A VLSI Design for Digital Pre-distortion with Pipelined CORDIC Processors

Jong Kang Park[1], Jun Young Moon[2], Kyunghoon Kim[2], Youngoo Yang[1], and Jong Tae Kim[1,*]

Abstract—**In a wireless communications system, a pre-distorter is often used to compensate for the nonlinear distortions that result from operating a power amplifier near the saturation region, thereby improving system performance and increasing the spectral efficiency for the communication channels. This paper presents a new VLSI design for the polynomial digital pre-distorter (DPD). The proposed DPD uses a Coordinate Rotation Digital Computing (CORDIC) processor and a PD process with a fully-pipelined architecture. Due to its simple and regular structure, it can be a competitive design when compared to existing polynomial-type and approximated DPDs. Implementing a fifth-order distorter with the proposed design requires only 43,000 logic gates in a 0.35 µm CMOS standard cell library.**

Index Terms—**Power amplifier, pre-distorter, digital pre-distorter, CORDIC, pipelined pre-distorter**

I. INTRODUCTION

Conventional radio-frequency power amplifiers (PA) operating with wideband signals, such as that of wideband code division multiple access (WCDMA) and wireless local area network (WLAN), require high linearity due to spectral efficiency provided. PA linearity that is used in variable amplitude modulation methods is also important with respect to transmission errors and interference to adjacent channels [1]. Generally, linear PA designs have been limited by their low energy efficiency. The increase in power consumption that comes as a result of the use of such linear PAs is an important issue for battery-powered mobile devices. It would be clearly advantageous to manage the trade-offs between linearity and energy efficiency, if the non-linearity of PAs can be reduced via explicit signal processing techniques.

One effective solution to compensate for the non-linear response of the PAs is to distort the signals prior to arriving at the PA [2]. The cascade composed of this pre-distorter (PD) and PA give a linear characteristic to the output of the PA. For pre-distorting needs to have a mathematical model that can express the relation between the amplitude (AM)/phase (PM) of its input and output signals. Pre-distortion can be established by emulating the inversion of the characteristic function. A PD can be implemented as an analog or a digital circuit. The analog PD is designed with passive elements [3, 4] and is simpler than the digital PD. However, it cannot change its distortion characteristic and suffers from aging as well as from temperature drift. On the other hand, a digital PD (DPD) has flexibility in coping with environmental changes and the characteristics of the passive elements [1, 5]. While analog PDs are stationary, adaptive DPDs can be used to adjust the LUT (Look-Up Table) elements or polynomial coefficients that determine the specific pre-distortion characteristics as the signal changes. A DPD can be re-configured so as to conform to the various communications standards.

The polynomial-type DPD [6-8] has been implemented using analog or digital components. Direct

Manuscript received May. 5, 2014; accepted Oct. 24, 2014

[1] School of Electronic and Electrical Eng., Sungkyunkwan Univ., 300 Cheoncheon-dong Jangan-gu, Suwon, Gyeonggi-do 440-746, South Korea

[2] Dept. of IT Convergence, Sungkyunkwan Univ., 300 Cheoncheon-dong Jangan-gu, Suwon, Gyeonggi-do 440-746, South Korea

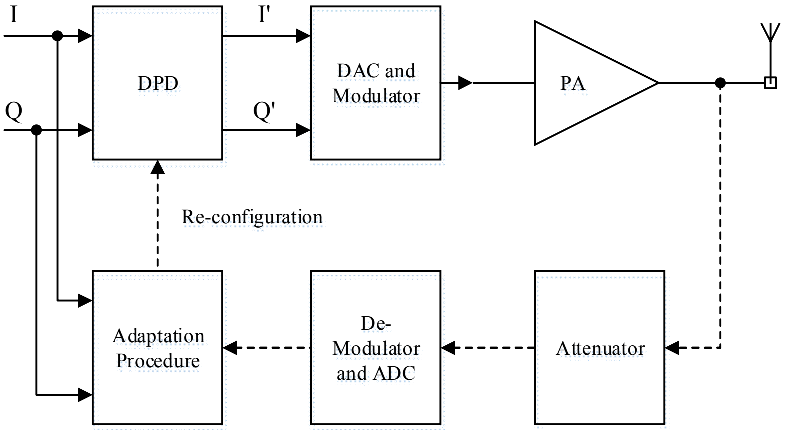
E-mail : jtkim@skku.edu

implementation of higher order polynomials to compensate for the strong non-linearity of the PAs requires increasing computational complexity with comparatively large mathematical units [1]. Piecewise linear approximation [9] enables the application of such higher order polynomials, but degrades the noise floor of the pre-distortion. In a simpler approach, DPD can be easily implemented via LUTs [10-13]. The LUT stores the pre-distorted AM/AM and AM/PM data of the PA that are mapped to the corresponding inputs. When hundreds of samples or more are stored, additional static memory blocks or register files are required, increasing the number of transistors needed considerably.

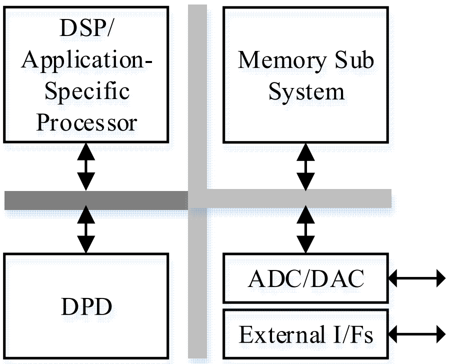
This paper introduces a new VLSI design for implementing pre-distortion polynomials in DPD. It employs fully pipelined processing elements including a Coordinate Rotation Digital Computer (CORDIC) processor to cover domain conversion between AM/PM and in-phase and quadrature (IQ) components. The CORDIC processor is used in calculating pre-distortion functions. Support for higher-order polynomial calculations can be achieved with its regular structure. In Section 3, we show that the implemented integrated circuit (IC) is more effective than a stand-alone high-performance digital signal processor (DSP) solution.

II. H/W Design for Digital Pre-distortion

DPD is typically used in a feed-back structure that can be adaptively adjusted to reduce the errors between the input and the output of the PA. Fig. 1(a) shows the block diagram for baseband DPD, IQ modulation, and PA. Through the use of the inverse characteristics of non-linearity of PA, the DPD distorts the IQ components of the signal prior to the IQ modulation and PA. The pre-distortion errors can be successively converged to zero by comparison to the PA output which is fed back to the



(a) Adaptive pre-distortion flow to compensate for the non-linearity of the PA [16]



(b) Implementation of the pre-distortion from a system-on-a-chip perspective

Fig. 1. A pre-distortion architecture with DPD in a communications system.

input of the DPD. This iterative adaption procedure might be flexible and can vary according to the communication standards and relevant system requirements. Therefore, hardware (HW) and software (SW) co-design would be a desirable choice for such systems, similar to that of Refs. 17-19. The calculation for the PD is performed by a sophisticated H/W block, and the DPD is adaptively configured and managed by the SW code on the DSP as shown in Fig. 1(b). In this paper, we focus on the efficient design of the calculation units for the DPD, using the results of our previous study [20].

1. Proposed Technique for Implementing Pre-distortion

The block diagram for the proposed pre-distortion block is shown in Fig. 2. I/Q-channel digital signal from the analog-to-digital converter (ADC) goes to the input of the DPD. This discrete data is then processed by each block in the following order: pre-processor, multiplexer (MUX), CORDIC processor in vectoring mode, de-multiplexer (DEMUX), PD, MUX, CORDIC processor in rotation mode, and post-processor. The internal pipeline of the CORDIC processor in Fig. 2 performs two modes of operations simultaneously. The arc-tangent and square root functions in vectoring mode executes AM-to-PM conversion and the rotation mode of operation in CORDIC processor converts PM signals to the AM domain by its trigonometric functions. The details for each block are described as follows:

The pre-processor adjusts the fixed-point format of the external data conforming to the inner data that is used in the CORDIC and PD blocks. Conversely, the post-processor changes the internal fixed-point data according to the output of the DPD.

The MUX passes the data to the first unit of the CORDIC processor and controls its operational mode, whether the data comes from the pre-processor or the PD block. If it is from the pre-processor, MUX gives the data to the CORDIC block for the vectoring mode. Otherwise the block would work in the rotation mode. DEMUX delivers the pre-distorted data to the post-processor or the non-pre-distorted data to the PD block.

The CORDIC processor performs the conversion between the I/Q data and the AM/PM signal. It can be also used to implement the PD function, but the experimental chip design has been done by using a sophisticated calculation block. The CORDIC algorithm and its pipelined architecture are explained in Section 2.3.

The PD block executes the pre-distortion of the AM/PM signal from the CORDIC processor. This will also be further explained in Section 2.4.

2. Fixed-point Design for DPD

When performing the arithmetic operations in the DPD, the binary scaling in a fixed-point format where the integer calculation is done with significant numbers is applied to the results of the operation. In this perspective, Fig. 3 shows the data flow and the relevant data format

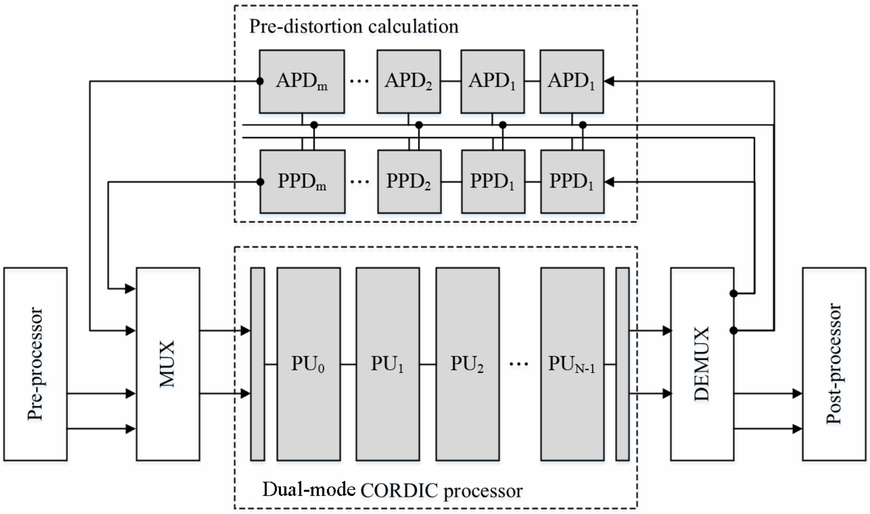


Fig. 2. Block diagram of the DPD.

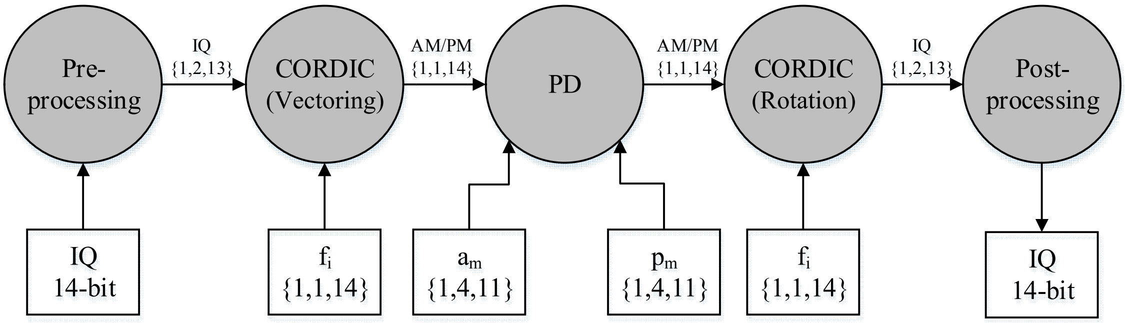


Fig. 3. Data flow and fixed-point design for the DPD.

used in the proposed DPD. The inputs of the DPD, which are the 14-bit ADC output data for the I/Q components, are expanded and formatted to 16-bit fixed-point data in the pre-processor. This format consists of one sign bit, two integer bits, and 13 fraction bits from the MSB to the LSB, and the format is marked as {1, 2, 13}. The data after the pre-process is of {1, 2, 13}, and the coefficients of the polynomial of the PD are used as {1, 4, 11} with a scaling factor, to be further explained in Section 2.4, in the form of {1, 1, 14}. The data that is reformatted by the pre-processor is converted to the AM/PM domain after applying the vectoring mode process of the CORDIC algorithm, and it is then distorted by the PD process. The data is then converted to I/Q form by the CORDIC rotation mode process and is finally changed to a 14-bit digital-to-analog converter (DAC) input format by the post-processor. The PA's input signal is then generated by the DAC.

3. Signal Domain Conversion by CORDIC Processor

CORDIC is a simple and efficient algorithm that calculates hyperbolic, trigonometric, and even linear functions [14]. Its pipelined design only requires adders, subtractors, and small LUTs to support the necessary functions. The generalized *i*-th CORDIC iterations are defined as follows,

$$\begin{aligned} x_{i+1} &= x_i - y_i \cdot \mu \cdot d_i \cdot 2^{-i} \\ y_{i+1} &= y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} &= z_i - d_i \cdot f(i)\,. \end{aligned} \tag{1}$$

Assume that X, Y, and Z are the final results after the CORDIC iterations where d_i, $f(i)$, and μ determine the modes of operation, circular/linear/hyperbolic and rotation/vector modes. The iteration starts with initial values, x_0, y_0 and z_0 and it ends with the final values X, Y, Z, respectively. Eq. (2) shows the results of the circular rotation mode for the CORDIC algorithm where K denotes a constant value.

$$\begin{aligned} X &= K(x_0 \cos z_0 - y_0 \sin z_0) \\ Y &= K(y_0 \cos z_0 - x_0 \sin z_0) \\ Z &= 0 \end{aligned} \tag{2}$$

Eq. (3) shows the results of the circular vectoring mode for the CORDIC algorithm.

$$\begin{aligned} X &= K\sqrt{(x_0{}^2 - y_0{}^2)} \\ Y &= 0 \\ Z &= z_0 + \tan^{-1}(y_0 / x_0) \end{aligned} \tag{3}$$

In the circular vectoring mode, as y_i gets closer to 0, each x_i and z_i converge to $K\sqrt{(x_0{}^2 - y_0{}^2)}$ and $z_0 + \tan^{-1}(y_0 / x_0)$, respectively. The constant K can be eliminated by multiplying with the inverse of K after the CORDIC iterations. In order to calculate the AM and PM components of the signals in the proposed DPD, x_0 and y_0 from Eq. (3) substitute the I and Q signals, respectively, where z_0 is initialized to 0. On the other hand, in the rotation mode, if x_0 and z_0 are initialized to the AM and PM components with z_0=0, we can obtain the I component on X and the Q component on Y by using Eq. (2).

Fig. 4 shows the dual-mode pipeline processing units of the CORDIC processor and the corresponding clock-by-clock operations proposed in our work. Both pre- and

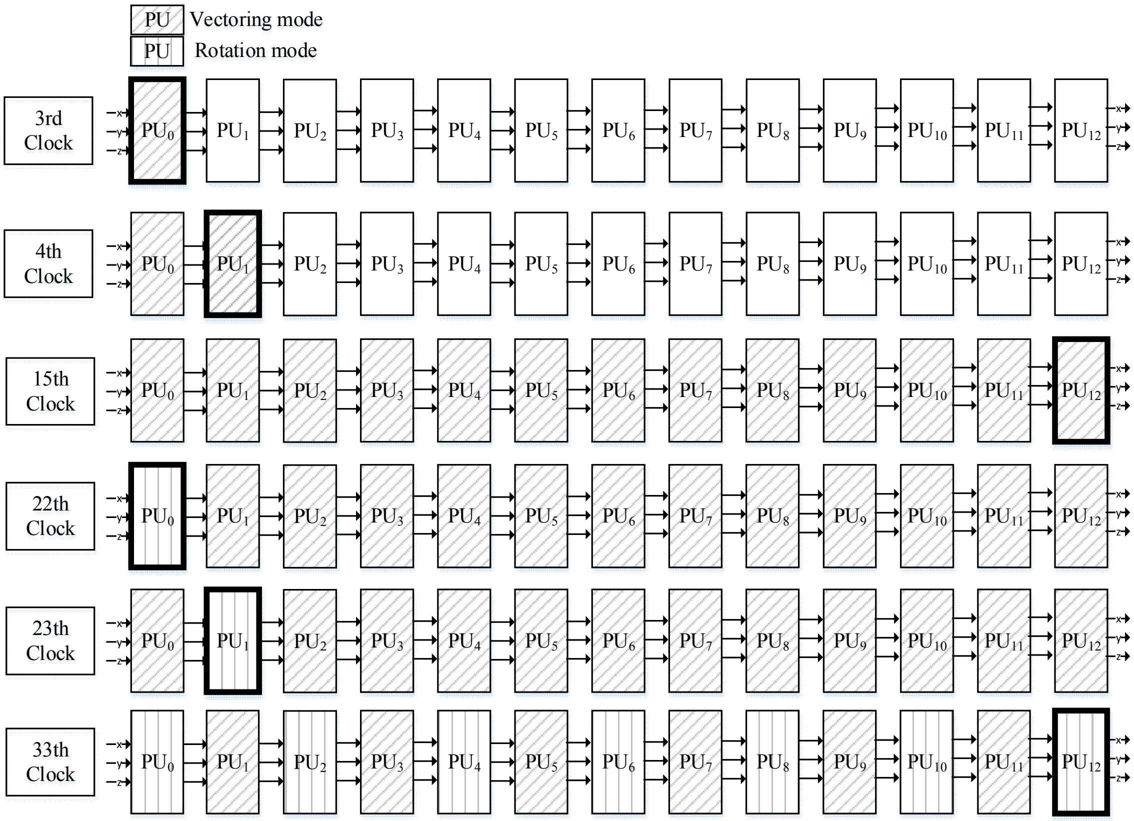


Fig. 4. Pipelined operations and clock latency in the dual-mode CORDIC processor.

post-processing for the PD are accomplished by the shared CORDIC processor which consists of 13 unit stages where the design is nearly identical to that of a typical pipeline architecture [15]. Each unit can work under one of two operational modes as a result of the mode control signal. As the DPD works with the successive input data, the MUX sends the data to be converted to the I/Q or the AM/PM domain alternately during each clock cycle to the first stage of the CORDIC pipeline. At the third clock cycle, PU$_0$ obtains the first I/Q component to be converted, as shown in the topmost pipeline stages of Fig. 4. At the fourth clock cycle, the output data of PU$_0$ moves to PU$_1$ and is still being converted using the vectoring mode. This conversion is finished at the 15th clock cycle, and then the corresponding AM/PM data can be fed into the PD block. After 7 clock cycles pass, the first PD output can enter PU$_0$ with the rotation mode. On the next clock, further I/Q data to be converted to AM/PM enter the first stage in the vectoring mode. Since the PD block produces distorted AM/PM signals that should be converted to I/Q signal every two clock cycles, the PU$_0$ operates with vectoring and rotation modes by turns. Finally, at the 33rd clock cycle, the conversion of the first output data would be finished by the completion of 13 iterations.

4. The PD block

The inverse function of the PA non-linearity can be defined as the m-th order the polynomial as follows,

$$A_{DPD}^{(m)} = \sum_{n=1}^{m} a_n A_D^n \tag{5}$$

$$P_{DPD}^{(m)} = \sum_{n=1}^{m} p_n A_D^n + P_D \tag{6}$$

In Eqs. (5) and (6), A_D and P_D denote the AM and PM of the original signal to be amplified by the PA. A_{DPD} and P_{DPD} represent the pre-distorted AM and PM components that compensate the non-linearity of the PA. a_n and p_n represent the coefficients of the n-th term. Eq. (5) can be re-constructed using Horner's recurrence formulation as follows,

$$\begin{aligned} A_{DPD}^{(1)} &= a_m A_D + a_{m-1} \\ A_{DPD}^{(2)} &= A_{DPD}^{(1)} A_D + a_{m-2} \\ &\vdots \\ A_{DPD}^{(m-1)} &= A_{DPD}^{(m-2)} A_D + a_1 \\ A_{DPD}^{(m)} &= A_{DPD}^{(m-1)} A_D + a_0 \end{aligned} \qquad (7)$$

Eq. (6) can be also rewritten in a similar way,

$$\begin{aligned} P_{DPD}^{(1)} &= p_m A_D + p_{m-1} \\ P_{DPD}^{(2)} &= P_{DPD}^{(1)} A_D + p_{m-2} \\ &\vdots \\ P_{DPD}^{(m-1)} &= P_{DPD}^{(m-2)} A_D + p_1 \\ P_{DPD}^{(m)} &= P_{DPD}^{(m-1)} A_D + p_0 \end{aligned} \qquad (8)$$

where a_0 and p_0 are initialized to 0 and P_D respectively. As a result of Eqs. (7) and (8), PD can be implemented by m pipelined units that calculate the form '*ax+b*' as shown in Fig. 5. PD also can be implemented by using the linear vectoring mode of the existing CORDIC processor. Each pipeline stage of the CORDIC is shown in Section 2.3, and it can be re-used to calculate Eqs. (7) and (8). However, we implemented PD block in our VLSI design that will be shown in Section 3, employed adders and multipliers with pipeline registers.

Sometimes the PA's output can be found to be smaller than the input by watching for the characteristic of non-linearity. In this case, the inverse function applied to the DPD would have larger values than the original ones. Generally, the output range of the ADC covers the entire range of the DPD input with high resolution and high accuracy. However, such inverse functions exceed the output range. Then, critical errors can occur. The scaling factor is used to prevent these over-ranged errors. For the inverse function to be applied, we calculate the largest values of the output, and we find the scaling factor that makes the output remain within the range of the DAC. The proposed DPD receives the pre-calculated scaling factor, and adjusts its output to be within the range by multiplying it with the determined value after the PD operation.

III. Experiments

In this section, we present the results of the experiments for the validation of our proposed DPD. We obtained the inverse function of the PA characteristic by the 5th-order polynomial, and the test data are also taken by analyzing one of the real PAs. The ADC and DAC have a 20 MPSP output rate with 14-bit resolution. The estimated function is defined as follows,

$$\begin{aligned} A_{DPD} &= 3.361A_D^5 - 6.346A_D^4 + 4.622A_D^3 - 1.447A_D^2 + 0.9892A_D \\ P_{DPD} &= -21.69A_D^5 + 74.06A_D^4 - 106.9A_D^3 + 77.29A_D^2 - 27.52A_D + P_D \end{aligned}$$

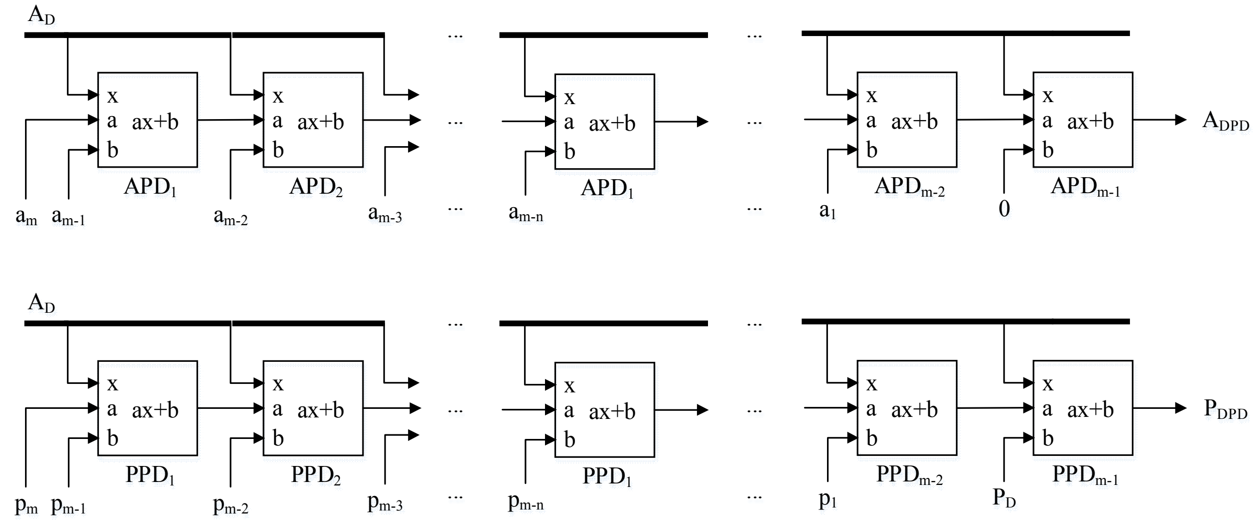
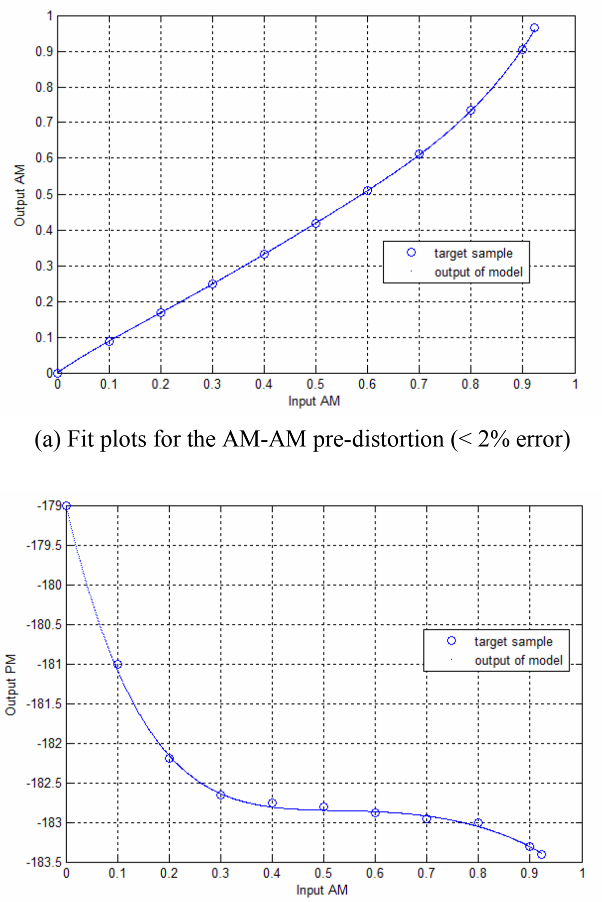


Fig. 5. A pipelined architecture for the PD block.



(a) Fit plots for the AM-AM pre-distortion (< 2% error)

(b) Fit plots for the AM-PM pre-distortion (< 0.04% error)

Fig. 6. Fitting results using a polynomial-based model.

where the maximum value of A_{DPD} is 1.183. Thus, the scaling factor was taken as 0.8392 from the inversion of the maximum value. Fig. 6 depicts the comparative plots for the target characteristic data and the estimated models above. The corresponding percentage errors are less than 2% and 0.04% for the AM-AM and AM-PM pre-distortion, respectively.

Table 1 summarizes the fixed-point error of the proposed DPD. The errors are calculated by comparing the computation results of the fixed point design to the simulation vectors representing the ideal floating numbers. The input range of the I/Q-component is from -0.5 to 0.5 and the corresponding range of the output amplitude is between 0 and $1/\sqrt{2}$ (≈0.707). As a result, the errors in Table 1 are small enough to be considered negligible. We evaluated the compensated linearity of the PA with the proposed DPD. Fig. 7 shows the power spectrum of the PA output. The blue line represents the original output signal of PA where the spectrum shows the distorted characteristics due to the non-linearity of the PA. The red line represents the result of the PA output using the pre-distorted DPD output. As a result of the nonlinearity of the PA, the PA output without the DPD (blue line) has spectral spreading with large side lobes adjacent to the central frequency. The pre-distorted signal as a result of the proposed design suppresses those out-band components of PA [20].

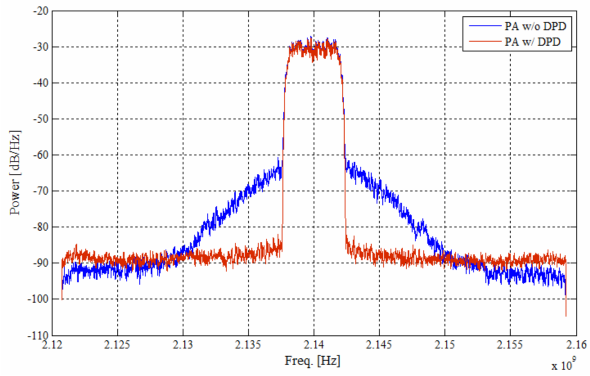


Fig. 7. PSD plot of the PA output with/without the proposed DPD.

Table 1. Fixed point design errors in the proposed DPD

Output Components	Error Amplitude (x10^{-4})	
	Avg.	Max.
I	6.34	11.89
Q	5.50	7.93

We designed our proposed DPD on the register transfer level (RTL) using Verilog HDL. An IC was fabricated for the DPD using a standard CMOS 0.35 μm cell library, as depicted in Fig. 8. The synthesized result of the logic show the total cell area to be estimated to have 43,000 gates, and the target clock speed was of 40 MHz. The implementation results are summarized in detail in Table 2. Using chip testing equipment, both functional and timing verification of the DPD IC were performed by comparing its outputs to the post-simulation results. We tested the implementation by varying the clock period from 15 ns to 35 ns, with a 1 ns interval. We also verified the operation correctness by changing the operating voltage from 2.2 V to 4.0 V. Each condition was tested for 30,000 input data sets. Consequently, we successfully validated the fabricated DPD with 40-50 MHz clock speed from 2.7 V to 4.0 V supply.

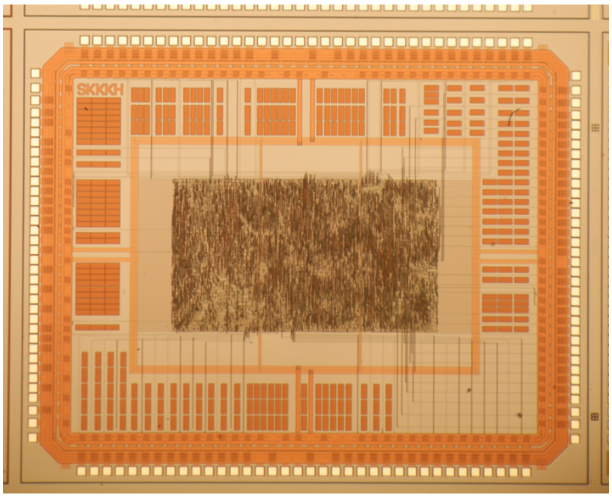


Fig. 8. Microphotograph for the fabricated DPD.

Table 2. VLSI implementation results for the proposed design

Technology		0.35 μm CMOS 1-poly, 4-metal
Operating clock freq./Voltage		40 MHz / 3.3V
Gate size	PD	29,272
	CORDIC	12,594
	others	673
	Total	42,539
Core size		2,349 μm x 1,348 μm
Maximum output data rate		20 MSPS
Power consumption (Active)		62.5 mW

Table 3. Comparison results with the DSP implementation result

Execution Time [μs]	Proposed H/W @40MHz	TMS320C6415 @780MHz
Pre/Post processor	0.2	< 0.1
I/Q – AM/PM conversion	2.8	5.1
PD	0.5	0.4
Initial latency	3.5	5.5
Pipeline latency	0.1	

Table 3 shows a comparison of the results for the proposed HW design at a 40 MHz clock rate with the SW implementation in a commercial high performance, fixed-point DSP (TI TMS320C6415) where the operating clock frequency was of 780MHz. Each variable for the SW code was also designed using a fixed-point level so that we could sufficiently utilize the computing capacity provided by the target DSP architecture. In our design, the number of internal pipeline stages is more than 30, so the initial latency for the first output of the DPD would be 3.5 μs. However, successive output can be generated every two clock cycles and therefore, the minimal pipeline latency can be of 100 ns. Conversely, the execution time of the identical SW functions on the DSP is more than 5 μs, in spite of the target operating clock frequency ranging in the hundreds of MHz. We can see that the main computation components for the DPD can be regularly structured by the pipelined VLSI design which is more effective than the SW solution running on the instruction set architecture.

To the best of our knowledge, there are no publicized results for DPD logic design based on standard cells that are comparable to our study. Instead, we use the existing FPGA (Field Programmable Gate Array) implementation results for DPDs. Table 4 summarizes the several experimental results on Xilinx FPGA devices. One of target the designs to be compared with is the direct form polynomial DPD [21-23] and the other type is the LUT-based distorter [22]. As shown in Table 4, our design was configured by all the same target FPGA devices that were already applied to the existing designs. For the comparison with the partial result of [22], there are no exact part names of the target FPGA device specified in [22], except for the family name as Virtex-6. We chose the one of Virtex-6 devices for these cases. For a specific DPD design, each column shows the adjacent channel power ratio, number of logic slice cells including flip-flops, specialized arithmetic unit cells (DSP48) including multipliers and memory blocks occupied in the target FPGA. Compared to the existing studies, our design has the low hardware complexity on digital logic circuits and memory blocks where its spectral regrowth is well suppressed.

IV. CONCLUSION

This paper presents a VLSI design for a polynomial DPD using a CORDIC processor. The CORDIC processor calculates a trigonometric function, converting an I/Q signal to AM/PM components and vice versa. We implement the CORDIC processor to support both two modes of operation, vectoring and rotation. Each pipelined unit changes its mode of operation every clock cycle. Including the PD block, the proposed DPD is a fully pipelined design supporting a high rate pre-distortion. By adjusting only the coefficients of the polynomial, it conforms to environmental change. The prototype IC was successfully fabricated and validated on 0.35 μm CMOS technology.

Table 4. Hardware complexity comparison with other FPGA implementation

Design	Adjacent channel power ratio [dBc]	FPGA	Logic size [slices]	DSP48 [slices]	Block RAMs
polynomial [21]	-22.5	XC4VSX35	2826	19	60
polynomial [22]	-60	Virtex-6	5934	260	17
polynomial [23]	-70	XC2VP50FF	2032	48*	Not used
LUT [22]	-60	Virtex-6	1220	16	18
This work	-60	XC4VSX35	1239	13	Not used
		XC2VP50FF	1239	13*	Not used
		Virtex-6 (XC6VCX75T)	693	13	Not used

* indicating the number of multipliers used in Virtex-2 device (XC2VP50FF)

ACKNOWLEDGEMENTS

This work was supported by IDEC (Integrated circuit Design Education Center) through the MPW (Multi-Project Wafer) Program.

REFERENCES

[1] I. Teikari, “Digital Pre-distortion Linearization Methods For RF Power Amplifiers,” Dissertation, Helsinki Univ. of Tech., 2008.

[2] J. K. Cavers, “Amplifier Linearization Using a Digital Pre-distorter with Fast Adaptation and Low Memory Requirements,” IEEE Trans. On Vehicular Tech., Vol. 39, No. 4, pp. 374-382, 1990.

[3] J. Namiki, “An automatically controlled pre-distorter for multilevel quadrature amplitude modulation,” IEEE Trans Commun., vol. COM-3 1, pp.707-712, 1983.

[4] D. Hilbom, S. Stapleton and J. Cavers, “An adaptive direct conversion transmitter,” Proc. of IEEE Vehicular Tech., 1992.

[5] Y. Nagata, “Linear amplification technique for digital mobile communications,” Proc. of IEEE Vehicular Tech., pp. 159-164, 1989.

[6] G. Baudoin and P. Jardin, “Adaptive polynomial pre-distortion for linearization of power amplifiers in wireless communications and WLAN,” Proc. of EUROCON’2001 Int. Conf. on Trends in Comm., vol. 1, pp.157-160, 2001.

[7] E. Westesson, L. Sundström, “Low-Power Complex Polynomial Pre-distorter Circuit in CMOS for RF Power Amplifier Linearization,” proc. of 27th ESSCIRC, pp. 486-489, 2001.

[8] M. Ghaderi, S. Kumar, D. E. Dodds, “Fast adaptive polynomial I and Q pre-distorter with global optimization,” proc. of IEE Comm., Vol. 143, No. 2, pp.78-86, 1996.

[9] P. Kenington, High-linearity RF amplifier design, Artech House, 2000.

[10] K. J. Muhonenm, K. Kavehrad, “Look-Up Table Techniques for Adaptive Digital Pre-distortion: A Development and Comparison,” IEEE Trans. On Vehicular Tech., Vol. 49, No. 5, pp.1995-2002, 2000.

[11] W. Woo, E. Park, K. U-yen and S. Kenny, “Wideband pre-distortion linearization system for RF power amplifiers using an envelope modulation technique,” Proc. of Radio and Wireless Conf., pp.401-404, 2003.

[12] J. Sills and R. Sperilich, “Adaptive power amplifier linearization by digital pre-distortion using genetic algorithms,” Proc. of IEEE Radio and Wireless Conf., pp.229-232, 2002.

[13] J. Lee, S. Jeon, J. Kim, and Y-W. Suh, “Adaptive HPA Linearization Technique for Practical ATSC DTV System,” IEEE Trans. On Broadcasting, Vol. 59, No.2, pp.376-381, 2013.

[14] B. Parhami, “Computer Arithmetic : Algorithms and Hardware Designs,” pp. 361-377, Oxford University Press,. 2000.

[15] Q. Zhengyu, A.C. Cabe, R.T. Jones, M.R. Stan and L. Charles, “CORDIC implementation with parameterizable ASIC/SoC flow, Proc. of IEEE SoutheastCon, pp.13-16, 2010.

[16] D.E. Aschbacher, “Digital Pre-distortion of Microwave Power Amplifiers,” Dissertation, Vienna Univ. of Tech., 2005.

[17] H. Gandhi, “A Flexible Volterra-Based Adaptive Digital Pre-Distortion Solution for Wideband RF Power Amplifier Linearization,” IEEE Long Island

Section, 2009.

[18] Texas Instruments, "GC5322 Wideband Digital Pre-Distortion Transmit IC Solution," TI datasheet, 2008.

[19] B. Ozgul, J. Langer, J. Noguera, and K. Visses, "Software-programmable digital pre-distortion on the Zynq SoC," IFIP/IEEE 21st Int'l Conf. on VLSI-SoC, pp.288-289, 2013.

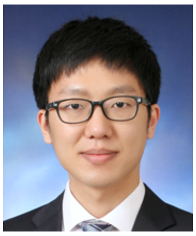
[20] K. Kim, S. Shim, J.T. Kim and J.T. Kim, "Digital Predistorter with Pipelined Architecture Using CORDIC Processors," World Academy of Science, Eng., and Tech., Vol.4, pp.803-806, 2010.

[21] S. Suranjana and A. Dinh, "FPGA Implementation of a Power Amplifier Linearizer for an ETSI-SDR OFDM Transmitter," ZTE Comm., No.3, pp.22-27, 2011.

[22] L. Guan and A. Zhu, "Low-Cost FPGA Implementation of Volterra Series-Based Digital Predistorter for RF Power Amplifiers," IEEE Trans. on Microwave Theory and Techniques, Vol.58, No.4, pp.866-872, 2010.

[23] N. Lashkarian and C. Dick, "FPGA Implementation of Digital Predistortion Linearizers for Wideband Power Amplifiers," Proc. of SDR, 2004.

Jong Kang Park received BS and MS degrees in Electric, Electronics and Computer Engineering in 2001, 2003 and Ph.D. degree in Electric and Electronics Engineering from Sungkyunkwan University, Korea in 2008. From 2008 to 2013, he was with Samsung Electronics as a senior engineer. He is now a research professor, school of Electronic and Electrical Engineering, Sungkyunkwan University. His current research interests include the digital logic design, sensor ICs, embedded system and soft error analysis and tolerance techniques for VLSI designs.



Jun Young Moon received BS degree in electric engineering in 2013 from Sungkyunkwan University, Korea. He is currently working toward the MS degree in the Department of IT Convergence in Sungkyunkwan University. His research interests include the digital logic design and embedded system platforms.

Kyunghoon Kim received a BS degree in electric and electronics engineering in 2009 and received a MS degree in department of mobile system engineering in 2011 from Sungkyunkwan University, Korea. From 2009, he joined Samsung Electronics as an engineer. His current research interests include the secured embedded system platforms and software solutions.

Youngoo Yang (S'99-M'02) was born in Hamyang, Korea, in 1969. He received the Ph.D. degree in electrical and electronic engineering from the Pohang University of Science and Technology(Postech), Pohang, Korea, in 2002. From 2002 to 2005, he was with Skyworks Solutions Inc., Newbury Park, CA, where he designed power amplifiers for various cellular handsets. Since March 2005, he has been with the School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Korea, where he is currently an associate professor. His research interests include power amplifier design, RF transmitters, RFIC design, integrated circuit design for RFID/USN systems, and modeling of high power amplifiers or devices.

Jong Tae Kim is a Professor at the School of Electronic and Electrical Engineering, Sungkyunkwan University, where he has been since 1995. He received the BS degree in electronics engineering from Sungkyunkwan University in Korea in 1982 and the MS and PhD degrees in electrical and computer engineering at the University of California, Irvine, in 1987 and 1992, respectively. From 1991 to 1993 he was with the Aerospace Corporation in Elsegundo, California. He was a full-time lecturer at Chunbuk National University in Korea from 1993 to 1995. His research interests include SoC design and design methodology, embedded systems, and multi-core processor architecture.