

Modern Digital Communications: A Hands-On Approach

A Brief Introduction to Software-Defined Radio

Dr. Nicolae Chiurtu

- Course material of Prof. Bixio Rimoldi -

Last revision: Sept. 17, 2021

In this course, we implement real (as opposed to simulated) wireless communication via software-defined radio.

What is software-defined radio and what makes it possible?

Software-Defined Radio: What It Is NOT

Many of today's communication devices have a programmable interface but this does not make them qualify as "software-defined radio" systems.

They do not qualify since the processing of the communication signals is, to a large extent, "hard-wired". It is specified during the design phase according to some standard (e.g. GSM, EDGE, LTE, 4G, 5G, ...) and it is implemented electronically in VLSI.

Such systems implement the standard(s) for which they have been designed. The development of such systems takes relatively long and evolutionary upgrades are not possible.

Software-Defined Radio: What It Is

In very general terms, when we communicate digital information, the transmitter may be seen as a map from bits to signals.

These signals are bandlimited — hence they are specified by samples taken at regular intervals (in the sense of the sampling theorem).

For the purpose of this class, software-defined radio means that

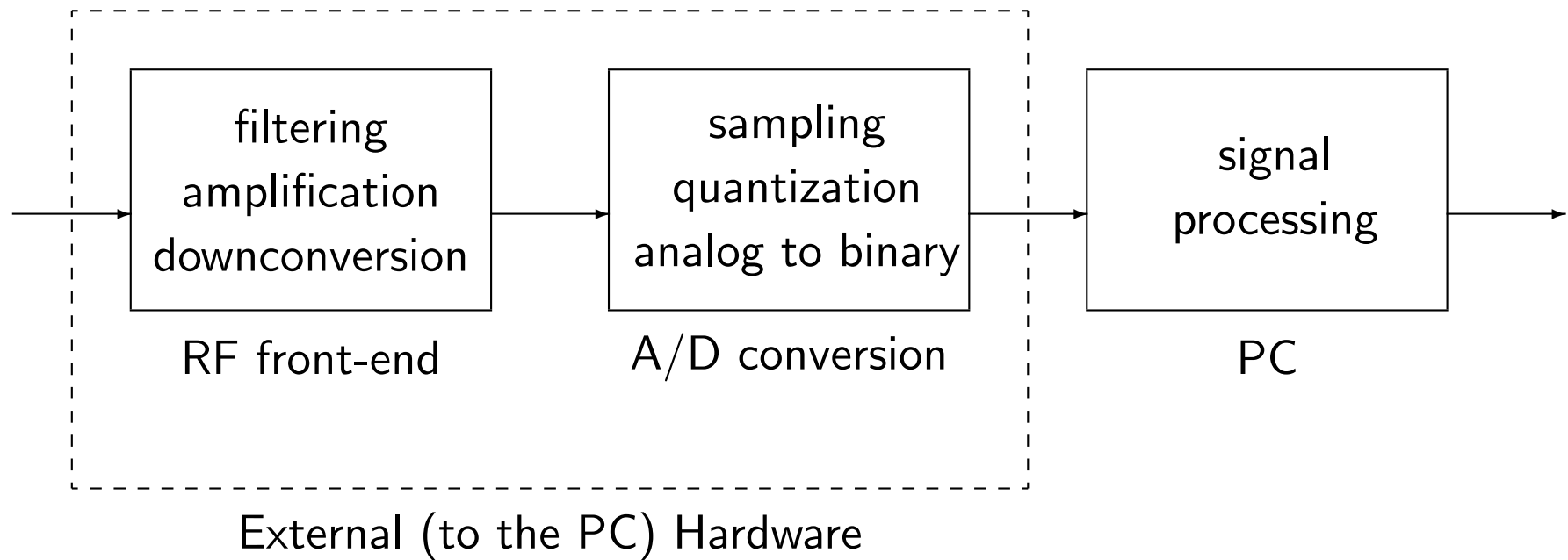
- at the transmitter, the mapping from the information-carrying bits to the samples of the corresponding signal is implemented as a computer program.
- at the receiver, the mapping between the samples of the received signal and the reconstructed information-carrying bits is implemented as a computer program.

The conversion from samples to signal at the transmitter back-end is done by a device which is unaware of the communication standard being used: it just implements the “interpolation” in the sense of the sampling theorem.

Similarly, the receiver front-end just takes samples in the sense of the sampling theorem.

This means that all we need in terms of hardware is a computer of adequate performance and a standard-independent hardware extension.

The following block diagram represents a software-defined radio receiver



Some Implications of Software-Defined Radio

1. Practical: You download/install a standard from the internet when needed.
2. Financial: The general purpose hardware is cheaper.
3. Evolutionary: New algorithms can reach the consumer much faster.
4. Know-How: Less/no EE knowledge required to implement a system (starting from the general purpose hardware).
5. Infrastructure: A small university lab can build one.
6. Flexibility: One device, unlimited number of standards.
7. There are also drawbacks (greater power consumption, larger hardware size).

What Makes Software-Defined Radio A Reality

1. The sampling theorem.
2. Inexpensive and powerful DSPs.
3. Inexpensive and powerful real-time operating systems (e.g. RT Linux).
4. Adequate signal processing development tools (e.g. MATLAB/Python).