



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2007년10월11일
(11) 등록번호 10-0765121
(24) 등록일자 2007년10월02일

(51) Int. Cl.

H04L 29/06(2006.01)

(21) 출원번호 10-2001-0073642

(22) 출원일자 2001년11월24일

심사청구일자 2005년11월18일

(65) 공개번호 10-2003-0042847

공개일자 2003년06월02일

(56) 선행기술조사문헌

WO 0062468 A2

(뒷면에 계속)

(73) 특허권자

엘지전자 주식회사

서울특별시 영등포구 여의도동 20번지

(72) 발명자

여운영

경기도안양시동안구비산동1109-4

셋별아파트612-1107

이승준

서울특별시강남구개포동대청아파트303동403호

이소영

경기도군포시금정동울곡아파트347동1401호

(74) 대리인

박장원

전체 청구항 수 : 총 13 항

심사관 : 박상현

(54) 송신버퍼의 프로토콜 데이터 유닛 폴링 방법

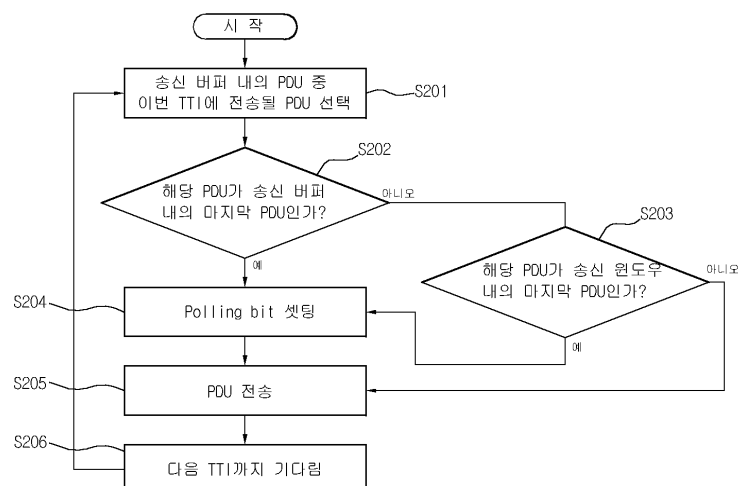
(57) 요약

본 발명은 3GPP 비동기식 이동통신 시스템의 송신측 RLC계층에서 송신측 RLC계층이 수신측 RLC계층으로 상태 정보의 전송을 요구하기 위한 폴링 비트 셋팅을 위해, 송신버퍼 내의 마지막 데이터 여부와 송신윈도우 내의 마지막 데이터 여부를 고려하여 셋팅할 수 있도록 한 것이다.

본 발명은 버퍼로부터 데이터 유닛을 선택하는 단계와; 상기 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 하나 이상에서 마지막 위치를 차지하는지 결정하는 단계와; 상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하지 않더라도, 상기 송신 윈도우 내에서 마지막 위치를 차지하는 것으로 결정되는 경우, 상기 데이터 유닛과 함께 폴링 정보를 전송하는 단계들을 포함하여 이루어진 것을 특징으로 한다.

이 같은 본 발명에 의하면, 송신버퍼 내에 마지막 프로토콜 데이터 유닛의 여부와 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 검사한 후 폴링 비트를 셋팅함으로써, 프로토콜 데이터 유닛의 전송 시스템에서 보다 안정적으로 시스템을 운용할 수 있다.

대표도 - 도6



(56) 선행기술조사문헌

KR1020020003194 A

KR1020020003233 A

W00021220 A1

2000062468

특허청구의 범위

청구항 1

삭제

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

버퍼로부터 데이터 유닛을 선택하는 단계와;

상기 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 하나 이상에서 마지막 위치를 차지하는지 결정하는 단계와;

상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하지 않더라도, 상기 송신 윈도우 내에서 마지막 위치를 차지하는 것으로 결정되는 경우, 상기 데이터 유닛과 함께 폴링 정보를 전송하는 단계들을 포함하는 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 9

제8항에 있어서, 상기 버퍼는 송신버퍼인 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 10

삭제

청구항 11

제10항에 있어서, 상기 결정 단계는

상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하는지 결정하는 단계와;

상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하지 않는 것으로 결정되는 경우, 상기 데이터 유닛이 상기 송신 윈도우 내에서 마지막 위치를 차지하는지 결정하는 단계로 이루어지는 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 12

삭제

청구항 13

제8항에 있어서, 상기 데이터 유닛은 프로토콜 데이터 유닛인 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 14

제8항에 있어서, 상기 폴링 정보는 상기 데이터 유닛에 폴링 비트를 셋팅하여 전송하는 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 15

제14항에 있어서, 상기 폴링 비트 셋팅은 제2프로토콜 계층에서 수행하는 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 16

제15항에 있어서, 상기 제2 프로토콜 계층은 응답모드(AM) 무선링크제어(RLC) 계층인 것을 특징으로 하는 데이터 유닛의 폴링 방법.

청구항 17

버퍼와;

상기 버퍼로부터 선택된 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 어느 하나 이상에서 마지막 데이터 유닛인지 결정하고, 상기 데이터 유닛이 상기 버퍼 내에서 마지막 데이터 유닛이 아니더라도, 상기 송신 윈도우 내에서 마지막 데이터 유닛으로 결정되면, 상기 데이터 유닛과 폴링 정보를 함께 전송하는 폴링비트 셋팅부를 포함하는 것을 특징으로 하는 송신기.

청구항 18

제17항에 있어서, 상기 버퍼는 송신버퍼인 것을 특징으로 하는 송신기.

청구항 19

삭제

청구항 20

삭제

청구항 21

제17항에 있어서, 상기 데이터 유닛은 프로토콜 데이터 유닛인 것을 특징으로 하는 송신기.

청구항 22

제17항에 있어서, 상기 폴링비트 셋팅부는 상기 폴링 정보를 상기 데이터 유닛에 폴링 비트를 셋팅하여 전송하는 것을 특징으로 하는 송신기.

청구항 23

삭제

청구항 24

제17항에 있어서, 상기 폴링 비트 셋팅부는 제2프로토콜 계층에서 수행되는 것을 특징으로 하는 송신기.

청구항 25

제24항에 있어서, 상기 제2 프로토콜 계층은 응답모드(AM) 무선링크제어(RLC) 계층인 것을 특징으로 하는 송신기.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- <8> 본 발명은 3GPP 비동기식(UMTS) 이동통신 시스템의 송신측 RLC 계층에서 송신측이 수신측 RLC 계층으로 상태 정보의 전송을 요구하는 폴링 방법 중 하나인 "송신버퍼에서 마지막 프로토콜 데이터 유닛(Last PDU in buffer)"의 폴링방법에 관한 것이다.
- <9> 상세하게는, 송신버퍼내에 있는 프로토콜 데이터 유닛이 송신윈도우의 범위 밖에 존재하는 경우 송수신측간의 통신 교착상태(deadlock)가 발생하는 것을 방지하기 위해 송신버퍼내의 마지막 프로토콜 데이터 유닛과 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 함께 검사하여, 폴링 비트를 셋팅할 수 있도록 한 송신버퍼의 프로토콜 데이터 유닛 폴링 방법에 관한 것이다.
- <10> UMTS(Universal Mobile Terrestrial System)는 유럽식 표준인 GSM(Global System for Mobile Communication s)시스템으로부터 진화한 제3세대 이동통신시스템으로, GSM 핵심망(Core Network)과 WCDMA (Wideband Code Division Multiple Access) 접속기술을 기반으로 하여 보다 향상된 이동통신서비스의 제공을 목표로 한다. UMTS의 표준화 작업을 위해, 1998년 12월에 유럽의 ETSI, 일본의 ARIB/TTC, 미국의 T1 및 한국의 TTA 등은 제3세대 공동프로젝트(Third Generation Partnership Project ; 이하, 3GPP라 약칭함)라는 프로젝트를 구성하였고, 현재까지 UMTS의 세부적인 표준명세서(Specification)를 작성 중에 있다.
- <11> 3GPP에서는 UMTS의 신속하고 효율적인 기술개발을 위해, 망 구성요소들과 이들의 동작에 대한 독립성을 고려하여 UMTS의 표준화 작업을 5개의 기술규격그룹(Technical Specification Groups; 이하, TSG라 약칭함)으로 나누어 진행하고 있다. 각 TSG는 관련된 영역내에서 표준규격의 개발, 승인, 그리고 그 관리를 담당하는데, 이들 중에서 무선접속망(Radio Access Network : 이하 RAN이라 약칭함)그룹(TSG-RAN)은 UMTS에서 WCDMA접속기술을 지원하기 위한 새로운 무선접속망인 UMTS무선망(Universal Mobile Telecommunications Network Terrestrial Radio Access Network;이하, UTRAN이라 약칭함)의 기능, 요구사항 및 인터페이스에 대한 규격을 개발한다.
- <12> TSG-RAN그룹은 다시 전체회의(Plenary)그룹과 4개의 운영그룹(Working Group)으로 구성되어 있다. 제 1운영그룹(WG1:Working Group 1)에서는 물리계층(제1계층)에 대한 규격을 개발하고, 제 2운영그룹(WG2 :Working Group 2)은 데이터링크계층(제2계층) 및 네트워크계층(제3계층)의 역할을 규정한다. 또한, 제 3운영그룹에서는 UTRAN 내의 기지국, 무선망제어기(Radio Network Controller; 이하, RNC라 약칭함) 및 핵심망(Core Network)간 인터페이스에 대한 규격을 정하며, 제 4운영그룹에서는 무선링크성능에 관한 요구조건 및 무선자원관리에 대한 요구사항 등을 논의한다.
- <13> 도 1은 3GPP 무선접속망 규격을 기반으로 한 단말과 UTRAN사이의 무선접속인터페이스 프로토콜의 구조를 나타낸다.
- <14> 도 1을 참조하면, 무선접속인터페이스 프로토콜은 수평적으로 물리계층(PHY), 데이터링크계층 및 네트워크계층으로 이루어지며, 수직적으로는 제어신호(Signaling)전달을 위한 제어평면(Control Plane)과 데이터정보 전송을 위한 사용자평면(User Plane)으로 구분된다. 여기서, 사용자 평면은 음성이나 IP 패킷의 전송등과 같이 사용자의 트래픽정보가 전달되는 영역이고, 제어평면은 망의 인터페이스나 호의 유지 및 관리 등의 제어정보가 전달되는 영역을 나타낸다.
- <15> 도 2의 프로토콜 계층들은 통신시스템에서 널리 알려진 개방형시스템간 상호접속 (Open System Interface; OSI)기준모델의 하위 3개 계층을 바탕으로 제 1계층(L1), 제 2계층(L2), 제 3계층(L3)으로 구분될 수 있다.
- <16> 제 1계층(L1)은 무선인터페이스에 대한 물리계층(PHY: Physical Layer)의 역할을 수행하고, 상위에 있는 매체접속제어(Medium Access Control; 이하 MAC이라 약칭함)계층과는 전송채널(Transport Channel)들을 통해 연결되어 있으며, 전송채널(Transport Channel)을 통해 물리계층으로 전달된 데이터를 무선환경에 맞는 다양한 코딩과 변조방식 등을 이용하여 수신측에 전달하는 역할을 담당한다. 물리계층(PHY)과 MAC계층사이에 존재하는 전송채널은 단말이 독점적으로 이용할 수 있는지, 또는 여러 개의 단말이 공유해서 사용하는지에 따라 각각 전용전송채널(Dedicated Transport Channel)과 공용전송채널(Common Transport Channel)로 구분된다.

- <17> 그리고, 제 2계층(L2)은 데이터링크계층(Data Link Layer)의 역할을 수행하고, 여러 단말들이 WCDMA망의 무선자원을 공유할 수 있도록 한다. 제 2계층(L2)은 MAC 계층, 무선링크제어(Radio Link Control; 이하 RLC라 약칭함)계층, 패킷데이터수렴프로토콜(Packet Data Convergence Protocol; 이하 PDCP라 약칭함)계층, 그리고 방송/멀티캐스트제어(Broadcast/Multicast Control; 이하 BMC라 약칭함)계층으로 나뉘어진다.
- <18> 상기 MAC계층은 논리채널과 전송채널간의 적절한 대응(Mapping) 관계를 통해 데이터를 전달한다. 논리채널들은 상위계층과 MAC계층을 연결시켜주는 채널들로 전송되는 정보의 종류에 따라 다양한 논리채널이 제공된다. 일반적으로 제어평면의 정보를 전송할 경우에는 제어채널(Control Channel)을, 사용자 평면의 정보를 전송하는 경우는 트래픽채널(Traffic Channel)을 사용한다.
- <19> RLC계층은 상위로부터 전송된 RLC SDU의 분할 및 연결(Segmentation and Concatenation)기능에 의해 전송에 맞는 적절한 RLC PDU를 구성하고, 전송 중 소실된 RLC PDU의 재전송을 담당하는 자동반복요구(Automatic Repeat request; ARQ) 기능을 수행할 수 있다. 상위로부터 내려온 RLC SDU를 처리하는 방식에 따라 투명모드(Transparent Mode), 무응답모드(Unacknowledged Mode), 응답모드(Acknowledged Mode)의 세 가지 방식으로 동작하고, RLC계층에는 상위계층에서 내려온 RLC SDU 또는 RLC PDU들을 저장하기 위한 RLC버퍼가 존재한다.
- <20> PDCP계층은 RLC계층의 상위에 위치하며, IPv4나 IPv6와 같은 네트워크 프로토콜을 통해 전송되는 데이터들이 RLC계층에서 전송되기에 적합하도록 만들어준다. 특히, IP패킷의 효율적인 전송을 위해 패킷의 헤더정보를 압축해서 전송하는 헤더압축(Header Compression)기법을 사용할 수 있다.
- <21> BMC계층은 CBS(Cell Broadcast Center)로부터 전달된 메시지를 무선 인터페이스를 통해 전송할 수 있도록 한다. BMC의 주된 기능은 단말로 전송되는 셀 방송 메시지(Cell Broadcast Message)를 스케줄링하여 전송하는 것으로, 일반적으로 무응답모드로 동작하는 RLC계층을 통하여 데이터를 전송한다.
- <22> 참고로, PDCP계층과 BMC계층은 사용자 데이터만을 전송하므로 사용자평면에만 위치한다. 이들과는 달리, RLC계층은 상위에 연결된 계층에 따라 사용자평면에 속할 수도 있고 제어평면에 속할 수도 있다. 제어평면에 속하는 경우에는 무선자원제어(Radio Resource Control; 이하 RRC라 약칭함)계층으로부터 데이터를 전달받는 경우에 해당되고, 그 외의 경우는 사용자 평면에 해당한다.
- <23> 일반적으로, 사용자 평면에서 제 2계층(L2)에 의해 상위계층으로 제공되는 사용자데이터의 전송서비스를 무선운반자(Radio Bearer; RB)라고 정의하며, 제어평면에서 제 2계층(L2)에 의해 상위계층으로 제공되는 제어정보의 전송서비스는 시그널링 무선 운반자(Signaling Radio Bearer; SRB)라고 정의한다.
- <24> 또한, 도 1에서 알 수 있듯이 RLC계층과 PDCP계층의 경우에는, 하나의 계층 내에 여러 개의 엔터티(Entity)들이 존재할 수 있다. 이는 하나의 단말이 여러 개의 무선운반자를 갖고, 하나의 무선운반자에 대하여 일반적으로 오직 하나의 RLC 엔터티 및 PDCP 엔터티가 사용되기 때문이다. RLC계층 및 PDCP계층의 엔터티들은 각 계층내에서 독립적인 기능을 수행할 수 있다.
- <25> 제 3계층(L3)의 가장 하부에 위치한 RRC계층은 제어평면에서만 정의되며, 무선운반자들의 설정, 재설정 및 해제와 관련되어 전송채널 및 물리채널들의 제어를 담당한다. 이때, 무선운반자가 설정된다(RB setup)는 것은 특정 서비스를 제공하기 위해 필요한 프로토콜 계층 및 채널의 특성을 규정하고, 각각의 구체적인 파라미터 및 동작방법을 설정하는 과정을 의미한다. RRC메시지를 통해 상위계층에서 전달되는 제어 메시지들의 전송도 가능하다.
- <26> 이하 상기의 RLC계층에 대해서 조금 더 자세히 살펴보도록 하자.
- <27> 상기의 RLC계층은 수행하는 기능에 따라 각각 투명모드(Transparent Mode, TM), 무응답모드(Unacknowledged Mode, UM), 그리고 응답모드(Acknowledged Mode, AM)의 세 가지 모드로 나누어 지는데, 이들 중 본 발명에 적용한 응답 모드(AM)에 대해서만 설명하기로 한다.
- <28> 응답 모드(AM)의 가장 큰 특징은 PDU의 전송 실패 시 재전송을 지원한다는 점이다. 즉, 송신측 RLC계층이 프로토콜 데이터 유닛(PDU: Protocol Data Unit)을 전송하면, 수신측은 각각의 PDU에 대해 그 수신 여부를 상태정보(Status information)에 실어 송신측으로 알린다. 송신측은 수신측으로부터 상태정보를 받으면 전송이 실패한 PDU를 수신측으로 재전송 하는 것이다.
- <29> 송신측 AM RLC에서 PDU 송신 과정을 도 2를 통해 좀더 자세히 알아보자. 도 2는 AM RLC의 송신부(100) 구조를 간단히 나타낸 것이다.
- <30> 도 2를 참조하면, PDU 생성장치(101)는 상위 계층으로부터 서비스 데이터 유닛(SDU: Service Data Unit)을 받으

면 이를 일정한 크기의 PDU로 만들기 위해 분할(Segmentation) 또는 연결(Concatenation)한 후 각각의 세그먼트에 RLC 헤더(header)를 붙여 PDU를 생성한다. RLC 헤더에는 일련번호(Sequence Number)가 포함되어 있으며, 이를 이용하면 각각의 PDU를 구별할 수 있다.

- <31> 이렇게 생성되는 PDU는 송신버퍼(102)와 재전송을 위해 재송신버퍼(103)에 저장된다. 송신측 AM RLC계층은 송신버퍼(102)에 저장되어 있는 PDU들을 매 전송 시간 간격(Transmission Time Interval, TTI) 마다 하위 계층에서 요구하는 갯수 만큼씩 하위 계층으로 내려보낸다.
- <32> 이때, 폴링 비트 셋팅부(104)에서는 전송하는 PDU 중 특정 PDU에 대해서 수신측에게 상태정보를 보내도록 요구하는 폴링 비트(Polling bit)의 셋팅 여부를 결정하게 되는데, 어떤 PDU에 폴 비트를 셋팅할 것인지는 사용되는 폴링 트리거(Polling Trigger)에 따라 다르다.
- <33> 이후, 상기 하위 계층으로 내려온 PDU들은 무선 구간을 통해 수신측으로 전송되며, 수신측 AM RLC에서는 PDU의 헤더에 있는 정보를 이용하여 SDU를 구성한 후 상위계층으로 전달한다.
- <34> 그런데, 만약 수신한 PDU 중 한 PDU에 폴링 비트(Polling bit)가 셋팅되어 있을 경우에는 그때까지 자신이 수신한 PDU들에 대해 제대로 수신이 되었는가를 검사하여 그 상태정보를 송신측으로 전송한다. 상태정보를 받은 송신측 AM RLC에서는 성공적으로 전송된 PDU들을 재송신버퍼(102)에서 삭제하고, 전송에 실패한 PDU들은 송신버퍼(102)로 보내 이후에 재전송 한다. 재전송은 부정응답(Negative acknowledgement)을 받은 PDU에 대해서만 가능하며, 재전송되는 PDU들은 전송이 성공할 때까지 계속 재송신버퍼(103)에 그대로 남겨놓는다. 재전송되는 PDU는 처음 전송되는 PDU들보다 우선 순위가 높으며, 재전송 PDU에도 폴링 비트(Poll bit)를 셋팅할 수 있다.
- <35> 그리고, AM RLC에서는 PDU의 송수신을 위해 송신측과 수신측에는 각각 송신윈도우와 수신윈도우가 존재하며, 일반적으로 송신윈도우와 수신윈도우의 크기는 같다.
- <36> 상기 송신윈도우는 수신측으로부터의 상태정보를 받지 않은 상태에서 최대로 보낼 수 있는 PDU의 갯수에 해당한다. 송신측은 항상 송신윈도우 내의 PDU에 대해서만 전송 가능하며, 그 이후의 PDU들은 송신윈도우가 갱신될 때까지 기다렸다가 전송된다. 송신윈도우의 갱신은 수신측으로부터 상태정보를 받으면 이루어지며, 첫 번째 부정응답(Negative acknowledgement)에 해당하는 PDU까지 송신윈도우를 이동시킨다.
- <37> 예를 들어, 송신윈도우의 위치가 1~100이고(송신윈도우 크기는 100), 1~50에 해당하는 PDU를 전송했을 때, 수신측으로부터 15, 20, 40번의 PDU는 전송 실패이고 나머지는 모두 성공했음을 전달받은 경우, 송신윈도우는 15~115로 갱신되는 것이다. 이때 다음 번에 전송하는 PDU의 순서는 15, 20, 40, 51, 52, 53 순서로 전송된다.
- <38> 또한, 수신측에서는 수신윈도우가 존재하며, 이는 수신 PDU의 유효 범위를 뜻한다. 수신측은 수신윈도우 내의 일련번호(Sequence Number)를 가진 PDU에 대해서만 수신하며, 수신윈도우의 범위를 벗어난 PDU에 대해서는 수신 즉시 폐기시킨다. 수신윈도우의 갱신은 수신윈도우 내에서 in-sequence로 오는 새로운 PDU를 받을 때 이루어진다.
- <39> 그 예로서, 수신윈도우가 1~100이고(수신윈도우 크기는 100), 1~50에 해당하는 PDU를 수신하였으나, 15, 20, 40번 PDU는 수신 실패한 경우 수신윈도우는 15~115로 갱신되는 것이다. 여기서, 수신 실패한 경우로는 PDU를 성공적으로 수신하지 못한 모든 경우로서, PDU 자체를 수신하지 못했거나, PDU를 수신했지만 에러가 있으면 모두 수신 실패이다.
- <40> 여기서, 폴링(Polling)이란 송신측이 수신측으로 상태정보를 요구하는 것을 말하며, 수신측은 송신측으로부터 폴링(Polling)을 받았을 경우 그 때까지의 PDU 수신 상태를 파악하여 송신측으로 알려야 한다. 구체적으로는, 폴링(Polling)을 위해서 송신측은 PDU 전송 시에 PDU 내에 폴링 비트를 셋팅해 전송하며, 수신측에서는 폴링 비트가 셋팅된 PDU를 받았을 당시의 수신버퍼 상태를 검사하여 각 PDU의 상태정보를 송신측에 알리는 것이다.
- <41> 그런데, 상태정보를 전송하는 것은 그 자체로 무선 자원을 낭비하는 것이기에 상태정보의 전송은 적절한 방법에 의해 조정되어야 한다. 즉, 송신측은 모든 PDU 마다 상태정보를 요구하는 것이 아니라, 어떠한 규칙을 정해 놓고 이 조건을 만족시키는 PDU에 대해서만 폴링 비트를 셋팅해야 하며, 이러한 규칙을 폴링 트리거(Polling trigger)라고 한다.
- <42> 폴링 트리거(Polling Trigger)에는 여러 가지 방법이 사용되는데, 그 중 하나가 "송신버퍼에서 마지막 프로토콜 데이터 유닛>Last PDU in buffer)"을 이용한 방법이다. 이 방법은 송신버퍼 내에 있는 PDU 중 마지막 PDU가 전송될 때에 폴링 비트(Polling bit)를 셋팅한다. 즉, 도 3과 같이 송신버퍼 내에 1~50에 해당하는 PDU가 저장되

어 있을 경우, 일련번호가 50인 PDU를 전송할 때 50번째 PDU에 폴링 비트를 셋팅하는 것이다.

- <43> 도 4는 종래의 송신버퍼에서 마지막 프로토콜 데이터 유닛을 이용한 폴 비트 셋팅 방법을 나타낸 플로우 차트이다.
- <44> 도 4를 참조하면, 하위 계층에서 요구하는 PDU 개수 만큼 송신버퍼내의 프로토콜 데이터 유닛을 이번(현재) 전송시간간격(TTI) 마다 순차적으로 선택하고(S101), 각각 선택된 프로토콜데이터유닛에 대해 해당 프로토콜 데이터 유닛이 송신버퍼 내의 마지막 프로토콜 데이터 유닛인가를 확인한다(S102).
- <45> 상기 확인결과 각각의 프로토콜 데이터 유닛에 대해 해당 프로토콜 데이터 유닛을 검사하여, 마지막 프로토콜 데이터 유닛일 경우에는 해당 프로토콜 데이터 유닛에 폴링 비트를 셋팅하고(S103), 전송할 프로토콜 데이터 유닛을 전송하며(S104), 해당 프로토콜 데이터 유닛이 송신버퍼 내의 마지막 프로토콜 데이터 유닛이 아닐 경우에는 상기 선택된 프로토콜 데이터 유닛을 전송해 준다(S104). 이후 다음 전송시간간격까지 대기하고(S105), 상기의 과정을 반복한다.
- <46> 그러나, 종래의 송신버퍼에서 마지막 프로토콜 데이터 유닛을 이용한 폴링 비트 셋팅방법에서는 송신윈도우를 고려하지 않았기 때문에, 특정한 경우 송수신 AM RLC들은 교착상태(Deadlock)에 빠지게 된다.
- <47> 즉, 송신윈도우의 범위를 넘어가는 PDU가 송신버퍼 내에 저장되어 있을 경우, 이들 PDU들은 송신윈도우가 갱신되지 않는 한 전송될 수가 없으며, 따라서 송신버퍼 내의 마지막 PDU 역시 전송될 수가 없다. 송신윈도우 내의 모든 PDU가 송신버퍼 내의 마지막 PDU가 아니므로 Poll bit이 셋팅되지 않은 채로 전송되며, 따라서 수신측에서는 이들 PDU를 성공적으로 받았더라도 상태정보를 보내지 않기 때문에, 송수신 모두 교착상태에 빠지는 것이다.
- <48> 예를 들어, 도 5와 같이 송신윈도우의 위치가 1~100 (송신윈도우 크기는 100)이고, 상위 계층으로부터 1~150에 해당하는 PDU를 받은 경우, 송수신측은 모두 교착 상태에 빠진다. 그 이유는 송신버퍼의 마지막 PDU는 150번이므로, "Last PDU in buffer" 방법을 사용하면 150번 PDU를 전송할 때에 폴링 비트를 셋팅하게 된다.
- <49> 그러나, 송신윈도우의 위치는 1~100 이므로, 송신측은 1~100까지의 PDU만 전송 가능하며, 이들 중 조건에 맞는 PDU는 없기 때문에 전송되는 모든 프로토콜 데이터 유닛은 폴링 비트(Polling bit)를 셋팅하지 않은 채로 전송된다.
- <50> 그러면, 수신측에서는 수신한 PDU 중 폴링 비트가 셋팅되어 있는 PDU가 없기 때문에 상태정보를 송신측으로 보내지 않으며, 수신윈도우를 갱신시킨 상태로 다음 프로토콜 데이터 유닛의 수신을 기다리게 된다.
- <51> 송신측은 수신측으로부터의 상태정보가 없기 때문에 송신윈도우를 갱신하지 못하며, 또한 부정응답을 받지 못했기 때문에 재전송도 할 수가 없게 된다. 따라서, 이후에는 송수신 모두 아무런 동작을 할 수 없으므로 교착상태에 빠지게 되고, 이는 망의 상황이 전송 가능한 상태임에도 불구하고 전송을 하지 않게 함으로써, 망 자원을 비효율적으로 사용하게 하며 불필요한 지연 상태를 유발한다.

발명이 이루고자 하는 기술적 과제

- <52> 본 발명은 상기한 종래의 문제점을 해결하기 위해 안출된 것으로서, 송신측 AM RLC 계층에서 수신측으로 전송할 특정 프로토콜 데이터 유닛에 대한 폴링 비트 셋팅을, 송신버퍼내의 마지막 프로토콜 데이터 유닛 여부의 검사를 통한 폴링 비트 셋팅과, 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부의 검사를 통한 폴링 비트 셋팅을 함께 수행하여 전송할 수 있도록 한 송신버퍼의 프로토콜 데이터 유닛 폴링방법을 제공함에 그 목적이 있다.
- <53> 본 발명의 특징은 전송 가능한 프로토콜 데이터 유닛들 중 송신버퍼 내의 마지막 프로토콜 데이터 유닛이거나, 송신윈도우 내의 마지막 프로토콜 데이터 유닛에 대하여 폴링 비트를 셋팅함으로써, 송신버퍼내에 있는 적어도 하나의 프로토콜 데이터 유닛에 대하여 폴링 비트를 셋팅할 수 있도록 한 송신버퍼의 프로토콜 데이터 유닛 폴링방법을 제공함에 그 목적이 있다.
- <54> 다른 특징은 응답 모드에서 폴링 비트 셋팅이 이루어 질 수 있도록 한 송신버퍼의 프로토콜 데이터 유닛 폴링방법을 제공함에 그 목적이 있다.

발명의 구성 및 작용

- <55> 상기한 목적 달성을 위한, 본 발명에 따른 송신버퍼의 프로토콜 데이터 유닛 폴링방법은,
- <56> 버퍼로부터 데이터 유닛을 선택하는 단계와;

상기 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 하나 이상에서 마지막 위치를 차지하는지 결정하는 단계와;

- <57> 상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하지 않더라도, 상기 송신 윈도우 내에서 마지막 위치를 차지하는 것으로 결정되는 경우, 상기 데이터 유닛과 함께 폴링 정보를 전송하는 단계들을 포함하는 것을 특징으로 한다.
- <58> 삭제
- <59> 삭제
- <60> 삭제
- <61> 바람직하게, 상기 결정하는 단계는 상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하는지 결정하는 단계와; 상기 데이터 유닛이 상기 버퍼 내에서 마지막 위치를 차지하지 않는 것으로 결정되는 경우, 상기 데이터 유닛이 상기 송신 윈도우 내에서 마지막 위치를 차지하는지 결정하는 단계로 이루어질 수 있다.
- <62> 바람직하게, 상기 선택된 특정 프로토콜 데이터 유닛에 대하여, 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 먼저 검사한 후, 마지막 프로토콜 데이터 유닛이 아닐 경우 송신버퍼 내의 마지막 프로토콜 데이터 유닛 여부를 나중에 검사하는 것을 특징으로 한다.
- <63> 바람직하게, 상기 프로토콜 데이터 유닛은 이동 통신 시스템의 이동 단말 또는 네트워크 장치 등의 무선 링크 제어 계층에서 정하는 단위의 데이터 인 것을 특징으로 한다.
- <64> 바람직하게, 상기 폴링 정보는 상기 데이터 유닛에 폴링 비트를 셋팅하여 전송하는 것을 특징으로 한다.
바람직하게, 상기 프로토콜 데이터 유닛에 폴링 비트를 셋팅하는 경우는 전송하는 프로토콜 데이터 유닛이 송신 버퍼 내의 마지막 프로토콜 데이터 유닛이거나 송신윈도우 내의 마지막 데이터 유닛인 경우에 셋팅하는 것을 특징으로 한다.
- <65> 바람직하게, 상기 프로토콜 데이터 유닛에 폴링 비트를 셋팅하기 위해서, 송신윈도우의 정보 및 송신버퍼 내에 존재하는 프로토콜 데이터 유닛의 정보를 반영하여 폴링 비트 셋팅 여부를 검사하는 것을 특징으로 한다.
- <66> 바람직하게, 상기 송신측이 수신측으로 폴링여부를 검사하는 과정은 매 전송 시간 간격마다 반복되는 것을 특징으로 한다.
한편, 상기한 목적 달성을 위한, 본 발명에 따른 송신기는 버퍼와; 상기 버퍼로부터 선택된 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 어느 하나 이상에서 마지막 데이터 유닛인지 결정하고, 상기 데이터 유닛이 상기 버퍼 내에서 마지막 데이터 유닛이 아니더라도, 상기 송신 윈도우 내에서 마지막 데이터 유닛으로 결정되면, 상기 데이터 유닛과 폴링 정보를 함께 전송하는 폴링비트 셋팅부를 포함하는 것을 특징으로 한다.
- <67> 상기와 같은 본 발명에 따른 송신버퍼의 프로토콜 데이터 유닛 폴링 방법에 대하여 첨부된 도면을 참조하여 설명하면 다음과 같다.
- <68> 먼저, UMTS시스템에서 무선링크제어계층의 송신버퍼에서 마지막 프로토콜 데이터 유닛의 폴링(Last PDU in buffer Polling) 방법을 사용할 때, 송신윈도우의 범위를 넘어가는 PDU가 송신버퍼에 있을 때에도 정상적인 동작을 하도록 한 것이다.
- <69> 이를 위해서, 전송되는 해당 PDU가 송신윈도우 내의 마지막 PDU인지 한번 더 검사하여, 만약 송신윈도우 내의 마지막 PDU이면 송신버퍼 내의 마지막 PDU가 아니더라도 폴링 비트를 셋팅하고자 한다.
- <70> 상세하게 도 6을 참조하면, 송신버퍼에서는 이번 전송 시간 간격에 하위 계층에서 요구하는 개수 만큼의 프로토콜 데이터 유닛을 순차적으로 선택하여 폴링 비트 셋팅부로 전달한다(S201).
- <71> 그리고, 폴링 비트 셋팅부는 해당 프로토콜 데이터 유닛이 송신버퍼 내의 마지막 프로토콜 데이터 유닛인가를 검사하게 되며(S202), 검사결과 프로토콜 데이터 유닛이 마지막 프로토콜 데이터 유닛이면 폴링 비트를 셋팅하

고(S204), 셋팅된 프로토콜 데이터 유닛을 전송하고(S205), 다음 전송 시간 간격(TTI)까지 대기하며(S206), 상기의 과정을 전송 시간 간격마다 반복한다.

<72> 한편, 상기 폴링 비트 셋팅부는 상기 단계 S202의 검사결과 해당 프로토콜 데이터 유닛이 송신버퍼내의 마지막 프로토콜 데이터 유닛이 아니면, 해당 프로토콜 데이터 유닛이 송신윈도우 내의 마지막 프로토콜 데이터 유닛인가를 검사한다(S203). 단계 S203의 검사결과 해당 프로토콜 데이터 유닛이 송신윈도우내의 마지막 프로토콜 데이터 유닛이면 해당 프로토콜 데이터 유닛에 폴링 비트를 셋팅하고 전송하게 되며, 다음 전송시간 간격까지 대기한다.

<73> 상기 단계 S203의 검사결과 해당 프로토콜 데이터 유닛이 송신윈도우 내의 마지막 프로토콜 데이터 유닛이 아니면, 해당 프로토콜 데이터 유닛을 전송하고(S205), 다음 전송시간간격(TTI) 까지 대기한다.

<74> 이러한 폴링 비트 셋팅을 위해서, 폴링 비트 셋팅부는 송신버퍼의 상태와 송신윈도우의 크기를 알고, 전송하기 위해 선택된 프로토콜 데이터 유닛이 송신버퍼 내의 마지막 프로토콜 데이터 유닛인지를 검사한다. 만약 해당 PDU가 송신버퍼 내의 마지막 PDU이면 폴링 비트를 셋팅하여 전송하고, 아니면 해당 PDU가 송신윈도우 내의 마지막 PDU인지를 다시 한번 검사한다. 이때 해당 PDU가 송신윈도우 내의 마지막 PDU이면 폴링 비트를 셋팅하여 전송하고, 아니면 그대로 전송한다. 이후, 다음 전송시간간격까지 기다린 다음, 상술한 과정을 전송시간간격마다 반복한다.

<75> 실시 예로서, 상기 송신버퍼 내의 마지막 프로토콜 데이터 유닛 여부를 검사하는 단계를 먼저하고 송신버퍼내의 마지막 프로토콜 데이터 유닛이 아닐 경우의 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 나중에 검사하여, 마지막 프로토콜 데이터 유닛에 대하여 폴링 비트를 셋팅하는 방법과, 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 먼저 검사하고 송신윈도우 내의 마지막 프로토콜 데이터 유닛이 아닐 경우 송신버퍼내의 마지막 프로토콜 데이터 유닛 여부를 나중에 검사한 후, 마지막 프로토콜 데이터 유닛에 대하여 폴링 비트를 셋팅할 수 있도록 하는 제 2 방법을 제시한다.

<76> 도 7은 본 발명에 따른 폴링 비트의 셋팅 예를 나타낸 실시 예이다.

<77> 도 7의 (a)는 송신윈도우의 위치가 1~160(송신윈도우 크기는 160)이고, 송신버퍼에는 1~150에 해당하는 PDU가 존재하는 경우로서, 이 때는 송신윈도우 범위 내에 PDU가 존재하기 때문에 기존의 방법과 같이 송신버퍼 내의 마지막 PDU인 150번 PDU에 대해 폴링 비트를 셋팅한다.

<78> 도 7의 (b)의 경우에는 송신버퍼에는 (a)와 같이 1~150에 해당하는 PDU가 존재하지만, 송신윈도우의 위치가 1~100(송신윈도우 크기는 100)이기 때문에, 송신윈도우 범위 밖에 PDU가 존재하게 된다. 이 때는 송신버퍼 내의 마지막 PDU인 150번 PDU에 폴링 비트를 셋팅하는 것이 아니라, 송신윈도우 내의 마지막 PDU인 100번 PDU를 전송할 때 폴링 비트를 셋팅하게 된다.

한편, 본 발명에 따른 송신기는 도 2와 유사하게 버퍼와; 폴링비트 셋팅부를 포함한다. 상기 폴링 비트 셋팅부는 상술한 바와 같이 상기 버퍼로부터 선택된 데이터 유닛이 상기 버퍼 또는 송신 윈도우 중 적어도 어느 하나 이상에서 마지막 데이터 유닛인지 결정하고, 상기 데이터 유닛이 상기 버퍼 내에서 마지막 데이터 유닛이 아니라도, 상기 송신 윈도우 내에서 마지막 데이터 유닛으로 결정되면, 상기 데이터 유닛과 폴링 정보를 함께 전송하도록 한다.

발명의 효과

<79> 이상에서 설명한 바와 같이, 본 발명에 따른 송신버퍼의 프로토콜 데이터 유닛 폴링 방법에 의하면, 기존에 송신윈도우의 범위 밖에 송신버퍼의 프로토콜 데이터 유닛이 존재하는 특정 상황에서 시스템이 교착상태에 빠지는 문제점을 해결할 수 있도록, 송신버퍼 내에 마지막 프로토콜 데이터 유닛 여부와 송신윈도우 내의 마지막 프로토콜 데이터 유닛 여부를 검사한 후, 적어도 하나의 프로토콜 데이터 유닛에 대해 폴링 비트를 셋팅함으로써, 프로토콜 데이터 유닛의 전송 시스템에서 보다 안정적으로 시스템을 운용할 수 있다.

도면의 간단한 설명

<1> 도 1은 3GPP 시스템의 무선접속 인터페이스 프로토콜의 구조.

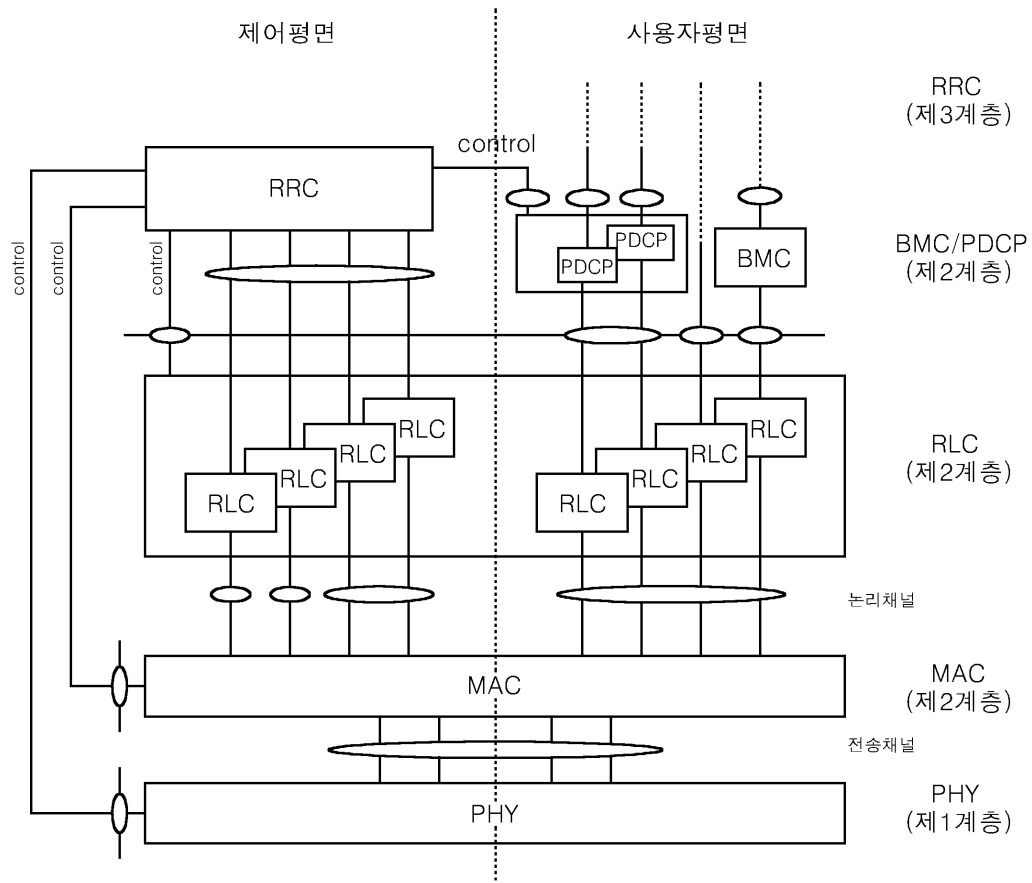
<2> 도 2는 AM RLC 송신부의 개략 구성도.

<3> 도 3은 송신버퍼에서 마지막 프로토콜 데이터 유닛의 폴링방법을 사용한 폴링의 예를 나타낸 도면.

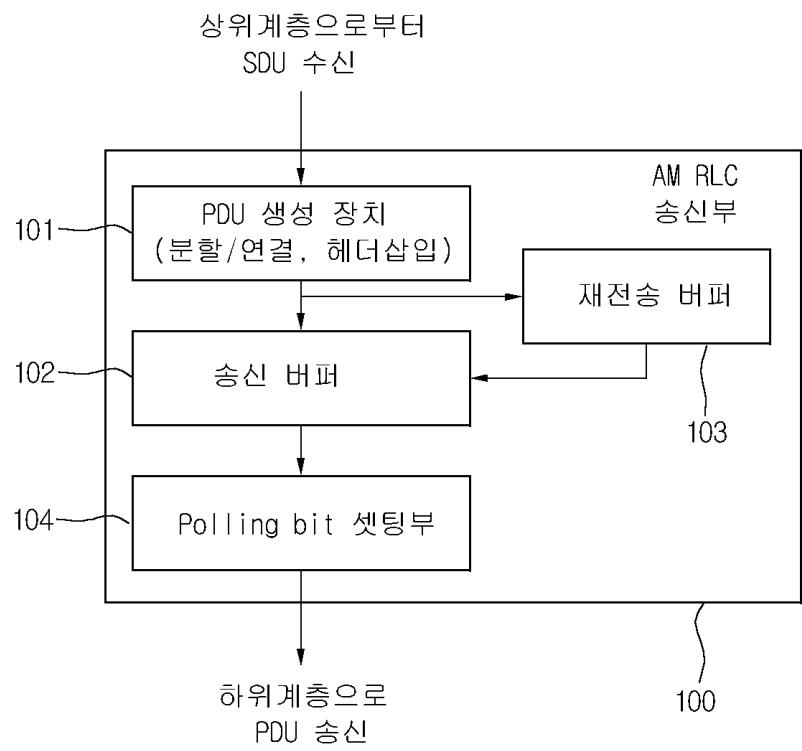
- <4> 도 4는 종래 송신버퍼에서 마지막 프로토콜 데이터 유닛의 폴링 방법을 나타낸 플로우 차트.
- <5> 도 5는 종래의 송신버퍼에서 마지막 프로토콜 데이터 유닛의 폴링방법을 사용할 때 교착상태에 빠지는 예시도.
- <6> 도 6은 본 발명 실시 예에 따른 송신버퍼의 마지막 프로토콜 데이터 유닛의 폴링 방법을 나타낸 플로우 차트.
- <7> 도 7의 (a)(b)는 본 발명에 있어, 프로토콜 데이터 유닛이 송신윈도우 범위 내에 존재 여부에 따른 폴링 비트 셋팅 예시도.

도면

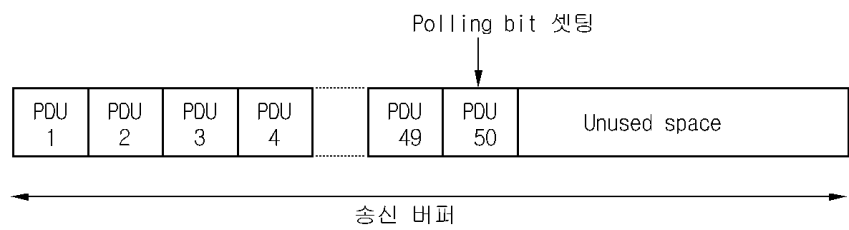
도면1



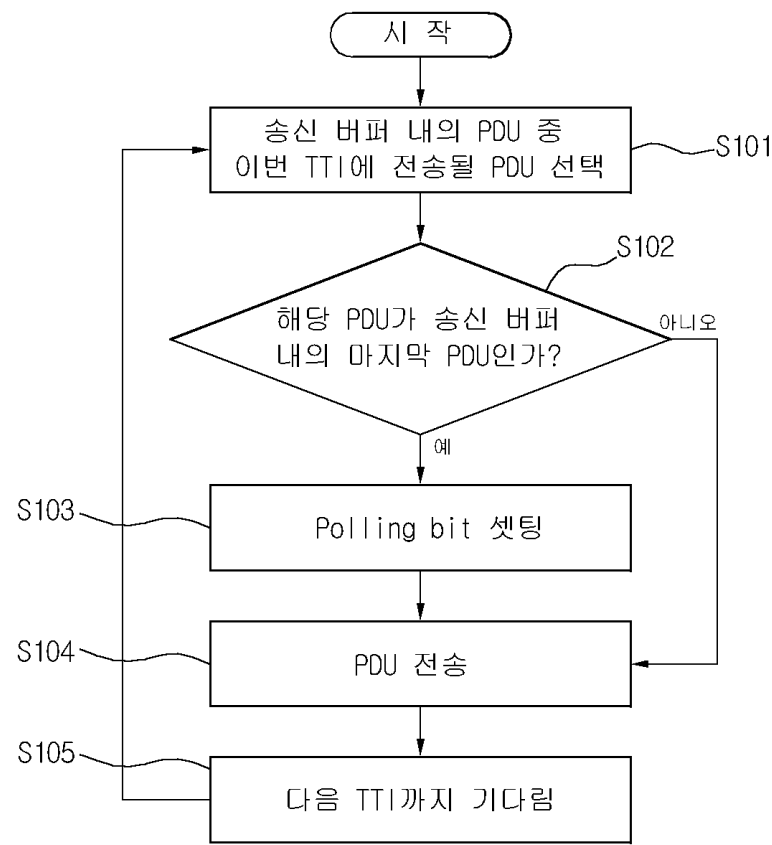
도면2



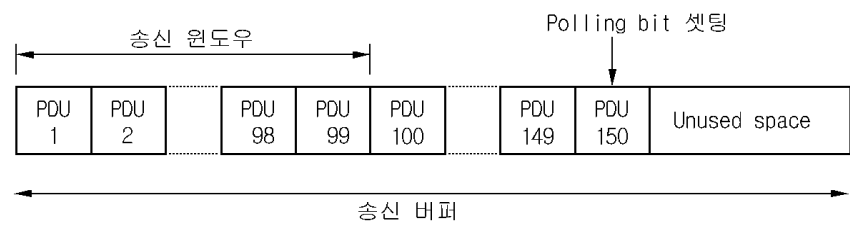
도면3



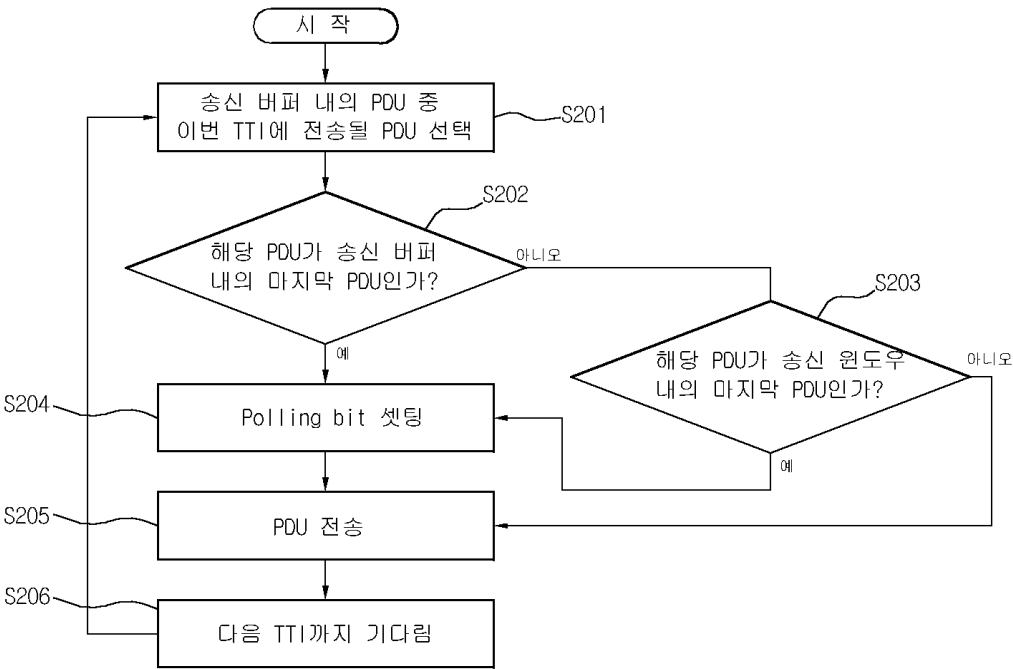
도면4



도면5



도면6



도면7

