Hardware and software architectures for the Software Defined Radio

Mots clés :
- Directeur de thèse : RENAUD PACALET
- Co-encadrant(s) :
- Unité de recherche : Laboratoire Traitement et Communication de l'Information
- Ecole doctorale : Ecole Doctorale Informatique, Télécommunications, Électronique de Paris
- Domaine scientifique principal: Divers

Résumé du projet de recherche (Langue 1)

Upcoming wireless standards like LTE or WIMAX, while promising enhanced throughputs impose new requirements on wireless transceivers in terms of scalability, inter-operability and down-compatibility. Such requirements are typically incompatible with low-power and area efficiency required in mobile terminals. Targeted data rates for LTE are foreseen to be up to ten times higher than today's wireless standards through use of advanced signal processing and coding techniques. This directly influences ASIC size and power consumption. Unfortunately, CMOS technology improvements predicted by Moore's Law are such that just scaling down architectures to reduce size and power within new technologies will never be enough to sustain the increased complexity. The down-compatibility and the inter-operability is also a mandatory feature in the wireless industry. Once again architecture size and cost makes it impossible to simply duplicate dedicated modem architectures. New architectures are therefore required to fill-in the gap between these requirements of new modems and state-of-the-art implementation technology. Several studies have already been carried out in the area of Scalable Architectures but nothing conclusive has yet been proposed for the specific requirements of low-power mobile wireless transceivers. Optimized architectures therefore need to be defined associating various techniques to take care of the specific constraints of wireless devices. The Mobile Communication department of Eurecom Institute (http://www.eurecom.fr/) and TELECOM ParisTech (http://www.telecom-paristech.fr/) work together to design an open prototype for a multi air interfaces mobile terminal. The multi air interfaces feature will cover OFDM and W-CDMA based interfaces. Most functions are common to the different standards but with different parameters. The digital base band processing (physical layer) of many air interfaces were analyzed and this led to the identification of several ASIC-like processing engines. These engines are highly parametrizable and offer a whole range of similar capabilities that can be used to implement all the different targeted standards. They are plugged together on an interconnect, embed a local memory space that is also mapped in the global memory map of the system, and are driven and synchronized by a 32 bits micro-controller. In order to reduce the workload for the CPU, the engines also embed a DMA engine and an 8 bits micro-controller. Thanks to these, an OFDM-like channel estimation, for instance, is launched with a simple call to one of the processing engines, the Front End Processor (FEP). The 8 bits local micro-controller runs a small routine that chains together all the DMA transfers and elementary processing. When completed the routine raises an interrupt and the main CPU takes the appropriate next actions. This architecture is a complex mixture of software and hardware components. Programming such a machine to support complex situations like, for instance, a vertical soft handover between a UMTS TDD and a LTE communication, is thus a challenging problem. This thesis proposes solutions to the software design of advanced, next generation, wireless systems. We defined formally DiplodocusDF, a domain-specific modelling profile for SDR applications. It extends the UML profile DIPLODOCUS to describe data flow applications. Also, it extends the syntax and semantics of DIPLODOCUS to support automatic code generation of executable code (DIPLODOCUS was initially intended for design space exploration). We propose in this thesis a compilation environment which translates models described in DiplodocusDF to applications in C language. In the generated code, the main control software runs on a main CPU and controls the entire system, distributing and synchronizing all the processing tasks among the different digital signal processors. The local micro-controllers run composite tasks (pieces of software to chain sequences of processing tasks) thus free the main CPU from fine grain control. Several case studies were conducted on different OFDM based standards to demonstrate our methodology and compilation environment.