Enable P1.1 to enable second pipeline
- Step4-Disable p1.1 and repeat step 2 to step 4 in the entire process.

V. CONCLUSIONS

The relative error of the frequency measurement almost zero when compare with other method measurements. The system can also measure the small readings which are nearer to zero value. The clock distribution becomes simpler by controlling clock signals by internal ports of microcontroller. Power consumption is small when compared to other methods. A simple control is implemented on the clock skew. Simple software logic is used to control the ports of microcontroller to generate the clock signals higher performance can be achieved using

smaller number of pipeline stages. The proposed pipeline scheme operates on more than one pulse simultaneously.

VI. REFERENCES

- 1) Suryanarayana B. Tatapudi, Student Member, IEEE and José G. Delgado-Frias, Senior Member, IEEE, "A Mesynchronous high performance digital systems", VOL. 53, NO. 5, MAY 2006
- 2) Thomas gay, "Timing constraints for wave pipelined systems" IEEE transactions on Computer aided design of integrated circuits, vol13, no.8, august 1994
- 3) Jabulani Nyathi, "High performance hybrid wave pipelined linear feedback shift register with skew tolerant clocks", IEEE, 1384- 1387, 2004
- 4) Mohammad Maymandi, "A digital programmable delay element: Design and analysis", IEEE transaction VLSI systems, Vol.11, no.5, October 2003
- 5) David E. Duarte, "A Clock Power Model to Evaluate Impact of Architectural and Technology Optimizations", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 10, NO. 6, DECEMBER 2002
- 6) Wayne P. Bureson, "Wave-Pipelining: A Tutorial and Research Survey", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 6, NO. 3, SEPTEMBER 1998
- 7) L. Cotten, "Maximum rate pipelined systems," in Proc. AFIPS Spring Joint Comput. Conf., 1969.
- 8) EBY G. FRIEDMAN, "Clock Distribution Networks in Synchronous Digital Integrated Circuits", Invited paper, PROCEEDINGS OF THE IEEE, VOL. 89, NO. 5, MAY 2001 665
- 9) Douglas V hall, "Microprocessor Interfacing: Programming and hardware", 2nd edition, TMH.
- 10) G.A.Gibson, "Micro Computer systems, The 8086/8088 family Architecture, PHI, 2nd edition
- 11) Barry B. Brey, "The Intel Microprocessors, 8086/8088, 80186/80188, 80286, 80386, 80486 and pentium processors, Architecture, programming interfacing"
- 12) Keneth J. Ayala, "8086 Microprocessor, Penram International/Thomson, 1995.
- 13) A.K. Ray, Burchandi, "Advanced Microprocessors and Pheripherals", Second edition, Tata McGraw-Hill.