第21卷第11期　　半　导　体　学　报　　Vol. 21, No. 11
2000年11月　　CHINESE JOURNAL OF SEMICONDUCTORS　　Nov., 2000

# A Novel Memory Compress Algorithm for Arbitrary Waveform Generator

LU Tie-liang(吕铁良) and QIU Yu-lin(仇玉林)

(*The Microelectronics R&D Center, The Chinese Academy of Sciences, Beijing* 100029, *China*)

**Abstract**: A memory compress algorithm for 12-bit Arbitrary Waveform Generator (AWG) is presented and optimized. It can compress waveform memory for a sinusoid to 16×13bits with a Spurious-Free Dynamic Range (SFDR) 90.7dBc (1/1890 of uncompressed memory at the same SFDR) and to 8×12bits with a SFDR 79dBc. Its hardware cost is six adders and two multipliers. Exploiting this memory compress technique makes it possible to build a high performance AWG on a chip.

**Key words**: arbitrary waveform generator; direct digital synthesis; memory compress algorithm

**EEACC**: 1230; 7250E

**CLC number**: TN402　**Document code**: A　**Article ID**: 0253-4177(2000)11-1075-05

## 1 Introduction

Arbitrary Waveform Generators (AWG) are widely used to generate stimuli for the under-testing devices, such as telecommunication system, magnetic storage devices and telemetry satellite. A modern high performance AWG is typically based on the direct digital synthesis (DDS) system, which consists of four major parts: the phase accumulator, the waveform lookup table, the digital to analog converter (DAC) and the anti-alias filter (LPF).

A typical AWG block diagram is shown in Fig. 1. The phase accumulator generates a phase sequence for waveform look up table (WLT), which contains the digital amplitude information of the output waveform; DAC converts the digital amplitude from WLT into analog waveform. As a DDS system is a quantized sampled system, the noises (quantization noise and aliasing) in the DAC output must be filterd by a LPF.

The advantages of DDS systems include fast switching time, smoothing frequency transitions, very fine output frequency resolution ($f_c/2^n$) and very low phase jitter.

LU Tie-liang (吕铁良) male, Ph. D. student. He was born in 1963. He is working on the design of high performance IC.

QIU Yu-lin(仇玉林) male, Professor. He was born in 1942.

Received 17 April 2000, revised manuscript received 29 June 2000

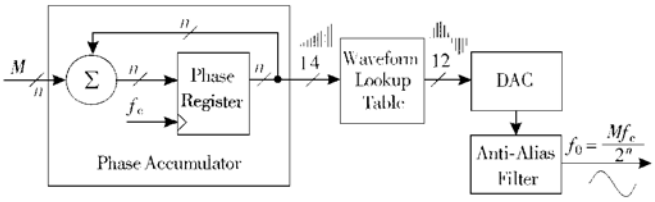

FIG. 1 Block Diagram of DDS System

The disadvantage of DDS systems mainly caused by the large volume of the WLT. A 12bits DDS system needs $2^{14}$ or $2^{15}$ words′ memory for the storage of a sine wave. As AWG is often applied to synthesize more complex waveforms than sinusoid, it is typically equipped several mega-words′ RAMs for its WLT. For an instance, in Chase Scientific Company′s AWG1200, the optional waveform memory is 1M words standard and up to 8M[1]. Such large volume RAMs make it very difficult to achieve a high system performance or reduce the system cost.

In recent years, the volume of WLT in DDS system has been reduced successfully for sinusoid synthesis[2—4]. Because the methods of generating waveform data in these studies are based on the characteristics of sinusoid, they can not be used to generate arbitrary waveforms.

In this paper, a high rate compress algorithm for arbitrary waveforms has been presented and optimized. The result shows that for 12-b sinusoid compression, the compression rate is over 1890∶1 with a spurious-free dynamic range 90.7 dBc which is nearly the ideal result for 12-b sine wave.

## 2 Memory Compression Algorithms

Let us suppose the DIFF interpolate algorithms:

On the identical distributed nods: $x_k(k=0,1,\cdots,n)$. Let the function $y=f(x)$ have the value $y_k=f(x_k)$. For an interpolate point $(x,y)$ over the interval$[x_m,x_{m+1}]$ $(m=0,1,\cdots,n-1)$, the function value $y$ can be calculated by polynomial:

$$\begin{aligned} y &= P_n(x)+R_n(x) \\ &= y_0+\frac{1}{h}\Delta y_0(x-x_0)+\frac{1}{2h^2}\Delta^2 y_0(x-x_0)(x-x_1) \\ &\quad+\frac{1}{6h^3}\Delta^3 y_0(x-x_0)(x-x_1)(x-x_2) \\ &\quad+\cdots+\frac{1}{(n!)h^n}\Delta^3 y_0(x-x_0)(x-x_1)\cdots(x-x_{n-1})+R_n(x) \end{aligned}$$

where, $h=x_k-x_{k-1}$

the first order DIFF is: $\Delta y_0=y_1-y_0$;

the second order DIFF is: $\Delta^2 y_0=\Delta y_1-\Delta y_0=y_2-2y_1+y_0$

……………

the $k$th order DIFF is:

$$\Delta^k y_0=\Delta^{k-1}y_1-\Delta^{k-1}y_0=\sum_{j=0}^{k}(-1)^j C_j^k y_{k-j}$$

the remainder is:

$$R_n(x) = \frac{f^{(n+1)}(\xi)^3}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n) \quad \xi \in [x_m, x_{m+1}]$$

As an approximation, we consider it linear, parabolic and cubic interpolation, when $n$ equals to 1, 2, 3, respectively, and the interpolation formulas become:

Linear:

$$y = y_0 + \frac{x - x_0}{h}\Delta y_0 \tag{1}$$

Parabolic:

$$y = y_0 + \frac{x - x_0}{h}\left[\Delta y_0 + \frac{x - x_1}{2h}\Delta^2 y_0\right] \tag{2}$$

Cubic:

$$y = y_0 + \frac{x - x_0}{h}\left[\Delta y_0 + \frac{x - x_1}{2h}\left[\Delta^2 y_0 + \frac{x - x_2}{6h}\Delta^3 y_0\right]\right] \tag{3}$$

## 3 SFDR of Interpolate Sinusoids

In many AWG applications, the spectral purity of the DAC output is of primary concern. For a 12bits AWG system, the interpolation algorithms can interpolate the phase points to $2^{15}$ or more. The SFDR effect due to the phase truncation error can be exclude. In interpolate sinusoids, the calculation error introduces additional noise. There are several factors affecting the calculation accuracy, including the compress algorithms, compress ratio and rounding.

Figure 2 shows the simulation result of compressed sinusoid SFDR vs the length of the sinusoid using the different compress algorithms. To minimize the complexity of memory managing circuit, the length of one cycle of the sinusoid's data table should be set to $2^n$, i. e. 32, 64, 128 etc. Under the condition of such a limitation with the cubic interpolate algorithm, the data table can be compressed to 32 points without any effects of SFDR; with the parabolic interpolate algorithm, it can be compressed to 32 points with about 6dB decline of SFDR; while with the linear interpolate, it can be done to 128 points.

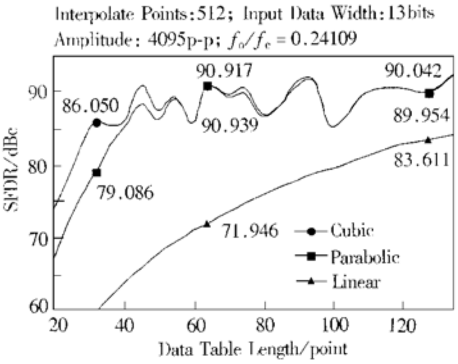

FIG. 2　Compressed Sinusold's SFDR vs Sinusoid Look Up Table Length.

The input and output data rounding induces the rounding noise. To minimize the hardware cost, the input data should be in a fixed-point format, which contains 1 LSBp-p's rounding error (peak to peak). During the interpolate calculation, the error would increase. After output data rounding, the error would grow to 1. 5—2 LSBp-p. The rounding noise can be reduced by increasing input data width, thereby reducing the input data error. As shown in Fig. 3, for a 12 bits DDS system, if the input data width is 12bits, the noise floor is near − 83dBc; if the input data width is 13 bits, the noise floor is about − 91dBc. At a high compress rate, the noise floor is determined by the calculation inaccuracy noise; increasing input data width cannot improve the spectral purity.

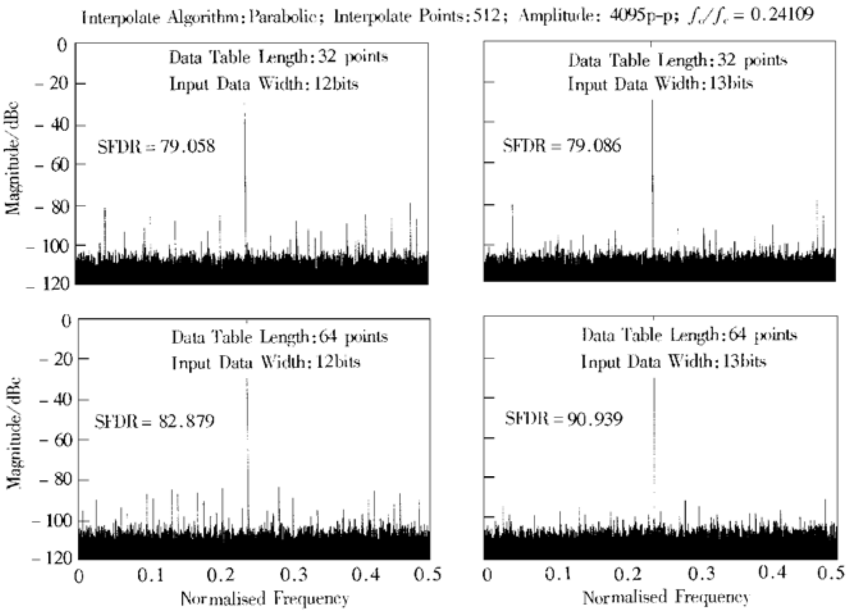

FIG. 3 Spectra of Interpolated Sinusoids

There has to be a trade-off between compress ratio and hardware cost in order to achieve the better system performance and lower hardware cost. Because the AWGs need more memories than DDS systems for the sinusoid, there should be an imbalance, in which more compress ratio is necessary. Among the compress algorithms, cubic interpolation gives the best SFDR at a high compress ratio, but the hardware cost is rather high: eight adders, three multipliers and one divider; the linear interpolation has minimum hardware cost but its compress ratio is only one-fourth of that of cubic interpolation. As a tradeoff, the parabolic interpolation having six adders and two multipliers can provide the same compress ratio as cubic interpolation do and has small effect on SFDR.

Although the hardware cost of parabolic interpolation is relatively low, the algorithm still need to be optimized. From equation (2), it is clear that the result of the data width of the first multiplier is doubled, as makes the second multiplier more complex. Because the contribution to the accuracy of second order DIFF is relatively small, rounding the additional bits of its product off does not deteriorate SFDR very much. Remaining one bit of its friction, as results in a good SFDR (as shown in Fig. 4), the multipliers can be simplified to an 8-b(signed) by 9-b(unsigned) and a 12-b(signed) by 9-b(unsigned).

## 4　Conclusion

A high rate memory compress algorithm has been presented, which sharply reduces the AWG's waveform look up table memory at low hardware cost. For an uncompressed waveform look up table, a phase precision of 14-b or 15-b can provide − 84.3 or − 90.3dBc of rejection of phase truncation noise accordingly. At − 90dBc spurious level, parabolic interpolation can provide compression rate of 472.6∶1. Another compression technique employed is the well-known quarter-wave symmetry compression. This technique is to store only $\pi/2$ rad of sine information, but the full range of $2\pi$ can be calculated by exploiting the symmetry of the sine function. Using this technique, the compression rate is over 1890∶1. Table 1 shows the information for comparison.

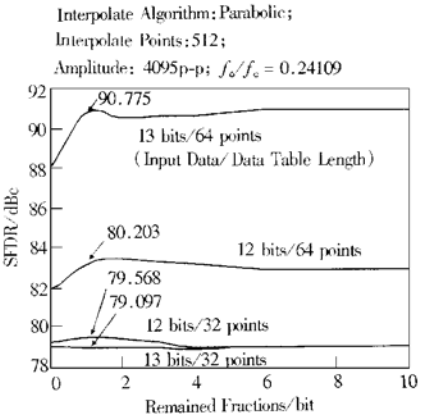

FIG.4　Calculation Rounding vs SFDR

**Table 1　Memory Compression Technique**

| | Memory for Sinusoid | SFDR (dBc) | Hardware Cost(not Includes Quarter Wave Logic*) | Note |
|---|---|---|---|---|
| Uncompressed Memory | $2^{15}\times$12bits | 90.3 | – | Reference |
| | $2^{14}\times$12bits | 84.29 | – | Reference |
| Coarse-Fine Segmentation[2] | $2^8\times$9bit+ $2^8\times$3bits* | 90.3 | Simple | Sinusoid only |
| Taylor Series Approximation[3] | $2^7\times$7bits+ $2^7\times$3bits* | 73.28 | 2 Adders<br>3 Multipliers | Available for Arbitrary Waveform |
| This Work | $2^4\times$13bits* | 90.78 | 6 Adders<br>2 Multipliers | Available for Arbitrary Waveform |
| | $2^4\times$12bits* | 80.20 | | |
| | $2^3\times$12bits* | 79.57 | | |

* Using Quarter-Wave Symmetry Compression

The AWG is more competitive and widely applied in many fields, compared with the traditional signal generator. Unfortunately, it's complex and the system cost is very high. Exploiting the high rate memory compress technique makes it possible to fabricate a high performance AWG on a chip and reduce the system cost to a large extend.

## References

[1] User Manual V1.3 for AWG1200, Chase Scientific Company, 1998.

[2] H.T.Nicholas and H.Samueili, IEEE J. Solid-State Circuits, 1991, **26**: 1959—1969.

[3] L.A.Weaver and R.J.Kerr, High Resolution Phase to Sine Amplitude Conversion, U.S. Patent 4905177, Feb. 27, 1990.

[4] Avanindra Madisetti and Alan Y. Kwentus, IEEE J. Solid-State Circuits, 1999, **34**: 1034—1042.