# Chapter 12
# The Future of Software-Defined Radio: Recommendations

**Waqar Hussain, Jouni Isoaho, and Jari Nurmi**

An efficient Software-Defined Radio solution comes when all the aspects of system design are collectively addressed under application specifications and constraints. It includes all—the efforts to design wideband antennas, powerful software to process huge bandwidth of information, optimizations at hardware to maximize performance, and nevertheless to mention compilers and operating systems. It is important that every engineer or a scientist working on a particular block of SDR should have a bare-minimum understanding of the entire design stack. There is a need to have clear vision about the targets to be achieved, trade-offs to be made, and a unified approach so that all the objectives are measurable to enable a qualitative and quantitative analysis.

The contributions as chapters in the books provide conclusions and based on those we can indicate recommendations for future designs. Computationally intensive parallel processing tasks are suitable for homogeneous/heterogeneous multicore architectures. The cores in such platform can be chosen based on application scenarios, for example a general-purpose task is more suitable for a RISC processor to handle or signal processing task to a DSP. Special logic can be implemented on FPGAs and rASIPs can be used to allow maximum flexibility. The cores can communicate with each other over a network-on-chip or on an efficient bus architecture. If the objective is to provide maximum instruction-level parallelism, then multiple VLIW cores can be integrated together over

W. Hussain (✉) • J. Nurmi
Department of Electronics and Communications Engineering,
Tampere University of Technology, Tampere, Finland
e-mail: waqar.hussain@tut.fi; jari.nurmi@tut.fi

J. Isoaho
Department of Applied Physics, Laboratory of Electronics and Information Technology,
University of Turku, Turku, Finland
e-mail: jisoaho@utu.fi



W. Hussain et al. (eds.), *Computing Platforms for Software-Defined Radio*,
DOI 10.1007/978-3-319-49679-5_12

a high-performance communication infrastructure. In such cases, LLVM-based C-compiler StreamIt-based compilers are optimal choices. Reconfigurable MPSoCs is also a good solution for SDRs as they are highly programmable and require a minimum knowledge of the underlying hardware by the SDR programmers. Open source libraries such as GNU radio will attract more programmers towards the SDR development. From a scalable communications core, we can expect that it yields several advantages including architectural efficiency, efficient interconnects, targeted flexibility and control as well as programmability. Tool development is also very important for consistent SDR evolution. We need tools that can do rapid prototyping, provide early synthesis results and near-accurate performance estimates. The tools should contain support for multiple compilers of different programming languages.

In the end, we can emphasize there is a need to continuously invent new applications to keep a high demand for SDR technology. All those interesting features that this books highlights come at a cost, i.e., power and area/resources. An SDR world—present day Internet-of-Things can become an environmental hazard due to the unprecedented energy requirements. It is urgently required that special emphasis should be given to the global view and idea of SDR design paradigm from all fields of technology.