

Implementation of component based wireless communication protocol for SDR AT

Junsik Kim¹, Sangchul Oh¹, Hongsoog Kim¹, Namhoon Park¹, and Nam Kim²

¹ Mobile Telecommunication Research Division
Electronics and Telecommunications Research Institute, Daejeon, Korea
{junsik, scoh, kimkk, nhpark}@etri.re.kr

² Dept. of Comput. & Commun. Eng., Chung-Buk Nat. Univ., Cheongju, Korea
namkim@chungbuk.ac.kr

Abstract. In this paper, we propose the SCARLET (SCA Reconfigurable middleware of ETRI) which is a middleware platform for the SDR (Software Defined Radio) AT (Access Terminal) and supports the SCA standard (Ver2.2) of JTRS. It guarantees the portability and reusability of the software, hardware, and also ensures the compatibility of mutual software component among products based on the SCA standard. The application component implementing the wireless communication protocol is arranged to the common terminal hardware based on this policy and requirement. Specifically, we propose and implement a design of the SDR AT middleware and the wireless protocol application components.

Keywords: SCA; SDR; Reconfiguration; Software Component; Middleware;

1 Introduction

It is expected that multiple radio access standards and systems will coexist in the same environment beyond 3G. SDR has made itself a key enabling technology in order to realize such a flexible and reconfigurable radio system. Therefore, it is considered as the core technology for positively coping with the change of this mobile communications market to one among the solutions working out the problem of not only the communications market but also the broadcasting market being faced. The SDR technology introduces the open architecture to the programmable digital radio technology. When it implements all kinds of the modeled devices, it could maintain an independency between each function module (the hardware, and the software). It provides the hardware reconfiguration ability, software programmability, flexibility, scalability, etc. It is consequently expected to form a new framework for the wireless telecommunications system implementation in the future.

In this paper, we propose a SDR AT developed on the basis of the SCA. The AT is built on the advanced SDR hardware platform on which provides an inherent

* This work was supported by the IT R&D program of MIC/IITA. [2006-S-012-01, Development of Middleware Platform Technology based on the SDR Mobile Station]

flexibility to support multiple air interface standards with different capacity ranges. Specifically, we propose a design of the SDR AT middleware and wireless protocol application components. The remainder of this paper is as follows. We describe an overview of the SCA which is a software platform for SDR and a flexible and reconfigurable environment in Section 2. In Section 3, we present the software, hardware architecture and functionalities of SDR AT, and section 4 includes application components and software platform architecture. In section 5, we describe the test environment and procedure of application component operation. Finally, conclusions are drawn.

2 SCA Overview

Ease of technology insertion, software reuse of waveform, which is radio access protocol and application, and software porting are feasible if the software architecture hides low-level system details[1]. The SCA hides a low level hardware and software details via;

- Encapsulation of hardware dependencies exclusively in SCA Device software which presents common interfaces defined in the Common Object Request Broker Architecture (CORBA) Interface Definition Language (IDL).
- The use of CORBA to hide details of the architecture, particularly the number and type of processors, their operating systems and communication mechanisms.
- A standard mechanism to describe the system and application configuration, the SCA Domain Profile[2]-[4].

The SCA defines an Operating Environment (OE) and specifies the services and interfaces that the applications use from the environment. The interfaces are defined by using CORBA IDL and graphical representations are made by using Unified Modeling Language (UML). The OE consists of a Core Framework (CF) that is the essential set for the open interface, a CORBA middleware and a POSIX-based OS.

The CF describes the interfaces, their purposes and operations [5]. It provides an abstraction of the underlying software and hardware layers for software application developers. The SCA compatible systems must implement these interfaces. The interfaces are grouped into Base Application Interfaces, Framework Control Interfaces and Framework Services Interfaces. The CF uses a Domain Profile to describe the components in the system. The Domain Profile is a set of XML files that describe the identity, capabilities, properties, inter-dependencies, and location of the hardware devices and software components that make up the system. The application components of the SCA are divided into two parts; CORBA components and Non-CORBA components. The communication between CORBA components and Non-CORBA components is made through a SCA adapter [6]-[8].

3 Platform for SDR AT

3.1 Hardware Platform

The functional modules on hardware platform are composed of mother board, processor board with GPP(General Purpose Processor), FPGA(Field Programmable Gate Array) for base band modem, IF board, RF module and reserved DSP board. For user interface ports, Ethernet port, USB2.0 port, UART port and JTAG for FPGA downloading/debugging are provided. The main control processor, PowerPC embedded in XC2VP30 operates at 150 MHz (Upto 300MHz is available according to Xilinx data sheet). 128 Mbyte FLASH is used for ROM and 128 Mbyte SDRAM is used for RAM. Two 1Mbit DPRAM(CY7C028) are provided for reserved data memory for MAC hardware. The both of RF and HSDPA modem are under construction, therefore, at the HSDPA mode, the lower transport layer of MAC is connected to LAN through a modem simulator and tested.

3.2 Software Platform

The main function elements of SCARLET are described in Fig. 1. They are comprised of the operating system, the CORBA middleware part, function components of the CF operated on CORBA that is the domain management unit, the device management unit, the XML configuration file and parsing unit, the application management unit, the component updating unit, and the external interface unit for the interface with Waveform Manager.

- **DomainManager** : The domain management unit performs the control and configuration of the file management and system domain.
- **DeviceManager** : The device management unit performs the function of managing the device component and interface with Device, loadableDevice, executableDevice, aggregateDevice of Framework Control Interface [1].
- **ApplicationManager** : The application management unit is comprised of compositional elements relating to the application management. It is the Application and the ApplicationFactory interface among the framework interface for the control of the system. The AssemblyController component does the resource controller part for the assembly of the components comprising one application. .
- **XML Parser** : The XML parsing part performs the function of the general XML parser. It converts configuration file into data type in which the program within SCARLET can use the Domain profile prepared to the XML format. For this, the XML parsing part provides the interface which the external program can use in the form of library. Moreover, it performs the function of detecting the error of the XML file, etc.
- **Component Update** : It is the function of renewing the application component of an error or the old version with the new application component. This function interfaces with the Core Framework part, and is not in the open interface of the

SCA CF standard but is the proprietary interface. The external operation from a user is received and started. After the operation of the application component is stopped and the corresponding component is removed, the component which it newly renews is inserted in the removed component column and it operates. An application has to be the CORBA component for the alternate interaction between the newly updated application, and existing application. And an interface has to be the open standard between the components.

- **External Interface** : This function manages an interface with the Waveform Manager in case the Waveform Manager is the Non-CORBA based system. The external interface part converts the message of the Non-CORBA based human-machine interface part into the interface of the CORBA based application component and delivers. It performs the message conversion function which is similar in the message of the opposite direction. The external interface part can omit in case the Waveform Manager is the CORBA based information network system.

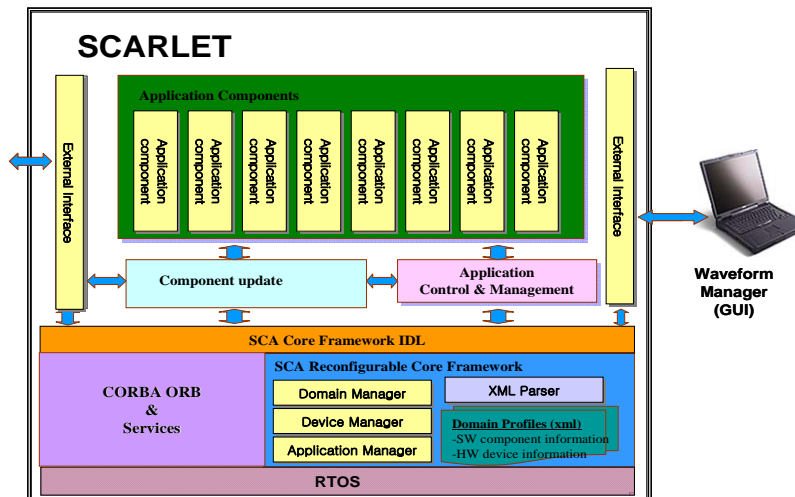


Fig. 1. SCARLET architecture

3.3 Waveform Manager

The waveform manager performs the management and the control function including the user interface for SCARLET middleware platform reconstructing the SCA base, and diagnostic function for the SDR AT. All kinds of manager software exist in the CF hierarchical layer for the reconfiguration of the component based application software. And it operates through the standard interface. SDR AT consists of TE(Terminal Equipment) and UE(User Equipment). The Waveform Manager is connected to the SDR TE, and it controls and manages the operation of the SCA application component software through an interface with the system. Moreover, it is developed in the GUI(Graphic User Interface) software environment(Fig. 2.) of the

CORBA based and the system information is given and taken through the SCA application component and CORBA interface. And the event information is outputted. The SDR AT and Waveform Manager are connected with LAN and the CORBA based message delivery is made.

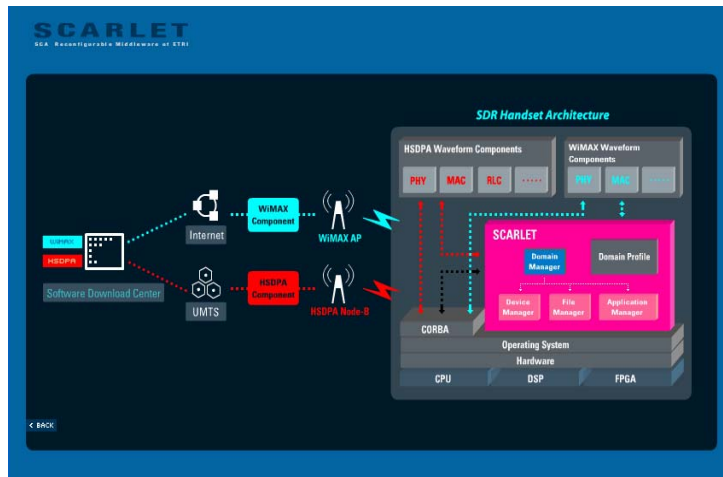


Fig. 2. Waveform manager GUI

4 Application component design

4.1 Application component architecture

Component software technology for maximizing software reuse is greatly helpful to overcome the extreme complexity of embedded software and reduce time-to-market. The component software technology is aimed at creating new software systems through the combination of deployable software, as opposed to ground-up development. We adopt the OMG's (Object Management Group) [9] definition of a component: a modular, deployable, and replaceable part of a system that encapsulates implementation and exposes a set of interfaces [10]. Components are typically in the form of shared library, depending on the deployment environment, and could be distributed as binaries, byte code, or even source files written in a scripting language.

The components for WiMAX and HSDPA were developed (Fig. 3), and performed under the SCARLET environment. The communication protocol of each mode utilized the used legacy protocol program which was developed before. Moreover, XML defining the connection structure between the application component, an attribute, etc. was built. It could be seen through a coupling with a middleware that complied with the standard of SCA in each application components.

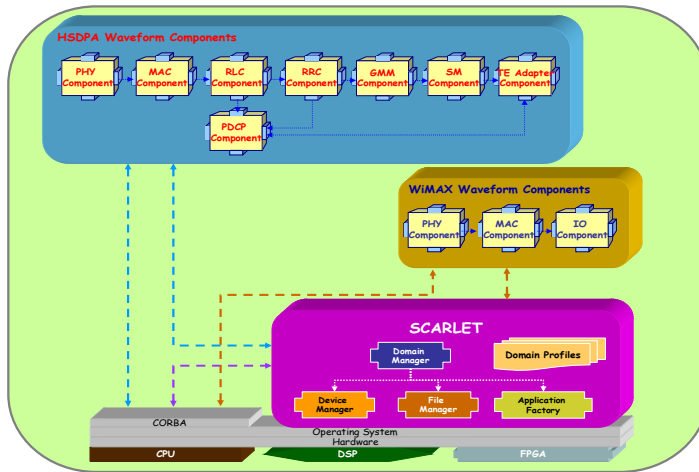


Fig. 3. Application component and middleware

4.2 Common function

The SCARLET system can be classified into the WF(Wave Form) subsystem and the CF subsystem. The software application components belonging to the WF subsystem operate with the CF subsystem through the interface as depicted in Fig. 4.

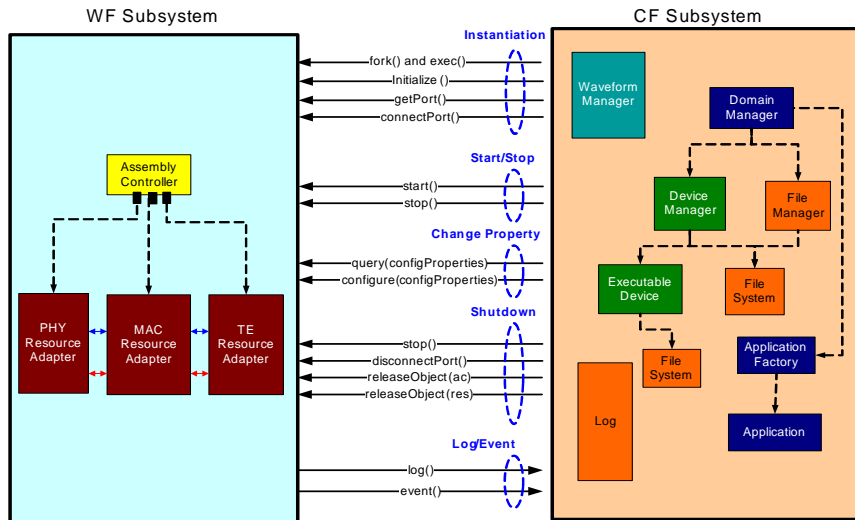


Fig. 4. Interface with CF

- **Instantiation** : The application components operate on the Executable Device with the `fork()` and `exec()` and are initiated by application with `Initialize()`.

Moreover, the getPort() and connectPort() are used for the port information reference and attach of each component.

- Start/Stop : In Application, each application component is stopped through the stop(), and start() interface with a beginning the service.
- Change Property : In Application, the query() and configure() interface are used in order to inquire and change the property of the application component.
- Shutdown : In Application, the stop(), and the disconnectPort() and releaseObject() are used in order to make the application component shutdown.
- Log/Event : The application component delivers its own log information and event information to the log component and Application of the CF subsystem, through the log() and event() interface.

4.3 WiMAX Application component

As follows the Fig. 5, the WiMAX waveform application components are the WACB block which is assembly controller block, the WPAB block which is physical hierarchical layer adapter block, the WMAB which is MAC layer adapter block, and WIAB block which is the adapter block for the IO Resource and TE. The WACB block mutually interfaces with the WPAB, WMAB, and WIAB, which is the application resource of the WiMAX mode with WPAB, WMAB, and WIAB in order to perform the start / stop / property control (Fig. 5). WPAB, WMAB, and WIAB take charge of the interface function with the legacy software performed through the protocol layer of WiMAX and the SCA. Each one changes the existing legacy protocol software into the application component using adapter method complying with the standard of SCA and operates under the SCARLET environment.

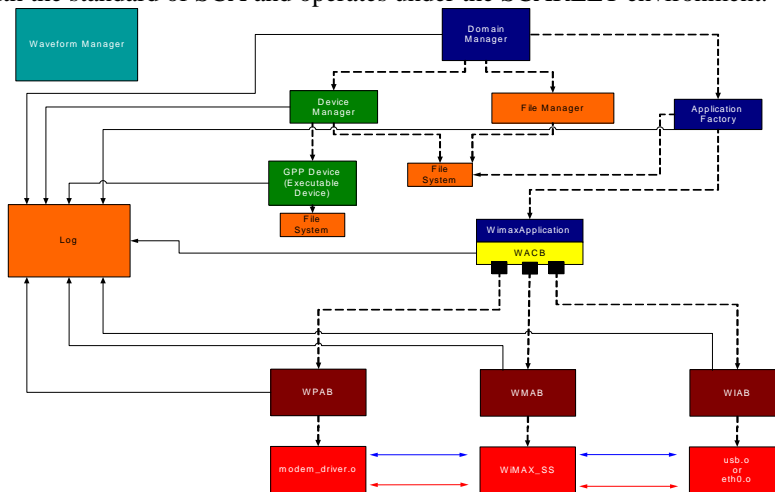


Fig. 5. WiMAX Architecture and Interface

4.4 HSDPA Application component

As follows the Fig. 6., the HSDPA application consists of 9 blocks. The HACB (HSDPA Assembly Controller Block) performs the start / stop / property control function of the HPHYB, HMAACB, HRLCB, HRRCB, HPDCPB, HGMMB, HSMB, and HTEAB resources which is the components of the HSDPA waveform application. It is installed on the SCARLET domain. It registers itself at the Naming Service so that the other components can approach itself.

The HTEAB (HSDPA TE Adapter Resource Block) performs the role of interface between TE and UE. The HSMB (HSDPA SM Resource Block) performs the role of setting up the packet switched call and release. The HGMMB (HSDPA GMM Resource Block) performs the registration for the packet switched service and the authentication function. The HRRCB (HSDPA RRC Resource Block) performs the delivery of all parameters for using the radio resource between UE and UTRAN, and it controls each. In the HRLCB (HSDPA RLC Resource Block) performs trustworthy data transmission between higher layer and MAC, the HPDCPB (HSDPA PDCP Resource Block) performs the function it compresses IP packet header through the traffic path generated with RRC. The HMAACB (HSDPA MAC Resource Block) classifies the performing function and the MAC protocol which transmits the data falling through the logical channel of the upper layer with the transport channel of the lower or transmits the data coming up to the transport channel of the lower with the upper layer through the logical channel of the upper layer according to MAC-d, MAC-c, and the MAC-hs entity and manages the respectively different transport channel and the MAC-hs protocol moreover handles transmitted data through HS-DSCH[11]. It manages the physical resources allocated for HSDSCH. In MAC, the HPHYB (HSDPA PHY Resource Block) transmits the data falling through the transport channel with the physical channel or performs the function of transmitting the received data with the physical channel through the transport channel to MAC.

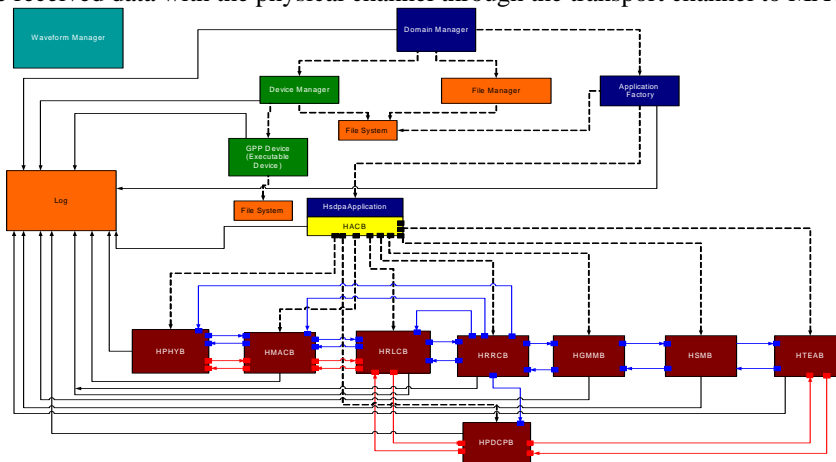


Fig. 6. HSDPA Architecture and Interface

5 TEST

The tests for verifying the SCARLET system consist of the application component test for the SCARLET, and middleware platform test. Fig. 7. shows the both test environment. The middleware platform and application component are verified on the test hardware platform. Moreover, the remote waveform Manager operates with the SCARLET middleware in terminal platform. After the HSDPA application components are set up on the test hardware in case a user desires the HSDPA service, the HSDPA service is provided. And after the WiMAX application components are set up on the test hardware platform in case a user desires the WiMAX service, the WiMAX service is provided.

Developed application components were smoothly performed under the common hardware platform and middleware. And it confirmed that the reconfigure operations including the loading / cancel/ release / perform / pause / mode switching, etc. were smoothly performed according to the direction of the CF. The applications like the web browser or the video player for the actual service, were performed in the TE.

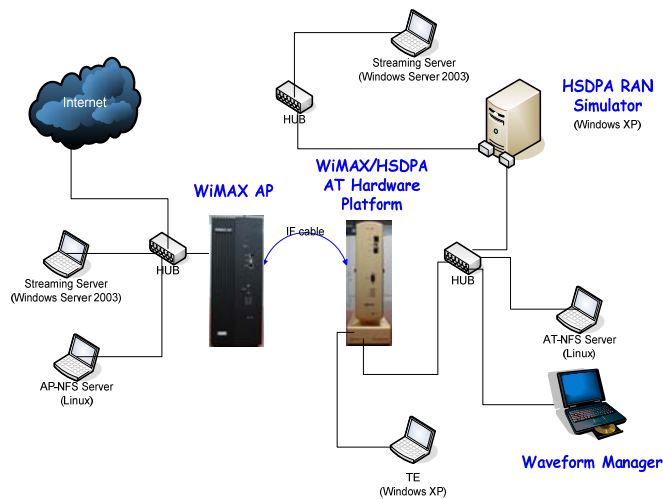


Fig. 7. Test environment

Using the Waveform Manager, the application components of the target mode are activated, and call processing is performed. The video on demand test which uses the internet service and Streaming Server are performed on TE. And, the WiMAX / HSDPA terminal platform and WiMAX AP were performed by using the NFS (Network File System) individual Server for the convenience of development.

Finally, by performing each service for each mode, right operation of the waveform application components and SCARLET were verified.

5 Conclusion

This paper represents a reconfigurable SDR AT developed on the basis of the SCA standard architecture and the WiMAX and the HSDPA service. We choose the bidirectional communication service combination of HSDPA and WiMAX in consideration of usability, performance and new wireless mobile communication technology. In order to support the standards of SCA for waveform application component, existing protocols were changed by using Adapter pattern. It currently adopted an interim architecture using a SCA adaptor. However, if there are more advances in the SCA technology as well as the SDR hardware, the architecture of the SDR AT will be upgraded for the system performance. Specifically, we designed SCA-based SDR AT software platform. The further research of reducing the reconfiguration time, FPGA componentization, and software modem improvement should be done.

References

1. Software Communication Architecture(SCA) Specifications MSRC-5000SCAv2.2, 17 Nov. (2001)
2. Chia-Ching Lin, Hung-Lin Chou, Chin-Lien Chiu, Min-Chiao Wang and Shiao-Li Tsao: Design of a SDR Software Framework, SDR Forum document SDRF-02-I-0019-V0.00.
3. Shiao-Li Tsao, Chia-Ching Lin, Chin-Lien Chiu, S. Hung-Lin Chou and Min-Chiao Wang: Design and Implementation of Software Framework for Software Defined Radio System, Proceedings of the Vehicular Technology Conference 2002-Fall, Vol.4, September 24-28, (2002) 2395-2399
4. Saehwa Kim, Jamison Masse, Seongsoo Hong and Naehyuck Chang : SCA-based Component Framework for Software Defined Radio, Proceedings of the IEEE Workshop on Software Technologies for Future Embedded Systems 2003, May 15-16, (2003) 3-6
5. Joint Tactical Radio Systems SCA Developer's Guide, Contract No. DAAB15-00-3-0001, V1.1, June (2002)
6. JTRS website, <http://www.jtrs.army.mil>
7. <http://www.sdrforum.org>
8. Eun-Seon Cho, Chang-Ki Kim, Yeon-Seung Shin and Jin-Up Kim : SCA-based multi-LAN application development, Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th Volume 3, 26-29 Sept. (2004) 1978 – 1982
9. Object Management Group (OMG), <http://www.omg.org>.
10. Unified Modeling Language Specification Version 1.4 Appendix B - Glossary, Object Management Group, September (2001)
11. <http://www.3gpp.org>