A Practical Step Forward Toward Software-Defined Radio Transmitters

Essam Atalla*, Imran Bashir†, Poras Balsara*, Kamran Kiasaleh*, Robert Bogdan Staszewski†,
*Department of Electrical Engineering, University of Texas at Dallas
†Texas Instruments Inc., Dallas, Texas

***Abstract*— After many years of research and development in the wireless communication community, software defined radio (SDR) is no longer an unachievable dream. In this paper, we present a significant step forward toward practical SDR transmitters. Motivated by the reconfigurability, programmability and available computational power in a commercial Digital RF Processor (DRP^{TM})-based single-chip GSM/EDGE radio, we succeeded to modulate the RF carrier with P25 compliant C4FM (continuous 4-level FM) data. P25 is a digital public safety standard that operates in the 746-806 MHz frequency band, which is different from the normal operation band of the GSM/EDGE chip. The modulation is based completely on software without need for any hardware modifications. The measurement results show that the transmitted signal spectrum is compliant with the P25 standard specifications. We also show that the work presented in this paper can be extended to provide more elaborate modulation schemes.**

I. Introduction

In the last few years, there has been great development in semiconductor technology that allowed RF and digital baseband integration in a complete system-on-chip (SoC). This was accompanied with the development of wireless communications standards such as WLAN, WiMAX. In addition, cellular communications standards have been modified to provide services beyond voice communications such as high data rate and video communications. For example, EDGE and GPRS are two enhancements that provide high data rate capability for the GSM standard that was originally designed for voice communications. The huge end user demand for integrating different services on a single handset, in addition to the requirement of global roaming, led the manufacturers to strive to develop software-defined radio (SDR).

Despite the great development in semiconductor technology, circuit design techniques are still not capable of providing a full spectrum of SDR operation. That's why different terminology appeared, like multiband, multistandard or multimode, and all are considered under the umbrella of SDR, although they are really not [1]. A transceiver can be referred to as SDR if its communication functions are realized as programs running on a suitable processor. Based on the same hardware, different transmitter/receiver algorithms, which usually describe transmission standards, are implemented in software [2].

In this paper, we will describe an example of a real SDR experience. A commercially available single-chip GSM/EDGE phone based on a Digital RF Processor (DRP^{TM}) technology [3] [4] is software programmed to provide P25 public safety standard transmission, without any hardware modification. Section II gives an overview of the P25 standard and the GSM/EDGE transmitter that we used. Section III describes the software modulation. The measurement results are given in Section IV. Finally, conclusions are given in Section V.

II. P25 Standard and the GSM Transmitter

A. Overview of P25 Standard

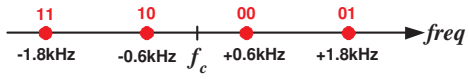

Fig. 1. C4FM modulation in frequency plane showing symbol mappings.

P25, which stands for Project 25, is an open architecture, user driven suite of system standards that define digital radio communications system architectures capable of serving the needs of Public Safety and Government organizations. The P25 suite of standards involves digital Land Mobile Radio (LMR) services for local, state/provincial and national (federal) public safety organizations and agencies [5]. A P25 radio is any radio that conforms to the P25 standard in the way it functions or operates. That is why there is great interest in integrating P25 capability in cellular handsets. P25 compliant technology is being deployed in several phases. Phase 1 uses continuous 4-level FM (C4FM) non-linear modulation for digital transmission in a 12.5 kHz channel. Phase 2 uses CQPSK modulation to transmit digital data over a 6.25 kHz channel. Both Phase 1 and Phase 2 use 9.6 kbps data rate. This is translated into 4.8 kHz symbol rate (2 bits per symbol). In this paper, we focus on Phase 1 only. Figure 1 shows the 4 frequency deviations corresponding to the 4 data symbols.

Phase 1 P25 transmitter employs root raised-cosine pulse shaping filter before modulating the RF carrier. Since the filter impulse response, which decides the pulse shape, cannot be realized as infinite in time, we have to choose the length of the filter impulse response (in number of symbols) that will provide proper filtering action in the frequency domain. Proper filtering action means that the frequency response of the filter satisfies the noise profile of the P25 standard. Figure 2 shows the frequency response of the root raised-cosine pulse shaping filter having impulse response length of 2, 3 and 5 symbols. It is quite clear that 2 symbols are sufficient. Yet, using 3 symbols is better because it provides some extra margin.

978-1-4244-1680-6/07/$25.00 ©2007 IEEE

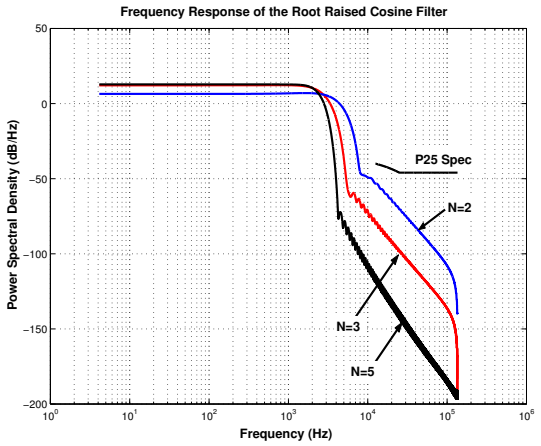

Fig. 2. Effect of the length of pulse shaping filter impulse response.

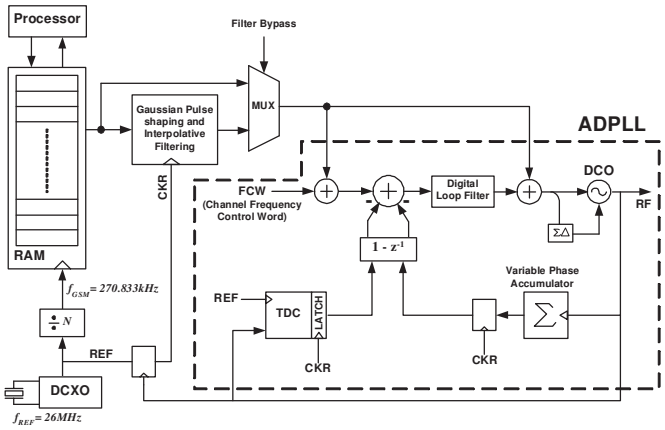

Fig. 3. All-digital PLL based GSM transmitter.

B. Description of the GSM/EDGE Radio Chip

The GSM/EDGE radio IC that we use in this work is a commercially available chip that has extensive reconfigurability and programmability features. In this paper, our focus is mainly on the transmitter [3]. Figure 3 shows a simplified block diagram for the GSM transmitter. The AM modulation path of the polar transmitter, which is used in EDGE transmission, is not shown in this block diagram, since in this Phase 1 work we are only interested in the C4FM frequency modulation of the P25 standard. The core of the transmitter is an all-digital PLL (ADPLL), which is formed of digital phase detection and filtering built around a digitally-controlled oscillator (DCO). The digital phase detection circuit provides an integer phase resolution by accumulating clock edges of the frequency reference (FREF) and RF signals and performing a fixed-point subtraction. In order to account for the frequency difference between the FREF and RF signals, the input to the accumulator clocked by the retimed FREF clock (CKR) signal is a frequency control word (FCW). The FCW is the ratio between the desired RF frequency and the reference frequency. The time-to-digital converter (TDC) provides fractional (i.e., sub DCO period) timing resolution. The phase error at the output of the phase detection and the TDC is filtered in a high order digital filter and fed to the DCO as a frequency tuning word. Frequency modulation is done by adding a modulation frequency word to the FCW. The value of the modulation frequency word is equal to the required frequency deviation from the carrier. For example, a digital word that corresponds to a maximum frequency deviation of ± 67 kHz is added to FCW in the GSM modulation.

The modulation data word is read real-time from a random-access memory (RAM) that is clocked at the GSM data rate equal to 270.833 kHz. This means that a new modulation data word is read every 3.692 μsec. Before being added to the FCW at the ADPLL input, the modulation data word is filtered in a Gaussian interpolative filter that provides pulse shaping as defined in the GSM standard [6].

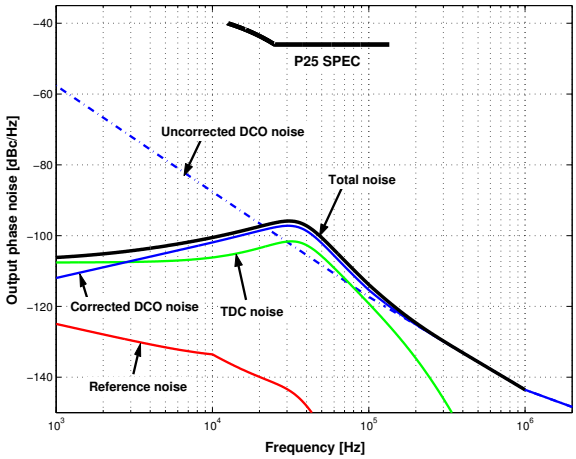

Fig. 4. ADPLL phase noise components at the nominal loop filter settings.

Figure 4 shows simulation results for the phase noise contributions of the different ADPLL components at its output. It also shows the total phase noise at the ADPLL output. Although the PLL bandwidth is 30 kHz, which is relatively wide corresponding to the P25 maximum frequency deviation of 1.8 kHz, the total ADPLL phase noise is far below the P25 spec. This guarantees that the P25 modulation can be done using this chip. This is indeed very encouraging, since it means that the P25 standard can be integrated into GSM cellular handsets with almost no additional cost.

III. Software Modulation

In the GSM/EDGE transceiver chip, there are embedded processors that perform different configuration and control functions. For example, one task of one of the processors is to set the default values for all the registers in the chip, such as the registers that store the default values of the ADPLL loop filter parameters. One of the embedded processors is shown as part of Figure 3. It is very important with respect to the focus of this paper because it controls the modulation process. It loads the modulation data control word into the RAM. After that, the data word is filtered and applied to the ADPLL to modulate the RF carrier. The best way to

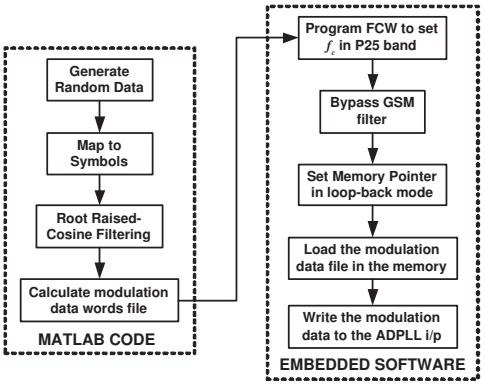

Fig. 5. Steps for performing P25 modulation on GSM chip

implement a software radio transmitter is to tap the powerful computational power [7] such as DSP and/or ARM processors which are already available on the SoC. While bypassing the GSM gaussian pulse shaping filter (by controlling the MUX shown in Figure 3), a processor can be programmed to handle different modulation schemes. For example, in order to generate P25 C4FM modulation data, the embedded processor should read digitized audio data from a CODEC chip at its input interface, and then perform software root raised-cosine filtering. The filtered data is then written to the RAM and applied directly to the ADPLL (added to FCW). But in this paper we use a slightly different approach. Instead of relying completely on the embedded processor, we used Matlab to generate random data, map the bits to symbols, perform software root raised-cosine cosine filtering, and finally generate the corresponding frequency modulation data word. The results are saved in a file. The embedded processor loads the file into the RAM (Fig. 3) and then writes the frequency modulation data words one-by-one to the ADPLL input. In this way, the output of the ADPLL is P25 compliant C4FM modulated RF carrier. It should be worth mentioning here that the memory pointer, which refers to the memory location from which the modulation data word is read, is configured in a loop-back mode. This means that if we have 100 memory locations storing modulation data words, the pointer will loop back to location 1 after reading from location 100. This guarantees continuous real-time modulation, but the 100 data words must be selected carefully to avoid any periodicity in the output and to guarantee P25 compliant transition from the last symbol (location 100) to the first symbol (location 1). Figure 5 shows a chart that summarizes all the steps in order to perform C4FM modulated carrier.

A. Limitations of this Approach

The first limitation in the approach we explained above is the memory-read clock frequency. The root raised-cosine pulse shaping filter upsamples the 4.8 kHz symbols, and produces samples at a rate equal to $r\times$4.8 kHz, where r is the oversampling ratio. Figure 6 illustrates this concept assuming $r = 4$. The modulation data words are read from the memory at the GSM symbol rate equal to 270.833 kHz

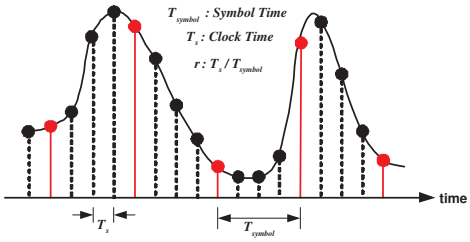

Fig. 6. Illustration of pulse shaping filter action with OSR $r = 4$

(because the chip was originally designed for GSM standard). This means that if the ratio between the GSM symbol rate and the P25 symbol rate is x, then each P25 (pulse-shaped) symbol must be represented by x GSM samples. Therefore, the best thing to do is to design the root-raised cosine pulse shaping filter with oversampling factor $r = x$. The problem appears when x is a fractional number. One solution is to use interpolative pulse shaping filter as suggested in [6]. Another solution is to downsample the P25 symbols by a factor D and then use oversampling factor I in the filter, such that the net oversampling ratio $r = I/D$ is equal to the non-integer ratio x. In this paper, for the sake of concept verification, although the ratio x = 270.833/4.8 = 56.423 is non-integer we approximated the filter upsampling factor r to 57. The described solutions are based on the fact that the memory read clock is fixed and equal to the GSM symbol rate as shown in Figure 3. A better solution is to have a programmable memory-read clock that can be chosen such that the ratio x is always an integer. This solution is not available in the specific chip we were using, since it was designed for GSM/EDGE applications. The second limitation in the approach we used is the memory size. Assuming that the memory size is n locations, the maximum number of P25 symbols that can be stored in the memory is equal to n/x where x is the number of GSM samples per P25 symbol rounded to the nearest integer. In our experiment, we had 1000 memory locations available, and so we were able to store 17 P25 pulse-shaped symbols. And these symbols are repeated in the loop-back mode as described in the last section.

IV. Measurement Results

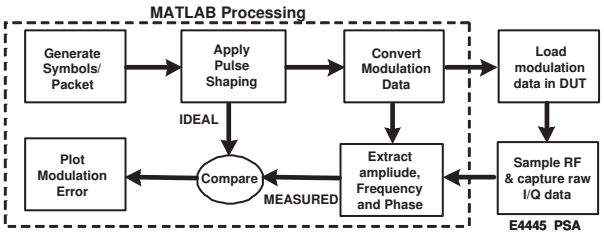

Fig. 7. Validation flow of P25 Phase I standard

Devising a common platform for system verification and testing of radios supporting diverse order of modulation schemes is a challenge. In most cases, the instruments are designed to support a specific set of most commonly used

wireless standards in the commercial sector. A more efficient approach to this problem, which was used in this project, is to use a generic vector spectrum analyzer as a sampler to output raw demodulated data. The modulation characteristics of the RF signal can then be determined by post-processing the raw data. This flow is graphically represented in Figure 7.

P25 C4FM modulation performance is analyzed using the stated flow. First, symbols are generated in MATLAB script from a random stream of bits. The data is then passed through a raised-cosine pulse-shaping filter with an upsampling factor of 56, roll-off factor of 0.25, and filter group delay of 4. This filter simulates the combined effect of the two root-raised cosine filters used in the transmitter and receiver. The DRP modulator has a fixed sampling clock of 270.833 kSps and when an integer upsampling of 56 is applied, the resulting data rate at transmitter output is 4.836 kSps.

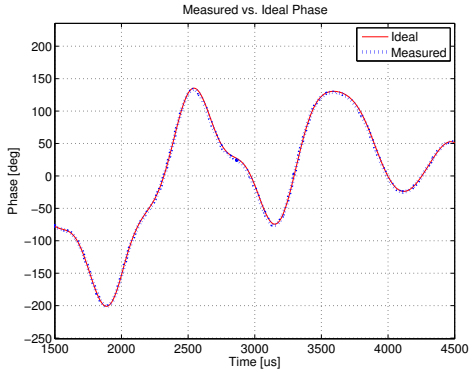

Fig. 8. Ideal and measured phase trajectory vs. time

Figure 8 shows the simulated and measured phase trajectory over the transmitted packet. The simulated phase is obtained from the MATLAB script and measured phase is determined from raw I/Q data from the vector spectrum analyzer.

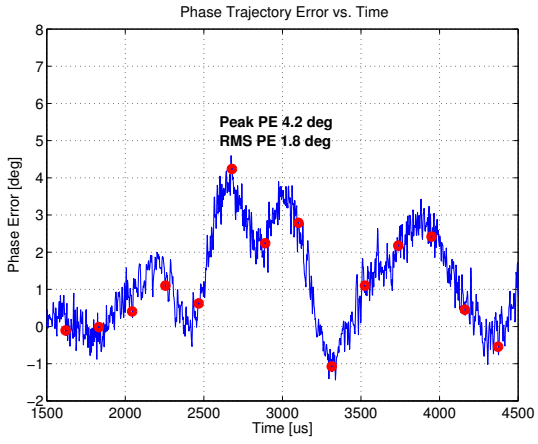

Fig. 9. Phase trajectory error vs. time

Figure 9 shows the phase trajectory error which is the numerical difference of ideal and measured phase over the transmitted packet. The red circles on the plot mark the position of the 14 symbols in the packet. The RMS phase error is calculated by taking the root-mean-square of the phase error measured at the symbols shown in the plot which is around 1.8^o. Similarly, the peak phase error is the maximum phase error over the symbols which is 4.2^o. Figure 10 represents the spectrum measured while C4FM modulation was applied to the transmitter. The spread of the carrier due to modulation is not symmetric. This is due to the fact that the data packet of 14 symbols is repeated over time and hence the data is not entirely random during the measurement. The phase noise at 12.5 kHz offset from the carrier, where the adjacent channel is located, is -62 dBc.

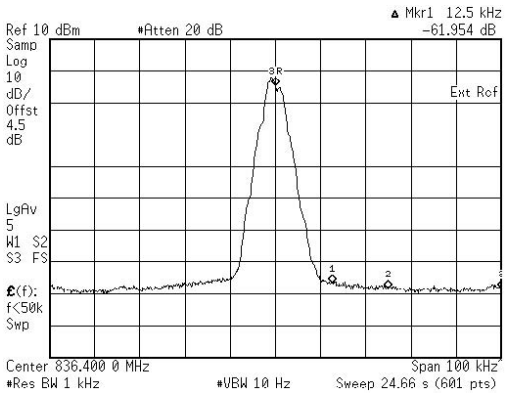

Fig. 10. P25 C4FM modulated spectrum

V. Conclusions

We have presented a practical realization of a software-defined radio (SDR) transmitter based on a GSM single-chip phone using the Digital RF Processor (DRP^{TM}) technology. The high reconfigurability and software programmability of the DRP transmitter allowed us to meet specifications of the Phase-I P25 public radio standard.

Acknowledgment

The authors would like to thank National Institute of Justice (NIJ) for funding this research project.

References

[1] R. Bagheri, A. Mirzaei, S. Chehrazi, *et al.*, "An 800MHz to 5GHz software-defined radio receiver in 90nm CMOS," in *Proc. of IEEE Solid-State Circuits Conf.*, 2006, pp. 480–481.

[2] F. Jondral, "Software-Defined Radio - Basics and Evolution to Cognitive Radio," *EURASIP Journal on Wireless Communications and Networking*, no. 3, pp. 275–283, 2005.

[3] R. B. Staszewski *et al.*, "All-Digital PLL and Transmitter for Mobile Phones," *IEEE Journal of Solid State Circuits*, vol. 40, no. 12, pp. 2469–2482, Dec. 2005.

[4] K. Muhammad *et al.*, "The First Fully Integrated Quad-Band GSM/GPRS Receiver in a 90-nm Digital CMOS Process," *IEEE Journal of Solid State Circuits*, vol. 41, no. 8, pp. 1772–1783, Aug. 2006.

[5] Daniels Electronics LTD. (2004) P25 training guide. [Online]. Available: http://www.p25.com/resources/P25TrainingGuide.pdf

[6] R. B. Staszewski and R. Staszewski, "Interpolative Pulse-Shape Filtering for a GSM/Bluetooth Transmitter," in *IEEE Dallas Circuits and Systems Workshop (DCAS)*, 2005, pp. 191–194.

[7] R. Staszewski *et al.*, "Software Assisted Digital RF Processor for Single-Chip GSM Radio in 90 nm CMOS," in *IEEE Custom Intergrated Circuits Conference (CICC)*, 2006, pp. 611–614.