

# 다중모드 SDR 단말을 위한 동적 컴포넌트 재구성 알고리즘 연구

권오준<sup>†</sup>, 오상철<sup>\*\*</sup>, 박남훈<sup>\*\*\*</sup>

## 요 약

본 논문에서는 WiMAX 와 HSDPA 모드를 모두 지원하는 SCA 기반 다중모드 SDR 단말의 컴포넌트를 설계하고 블럭도를 제시하였으며, 실제 실험을 통한 경량화 대상 분석을 통해 SDR 단말 구현 시 어느 구성요소를 경량화해야 하는 지 분석해 보았다. 또한, 컴포넌트 재구성 분석을 통하여 동적 재구성 전체 시간 측정을 위한 일반 식을 도출하였고, 분석 결과를 토대로 SCA 규격의 인터페이스와 수행절차를 위배하지 않는 범위 안에서 총 22단계에 걸친 컴포넌트 레벨 동적 재구성 알고리즘을 설계하였다.

## A Study on the Dynamic Reconfiguration Algorithm of Components for Multi-mode SDR Terminal

Oh-Jun Kwon<sup>†</sup>, Sang-Chul Oh<sup>\*\*</sup>, Nam-Hoon Park<sup>\*\*\*</sup>

## ABSTRACT

In this paper, we first design the block diagram of the components for multi-mode SDR terminal based on SCA technology. It support both WIMAX mode and HSDPA mode. Then from the result of the analysis with a real SDR terminal implementation, we analyze which components of it have to be more lightweight. In addition we derive the general formula for the measure of dynamic reconfiguration time on the reconfiguration of the components. Finally we present a dynamic component-level reconfiguration algorithm with 22 stages based on the result of our analysis without violating the interface and procedure of SCA specification.

**Key words:** SDR(소프트웨어 정의 무선), Dynamic Reconfiguration(동적 재구성), Lightweight(경량화)

## 1. 서 론

최근 이동통신 분야에서 사용자들은 다양한 서비스를 저가에 제공받기를 원하며 이러한 욕구를 충족시키기 위해 다양한 형태의 이동통신 규격이 개발되어 상용화되고 있다. SDR(Software Defined Radio)

기술은 이러한 이동통신 시장의 변화에 적극적으로 대처하기 위한 핵심기술로써 IP 기반 이동통신망의 기반으로 한 다양한 무선통신 환경(다중 모드, 다중 표준, 다중 대역, 다중 기능)에 유연하게 대처하기 위하여 하나의 공통 하드웨어 플랫폼에 사용자가 원하는 응용 소프트웨어(무선 프로토콜 규격)로 자유롭

\* 교신저자(Corresponding Author) : 권오준, 주소 : 부산광역시 부산진구 엄광로 995(614-714), 전화 : 051)890-1725, FAX : 051)890-2629, E-mail : ojkwon@deu.ac.kr  
접수일 : 2008년 4월 23일, 완료일 : 2008년 7월 4일

<sup>†</sup> 종신회원, 동의대학교 컴퓨터소프트웨어공학과 부교수

<sup>\*\*</sup> 한국전자통신연구원 이동통신연구본부 모바일엑세스연구팀 선임연구원

(E-mail : scoh@etri.re.kr)

<sup>\*\*\*</sup> 한국전자통신연구원 이동통신연구본부 모바일엑세스연구팀장(책임연구원)

(E-mail : nhpark@etri.re.kr)

\* 이 논문은 2007년도 정부(교육인적자원부)의 재원으로 한국대학교육협의회 대학교수 국내교류 연구비 지원에 의한 것임.

게 재구성할 수 있는 개방형 신호처리 기술이다.

1970년대 초에 초기 SDR 프로토타입이 등장 했는데, 그 당시에는 대부분 초저주파에서 시도된 것들이었다. 또한 안테나 모듈에서 직접 아날로그/디지털 변환을 했고, 여기서 나온 디지털 신호를 처리하기 위해 소프트웨어가 사용되었다. 한편, 오늘날 등장하는 3G/4G 기술에서의 SDR 구현은 두 가지 기본적인 문제를 가지고 있다. 첫째 3G/4G에 할당되는 주파수 대역은 2GHz 이상이다. 그런 높은 주파수에서의 아날로그/디지털 직접 변환은 현재 기술로는 상당히 어려워져서 기술적인 도전과제로 남아 있다. 둘째, 다중 표준들과 프로토콜에서 요구하고 있는 무선 인터페이스의 복잡성은 하드웨어 측면에서 상당히 많은 프로세싱 전력을 필요로 한다. 이러한 이유로 인해 초기에 소프트웨어 재구성 능력은 프로토콜 스택의 상위 계층에 있는 것으로 고려되었다. 그러나 DSP 처리기의 성능 및 저전력화 기술의 발전에 따라 이런 재구성 능력은 스택의 하위 계층으로 점점 옮겨가고 있다.

일반적으로, 실시간 응용들을 위한 임베디드 시스템들은 소프트웨어와 하드웨어가 혼용된 시스템으로 구현되고 있다. 소프트웨어 부분은 시스템 특성들과 변경이 가능한 융통성을 담당하고, 반면에 하드웨어 부분은 요구되는 성능과 전력 절약을 위해 이용된다. 한편, 개선된 하드웨어 부분의 프로세싱 전력과 효율성으로 인해 고수준의 기능적인 프로토콜 요구 사항이 소프트웨어 부분으로 전환될 수 있었다. 또한 이것은 소프트웨어를 이용하여 다중 변조 형식들을 제어할 수 있게끔 하였다.

SDR 시스템의 기술 개발 촉진을 위하여 1996년 MMIT'S(Modular Multifunction Information Transfer System) Forum이 만들어졌으며, 1998년 SDRF(SDR Forum)으로 개칭되었다[1]. SDRF에서는 SDR시스템을 기능적 관점에서 정의하는 노력을 통하여, 상위 수준 기능 모델인 SDR 소프트웨어 참조 모델을 제시하였으며, 미국방성 JTRS(Joint Tactical Radio System) 프로젝트에서는 SDRF의 SDR 소프트웨어 참조 모델에 기반하여 컴포넌트 기반 프레임워크인 SCA(Software Communication Architecture)를 설계하였고 SDRF에 의해 SDR 소프트웨어 표준으로 채택되었다[2]. 이러한 SCA의 본래 목적은 새로 개발된 무선기술의 빠른 수용이 가능

하도록 하기 위해 하드웨어로부터 소프트웨어의 독립성을 최대한 보장하는 것에 있었다.

본 논문에서는 IEEE 802.16 WiMAX(Worldwide interoperability for Microwave Access) 모드[3]와 3GPP HSDPA(High-Speed Downlink Packet Access) 모드[4]를 둘 다 지원하는 다중모드 SDR 단말을 SCA 기반으로 구현할 경우에 필요한 컴포넌트들을 설계하고, SDR 단말 구현 시 어느 구성요소를 경량화해야 할 지 분석한다. 또한, 컴포넌트 재구성 분석을 통하여 동적 재구성 전체 시간 측정을 위한 일반 식을 도출하고, 통신모드 변경 시 반드시 발생하게 되는 컴포넌트 재구성에 필요한 알고리즘을 설계한다.

## 2. SCA 기반 다중모드 SDR 단말

지금까지도 다중모드 SDR 단말을 구현함에 있어서 SCA의 적용 유무는 많은 논란의 여지가 있을 수 있으나 본 논문에서는 이를 적용한 다중모드 SDR 단말을 가정하기로 한다.

그림 1은 본 논문에서 고려한 SCA가 적용된 다중모드 SDR 단말의 구조를 나타내고 있다. SDR 단말은 크게 TE(Terminal Equipment)와 AT(Access Terminal)로 구분되며 AT에는 실제 무선 프로토콜이 탑재되어 동작하게 되며, 모드 변경 시 XML 형태의 domain profile에 명시된 대로 CF(Core Framework)의 제어를 받아 CPU, DSP, FPGA 등 적절한 디바이스에 재배치되어 관련 재구성 절차를 수행하게 된다.

다중모드 SDR 단말의 재구성 요구 발생 시 가장 핵심이 되는 SCA CF는 어플리케이션 컴포넌트를 위한 표준화되고 개방적인 인터페이스와 서비스를

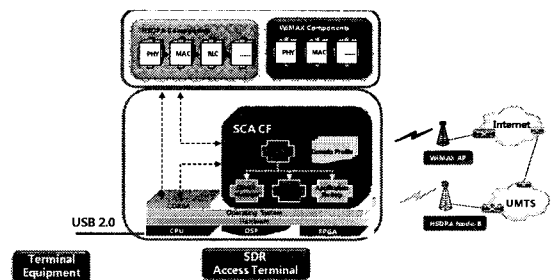


그림 1. SCA 기반 다중모드 SDR 단말의 구조도

제공함으로써, 어플리케이션 컴포넌트 설계자에게 하부 소프트웨어와 하드웨어 계층을 추상화해서 설계하도록 도와준다. SCA CF의 인터페이스와 서비스를 크게 구분하면 다음과 같다.

- **BAI(Base Application Interface):** 어플리케이션 컴포넌트가 구현하고 제공해야 하는 인터페이스다. Port, LifeCycle, TestableObject, PropertySet, PortSupplier, ResourceFactory, Resource와 같은 인터페이스가 있다.

- **FCI(Framework Control Interface):** 시스템 제어를 위해 SCA CF 내부에서 사용되는 인터페이스이고, 코어 프레임워크부 내부 기능요소들간에 제공/사용되는 인터페이스와 서비스들이다. Application, ApplicationFactory, DomainManager, Device Manager와 같은 인터페이스가 있다.

- **BDI(Base Device Interfaces):** 도메인내 logical 디바이스의 관리 및 구현을 제공하는 인터페이스이다. Device, LoadableDevice, Executable Device, AggregateDevice, DeviceManager와 같은 인터페이스가 있다.

- **FSI(Framework Service Interface):** File, FileSystem, FileManager, Timer와 같이 코어 프레임워크부 내부 응용 프로그램이나, 다운로드 관리부, 어플리케이션 컴포넌트와 같이 외부 응용 프로그램에 의해 사용되는 서비스이다.

- **Domain Profile:** SDR 단말장치의 하드웨어 속성을 정의한 디바이스 프로파일과 소프트웨어 컴포넌트를 위한 소프트웨어 프로파일이 있다. 형상파일은 XML 양식으로 작성된다.

WiMAX 나 HSDPA 웨이브폼(waveform)은 하나의 어플리케이션 단위로 패키징(packaging)되어 SCA CF에 의해 관리되고 실행되며, 어플리케이션 하나당 하나의 SAD(Software Assembly Descriptor) 프로파일을 갖는다. 또한 CORBA의 Naming 서비스와 Event 서비스를 받기 위해 CORBA와 연동한다. 컴포넌트간 IPC 통신은 CORBA의 IDL을 이용한 SCA Port 인터페이스를 이용한다.

### 3. 컴포넌트 설계

#### 3.1 WiMAX 웨이브폼

그림 2는 WiMAX 웨이브폼을 SCA V2.2.2 기반으

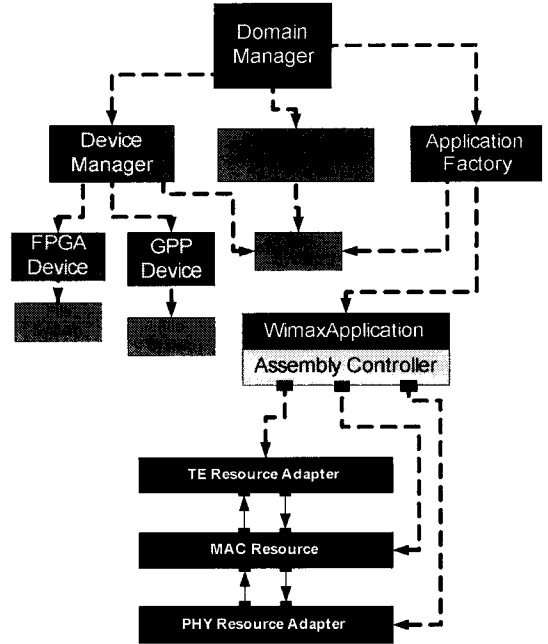


그림 2. WiMAX 웨이브폼을 위한 컴포넌트 블럭도

로 회로도를 설계했을 때 컴포넌트 블럭도를 나타내고 있다. SCA V2.1에서는 별도의 Log 컴포넌트를 정의했으나 SCA V2.2.2에서는 CORBA light-weight-Log 서비스로 대체 되었으므로 Log 컴포넌트를 설계에서 배제하였다.

IEEE 802.16에서 요구하는 PHY 기능을 GPP (General Purpose Processor)에서 수용하기에는 불가능하므로 순수 PHY 기능 컴포넌트는 FPGA로 구현하고 나머지 MAC 이상의 프로토콜을 GPP에서 동작하도록 설계하였다[3].

또한, 현재 SCA OE(Operating Environment)가 FPGA 지원이 불가능하므로 GPP에 있는 MAC 프로토콜 컴포넌트와 SCA Adapter 기능을 갖는 PHY Resource Adapter를 통해 통신을 한다. 그리고 마찬가지로 SCA OE가 제공되지 않는 TE와의 USB 통신을 위해 TE Resource Adapter를 둔다. SCA에서 정의하고 있는 기본 connection이 단방향이므로 양방향 통신을 위해 PHY Resource Adapter는 connection 2개를, MAC Resource는 connection 4개를, TE Resource Adapter는 connection 2개를 갖는다.

그리고 각 컴포넌트는 Assembly Controller를 통해 CF로부터 전달된 start()/stop() 등의 제어 정보를 수신한다.

3.2 HSDPA 웨이브폼

그림 3은 HSDPA 웨이브폼을 SCA V2.2.2 기반으로 회로도를 설계했을 때 컴포넌트 블록도를 나타내고 있다. 마찬가지로 SCA V2.1에서는 별도의 Log 컴포넌트를 정의했으나 SCA V2.2.2에서는 CORBA lightweight-Log 서비스로 대체 되었으므로 Log 컴포넌트를 설계에서 배제하였다.

IEEE 802.16 WiMAX와 마찬가지로 HSDPA도 3GPP 규격에서 요구하는 PHY 기능을 GPP에서 수용하기에는 불가능하므로 순수 PHY 기능 컴포넌트는 FPGA로 구현하고 MAC, RLC, RRC, PDCP, SM, GMM 등 나머지 MAC 이상의 프로토콜을 GPP에서 동작하도록 설계하였다[4].

또한, 현재 SCA OE가 FPGA 지원이 불가능하므로 GPP에 있는 MAC 프로토콜 컴포넌트와 SCA Adapter 기능을 갖는 PHY Resource Adapter 를 통해 통신을 한다. 그리고 마찬가지로 SCA OE가 제공

되지 않는 TE와의 USB 통신을 위해 TE Resource Adapter를 둔다. SCA에서 정의하고 있는 기본 connection이 단방향이므로 양방향 통신을 위해 PHY Resource Adapter 는 connection 2개를, MAC Resource는 connection 4개를, RLC Resource는 connection 6개를, RRC Resource 는 connection 8개를, PDCP Resource는 connection 6개를, SM/GMM Resource는 각각 connection 6개를, TE Resource Adapter는 connection 6개를 갖는다.

그리고 각 컴포넌트는 Assembly Controller를 통해 CF로부터 전달된 start()/stop() 등의 제어 정보를 수신한다.

4. 경량화 대상 분석

SDR 단말을 고려하였을 때 기지국보다는 모든 것이 열악한 환경인 단말에서 그 구성요소의 전체 크기가 좀 더 심각하게 다가온다. 또한 SCA를 단말에 적용함에 있어 그 크기에 대한 오버헤드를 우려하기도 한다. 그리고 적은 메모리와 낮은 프로세싱 파워는 잘 알려진 단말이 갖는 제약사항중에 하나다. 따라서 3장에서 설계한 컴포넌트 세부 회로의 블록도를 기준으로 IEEE 802.16 과 3GPP 실제 규격을 준수하여 구현해 봄으로써 단말 적용 시 경량화 대상에 포함되는지 아닌지에 대한 판단의 기준으로 삼고자 한다. 물론, 본 분석에 사용한 Binary 파일은 실제 WiMAX 서비스 또는 HSDPA 서비스 시험을 거친 결과물을 사용한다.

실험에 사용한 SCA OE 환경은 표 1과 같다. SCARLET(SCA Reconfigurable middleware of ETRI)은 한국전자통신연구원에서 SCA V2.2.2와 Minimum CORBA V1.0을 준수하여 개발한 제품을 사용하였다.

표 1. SCA OE 환경

구성요소	사양
CPU	Xilinx XC2VP30 200 MHz
SDRAM	128 MBytes
OS	MontaVista Linux Pro 3.1
CORBA	SCARLET ORB v.1.0
Parser	SCARLET XML Parser v.1.0
CF	SCARLET v.1.0

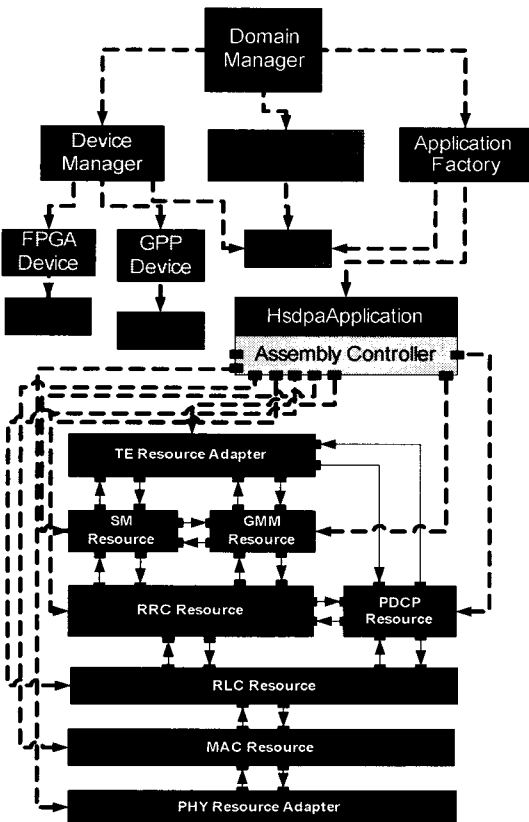


그림 3. HSDPA 웨이브폼을 위한 컴포넌트 블록도

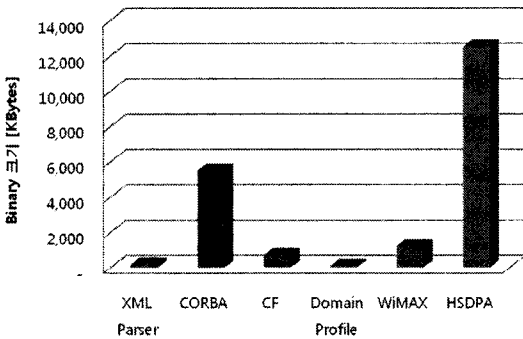


그림 4. SDR 단말 구성요소의 Binary 크기 분포도

그림 4는 본 실험의 결과로 나온 SDR 단말 구성요소의 Binary 크기 분포도를 보여 주고 있다.

본 결과에 따르면 기존의 우려와 달리 XML parser, domain profile, CF는 단말 적용에 있어서 그 크기 면에서 별 문제가 없으며 오히려 CORBA가 CF에 비해 7.6배정도 더 무거운 것을 알 수 있었다. 이는 SCA를 단말에 적용함에 있어서 경량화 대상이 CF가 아니라 CORBA라는 것을 의미한다.

WiMAX와 HSDPA 웨이브폼을 비교하면 HSDPA 웨이브폼이 10배 정도 더 무거운 것으로 나타났다. 이것은 단말 적용 시 HSDPA 웨이브폼 컴포넌트를 경량화 대상으로 삼아야 한다는 것으로 의미한다. 또한 WiMAX 웨이브폼은 오히려 CORBA보다도 4.5배정도 더 가벼운 것으로 나타났다.

결과적으로 본 실험을 통해 CORBA와 HSDPA 웨이브폼 컴포넌트를 경량화 대상에 포함시켜야 한다는 것을 알 수 있었다.

## 5. 재구성 분석

재구성이란 현재 시스템을 구성하고 있는 컴포넌트의 교체 또는 컴포넌트간 연결 관계의 변경을 말한다. 시스템의 재구성은 크게 정적 재구성과 동적 재구성으로 분류할 수 있다. 정적 재구성은 동작 중인 시스템을 정지하고, 재구성한 후 시스템을 다시 시작하는 과정으로 이루어진다. 반면 동적 재구성은 시스템을 다시 시작하지 않고 런타임 동안에 재구성하는 것을 의미한다.

정적 재구성은 재구성 중에 시스템을 정지해야 하기 때문에 동적 재구성에 비해 가용성(availability)에 큰 손실을 가져온다. 끊임 없이 다중모드 서비스

를 제공해야 하는 SDR 단말에 있어 가용성의 손실은 바람직하지 않다. 또한 정적 재구성 과정은 단순한 시스템 정지 및 재 시작으로 이루어지므로 개선의 여지가 적어 최적화 효과를 크게 기대하기 어렵다.

동적 재구성의 목적은 시스템 재 시작 없이 시스템을 현재 동작 중인 구성으로부터 다른 구성으로 변경하는 것이다. 시스템의 구성이라 함은 현재 시스템을 이루고 있는 소프트웨어 컴포넌트들간의 연결 구조를 말한다. 동적 재구성은 한 구성으로부터 다른 구성으로 변경할 때 시스템에 영향을 가능한 작게 미치도록 해야 한다.

### 5.1 동적 재구성 조건

동적 재구성은 런타임 동안에 컴포넌트를 생성하고 제거하거나 교체하는 등의 작업을 수반한다. 이와 더불어 컴포넌트간의 연결관계도 끊어지거나 새로 연결된다. 아무런 제약 사항 없이 이러한 작업들을 하게 되면 시스템의 일관성을 해치게 되어 결과적으로 동적 재구성 후의 시스템이 오작동을 일으킬 가능성이 있다.

따라서 시스템의 일관성 유지는 동적 재구성이 올바르게 동작하기 위한 가장 중요한 요구사항이다. Goudarzi는 그의 연구에서 동적 재구성 후에 시스템이 정확하게 동작하기 위해서 필요한 일관성에 관한 요구사항을 아래의 세 가지로 분류하였다[5].

- 시스템의 구조가 보전되어야 한다.
- 시스템을 구성하는 컴포넌트들은 상호 일관성을 유지해야 한다.
- 응용의 상태 정보가 유지되어야 한다.

동적 재구성에 관한 연구의 토대를 마련한 것은 Kramer와 Magee에 의해서였다. [6]에서 그들은 컴포넌트의 안전 상태를 정의하였다. 또한 컴포넌트가 안전 상태에 도달하였음을 보장하기 위해서 컴포넌트의 수동 상태를 정의하여 어떤 컴포넌트에 영향을 주는 다른 컴포넌트들을 수동 상태로 만듦으로써 컴포넌트를 안전 상태로 만들 수 있음을 보였다.

그러나 Kramer와 Magee의 방법은 하나의 컴포넌트를 안전 상태로 만들기 위해 과도하게 많은 컴포넌트들을 수동 상태로 만들어 수행을 방해한다는 문제점이 있었다. Goudarzi는 그의 연구[5]에서 서비스 요청들을 일시적으로 대기 옆에 넣어 처리를 지연시

김으로써 컴포넌트를 안전 상태로 만들 수 있음을 보였다. 그러나 무조건 서비스 요청의 처리를 지연시키면 교착상태 문제를 발생시킬 수 있다. Goudarzi는 교착상태 문제를 실제로 안전 상태로 도달할 필요가 없는 컴포넌트들의 수행을 일부 방해함으로써 해결하였다.

이후 Almeida는 그의 연구[7]에서 Goudarzi의 연구가 야기하는 교착상태 문제를 선택적 대기 열을 이용하여 해결하였다. 이를 이용하면 교착상태 문제를 해결하기 위해 추가로 다른 컴포넌트의 수행을 방해할 필요가 없어져 결과적으로 시스템의 충격을 줄일 수 있다.

5.2 동적 재구성 시간

그림 5는 컴포넌트 레벨 동적 재구성에 대해 설명하고 있다. 컴포넌트 B를 B+로 교체하는 경우 A와 C는 stop 상태에 있어야 하고, 교체 후에는 B+와 연결 설정을 다시 맺어야 한다. 좀 더 자세한 재구성 시간은 아래의 식을 이용해 설명할 수 있다.

식(1)은 전체 재구성 시간을 나타내고 있다. 여기서,  $T_{reconfiguration}$ 은 전체 재구성 시간을 나타내며,  $T_{old-stop}$ ,  $T_{old-disconnect}$ 과  $T_{old-shutdown}$ 은 각각 old 컴포넌트(B)를 stop 하는데 걸리는 시간, 연결을 끊는데 걸리는 시간, shutdown 하는데 걸리는 시간을 의미한다. 그리고 수식 중에서  $T_{new-instantiation}$ ,  $T_{new-connect}$ 와  $T_{new-start}$ 는 각각 new 컴포넌트(B+)를 instantiation하는데 걸리는 시간, 연결을 설정하는데 걸리는 시간, start하는데 걸리는 시간을 의미한다.

$$T_{reconfiguration} = T_{old-stop} + T_{old-disconnect} + T_{old-shutdown} + T_{new-instantiation} + T_{new-connect} + T_{new-start}$$

식 (1)

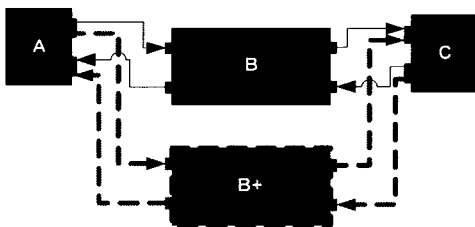


그림 5. 컴포넌트 레벨 동적 재구성

여기서 사용자로부터 재구성 명령에 의해 stop 명령을 받으면 컴포넌트 A, B, C는 모두 자신이 현재 수행하고 있던 서비스를 완전히 끝낸 후에 stop 상태로 갈 수 있으므로, 전체 컴포넌트 A, B, C 중 최대 지연시간을 갖는 컴포넌트가 최종적으로  $T_{old-stop}$ 에 영향을 미치게 된다. 그러므로  $T_{old-stop}$ 은 아래 식(2)와 같이 정의될 수 있다. 여기서 n은 stop 상태로 가야 할 전체 컴포넌트의 개수를 나타낸다. 그림 5의 경우 n=3.

$$T_{old-stop} = \max\{T_{old-stop}(1), \dots, T_{old-stop}(n)\}$$

식 (2)

결과적으로, 식(1)은 아래의 식(3)으로 나타낼 수 있다.

$$T_{reconfiguration} = \max\{T_{old-stop}(1), \dots, T_{old-stop}(n)\} + T_{old-disconnect} + T_{old-shutdown} + T_{new-instantiation} + T_{new-connect} + T_{new-start}$$

식 (3)

6. 재구성을 위한 CF 인터페이스 분석

그림 6는 웨이브폼과 CF 사이의 연동 인터페이스에 대한 설명을 나타내고 있다. SCA 기반 다중모드 SDR 단말은 이와 같이 WF(Waveform) 서브시스템과 CF(Core Framework) 서브시스템으로 구분할 수 있다. WiMAX나 HSDPA 웨이브폼 컴포넌트는 WF 서브시스템에 속한다. 이들은 CF 서브시스템과 표 2와 같은 인터페이스를 통해 연동한다.

그림 7은 그림 5에 있는 컴포넌트 B를 B+로 교체하는 경우, 본 연구에서 분석한 내용을 토대로 설계한 컴포넌트 레벨의 동적 재구성 알고리즘을 설명하고 있다. 그림 7과 같이 본 컴포넌트 레벨 동적 재구성 알고리즘은 미국 DoD의 JTRS에서 제시한 SCA 규격 V2.2.2를 준수하여 SCA의 최소 요구사항인 기능 계층간의 인터페이스 규격(Interface Specification), 애플리케이션 프로그램 인터페이스(Application Program Interface: API) 동작에 관련된 모든 명세 및 규칙(Behavioral Specification and Rule)을 준수하였다[2]. 제시된 알고리즘 단계별 설명은 아래와 같다. 아래의 단계별 번호는 그림 7의 단계별 번호와 일치한다.

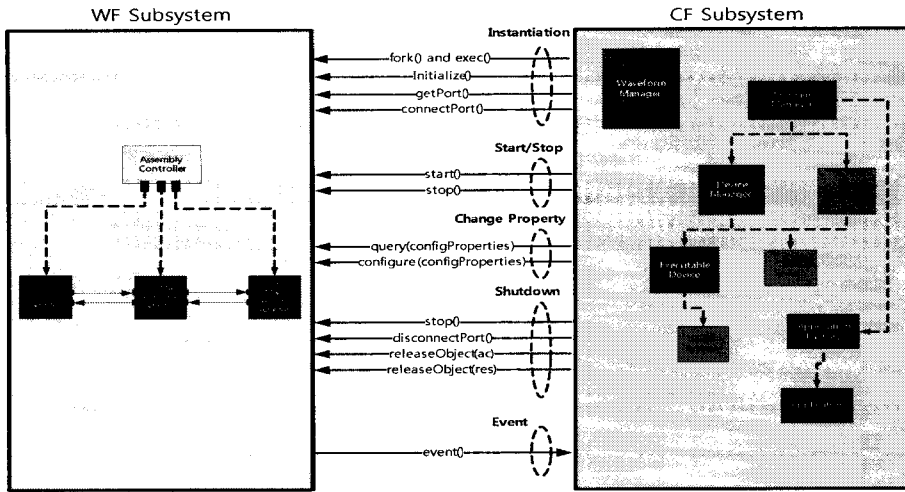


그림 6. CF 연동 인터페이스

표 2. 인터페이스 설명

인터페이스	설 명
Instantiation	Install 된 실행 파일을 특정 디바이스에 재배포 및 구동 웨이브폼 컴포넌트들은 Executable Device 에 의해 fork() 및 exec() 되고 Application 에 의해 Initialize() 된다. 또한 각 컴포넌트의 포트(port)정보 조회와 연결설정을 위해 getPort() 및 connectPort()를 사용한다.
Start	Instantiation된 실행 파일을 시작
Stop	Instantiation된 실행 파일을 중지
ChangeProperty	웨이브폼 컴포넌트의 property를 조회하고 변경
Shutdown	Instantiation된 실행 파일을 특정 디바이스에서 해제(종료 및 제거)
Event	웨이브폼 컴포넌트는 자신의 이벤트정보를 event() 인터페이스를 통해 CF 서브시스템의 Application으로 전달

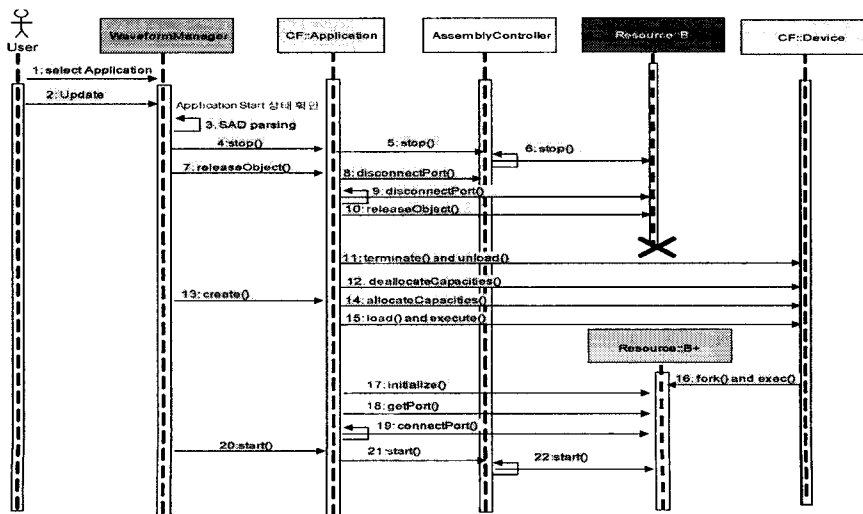


그림 7. 컴포넌트 레벨 동적 재구성 알고리즘 설계

1. 사용자가 update하고 싶은 application을 waveform manager GUI 화면에서 선택한다.
2. waveform manager GUI 화면에서 해당 application에 update 명령을 내린다.
3. waveform manager는 해당 application이 현재 이미 start된 상태인지 아닌지 검사한다. Start된 상태이면 domain profile을 parsing 한다. 이전 application에 Parsing 결과와 비교하여 update 되어야 할 컴포넌트만 추출한다.
4. 해당 application을 stop 시킨다.
5. Assembly Controller를 stop 시킨다.
6. 컴포넌트 B를 포함한 컴포넌트 A, C 모두를 stop 시킨다.
7. 해당 application에게 컴포넌트 B를 shutdown 시키도록 명령한다.
8. Assembly Controller에게 컴포넌트 B와 연관된 제어용 connection 을 해제하도록 명령한다.
9. 컴포넌트 A, B, C에게 컴포넌트 B와 연관된 connection을 해제하도록 명령한다.
10. 컴포넌트 B를 destruction 시킨다.
11. 해당 process를 terminate 시키고 메모리상에서 unload 시킨다.
12. 컴포넌트 B가 점유하였던 디바이스 자원을 반환한다.
13. 컴포넌트 B+을 생성하도록 명령한다.
14. 컴포넌트 B+가 사용하게 될 디바이스 자원을 할당한다.
15. 컴포넌트 B+를 메모리상에 load 하고 실행시키도록 명령한다.
16. 해당 디바이스의 child process로 컴포넌트 B+를 실행한다.
17. 컴포넌트 B+를 초기화한다.
18. 컴포넌트 B+의 Port 정보를 가지고 온다.
19. 컴포넌트 A, B+, C에게 컴포넌트 B+와 연관된 connection을 설정하도록 명령한다.
20. 해당 application을 start 시킨다.
21. Assembly Controller를 start 시킨다.
22. 컴포넌트 B+를 포함한 컴포넌트 A, C 모두를 start 시킨다.

## 7. 결 론

본 논문에서는 WiMAX와 HSDPA 모드를 둘 다

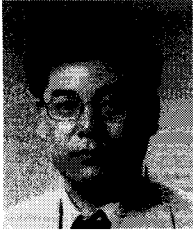
지원하는 다중모드 SDR 단말을 SCA 기반으로 구현할 경우, 필요한 각 모드별 컴포넌트 회로도를 설계하고 블럭도를 제시하였다. 이러한 컴포넌트 설계를 통하여 WiMAX 의 경우 총 4개의 SCA connection이 필요하고, HSDPA 의 경우 총 22개의 SCA connection이 필요함을 알 수 있었다. 그리고 각 구성요소의 경량화 대상 분석을 통해 CORBA와 HSDPA 웨이브폼이 주요 경량화 대상임을 알 수 있었다.

한편, 모드 변경 시 재구성 처리 과정이 반드시 필요하며, 여기에는 정적 재구성 방법과 동적 재구성 방법이 있으나 정적 재구성 방법의 경우 단순하고 시스템의 재 시작이 필요하다는 단점이 있다. 따라서 본 논문에서는 앞에서 설계된 내용에 대한 분석을 토대로 동적 재구성 방법에 대한 컴포넌트 레벨의 재구성 알고리즘을 SCA 규격에서 정의하고 있는 인터페이스와 수행절차를 해치지 않는 범위에서 총 22 단계에 걸쳐 설계하였다. 그리고 이러한 동적 재구성 전체 시간 측정을 위한 일반식을 제시하였다.

## 참 고 문 헌

- [1] <http://www.sdrforum.org>, Software Defined Radio Forum
- [2] <http://sca.jpeojtrs.mil/>, SCA Specification, SCA V2.2.2, May. 2006.
- [3] IEEE Std., Part 16: Air Interface for Fixed Broadband Wireless Access Systems, P802.16-REVd/D5, May. 2004.
- [4] <http://www.3gpp.org/>, 3GPP Web site
- [5] Goudarzi K.M. and Kramer J., "Maintaining node consistency in the face of dynamic change," *Configurable Distributed Systems, Proceedings of third International Conference*, pp. 62-69, 1996.
- [6] Kramer J. and Magee J., "The evolving philosophers problem: dynamic change management," *IEEE Transactions on Software Engineering*, Vol.16, Issue.11, pp. 1293-1306, Nov. 1990.
- [7] Almeida J.P.A., Wegdam M., van Sinderen M., and Nieuwenhuis L., "Transparent dynamic reconfiguration for CORBA," *Distributed Objects and Applications, Proceedings of 3rd International Symposium*, pp. 197-207, 2001.





권 오 준

1986년 경북대학교 전자공학과 (공학사)  
 1992년 충남대학교 전산학과 (이학석사)  
 1998년 포항공대 전자계산학과 (공학박사)  
 1986년~2002년 한국전자통신연구원 선임연구원

2000년~현재 동의대학교 컴퓨터소프트웨어공학과 부교수

2007년 7월~현재 한국전자통신연구원 이동통신연구본부 초빙연구원

관심분야 : 컴퓨터네트워크, 정보보호, 무선 인터넷, 패턴 인식, 인공지능



박 남 훈

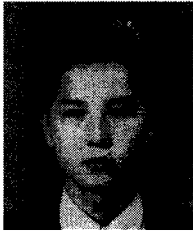
1984년 전남대학교 계산통계학과 이학사  
 1987년 중앙대학교 컴퓨터공학과 공학석사(시스템소프트웨어 전공)  
 1999년 충남대학교 컴퓨터과학과 이학박사(전산과학 : 이동통신 전공)

1995년 전자계산기조직응용기술사(P.E)

2002년 2월~2003년2월 : 한국무선인터넷포럼 모바일 표준플랫폼 분과위원장

1989년~현재 한국전자통신연구원 이동통신연구본부 모바일액세스연구팀장(책임연구원)

관심분야 : 컴퓨터 네트워크, 광대역 통신망/신호망, ATM 기술, 이동통신망, 차세대무선인터넷, 모바일 컴퓨팅기술, 이동통신 단말 기술, SDR 및 Cognitive Radio기술



오 상 철

1995년 광운대학교 전자통신공학과(학사)  
 1997년 광운대학교 전자통신공학과(석사)  
 1997년~2000년 대우통신(주)

2000년~현재 한국전자통신연구원 이동통신연구본부 모바일액세스연구팀 선임연구원

관심분야 : 3G/4G 이동통신망, SDR, 핸드오버, 유무선 프로토콜