Reducing Power Dissipation in Pipelined Accumulators

Gian Carlo Cardarilli[1], Alberto Nannarelli[2] and Marco Re[1]

[1] Department of Electronic Eng., University of Rome Tor Vergata, Rome, Italy
[2] DTU Informatics, Technical University of Denmark, Kongens Lyngby, Denmark

Abstract— Fast accumulation is required for units such as Direct Digital Frequency Syntehesis (DDFS) processors which, together with a digital to analog converter, generate periodic waveforms. In these units, waveforms with high frequency resolution are obtained if the clocking frequency of the digital processor is high (GHz range in today's technologies). Accumulators necessary for DDFS are then deeply pipelined down to the bit-level with two main consequences: high power dissipation, due to the large number of latches/flip-flops, and large latency dependent on the granularity of the applied pipelining. In this work, we address the two issues of reducing the power dissipation in the accumulator by applying selective clock gating, and reducing the accumulation latency by pipelining the adder to adapt the delay of the carry-chain to the necessary clock period.

I. INTRODUCTION

Direct Digital Frequency Synthesis (DDFS) is playing a role of growing importance in modern digital communications due to fast frequency switching, fine frequency resolution, large bandwidth, good spectral purity and fast evolution of the Digital-to-Analog Converter (DAC) technology. High performance DDFS requires very complex digital circuits at the cost of very high power consumption limiting its use in portable communication applications. In the literature, several papers have been presented to face low power consumption issues in DDFS. In [1], low power consumption is obtained by reducing logic resources (for example by reducing the size of the look-up-tables), and maintaining the DDFS performances. In [2] the authors show that most of the power in a DDFS circuit is used by the accumulator. Consequently, most of the research efforts have been focused in this direction. In [3] power consumption is reduced both by pipelining and parallelizing the accumulator (progression of state technique [4]). In [5], the power dissipation is lowered by identifying the redundant circuitry and by using a dynamic delay element. Moreover, a hybrid adder cell is also used.

The basic block of a DDFS is a fast accumulator. In this work, we review some of the basic concepts of design of DDFS and deeply pipelined accumulators. Then, we reduce the power consumption in the accumulator by applying selective clock gating [6].

Moreover, we apply tradeoffs between accumulator speed and latency and design accumulators with reduced pipelining depth.

The results of the implementations show that the power dissipation can be significantly reduced by clock gating and that by using radix-2^k adders in place of full-adders in the implementation of a systolic accumulator, not only reduces the latency, but also the power dissipation.

II. DIGITAL DIRECT FREQUENCY SYNTHESIZER BASICS

In this section, the DDFS architecture and its main design parameters are introduced. Moreover, some applications using DDFS are analyzed together with a short review of the DACs state of the art.

The general architecture of a DDFS is shown in Fig. 1. It is composed by a n bit accumulator whose rate of increment is controlled by the frequency word Δf clocked at F_{CLK}[1]. The function evaluation block, addressed by m bits ($m \leq n$, phase truncation) of the accumulator, computes a p bit sample of the output sinusoid and a DAC converts it in an ouput voltage. A low pass filter is used to reconstruct the sinusoid [7].

The basic parameters taken into account in the design of a DDFS are

1) **Frequency resolution:** it depends on the clock frequency and on the number of bits used in the accumulation loop:

$$f_{res} = \frac{F_{CLK}}{2^n} \quad [Hz]$$

where n is the accumulator number of bits.

2) **Spurious Free Dynamic Range (SFDR):** the spurious free dynamic range is affected by two major contributions: phase truncation and finite number of bits used to represent the sinusoid samples. The phase truncation is introduced in order to reduce the complexity of the function generator block (the number of locations in the case of a Look-Up Table (LUT) approach). The maximum number of bits used for the representation of the sinusoid output sample is related to the DAC number of bits. The SFDR is modeled by the following two equations

- **Phase truncation**:

$$SFDR = 6.02 \cdot m - 3.92 \quad [dB]$$

(worst case) i.e. the amplitude level of the spurious components in the output signal is reduced as the

---

[1]We indicate with upper case F the frequency of the DDFS clock, while we indicate with lower case f the frequency of the output waveforms.
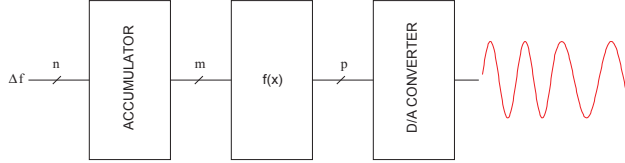
Fig. 1.   DDFS general architecture.

Company	DAC	channels	Speed	n.bits
Analog Devices	AD9779	1	1 GSPS	16
Fujitsu	MB86065	2	1.3 GSPS	14
Analog Devices	AD9736	1	1.3 GSPS	14
Maxim	MAX5881	4	4.3 GSPS	12

TABLE I

DAC: STATE OF THE ART.

number of bit used to address the function generation block increases.

- **Output Sample Quantization**:

$$SNR = 6.02 \cdot p + 1.76 \quad [dB]$$

By comparing these two formulas the value of m such that SFDR and SNR are equal is $m \approx p + 1$ (usually $m = p + 2$ is chosen). In this case the amplitude quantization error is larger than the phase quantization error.

3) **Output frequency:**

$$f_{out} = \frac{\Delta f \cdot F_{CLK}}{2^n}$$

Normally, the maximum frequency that can be obtained maintaining a good signal quality is $F_{CLK}/4$ (depending on the quality factor of the low pass filter at the output).

4) **Maximum latency:** the maximum allowed latency depends on the application.

A. Example of DDFS use

The Bluetooth specification requires a channel bandwidth of 1 MHz which is frequency hopped over 79 channels every 625μs, with frequency resolution of 1 ppm. The DDFS is a good candidate for the frequency synthesizer in a Bluetooth system as it is capable of generating signals of high frequency resolution, low phase noise and fast frequency hopping. Additionally the DDFS is controlled completely by digital signals providing a good interface to digital baseband systems.

B. State of the art DACs

The state of the art of commercial of the shelf DACs is illustrated in Table I. In these architectures when the data rate exceeds 1 GSPS[2] parallelism has been exploited. To support rates of 1 GSPS, LVDS signaling is used. In the following, a design example of a very high performance DDFS is shown to obtain a rough evaluation of the number of bits needed for the accumulator.

Example 1: 16 bit DAC, 1 GSPS, resolution 10^{-6} Hz → about 50 bits. If we have 16 bits representing the output samples, the maximum useful phase word is 18 bits.
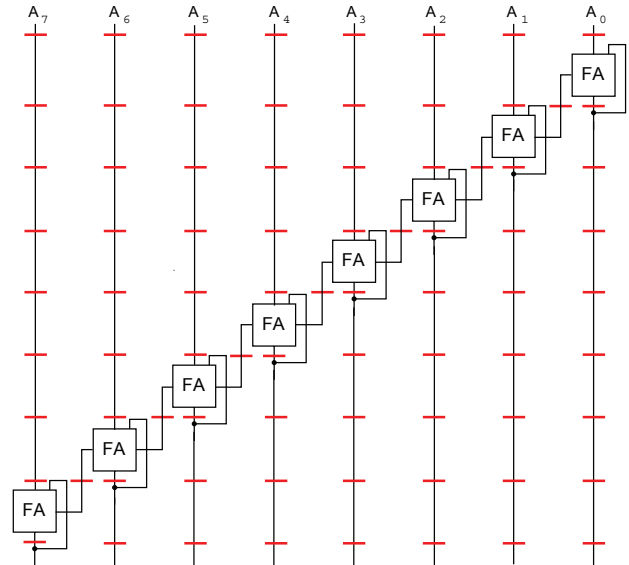
[2]Giga Sample Per Second.

Fig. 2.   8-bit pipelined accumulator.

Example 2: 12 bit DAC, 4.3 GSPS, resolution 10^{-6} Hz → about 52 bits. If we have 16 bits representing the output samples, the maximum useful phase word is 14 bits.

This examples shows that 50 bits accumulators generating frequencies at hundreds of MHz are frequently required in applications.

In this case pipelined adders are used. The pipelining can be exploited at different levels as a function of the maximum working frequency. The maximum speed is reached by pipelining at single bit level obtaining a systolic adder. The price to pay, in this case, is the high number of registers, and, consequently, power consumption. If pipelined adders are used, latency becomes an important issue that must be managed at compiler level. Low latency and very fast accumulators should be designed.

III. PIPELINED ACCUMULATORS

Pipelined adders have been extensively described in [8]. Here we focus on the configuration as an accumulator: one addend is the sum obtained at the previous iterations. The fastest accumulator can be obtained by pipelining the adder at full-adder (FA) level. An example for a 8-bit pipelined accumulator is shown in Fig. 2. In the figure, the thicker horizontal marks represent 1-bit registers (flip-flops in our case). Beside the first row of flip-flops (FF) which hold the value of the increment A (Δf in case of a DDFS), the circuit
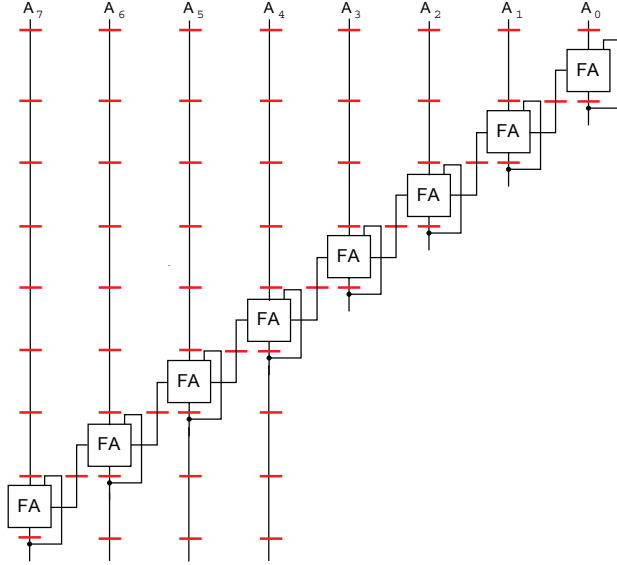
Fig. 3.   8-bit pipelined accumulator addressing 4-bit LUT.

Fig. 4.   Clock gating: enabling function.

is dominated by the large number of FFs. In the configuration of Fig. 2, the number of FFs for a n-bit accumulator is

$$n + n^2 + (n - 1)$$

where the first n holds the value of the increment A, n^2 are the FFs in array along the n columns, and $n - 1$ are the FFs necessary to store the carries of the accumulator.

Because, as previously explained, only m bits of the sum are used to address the LUT in a DDFS, it makes sense to eliminate the FFs corresponding to those bits not used in the table. This is shown in Fig. 3 for an 8-bit resolution 4-bit LUT accumulator ($n = 8$, $m = 4$).

By comparing Fig. 2 and Fig. 3, we can see that the number of FFs has been reduced by $(n + m - 1)(n - m)/2$. That is, 22 FFs less for the example shown in the figures.

IV. DESIGN FOR LOW POWER

From both Fig. 2 and Fig. 3 it is clear that a large portion of the energy dissipated by the accumulator is due to the FFs. Flip-flops consume power both when the data input switches and when it does not. In the latter case, the power is dissipated in the FF internal clock network consisting of buffers and wires. For this reason, even if the data input of a FF is stable over a long period of time, the FF dissipates dynamic power if the FF clock input switches.

A. Clock gating

To reduce the power dissipation in the FFs, we can disable the clock of FFs that do not change state. This technique is known as clock gating. Fig. 4 shows an application of the gated flip-flop technique [6]. We introduce the activation function F, that enables the clock of the flip-flop only when it is needed. As described in [6], F must be ANDed with the clock signal
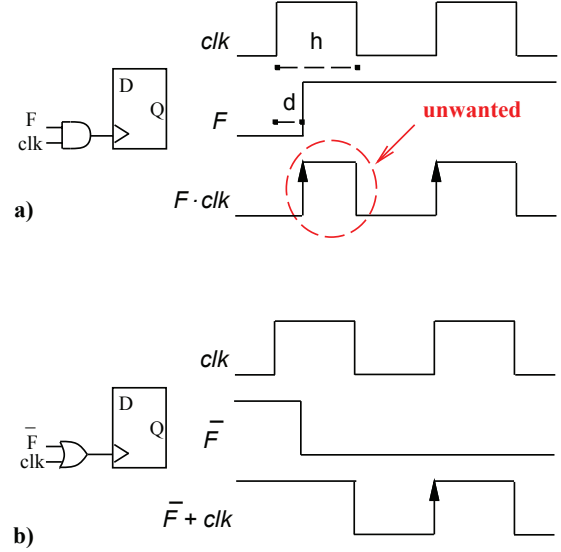
(clk) for trailing-edge-triggered flip-flops. For leading-edge-triggered (rising edge) flip-flops an AND gate cannot be used, to avoid a malfunctioning of the circuit if the delay (d) of F is shorter than the period the clock is high (h), as shown in Fig. 4.a ($d < h$). By making the flip-flop clock signal

$$cp = \overline{F} + clk \qquad (1)$$

we obtain the desired result for leading-edge-triggered flip-flops (Fig. 4.b). Note that the problem is still present if F changes when clk is low, but usually the delay d is shorter than the clock pulse width h. Expression (1) can be transformed by the De Morgan theorem into

$$cp = \overline{F \cdot \overline{clk}}$$

that is the NAND of the enabling function and the inverted clock (easy to obtain from the clock tree).

B. Applying clock gating to FFs holding increment A

The flip-flops located in the upper left triangle of the scheme of Fig. 2 and Fig. 3 have their data input change only when a new increment A is placed at the accumulator input. This value of the increment, that sets the f_{out} of the waveform in the DDFS, normally does not change too often, otherwise the output waveform will result highly distorted. We can apply clock gating to these FFs in the upper left triangle. We need to have an enabling function $F = 1$ when the increment changes. We assume that the value $F = 1$ is set by the same controller which sets the new increment. The scheme for clock gating is sketched for a portion of the accumulator in Fig. 5.

From Fig. 5, we can see that $n - 1$ extra NAND gates are required to generate the signals **clki** ($i = 1, \ldots, n - 1$). Moreover, we need to create a 1-bit delay line to signal the increment change $F = 1$ for the latency of the accumulator ($n - 1$).
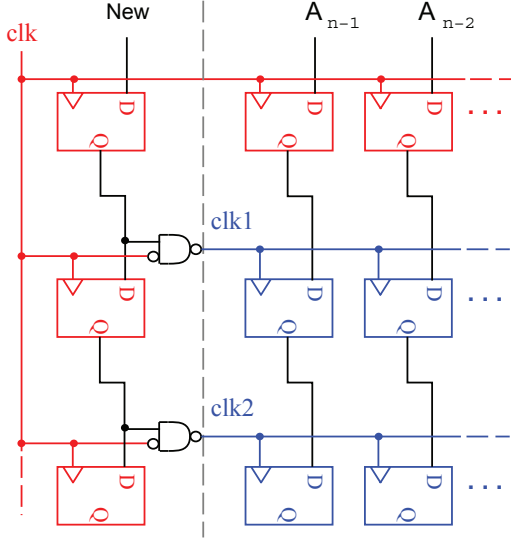
2100
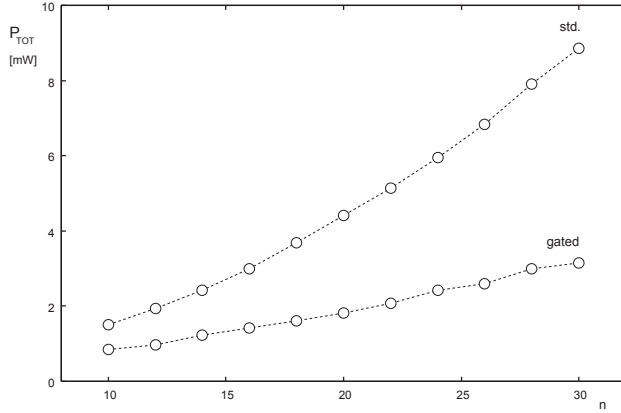
Fig. 5.   Clock gating in FFs holding increment A.

Fig. 6.   Power dissipation for accumulators at 1 GHz with $m = \frac{n}{2}$.

We applied clock gating to accumulator for different n (with $m = n/2$) and obtained the values shown in Fig. 6. In the experiment, we assumed Δf changed every 1,000 clock cycles (1 μs). By applying clock gating, we reduce the power dissipation for a 30-bit accumulator to about $1/3$ of that of the the standard pipelined implementation.

C. Individual clock gating

We can notice that the flip-flops below the diagonal of full-adders in Fig. 2 and Fig. 3 change their state when the accumulator sum increases. For small values of A the bits in the most-significant part of the adder do not change often. It might be reasonable to apply individual clock gating to those flip-flops. The enabling function for these FFs is shown in Fig. 7.

We show in Fig. 8 the reduction in total power dissipation obtained by applying individual clock gating to the FFs in the lower triangle of the accumulator for an accumulator with
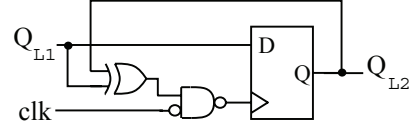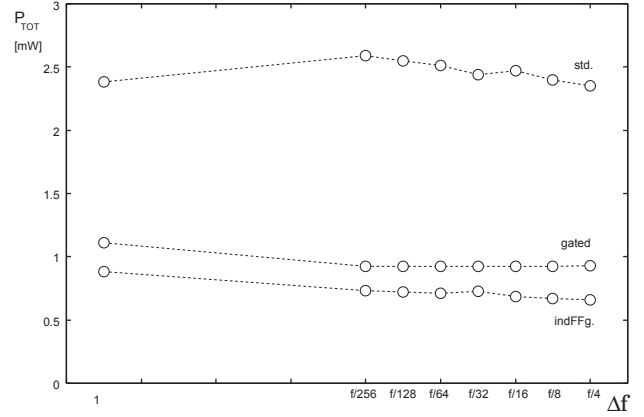

Fig. 7.   Individual FF clock gating.

Fig. 8.   Power dissipation for accumulator with $n = 16$, $m = 8$ at 1 GHz.

$n = 16$, $m = 8$. The different values in the figure are obtained for increments $\Delta f = 2^k$ with $k = \{0, 8, 9, 10, 11, 12, 13, 14\}$.

We can see from Fig. 8 that the reduction in power dissipation is not that high as in the case between standard and gated implementation, but still the extra gates (XOR and NAND) do not offset the saving obtained by turning off the clock signal when the FF's state does not change.

V. ACCUMULATORS WITH RADIX-2^k ADDERS

The application of clock gating to pipelined accumulators resulted in a significant reduction in power dissipation. However, the latency of systolic n-bit accumulators is n. We can try and reduce the latency by trading off the delay (minimum clock cycle) with the latency. Instead of storing the carry every bit, we can propagate the carry for a few bits and then store it. Fig. 9 shows how the latency can be reduced in a 8-bit accumulator by using radix-4 adders instead of full-adders. The delay of a radix-2^k adder, implemented with a carry look-ahead like scheme, is:

$$t_{r-2^k} = f(k) \approx c_1 \cdot k + c_0$$

and the critical path in an accumulator with radix-2^k adders is

$$t_{max} = t_{pFF} + t_{r-2^k} + t_{su}$$

where t_{pFF} is the propagation delay of the FF and t_{su} its set-up time.

Clearly the delay of a full-adder $t_{FA} < t_{r-2^k}$, but as long as

$$F_{CLK} = \frac{1}{T_{CLK}} = \frac{1}{t_{pFF} + t_{r-2^k} + t_{su}}$$

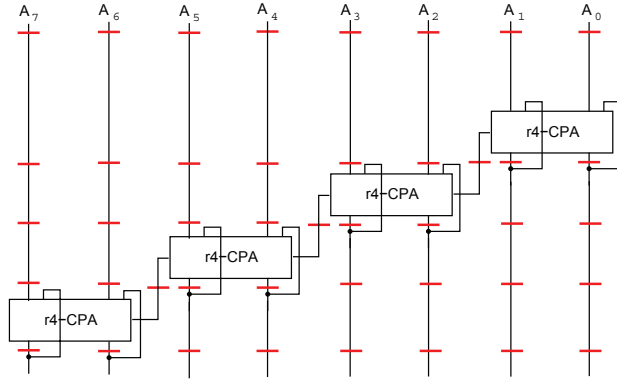the radix of the adder can be increased, and the latency decreases.

Fig. 9. 8-bit accumulator with radix-4 adders.

acc.	radix-2 P_{TOT}	radix-4 P_{TOT}	%	radix-16 P_{TOT}	%	radix-256 P_{TOT}	%
16-bit	3.13	0.25	8	0.12	4	0.06	2
24-bit	3.35	0.37	11	0.18	5	0.09	3
32-bit	3.38	0.51	15	0.25	7	0.13	4
	$[mW]$	$[mW]$		$[mW]$		$[mW]$	

TABLE II

RESULTS OF IMPLEMENTATIONS WITH DIFFERENT RADIX-2^k ADDERS.

For example, in the STM 90 nm library of standard cells [9], if we have $T_{CLK} = 1\ ns$, we can use up to radix-2^8 adders (propagate the carry 8 bits) to implement the accumulator.

The reduction in the number of FFs is significant. For an accumulator with radix-2^k adders the number of FFs is

$$n + \left(\frac{n}{k} - 1 \right) + n \cdot \frac{n}{k}$$

By comparing Fig. 2 and Fig. 9, the number of FFs is 79 for the systolic implementation and 43 for the radix-4 accumulator.

The reduction in the number of FFs has clearly an impact on the power dissipation as well. To evaluate the power reduction that can be obtained we implemented accumulators with the following characteristics:

1) 16-bit accumulator with $m = 16$
2) 24-bit accumulator with $m = 16$
3) 32-bit accumulator with $m = 16$

The three unit are implemented with radix 2 (systolic), $2^2 = 4$, $2^4 = 16$ and $2^8 = 256$ adders. The results, obtained for the same set of test vectors and with $F_{CLK} = 1\ GHz$ are reported in Table II.

From the table, we can see, for example, that when for the 32-bit accumulator we go from radix-2 to radix-256, the latency reduces from 32 to 4 and the power dissipation from $3.38\ mW$ to $0.13\ mW$, corresponding to only 4% of the power dissipated in the systolic implementation with clock gating.

VI. CONCLUSIONS

In this work we have addressed the issue of reducing the power dissipation in fast accumulators to be used in Digital Direct Frequency Synthesizers. To have high frequency resolutions, fast accumulators with a large word-length are required. To achive high speed for large word-lengths, accumulators are deeply pipelined with consequent large area and power dissipation. The power dissipation can be significantly reduced by applying clock gating to the flip-flops that do not change their state so often. Clock gating can be applied both to row of flip-flops that hold the accumulator increment, and to individual flip-flops that do not change their state often.

Moreover, if the speed requirements are not very tight, by implementing the accumulator with radix-2^k adders, both latency, area, and power dissipation can be greatly reduced.

REFERENCES

[1] A. Bellaouar, M. Obrecht, A. Fahim, and M. I. Elmasry, "A low-power direct digital frequency synthesizer architecture for wireless communications," *Proc. of IEEE Custom Integrated Circuits Conference*, pp. 593–596, 1999.
[2] F. Curticapean and J. Niittylahti, "Low-power direct digital frequency synthesizer," *Proc. of 43rd IEEE Midwest Symposium on Circuits and Systems*, pp. 822–825, Aug. 2000.
[3] B.-D. Yang, L.-S. Kim, and H.-K. Yu, "A high speed direct digital frequency synthesizer using a low power pipelined parallel accumulator," *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS 2002)*, vol. 5, pp. 373–376, 2002.
[4] M. Thompson, "Low-latency, high-speed numerically controlled oscillator using progression-of-states technique," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 113–117, Jan. 1992.
[5] M. Chappell and A. McEwan, "A low power high speed accumulator for ddfs applications," *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS 2004)*, vol. 2, pp. 797–800, 2004.
[6] T. Lang, E. Musoll, and J. Cortadella, "Individual flip-flops with gated clocks for low-power datapaths," *IEEE Transactions on Circuits and Systems*, June 1997.
[7] J. Tierney, C. M. Rader, and B. Gold, "A digital frequency synthesizer," *IEEE Transactions on Audio and Elettroacustics*, vol. 19, Mar. 1971.
[8] L. Dadda and V. Piuri, "Pipelined adders," *IEEE Transactions on Computers*, vol. 45, pp. 348–356, Mar. 1996.
[9] STMicroelectronics. 90nm CMOS090 Design Platform. [Online]. Available: http://www.st.com/stonline/prodpres/dedicate/ soc/asic/90plat.htm