The Front end of Software-Defined Radio: Possibilities and Challenges

Roel Schiphorst, Fokke W. Hoeksema and Cornelis H. Slump

University of Twente, Department of Electrical Engineering,
Laboratory of Signals and Systems,
P.O. box 217 - 7500 AE Enschede - The Netherlands
Phone: +31 53 489 4037 Fax: +31 53 489 1060
R.Schiphorst@el.utwente.nl

Keywords: Mobile communication, front-end receiver architectures, software-defined radio (SDR).

Abstract

The use of mobile telephony has shown a spectacular growth in the last 10 years. A side effect of this rapid growth is an excess of mobile system standards. Therefore, the Software-Defined-Radio (SDR) concept is emerging as a potential pragmatic solution: it aims to build flexible radio systems, which are multi-service, multi-standard, multi-band, re-configurable and re-programmable, by software.

First, this paper presents a global overview of SDR. Furthermore, it discusses several front-end architectures of SDR. The goal of this project is to generate knowledge about designing part of the functionality of SDR, implemented by rapid prototyping strategies. The focus is on the front end of SDR. The technological roadmap is taken into account to evaluate several architectures.

1. Introduction

Since the early 1980's the use of cellular mobile systems has grown enormously. Nowadays, mobile communication has become a major worldwide business. A side effect of this rapid growth is an excess of analog and digital mobile system standards such as TACS, GSM, DCS-1800, IS-95 CDMA, etc. In fact, every major country has its own standard(s). Efforts to define a unique worldwide standard result often in a new, extra standard. Furthermore new standards, such as UMTS are not one standard but a cluster of substandards.

The large number of standards is not only troublesome for manufacturers but also for consumers. Manufacturers have to develop a new telephone for each standard. This results in extra development costs and small divided markets. It is also disadvantageous for consumers because they cannot use their mobile telephones wherever they want. A unique common worldwide standard has benefits, but the competition between Asians, Europeans and American industries makes it very difficult.

It is for this reason that the software-radio concept is emerging as a potential pragmatic solution: a software implementation of the user terminal able to dynamically adapt to the radio environment in which the terminal is located [1, 2]. Aside from the standardization issues, one should also view the software radio concept as a means to make users, service providers, and manufacturers more independent of standards. The benefits of this approach are that air interfaces may, in principle, especially be tailored to the specific needs of a particular service for a particular user in a given environment at a given time. For a manufacturer, a single design is sufficient for the whole world and consumers can use their telephones in every country [3].

Because of the analog nature of the air interface, a software radio will always have an analog front end. In an ideal software radio, the analog-to-digital converter (ADC) and the digital-to-analog converter (DAC) are positioned directly after the antenna. Such an implementation is not feasible due to the power that such device would consume and other physical limitations [4]. It is therefore a challenge to design a system that preserves most properties of the ideal software radio while being realizable with current-day technology. Such a system is called a software-defined radio

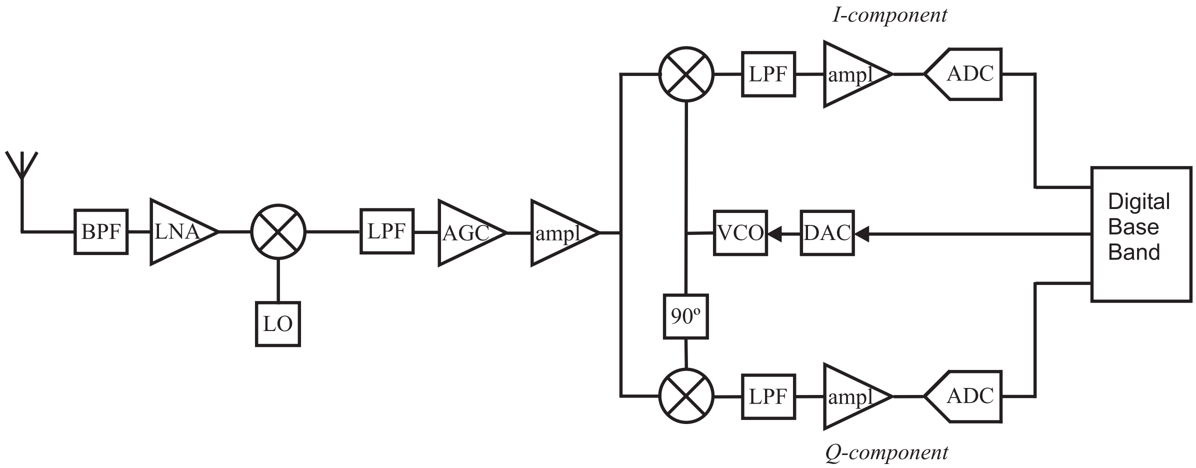


Figure 1: Conventional heterodyne receiver.

(SDR). The first serious attempts to build an SDR were made in the context of military applications (see e.g. [5]). In the last few years, a strong interest in the civil application of SDR has grown. This becomes clear from the long list of commercial companies that have joined efforts to cooperate on the standardization of such systems in the non-profit organization SDR Forum [6]. In addition, several consortia are actively involved in research in the field. An example is the SORT project financed by the European Union Advanced Communications Technology and Services (ACTS) programme [7].

An SDR will have one or more of the following properties [8]:

- a flexible transceiver architecture that can be controlled and programmed by software;
- radio functions that are mainly computed by digital signal processing;
- reprogrammability (the possibility to download new software) via the air interface;
- support of multiple modes and standards.

The reprogrammability can be used for:

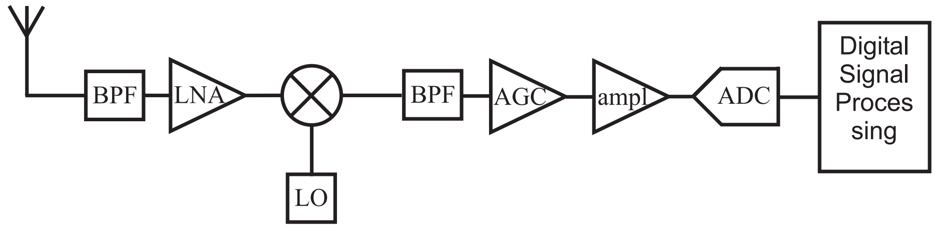
- the frequency band and the channel bandwidth;
- the modulation and coding scheme;
- the radio resource and mobility management protocols;
- user applications.

2. Front-end Architectures

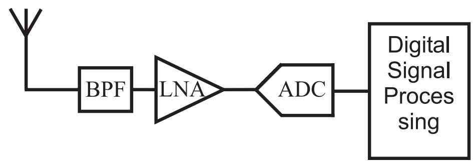
Nowadays, the heterodyne receiver architecture (Figure 1) is mostly used in mobile communications. It has several analog stages for selecting one user channel. Only the baseband (BB) stage is implemented digitally, usually built in dedicated hardware. In (Figure 1) the signal is picked up by the antenna. The next step is to filter the signal with a band-pass filter (BPF) and to amplify it with a low-noise amplifier (LNA). The resulting system band is converted to a lower frequency band by multiplying it with a local oscillator (LO). A low-pass filter (LPF) isolates the down-converted system band. Then the analog gain control (AGC) block intends to normalize the signal power for an optimal use of the analog digital converter (ADC). The next step is to isolate one channel from the system band. First the signal is multiplied with a voltage-controlled oscillator (VCO). The Digital-Base-Band block controls this VCO. A digital analog converter (DAC) is then used to convert the digital control signal of the Digital-Base-Band block to an analog signal. After the signal is multiplied with the frequency of the VCO, the signal is filtered with a LPF and finally sampled (ADC). Because some mobile system standards use quadrature modulation techniques, both the in-phase (I) and quadrature-phase (Q) component are extracted and sampled. These two bit streams are sent to the digital baseband processing. This block also controls the channel selection.

The advantages of this architecture are the proven concept and the low power consumption. The disadvantage of this architecture is the fixed narrow-band passive components that do not fit in a broad-band system with multi-mode operation. Therefore a general-purpose common Radio Frequencies (RF) stage is required for standards with different RF specifications. These specifications include carrier frequency, bandwidth, modulation scheme, and transmission power [9]. The ability of the software-radio architecture to support a communication waveform is predominantly determined by [10]:

- the largest instantaneous signal bandwidth (W);



(a) Digital IF



(b) Utopical software radio

Figure 2: Two front-end architectures

- the frequency range and bandwidth of the RF stage;
- the A/D-converter sampling rate (greater than 2W);
- the maximum dynamic range;
- Digital-Signal-Processor (DSP) throughput requirements including translation of Intermediate Frequencies (IF) to baseband, modulation, demodulation, coding, and decoding.

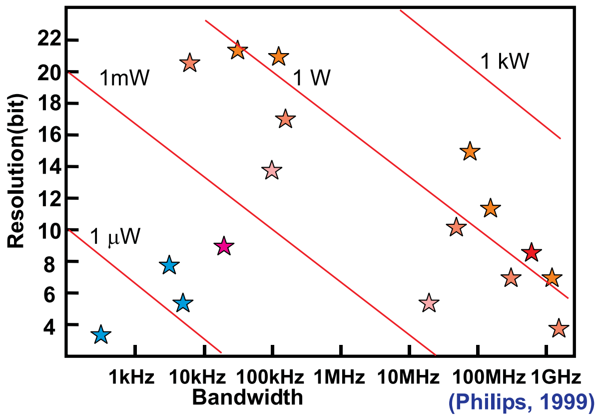
Figure 2 shows two designs of a general-purpose common RF stage. Figure 2(a) shows the digital IF architecture. The first stage of the receiver equals the heterodyne receiver architecture. Then the signal is converted into the digital domain by the A/D converter with digital channel. This design is more flexible than the conventional heterodyne receiver architecture but needs for each standard a separate front end. Figure 2(b) represents the ideal software radio architecture. The A/D converter is placed directly after the antenna in order to maximize the re-programmability of the system. This design is not feasible because it would require A/D converters which have a sample rate of more than 2 GHz (with a resolution of several bits). Figure 3(a)) shows several commercial available A/D converters. There is a tradeoff between resolution, bandwidth and power. Extrapolating current A/D-converter characteristics the A/D converter for SDR would consume about 1 kW. This is far too much for mobile applications. Furthermore the digital processing requirements are several thousands of MIPS. At this moment there are no DSPs or other processors which are capable of delivering several thousands of MIPS at an acceptable power level. (A typical power level for a mobile telephone is about 10 mW).

The progress in A/D converters with respect to power consumption is slow, about 1,5 bit in 8 years [4] at the same power level (at the same sample frequency). On the other hand the progress in the digital domain is very fast, the instruction-energy consumption is decreasing and the processing capabilities are increasing. Figure 3(b)) shows for several processors the instruction-energy consumption. This is mainly caused by down-scaling of the production process to smaller features sizes. So we may expect that there is enough processing power in the near future but no fast A/D converters. For that reason the ideal software radio (Figure 2(b)) will remain a utopy for a long time and another solution is required.

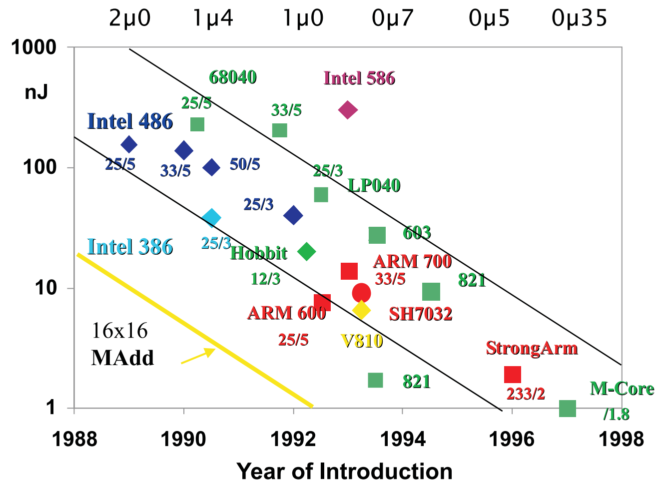
3. Challenges

The goal of this project is to find an architecture that is both flexible and feasible. Important questions arise such as:

- new standards, what will bring (e.g. other modulation techniques, smart antennas)?
- what architecture to choose?
- how to partition analog/digital parts?
- how to partition flexibility (ASIC/FPGA/DSP)?
- what should be implemented in hardware and what on a general processor in software?



(a) Commercial available A/D converters in 1999



(b) Roadmap digital signal processors: instruction-energy consumption for several processors

Figure 3: Technological roadmaps

The answers on these questions depend on the technological roadmap: the trend is to implement more and more functionality digitally. Furthermore, the architectural solution depends on analog and digital technology, so an approach from both analog and digital perspective is essential.

Acknowledgement

This research is carried out in cooperation with the Integrated Circuit Design Laboratory of the University of Twente. Furthermore, this research is supported by the PROGram for Research on Embedded Systems & Software (PROGRESS) of the Dutch organization for Scientific Research NWO, the Dutch Ministry of Economic Affairs and the technology foundation STW.

References

[1] J. Mitola, "The software radio architecture," *IEEE Communications Magazine*, pp. 26–38, May 1995.

[2] J. Mitola, *Software Radio Architecture: Object-Oriented Approaches to Wireless Systems Engineering*, Wiley, 2000.

[3] R. Schiphorst et al., "The exploration of the software-defined radio concept by prototyping transmitter and receiver functions on a digital signal processor," *PROGRESS 2000 Workshop on Embedded Systems, Utrecht, The Netherlands*, 2000.

[4] R.H. Walden, "Performance trends for analog-to-digital converters," *IEEE Communications Magazine*, pp. 96–101, February 1999.

[5] D.W. Lackey, R.J. Upmal, "Speakeasy: The military software radio," *IEEE Communications Magazine*, pp. 56–61, May 1995.

[6] "Software defined radio forum, a forum on open architecture reconfigurable wireless technology," `http://www.sdrforum.org/`.

[7] "Ac315: Sort - software radio technology," `http://www.ifn.et.tu-dresden.de/~sort/welcome.html`.

[8] E. Buracchini, "Sort and swradio concept," *4th ACTS Mobile Communications Summit (AMOS), Sorrento, Italy*, 1999.

[9] H. Tsurumi and Y. Suzuki, "Broadband rf stage architecture for software-defined radio in handheld terminal applications," *IEEE Communications Magazine*, pp. 90–95, February 1999.

[10] J. E. Gunn, "A low-power dsp core-based software radio architecture," *IEEE Journal on selected areas in communications*, vol. 7, no. 4, pp. 574–589, 1999.