Hardware Implementation of Software Radio Receivers

Sambhav Jain

ECE Division, NSIT, Delhi 110078

Abstract

This paper describes the hardware design and implementation of an inexpensive software defined radio (SDR). SDRs use a hardware front end but can change their operational frequency, bandwidth requirement and various standards by implementing various software algorithms. Rather than using analog blocks for design of radios, as is the convention, this paper details the use of digital signal processing for implementation of radio system. Such a solution provides inexpensive, efficient interoperability between frequencies and standards. This system is designed on BeagleBone Black equipped with AM335x ARM Cortex A8 Microprocessor with a RF front end based on Tayloe Detector.

Keywords: SDR, Beaglebone Black (BBB), ARM, Tayloe Detector

1. Introduction

Software Defined Radio aims to go one step ahead in the development of radios. Rather than using analog blocks for design of radios, as is the convention, this paper details the use of digital signal processing for implementation of radio system. SDRs are characterized by its flexibility: by modifying some lines of software can completely change its functionality. This ease upgrades to new standards and improved performance without replacing any hardware. There is the distinct difference between a radio system that uses software for some of its functions and a radio that can be completely redefined through software modification. The latter is a software-defined radio.

2. Organization of the Text

The system described in this paper uses a RF front end based on Tayloe Detector to down convert the required RF signals and the software processing is done on ARM

processor. This paper is organized as follows. First, Section II explains the basic concepts of any radio receiver with the insight of quadrature mixing. Then, a description of adopted Tayloe Detector technique in Section III. Then software and design approach with receiver development details in Section IV. Last, laboratory, simulation and field test results are analyzed in Section V, and conclusion on the receiver design and implementation in Section VI.

3. Basic Concepts

The goal of this paper is to build a standalone software defined radio receiver that require a little hardware and maximize its flexibility by changing software. To do so, it is require to down convert the RF carrier signal to audio frequencies in such a way that it allows removal of the unwanted mixing products or images. The simplest way to do this is to use Direct Conversion techniques to convert the modulated RF signal directly to baseband. We can down convert the RF signal by mixing a signal with signal generated by an oscillator which is tuned to the RF carrier frequency to translate the baseband signal to a 0-Hz (Ideal) intermediate frequency (IF) as shown in Fig 1.
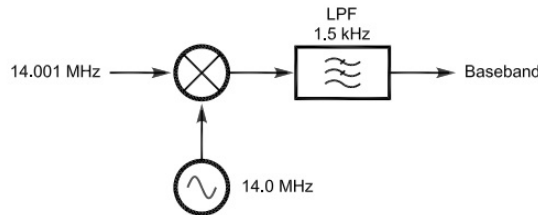
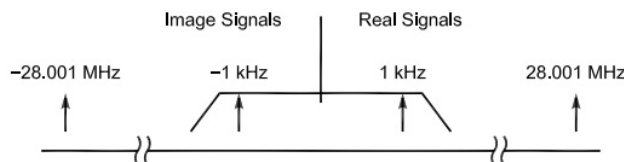**Fig. 1**: A down-conversion mixer with a low-pass filter.

Fig. 2: Real mixer's output spectrum showing the sum, difference and image frequencies.

Above figures shows a 14.001 MHz carrier signal mixed with a local oscillator of 14.000 MHz to translate the carrier signal to 1 kHz. If the cutoff frequency of low-pass filter is 1.5 kHz then any signal between 14.000 MHz and 14.0015 MHz would be within the pass-band of the direct-conversion receiver. The problem with this approach is that the receiver would also receive all signals between 13.99815 MHz and 14.000 MHz as unwanted signal within the pass-band, as illustrated in Fig 2.

To remove the image frequency signal we adopted quadrature mixing technique. Fig 3 shows the basic concepts of quadrature mixing, which shows a quadrature sampling I/Q mixer.
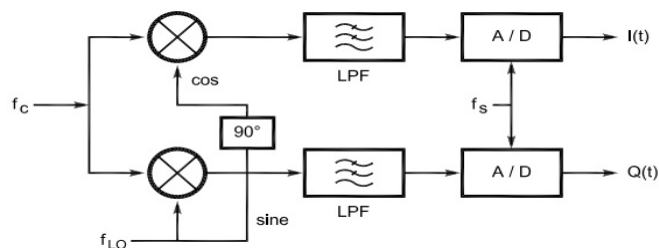
Fig. 3- Quadrature sampling mixer with A/D converters.

First, the RF input signal is applied to the two parallel mixer channels. We can generate a cosine wave by delaying the local oscillator wave by 90° to forms a quadrature oscillator. The RF input signal fc(t), is mixed with the respective sine and cosine wave local oscillators and is passed through low-pass filters to generate the quadrature, Q(t), and in-phase, I(t), signals. Effectively, the Q(t) channel is 90° phase-shifted relative to I(t) channel. To prevent aliasing, cutoff frequency of the low-pass filter should below the Nyquist frequency. The analog to digital convertor converts continuous-time signal to discrete-time sampled signal. Now that we have I and Q samples, we can perform the magic of digital signal processing.

4. Tayloe Detector

The product detector is shown in Fig. 4. Note that the detector is extremely simple in design. The incoming RF signal is routed via a common resistor and a RF multiplexer, to one of four sampling capacitors. This 1:4 multiplexer is commutated at a rate of 4 times the desired detection frequency. The 4x commutated frequency charges each capacitor to the quarter cycle of the input RF at the desired frequency.
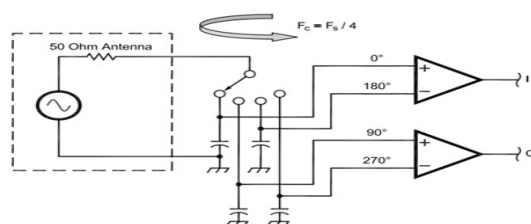

Fig. 4: Tayloe Detector to produce I and Q samples.

Mixers generally generate sum and difference outputs. In zero IF applications, the difference frequency is only used and the sum is thrown away. This design is not a mixer, but rather can best be described as "switching integrator", producing only a difference frequency. The input resistance and particular detection capacitor act together as an integrator, averaging the signal over the quarter cycle sample to the detection capacitor(0°, 90°, 180° and 270°).

If we differentially add the outputs of 0° and 180° detection capacitors with an op amp (see Fig. 4), the output would be a dc voltage equal to 2 times the value of the

individual sampled values when the switch rotation frequency is equal to the carrier frequency. The same would be true for the 90° and 270° capacitor as well. The 0°/180° summation forms the I channel and the 90°/270° summation forms the Q channel of the quadrature down conversion.

The real payoff in the Tayloe detector is its performance. It has been stated that the ideal mixer has a minimum conversion loss of 3.9dB [3, 4]. Typical high-level diode mixer have a conversion loss of 6-7dB. The Tayloe detector has less than 1dB of conversion loss, remarkably due to the design which uses discrete-time sampling theory which is similar to mixing but has its own unique characteristics.

5. Software and Design Approach

When the RF signal is converted to baseband audio signal, the output we can visualize as varying amplitude along a single axis as illustrated in Fig. 6. We will refer to this as the in-phase or I signal. In demodulation of AM signals, only amplitude information is required to fully extract the original signal. The problem is with other modulation techniques which also require the phase of the signal. Here quadrature detection technique can be used. We can find the instantaneous amplitude and phase of the original signal by using original in phase signal with a delayed copy of RF carrier signal by 90° (quadrature (Q) signal).
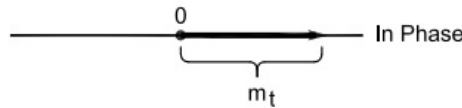

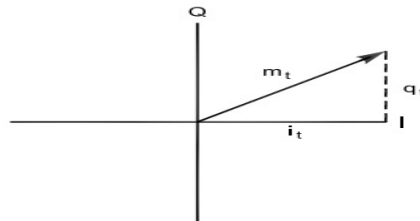**Fig. 6** An in-phase signal on the real plane.

Fig. 7 I and Q signal are shown in complex plane.

Fig. 7 illustrates a RF carrier with in-phase (I) signal plotted on the x-axis and the quadrature (Q) signal plotted on the y-axis of a plane. Now to extract magnitude and phase, we use the properties of right angled triangles.

$$m_t = \sqrt{I_t^2 + Q_t^2}$$ Eq. - 2

$$\phi_t = tan^{-1}(\frac{Q_t}{I_t})$$ Eq. - 3

The hypotenuse as magnitude with respect to time Eq. - 2. The instantaneous phase of the signal can be measured counterclockwise from the positive I axis and may be computed by Eq. - 3. Thus the in-phase and quadrature signal gives the ability to demodulate virtually any type of signal.

BeagleBone Black (BBB) is the processing power of the system. Using its onboard 12bit ADC, we converted the in-phase(I) and quadrature(Q) signal in digital codes and further processing is done on them. For designing SDR Receiver following functions are implemented in software:

Take the samples from ADC and implement a circular buffer structure for real time system.

Conversion from time domain to frequency domain using a fast-Fourier transforms (FFT).

Implementing amplitude and phase correction algorithm.

Cartesian-to-polar conversion of the signal vectors.

Sideband Selection.

Band-pass filters coefficient generation.

FFT fast-convolution filtering.

Conversion back to the time domain with an inverse fast Fourier transforms (IFFT).

Digital automated gain control (AGC).

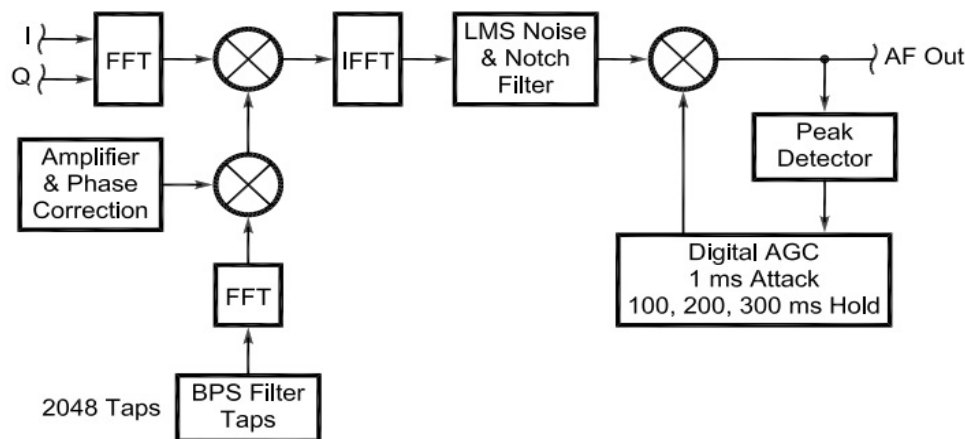Transfer of the processed signal to the output buffer for transmitting to DAC.

Fig. 8: DSP software architecture block diagram.

Digital signal processing for SDR requires that we be able to digitized I and Q data through BBB onboard ADC, process those signals and return them to the DAC in pseudo real time. This is called full duplex. For doing this, we implement circular double buffer in input as well as in output side to provide uninterrupted operation in real time.

Fig. 8 provides architecture of the DSP software. By changing few lines, this architecture can be used for both transmitter and receiver. But this paper is only describing the receiver part.

6. Simulation Results

Demodulation algorithm is implemented for amplitude modulated (Fig. 11).
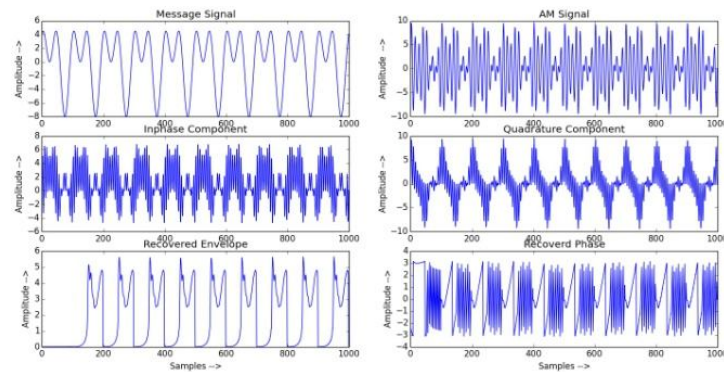

Fig. 11: AM Simulation Results.

Fig. 11 shows the amplitude modulated signal with extracted in-phase and quadrature component and by applying FFT convolution with filter coefficient we get envelope of the message signal.
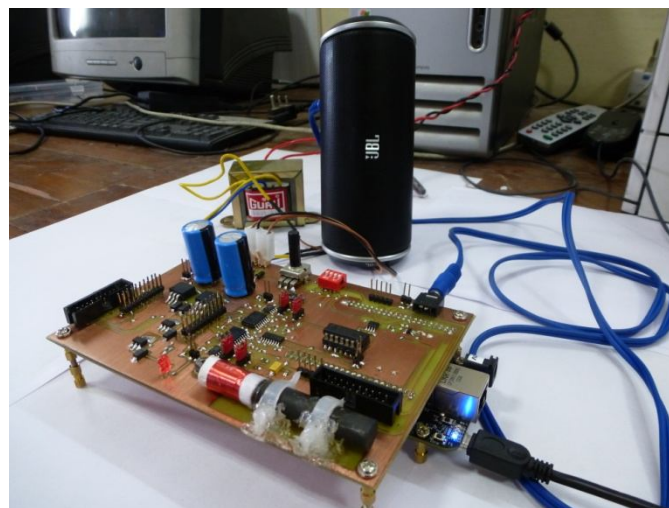

Fig. 13: SDR System. BeagleBone Black with RF Front End and Speaker.

7. Conclusion

This paper has presented a practical approach to high-performance SDR development. Concept of Tayloe detector is successfully implemented in this system with simulation and hardware results.

References

[1] Gerald Youngblood, "A software defined radio for the masses" QEX, Vol. 75, No 20, pp. 827-830, 2002.

[2] D. Tayloe, N7VE, "Letter to the Editor, notes on "Ideal" Commutating Mixers (Nov / Dec 1991), " QEX, March/April 2001H.

[3] D. Smith, KF6DX, "Notes on 'Ideal' Comutating Mixers," QEX 1999 Nov/Dec, pp 52-54.

[4] P. Chadwick, "Letter to the Editor, Notes on Ideal' Commutating Mixers" (Nov/Dec 1999), QEX, Mar/Apr 2000, pp 61-62.

[5] D. Hershberger, W9GR, and Dr S. Reyer, WA9VNJ, "Using the LMS Algorithm for QRM and QRN Reduction," QEX, 1992 Sep, pp 3-8.

[6] FFT convolution information can be found in the following references: R. Lyons, Understanding Digital Signal Processing, (Addison-Wesley, 1997) pp 435-436; M. Frerking, Digital Signal Processing in Communication Systems (Boston, Massachusetts: Kluwer Academic Publishers) pp 202-209; and S. Smith, The Scientist and Engineer's Guide to Digital Signal Processing (San Diego, California: California Technical Publishing) pp 311-318.

[7] Overlap/add method: Ibid, Chapter 18, pp 311-318; M. Freirking, pp 202-209.