

단말 프로토콜의 스트리밍 기반 데이터 전달 모델

진성근, 신경철, 김진업
한국전자통신연구원
(plukey, neuro, jukim)@etri.re.kr

Streaming based data transfer model for Mobile Equipment Protocol

Sung-Geun Jin, Gyung-Chul Shin, Jin-Up Kim
ETRI

요 약

이동 단말의 프로토콜 스택을 설계할 때 가장 많이 선호되는 기법은 프로토콜 스택의 계층을 타스크로 정의하고 타스크 사이의 통신을 위해서 메시지 큐를 이용하여 하위 계층에서 상위 계층으로 데이터를 전달하는 방식이다. 그러나 3세대 이상의 단말 시스템은 복잡한 응용 프로그램과 운영체제 환경을 가지게 되며 이러한 방식의 설계 방법은 더 이상 효율적이지 않게 되었다. 본 논문은 스트리밍 기술에 기반 한 단말 프로토콜 스택의 설계 방법에 대해서 소개하고 메시지 큐 방식에 기반 한 설계 모델보다 가용성이 높음을 보여준다.

1. 서 론

비동기 방식의 IMT-2000 시스템 개발 규격인 3GPP기반의 단말 시스템의 소프트웨어 계층은 물리계층에 대한 접속을 담당하는 L1계층과 데이터에 대한 재전송, 오류 검사, 암호화 등을 처리하는 L2계층 그리고 기지국에 대한 고 수준의 프로토콜 처리와 하위 계층에 대한 제어를 담당하는 L3계층으로 이루어진다. L1계층과 L2계층 L3계층에 대한 데이터 전달 방법은 각 계층사이의 프리미티브를 통해서 이루어지고 있으며 프리미티브는 3GPP규격을 통해서 정의되어있다.

프리미티브는 계층사이에 데이터를 전달하는 방법이며 동시에 계층사이의 제어 정보를 전달하는 방법이다. 규격에 정의되어 있는 프리미티브 방식을 충족시키는 단말 시스템을 위한 설계방법은 계층 사이의 데이터 전달 방법을 포함하고 있으며, 데이터 전달의 효과적인 처리는 단말 시스템의 성능에 영향을 준다.

프리미티브를 통해서 데이터 전달이 이루어질 때 데이터가 최상위 계층에 전달될 때까지 발생할 수 있는 지연은 계층의 연산을 통해서 이루어지는 지연과 계층 사이의 데이터 전달 시에 발생하는 지연으로 구분할 수 있다. 단말 시스템의 규격은 L1계층에서 L2 계층에 이르기까지 전달되는 모든 지연이 하나의 TTI내에서 이루어지도록 규정하고 있으며 단말 시스템이 열수 있는 모든 채널에 대해서 같은 조건을 요구한다[1].

계층 사이의 프리미티브 처리를 위한 설계 방법으로 메시지 큐를 이용하는 방법이 있다. 동기방식의 IS95 단말 시스템은 이러한 방법으로 프로토콜 스택을 개발 하였는데, 프리미티브를 처리하는 계층은 모두 타스크로 정의되어 있으며 타스크들은 하나의 메시지 큐를 가지고 있어서 다른

계층들이 보내오는 프리미티브를 처리한다. 메시지 큐 방식은 프로토콜 스택의 독립성을 지키며 데이터전달을 효과적으로 처리하는데 있어서 효과적인 방식이다. 그러나 IMT-2000시스템의 서비스는 단순히 음성 통화 서비스를 벗어나 게임, 화상 서비스, 인터넷 등 다양한 서비스를 목적으로 하고 있으며 다양한 서비스의 제공과 콘텐츠의 활용을 위해서는 복잡한 응용 소프트웨어가 수행되며, 이러한 요구를 충족시켜 주기 위해서는 매우 복잡한 운영체제가 채택된다. 과거의 음성통화만을 목적으로 프로토콜 스택만 수행되던 운영체제에서 벗어나 강력한 수행환경을 가지기를 요구하고 있다. 갈수록 범용화 되는 운영체제 환경에서 수행되는 메시지 큐 방식 기반의 프로토콜 스택은 계층사이의 데이터 전달을 위한 지연이 실시간성이 요구되는 프로토콜의 처리를 어렵게 한다. 새로운 개발 환경을 수용하고 단말 시스템의 계층간 프리미티브 처리를 위하여 새로운 방식이 요구된다. 본 논문은 스트리밍 기반의 프로토콜 스택을 이용한 프리미티브 처리 방법을 소개한다. 스트리밍 기반의 프로토콜 스택은 계층사이의 데이터 전달 시 발생할 수 있는 지연을 최소화 하며 데이터 처리를 위한 메모리 사용을 줄인다.

2장에서 메시지 큐를 이용한 프로토콜 스택과 문제점에 대해서 설명하며, 3장에서 대안으로써 스트리밍 기반의 프로토콜 스택에 대해서 소개한다. 4장에서 두 가지 방식 사이의 차이에 관해서 설명하며 5장에서 결론을 맺는다.

2. 메시지 큐 기반의 프로토콜 스택

메시지 큐 기반의 프로토콜 스택은 프로토콜의 계층 단위로 타스크를 정의하고 타스크 사이의 프리미티브 처리를 위한 기본 방식으로 메시지 큐를 이용한다. 프리미티브는 계층사이의 제어정보와 데이터를 포함하여 메시지를 통해서 타스크로 전달된다. 3GPP 방식의 단말 시스템에서 메시지 큐 기반의 프로토콜 스택은 [그림 1]과 같다.

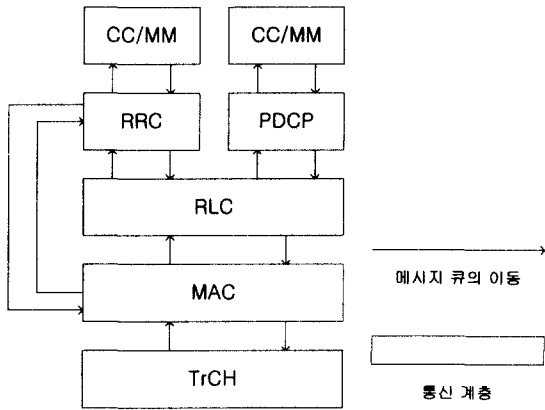


그림 1 프로토콜 계층 구조

메시지 큐 방식은 계층사이의 독립적 개발을 가능하게 하여 계층사이의 상관성을 줄여준다. 하나의 공유 가능한 자원에 대해서 계층사이의 경주 문제가 발생할 때 운영체제의 스케줄러에게 공유 자원에 대한 접근을 맡기는 효과가 되어 개발자의 부담을 덜어준다. 그러나 스케줄을 위한 타스크의 우선순위(priority)의 세밀한 조정이 필요하며 우선순위가 낮은 단말 시스템에서 수행될 다양한 응용 프로그램에 따라서 가변적인 값을 가지지만 그 값을 예측하기 힘들다. 뿐만 아니라 데이터 전달을 위하여 계층 사이의 지연을 유발한다. 메시지 큐를 통해서 데이터의 전달이 이루어지기 위해서는 데이터가 소스(source) 타스크가 관리하는 메모리에 포함되어 있다가 메시지 큐로 복사가 일어난 뒤에 다시 목적(destination) 타스크가 관리하는 메모리로 복사가 이루어져야 한다. 데이터 전달을 위하여 두 번의 복사가 필요하며 두 번의 복사를 위해서 소스와 목적 타스크에 같은 양의 메모리를 요구한다. TrCH계층에서 RRC계층까지 같은 작업이 반복적으로 발생하므로 각 타스크 사이에 필요한 메모리와 복사 횟수는 다음과 같다.

$$m = 3T \cdot L \cdot b \quad (\text{식 1})$$

$$n = 2L \quad (\text{식 2})$$

T 는 데이터의 점유시간이며, L 은 계층의 수, b 는 데이터의 요구 대역이다.

3. 스트리밍 기반의 프로토콜 스택

스트리밍 기술은 운영체제의 디바이스 드라이버를 위하여 발전된 기술이다. 스트리밍 기술의 목적은 재설정(reconfigurable) 가능한 디바이스 드라이버 개발과 드라이버의 계층화를 쉽게 하고 드라이버 사이의 데이터 전달을 표준화된 방법으로 운용하기 위한 것이다[2]. 3GPP 규격의 프로토콜 스택은 스트리밍 기술의 요소를 모두 가지고 있다. 계층화 되어 있으며 계층 사이의 데이터 전달이 이루어지며 수신을 위한 큐와 송신을 위한 큐가 분리되어 있다.

스트리밍 기술은 두개의 스트림 큐 즉 상향 스트림과 하향 스트림을 배대로 하여 계층에서 필요한 정보를 스트림을 통해서 얻은 후 계층의 처리결과를 다시 스트림으로 반환한다. 3GPP의 프리미티브는 데이터와 제어정보가 포함되어 있는 메시지이며 프로토콜 스택은 이러한 메시지를 통해서 처리되므로 스트림을 통해서 프리미티브를 지원하기 위하여, 프리미티브를 제어 정보와 데이터로 분리한다. 제어 정보는 스트림의 프로토콜 메시지이며 데이터는 스트림의 데이터로 매핑(mapping)된다. 단말 시스템 내부의 데이터 흐름을 기준으로 TrCH 단에서 RRC쪽은 RRC와 같은 수준의 계층으로 가는 상향식 스트림과 반대 방향으로 데이터 전달이 이루어지는 하향식 스트림을 가진다[그림 2].

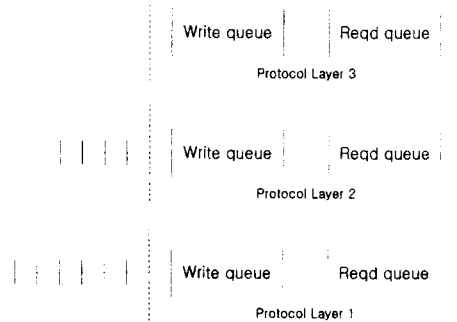


그림 2 프로토콜 스트림 구조

스트림을 통해서 전달 되는 메시지 프레임 구조는 [그림 3]과 같다.

TrCH를 통해서 데이터가 수신된 후 최상위 계층에 이르기까지 단일 큐 스트림을 통해서 데이터 전달이 이루어지며 단일 큐 내에서 메시지 프레임에 대한 접근은 포인터를 통해서 이루어지고 메시지 프레임 내에서의 계층 데이터에 대한 접근은 [그림 3]에서 정의된 메시지 접근 포인터를 통해서 이루어 지므로 데이터의 전달과 처리과정을 통해서 수신 및 송신 데이터의 실제적인 복사과정은 발생하지 않는다.

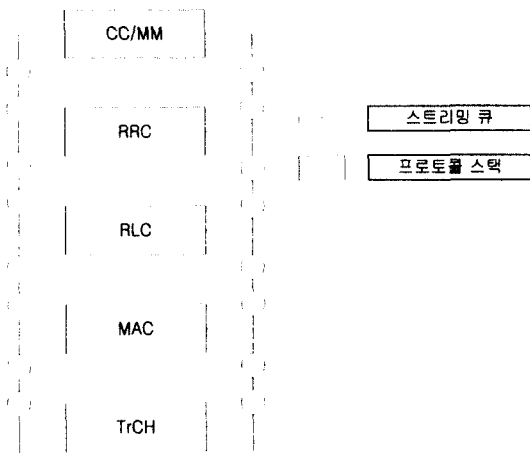


그림 3 스트리밍 기반 계층

데이터가 에어(air)를 통해서 단말 시스템에 수신된 후 최상위 계층에 이르기 까지 필요한 메모리와 복사 횟수 다 음과 같다.

$$m = T \cdot b \quad (\text{식 3})$$

$$n = 0 \quad (\text{식 4})$$

스트리밍 방식의 단말 시스템은 메시지 큐와 이를 이용 한 데이터 전달 방식을 사용하지 않으며 단말 시스템의 계 층을 task로 정의하지 않는다. 데이터 송수신 처리의 분 리와 계층의 능동적 처리 요구를 고려할 때 단말 시스템의 task는 데이터의 송수신을 위한 TrCH계층과 RRC, 음 성, 영상, 패킷 데이터의 생산 및 소비를 위한 task로 정 의한다.

4. 결론

2장에서 기존 시스템의 모델에 대해서 살펴보고 3장에 서 새로이 제안된 스트리밍 기술에 대해서 소개하였다. 스트리밍 기술을 기반으로 개발된 단말 시스템이 기존의 모델과 비교해 볼 때 메모리 자원의 낭비를 줄여주며 데 이터 전달로 발생하는 지연이 줄어들음을 보았다. 메시지 큐에 기반 한 방식은 계층별로 task를 정의하 고 메시지 큐를 정의하여 task 사이의 컨텍스트 스위 칭(context switching)과 메시지 큐의 처리를 위한 지연 과 자원을 필요로 한다. 그러나 스트리밍 task의 수가 줄어들며 task 사이의 메시지 전달을 필요로 하지 않 으므로 자원의 낭비가 덜하다. 스트리밍 기술은 계층 사 이의 데이터 전달 방법을 일반화 시켜서 특정한 규격 혹 은 계층별 알고리즘에 종속 되지 않도록 한다. 계층별로 독립적인 개발을 이룰 수 있는 장점은 계층이 재설정 가

능하도록 하는 방법을 제시한다. 이러한 장점은 스트리밍 기술에 기반 한 프로토콜 스택이 새로운 프로토콜 스택으로 재설정 가능하며 프로토콜 스택의 유지 및 갱신 이 가능하도록 한다.

참고 문헌

- [1] 3GPP TS 25.322 V3.3.0, p. 6, June, 2000
- [2] Ritchie, D.M., "A Stream Input-Output System," *AT&T Bell Laboratories Technical Journal*, vol 63, No. 8, 10/84