Programmable Spread Spectrum Clock Generation Based On Successive Phase Selection Technique

Ruchir Saraswat*, Uwe Zillmann†, Supriyanto Supriyanto‡, Guido Droege§, Ulrich Bretthauer¶

Germany Microprocessor Lab, Intel GmbH, Braunschweig, Germany.

Email: *ruchir.saraswat@intel.com,† uwe.zillmann@intel.com,‡ supriyanto.supriyanto@intel.com,
§guido.droege@intel.com,¶ulrich.bretthauer@intel.com

Abstract— A wideband frequency modulator capable of modulating the input frequency within a range of 0.5% to 5% with a programmable step of 0.5% is presented. The block has been built in 65nm technology with a low power consumption of 1mW. The proposed topology uses a period locking delay locked loop, phase interpolator and a digital controller to modulate the input frequency and gives the flexibility of controlling the shape of the modulating signal.

I. INTRODUCTION

Spread-Spectrum Clock Generation (SSCG) [1] is a technique to reduce the radiated emissions of digital signals and its harmonics. It intentionally broadbands normally narrow band signal by frequency modulating the clock signal. This paper proposes a novel approach for spreading a clock external to the PLL using a delay locked loop (DLL) and a phase interpolator (PI). The proposed block provides several advantages over conventional approaches. Firstly, it works at a low frequency, 33 MHz to 133MHz with a modulation frequency of 32KHz (compatible to the Intel microprocessor design guidelines [3]) and thus consumes less power than a PLL based spread spectrum clock generator. Secondly, the block does not contain any regulation loop in contrast to conventional PLL based designs [5]. Thus, the jitter profile introduced at the input is equivalent at the output. Thirdly, different blocks can be connected after a single PLL to cater to different clock spreading requirements eliminating the need of multiple PLL's. Section II discusses the algorithm of the block, the circuit architecture and the simulation results in $65nm$ technology. Section III reports the top level simulation results in $65nm$ technology. Section IV concludes the paper.

II. DESIGN

A. Design Considerations

To cater to different spreading ratios required on a single platform in a computer, we have a operating frequency target ranging from 30 MHz to 133 MHz with a programmable spreading range from 0.5% to 5% at a step of 0.5%. The target technology for the design is 65nm at 1.05V supply and 25^oC temperature.

Different modulation profiles have been proposed, [1] [5] [4]. A staircase modulation profile as shown in fig.1(a) is used in the current design where each step corresponds to a frequency change. Each step causes a peak in the frequency spectrum as the maximum energy will concentrate at $\frac{df}{dt} = 0$
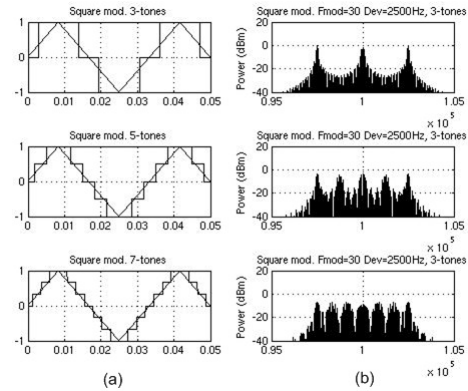

Fig. 1. (a) Modulation profile (b) Frequency spectrum of modulated clock

as shown in figure 1(b). An input frequency of 133 MHz with 5 modulation steps and 0.5% spread would require a resolution of 7.5ps. The higher the number of steps, the finer the resolution required. For the current case, we have 5 steps with each step corresponding to 0.5% frequency change.

B. Algorithm

Multiple phases of an input clock are generated such that the last phase is 360° out of phase (or in phase) with respect to the reference clock. The output clock is formed by selecting the delayed phases at every edge as shown in figure 2(a). If we keep on adding a delay of Δ_1 at every rising edge, a new stable time period of $T + \Delta_1$ is achieved. If there is an integer number g such that $g \times \Delta_1 = T$, then at g^{th} edge the input clock and the output clock edge will coincide. This is shown pictorially in figure 2(b) where each dot on the circle represents the change in phase. At the gth edge we complete a full circle. Anti-clock wise rotation corresponds to a $+\Delta$ delay leading to a decrease in frequency while a clockwise rotation corresponds to a $-\Delta$ delay leading to increase in frequency. Thus by rotating around the phase circle of figure 2(b), and controlling the Δ introduced a new stable frequency is generated.

C. Block Description

Figure 3 depicts major blocks of the design namely a period locking delay locked loop (DLL), a mux phase interpolator (MPI), a digital controller and a digital waveform generator.
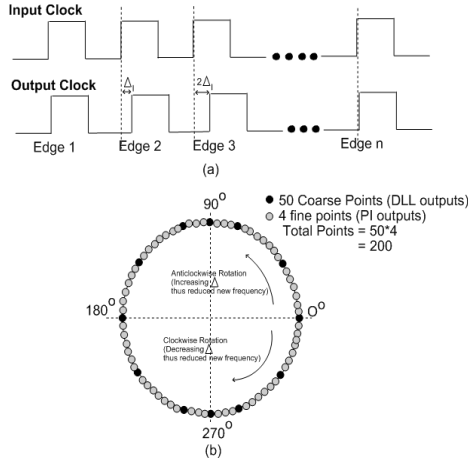
Fig. 2. (a) A addition of the Δ_1 to every edge leads to a change in the frequency by a constant amount (b) The coarse points on the phase circle correspond to the DLL output taps. These points are further subdivided into fine points by the phase interpolator. A anti-clockwise rotation corresponds to a positive Δ_1 while a clockwise rotation corresponds to a negative Δ_1

The DLL-MPI combination generates a delay Δ_t. A 133MHz input clock frequency with 5 step modulation profile would need a time step of 37.7ps to obtain a 0.5% frequency spread. To generate this delay using only one DLL, we would require $7.5ns/37.7ps = 200$ phases (delay stages) which leads to a large power/area consumption and mismatch problems. We introduced a phase interpolator to solve the problem. The DLL creates delayed phases which are represented by dark dots in the figure 2(b). The PI takes two adjacent phases from the DLL and creates equidistant finer delay phases. The current design implements 50 phases inside the DLL and the PI divides it into 4 finer phases resulting in total of 200 phases. The digital controller controls the switching of the Mux and phase interpolator.

1) Delay Locked Loop: The delay locked loop is a conventional period locking DLL and consists of a phase frequency detector (PFD), a charge pump (CP), a loop filter(LF) and a voltage controlled delay chain (VCDL) [6] with a modified delay stage implementation. We have used a differential based delay stage rather than the conventional CML stage as shown in figure 4 which leads to power savings. 50 phases are produced with a phase difference of $\frac{ckin_{time period}}{50}$. Due to
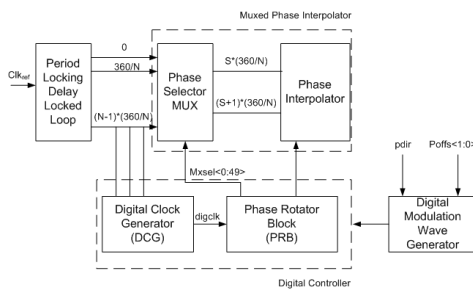
the wide range of operation and a long chain in the VCDL, the DLL is prone to false locking. A false locking prevention circuit has been incorporated for a fail safe operation.

2) Muxed Phase Interpolator (MPI): The delay locked loop (DLL) is followed by a MPI (figure 5). T_{O1} - T_{O5} performs the differential to single ended conversion. The MPI mixes two adjacent output phases from the DLL in a current steering circuit to generate the clock fineclk. T_{L1} - T_{L4} form the load to the PI. T_4 and T_7 forms the selection bit for the PI. T_1 and T_2 form the current steering legs of the PI. Each half of the current steering circuit is controlled by a 2 bit thermometer code generated by the digital controller, so four different phases (spaced by $\frac{\phi}{4}$) are produced. Since the transistors T_4 and T_7 are in saturation when selected, the sizing of the current steering legs follows equation (1)

$$\frac{W}{L}_{T_{6,3}} + \frac{W}{L}_{T_{7,4}} = \frac{W}{L}_{T_{8,5}} \qquad (1)$$

The top level system contains 50 PI-MUX legs. The magnitude of interpolation is controlled by the selected main branch(mb) set by sel$< 49 : 0 >$ and the next branch (nb) depending on the value of the intp¡1:0¿. (00=mb(100%)+nb(0%); 01=mb(75%)+nb(25%);10=mb(50%)+nb(50%);11=mb(25%)+ nb(75%)), where the % refers to the percentage of the total steered current. The bias voltage is generated by the bias generator in the DLL. Thus, the delay produced by the PI follows any variation in the DLL to achieve an automatic correction. As can be observed in figure 5 the current is steered into each of the legs to produce a delayed clock. The fineclk is then fed to the differential-to-single converter which does edge recovering as well as differential to single ended conversion of the output clock.

3) Digital Controller: The digital controller consists of Digital Clock Generator- DCG (custom logic) and the Phase Rotator Block- PRB (synthesized) blocks. The DCG generates a *digclk* which is fed to the PRB. As shown in figure 7, the DCG takes two clock differing by $\frac{\pi}{2}$. The $quad < 1 : 0 >$ defines the quadrant of operation in the phase circle (00 = quadrant I, 01 = quadrant II, 10 = quadrant III, 11 =

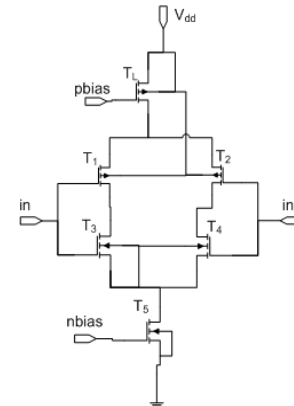Fig. 3. Top Level Block Diagram

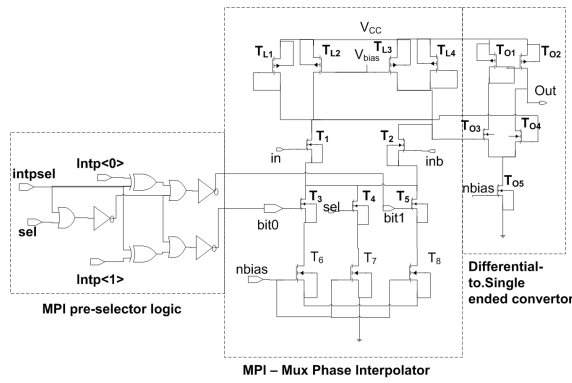Fig. 4. Voltage Controlled Delay Line Element
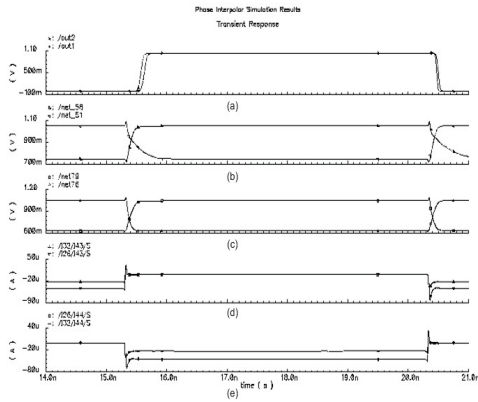
Fig. 5.   Muxed Phase Interpolator

Fig. 6.   Phase Interpolator Simulation Results. Single ended (a) Differential (b) (c) and Currents in the legs (d)(e) outputs corresponding to input Intp$< 0 : 1 >$ = 00 and Intp$< 0 : 1 >$= 01.

quadrant IV). During the quadrant change, due to the adjacent points there is a likelihood of a glitch. This is taken care of by creating an additional 45° phase addition/removal in the digclk inside the DCG every time the quadrant changes. in clockwise/anticlockwise direction, figure 8.

The PRB is a synthesized block and coded in Verilog. Table I describes the various pins of the PRB. Figure 9(a) depicts the delay between the reference clock and the output modulated clock by controlling the poffs1, poffs2 and the pdir input signals. Figure 9(b) shows the phase added into the clock while figure 9(c) shows the quadrature signals. Quad$< 1 : 0 >$ depicts the quadrant of operation of the system as defined earlier and shown in phase circle figure 2(b).

III. Top Simulation Results

A. Spectral Power Attenuation

Figure 10, 11 and 12 show the power spectral densities (PSD) of the output modulated clock for a 100MHz, 133MHz and 33MHz input unmodulated square pulse clock time domain transistor level simulations in 65nm process. It is interesting to note that in case of lower spreads there is an increase in the difference between the peak and the trough. This is due to the decrease in number of steps. For example, due to the fixed resolution of the delay defined by the DLL
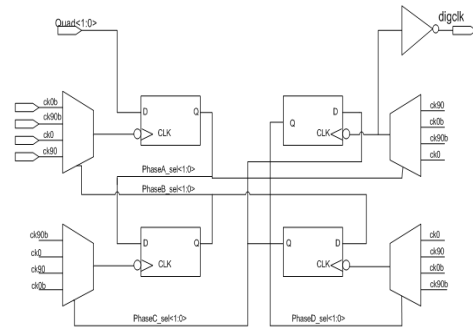

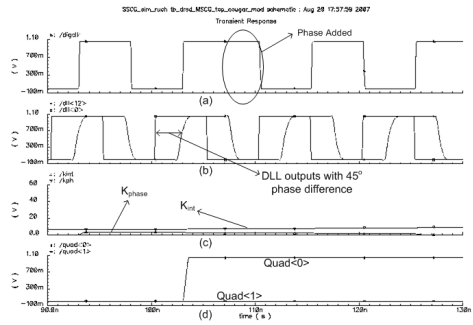Fig. 7.   DigitalClkGenerator. The digclk is generated depending on the quadrature of operation in the phase circle

Fig. 8.   Glitch Free Operation - Phase Added on quadrant change

TABLE I
PRB Pin Description

Pin	In/Out	Description
ckin	Input	Clock to PRB. This is shorted to digclk generated by the DCG
pdir	Input	0 = reverse rotation (increasing frequency). 1=forward rotation (decreasing frequency).
poffs1,poffs0	Input	Selects the amount of delay to be introduced in each step around the phase circle.
mxsel$< 0 : 49 >$	Output	Controls the Mux (sel pin of the PI) to enable the required PI.
Quad$< 1 : 0 >$	Output	Tells the DCG which quadrature the circuit is in the phase circle.
Intp$< 1 : 0 >$	Output	Forms the thermometer code to the PI.
kint,kphase	Output	Denotes the phase added to the input clock.

and the MUX-PI, we have a three step changes (symmetrical spread) for a 0.5% modulation while a 5% modulation has 5 steps. Thus, we have 3 distinct bigger peaks for 0.5% while 5 relatively flatter peaks for the 5% change.

B. Power Consumption

Table II shows the power consumption in the various blocks. The total power consumed is 1.11 mW for a 100 MHz reference clock. The power scales linearly with frequency. Table III compares a PLL based programmable SSC generator [2] with the current work. We expect that if the design [2] is
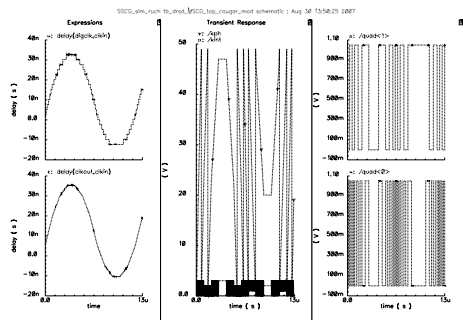
Fig. 9. (a) Delay between output clock and input clock (b) Accumulated phase in the output clock (c) quadrature signals defining the operational quadrature in the phase circle.

TABLE II
POWER CONSUMPTION (100MHz)

Block	Current Consumption(mA)	Power Consumption(mW)
DLL	0.7174	0.7533
MuxPI	0.02757	0.0289
Digital Clock Generator	0.2822	0.2896
Digital Controller + Waveform Generator	0.07646	0.0802
Total Consumption	1.10	1.1588

TABLE III
SUMMARY OF PERFORMANCE COMPARISON

SSCG	Triangular Modulation [2]	Step Modulation. This work
Process	0.18um CMOS	65nm CMOS
Supply Voltage	1.8V	1.05V
Input Frequency	1MHz-120MHz	33MHz-133MHz
Modulation Frequency	30KHz-50KHz	Independent
Power Number	100mW	1.15mW

scaled to 1.05V-65nm technology, the power might go down to about 20mW which is still an order of magnitude higher than the proposed solution.
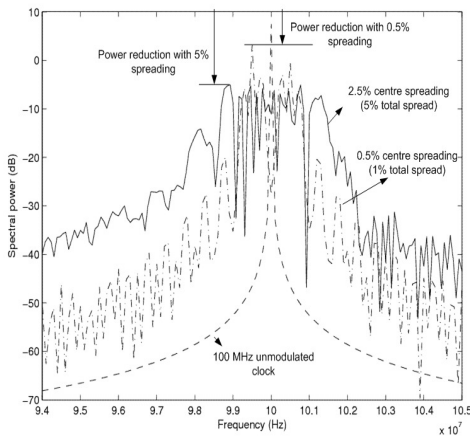

Fig. 10. Spectral power of (a) 100 MHz unmodulated clock (b)±0.5% spread clock (c) ±2.5% spread clock
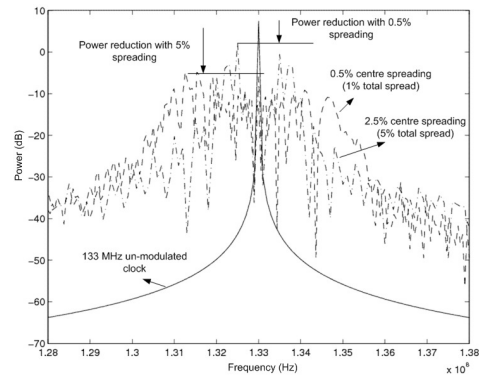

Fig. 11. Spectral power of (a) 133 MHz unmodulated clock (b)±0.5% spread clock (c)±2.5% spread clock
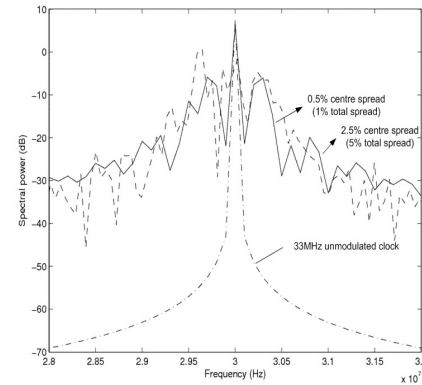

Fig. 12. Spectral power of (a) 33 MHz unmodulated clock (b)±0.5% spread clock (c)±2.5% spread clock

IV. CONCLUSION

The circuit implementation of a novel modular programmable spread spectrum modulator in 65nm based on new phase selection methodology is reported here along with the transistor level simulation results. Being a PLL independent solution, the power consumption, complexity and the accuracy are better.

REFERENCES

[1] K. Hardin, J. Fessler, and D. Bush, "Spread spectrum clock generation for the reduction of radiated emissions," *IEEE Int. Symp. Electromagnetic Compatibility*, pp. 227–231, 1994.
[2] H.-Y. Huang, S.-F. Ho, and L.-W. Huang, "A 64-MHz-1920-MHz programmable spread spectrum clock generator," *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 3363–3366, May 2005.
[3] Intel Corp., *Design for EMI-Application Note AP-589*, Feb. 1999.
[4] D.-S. Kim and D.-K. Jeong, "A spread spectrum clock generation PLL with dual-tone modulation profile," *IEEE Symposium on VLSI Circuits Digest of Technical Papers*, 2005.
[5] J. Kim, P. Jun, J.-G. Byun, and J. Kim, "Design guidelines of spread spectrum clock for suppression of radiation and interference from high-speed interconnection line," *IEEE Workshop on Signal Propagation On Interconnects*, pp. 189–192, 2002.
[6] J. Maneatis, "Low-jitter process-independent DLL and PLL based on self-biased techniques," *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 1723–1732, 1996.