

근거리 통신망과 사설교환기의 음성통신을 위한 게이트웨이의 구현

正會員 安 鏞 澈* 正會員 申 炳 喆**

Implementation of a Gateway Protocol between LAN and PABX for Voice Communication

Yong Cheol Ahn*, Byung Cheol Shin** *Regular Members*

要 約

패킷 음성 프로토콜은 지금까지 많이 연구되고 구현되어왔다. 하지만 음성통신을 위한 근거리 통신망과 사설교환기사이의 연동에 관한 연구는 아직 많은 편은 아니다. 본 논문에서는 근거리 통신망과 기존의 사설 교환기사이의 음성통신을 위한 게이트웨이를 설계하고 구현하였다.

구현한 게이트웨이의 프로토콜은 CCITT의 G.764 패킷 음성 프로토콜을 수정·사용하였다. 연동을 위한 하드웨어 시스템을 구현하였으며, 이는 전화선과의 인터페이스 부분, 음성처리부분, PC 인터페이스 부분 및 제어부분, 그리고 DTMF(Dual Tone Multiple Frequency) 접속부분으로 나누어진다. 그리고 소프트웨어는 근거리 통신망 접속용 네트워크 카드를 구동하는 패킷 드라이버를 이용하기위한 인터페이스 부분과 게이트웨이를 구동하는 드라이버, 그리고 프로토콜 처리부분으로 구성되어있다.

ABSTRACT

Packet voice protocols have been realized in many research works. But few studies for the interconnection of LAN and PABX to facilitate the voice communication have been done. In this paper, the gateway to interconnect the Ethernet LAN with the existing PABX telephone network for voice communication has been designed and implemented.

The implemented gateway protocol is a modified protocol based on CCITT's G.764 packetized voice protocol. To accomplish this goal the hardware system has been realized, which is divided into five parts: interface part with the telephone line, voice-processing part, PC interface part, controller part, and finally DTMF part. And the gateway software is divided into three parts:

*三星電子(株) 通信技術研究所
Telecom. R&D Center, Samsung Electronics Co., Ltd.

**韓國科學技術院 電氣 및 電子工學科
KAIST

論文番號: 94103

接受日字: 1994年 4月 4日

interface to make use of the packet driver which drives the network card, driver to drive the PABX gateway, and the protocol handling part.

I. 서 론

지난 수년간 컴퓨터의 확대 보급과 통신망 기술의 발전으로 근거리 통신망이 급속히 보급되었다. 이러한 근거리 통신망은 컴퓨터 사이의 파일(file) 전송, 값비싼 주변장치의 공유, 전자우편 등 광범위한 데이터 서비스를 제공하고 있다. 그러나 구내 음성교환 서비스는 보통 사설 교환기를 통하여 이루어지고 있다. 앞으로 정보화 사회로 발전하면서 각 사무실에서는 근거리 통신망을 통해 음성·데이터 및 영상까지 전송하면서 작업자들 사이에 자료를 교환하는 group work이나 멀티미디어 통신이 발달할 것으로 예상된다.

데이터 전송을 위해 발전된 근거리 통신망에서 음성을 전송하려면 먼저 음성의 특성을 잘 알아야 한다. 음성은 데이터와 달리 전송에러에 큰 영향을 받지 않으나 딜레이에는 민감하다. 일반적으로 데이터 전송시에는 에러가 없는 전송을 하기 위하여 재전송 알고리즘이 필요하다. 이와 달리 음성전송시에는 실시간으로 전송을 하면서 어느 정도의 에러를 허용하므로 재전송 알고리즘이 필요하지 않는 대신 음성의 질을 높이기 위한 음성재생 알고리즘을 필요로 한다. 이와 같이 음성과 데이터의 상이한 서비스 요구조건 때문에 기존의 데이터용 프로토콜과는 다른 음성 서비스를 위한 프로토콜을 따로 마련할 필요가 있다. 이러한 음성용 프로토콜은 많은 사람들에 의해 연구되었다[1][2][3][4][5]. 근거리 통신망 위에서 패킷 음성통신의 구체적 구현에는 다음과 같다. 참고문헌 [7]에서는 EAC(Ethernet Adaptor Card)와 TIC(Telephone Interface Card) 보드를 장착하고 음성 단말장치로는 상용 전화를 이용하였다.

본 논문의 목적은 이를 가장 보편화된 음성교환장치인 사설교환기(PABX)와 연결시키는 게이트웨이를 구현하는 것이다. 이를 위해서 CCITT의 G.764 packetized voice protocol을 수정하여 근거리 통신망 위에서 실험하였다. 본 논문의 의의는 다음과 같다. 컴퓨터망 및 사설교환기(PABX)망에서의 음성통신을 상호연동시키기 위하여 게이트웨이가 제안된 적은 있지만[6], 저자가 알고 있는 범위내에서 실제로 개발된 시스템은 확인된 것이 없다. 따라서 본 연구는 이 분야에서 몇 개 안되는 연구 중의 하나로 추

정된다.

본 논문의 구성은 다음과 같다. 2장에서는 패킷 음성통신에 관련된 연구동향과 게이트웨이의 개요에 대해서 살펴보았다. 3장에서는 사설교환기와 게이트웨이 사이에서 음성통신을 하기 위해 필요한 게이트웨이의 프로토콜, 하드웨어, 소프트웨어에 대해서 설명하고 4장에서 실험결과를 정리하고, 5장에서는 결론을 맺는다.

II. 관련 연구동향과 게이트웨이에 대한 개요

2.1 패킷음성통신에 대한 연구동향

1970년대 초반에 ARPANET 위에서 최초의 패킷 음성 실험이 실시되어 store-and-forwarded packet network 위에서의 음성통신 가능성이 제시되었다. 그 이후 수신된 음성패킷의 재생 문제는 기술적인 측면으로, 호 설정과 음성전송 문제는 프로토콜쪽으로 각각 발달되어왔다. 이러한 패킷음성통신에 대한 연구는 근거리 통신망, 장거리 통신망(WAN)을 비롯해 위성패킷통신, 이동통신, ISDN 등에서 이루어져 왔다.

최근 인터넷(Internet)에서 현재 가장 활발히 연구되고 있는 분야는 음성을 이용한 다자간 회의이다. 이에 대한 연구결과로서 1992년 3월에 열린 IETF(Internet Engineering Task Force) 미팅을 "audiocast"를 이용하여 원격지간(3개 대륙 20개 위치에 있는 참석자들에) 진행하였다. 인터넷에서 음성을 이용한 다자간 회의가 가능했던 요인으로는 첫째, 음성을 패킷화해 전송하고 이를 다시 수신해서 원래의 음성신호로 재생하는 하드웨어와 소프트웨어 기술이 이미 보편화되어 있고, 둘째, 인터넷에 일부이기는 하지만 멀티캐스트 라우팅 기능이 가능해졌으며, 셋째, 실시간 데이터가 시간지연없이 전송될 수 있도록 네트워크의 성능이 향상되었기 때문이다.

2.2 게이트웨이에 대한 고려사항[6]

게이트웨이는 근거리 통신망과 사설교환기사이의 중계역할을 하는 인터페이스이다. 게이트웨이는 근거리 통신망 접속 부분에서 패킷 음성 프로토콜을 처리하고 패킷화된 음성을 아날로그나 디지털 형태로

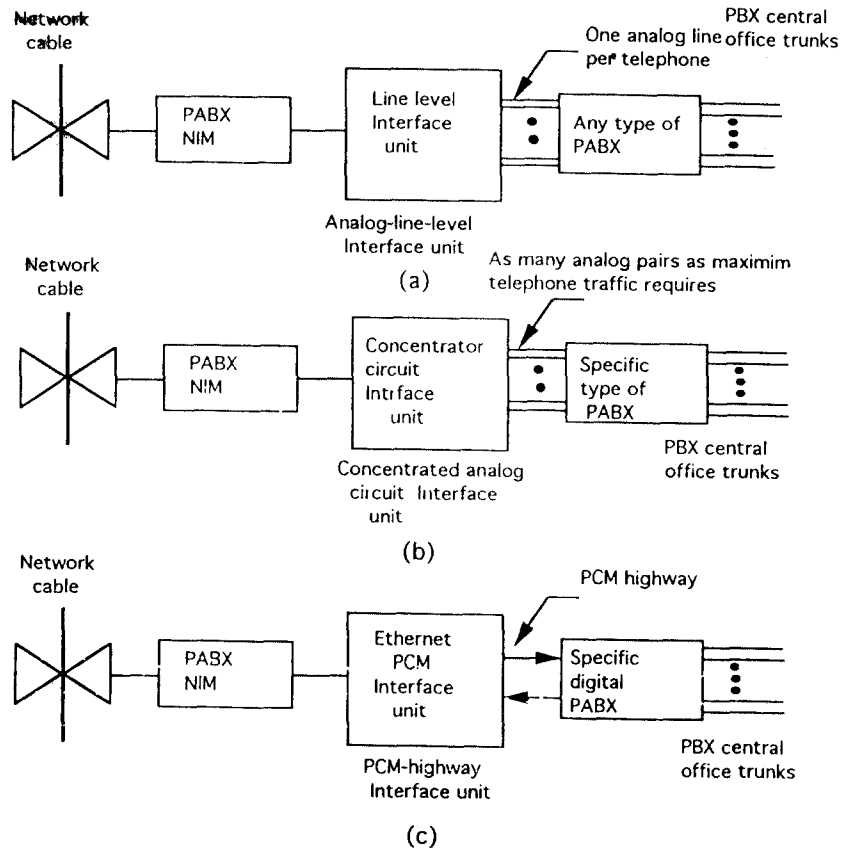


그림 2-1. PABX gateway

바꾸어 사실 교환기와 연결시켜준다.

근거리 통신망과 사실교환기를 연결시키는 방법에는 3가지가 있을 수 있다. 처음방법은 단순히 패킷망에 있는 전화수만큼의 전화선과 연결시켜 사실교환기가 보기에 전화처럼 보이게 하는 방법이다. 두번째 방법은 처음방법과 동일하나 전화통화량이 최고일때 소요되는 전화선 만큼만 전화회선을 연결시키는 방법이다. 세번째 방법은 사실교환기와 게이트웨이 사이에 하나혹은 여러개의 트렁크를 두어 전화선 수를 줄인다. 위의 방법중 3번째는 상용 사실교환기를 사용할 수 없고 특별하게 제작되어야만 한다. 그림 2-1에 각각의 경우에 대해서 나타나있다.

본 논문에서는 두번째 방법을 이용하여 2개의 전화선을 수용할 수 있는 게이트웨이를 구현하였다. 그림 2-2에 전체 시스템 구성도가 나타나있다.

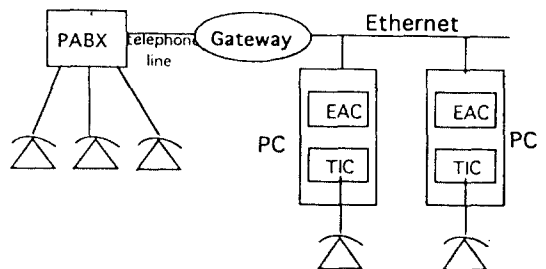


그림 2-2. 전체 시스템 구성도

2.2.1 샘플링된 음성과 Ethernet의 특징

PCM(Pulse Coded Modulation) 음성채널은 보통 3,200Hz의 대역폭(200~3,400Hz)을 가지고 있고 8kHz로 샘플링되어진다. 또 음성은 talkspurt 구간과

묵음(silence) 구간으로 나눌 수 있다. 음성의 talkspurt 구간은 40% 정도이기 때문에 talkspurt 구간만 패킷화해서 보낸다면 전송효율을 2배 이상 향상시킬 수 있으며, 묵음 구간을 데이터 트래픽이나 다른 음성 트래픽이 이용할 수 있으므로 채널의 이용도를 높일 수 있다.

Ethernet은 Contention 방식이므로 프레임을 전송할 수 있기까지의 시간 딜레이(delay)는 변화가 매우 심하다. 이런 상황때문에 음성을 패킷으로 만들어 보낼때 지터가 생길가능성이 많다. 이 지터를 예방하기 위하여 수신단에 버퍼를 설정하여 패킷을 저장한다. 송신단에서는 음성샘플을 패킷으로 만들어 전송하기 위하여 일정시간동안 버퍼링하여야만 하는데 이것이 딜레이를 가져온다. 수신단에서의 패킷 버퍼링(buffering)을 이용하여 음성 패킷의 end-to-end 지연 시간을 일정하게 유지하여 지연시간 지터를 없앤다. 그러나 전송단과 수신단에서의 버퍼링은 전체 지연시간(dealy)을 증가시켜 사설교환기와 연동시킬 때 에코(echo)의 주요한 원인이 된다.

2.2.2 에코와 딜레이에 대한 고려

패킷음성통신에서 샘플링된 음성을 패킷으로 만들 때와 수신단에서의 버퍼링, 그리고 network contention 때문에 딜레이가 생기는데 이것이 에코를 일으키는 주요한 원인이 된다. 인공위성을 이용한 국제통화는 약 660ms의 왕복 딜레이를 겪는데 이런 경우에는 echo cancellation을 이용하여 에코를 없앤다.

그림 2-3에 패킷망전화와 사설교환기사이에서 음성 통신을 할 때의 개략적인 블록도가 나타나있다. 여기에서의 지연시간은 근거리 통신망 케이블 위로 음성 패킷이 지나갈 때 겪게 되는 딜레이이다. 지금 패킷망전화 사용자가 말을 할 때 그 음성은 TIC의 하이브리드 회로를 거쳐서 송신단과 수신단이 따로 케이블 위로 채널이 형성된다. 음성패킷은 게이트웨이에 도착해서 하이브리드 회로를 거쳐서 2선식의 전화선을 통해서 PABX로 전해진다. 이 때 커플링의 감쇄가 충분하면 에코는 방지할 수 있다. 그러나 하이브리드 회로가 완전하게 전화선과 임피던스 정합이 이루어지지 않기 때문에 하이브리드 회로로 입력된 음성이 다시 출력된다. 게이트웨이는 이렇게 다시 되돌려진 음성신호도 PABX에서 오는 음성신호와 함께 샘플링하여 패킷으로 만들어 전송한다. 패킷망전화사용자는 이런 식으로 자신의 목소리를 다시 듣게된다.

이 에코에 대한 대책으로는 echo canceller나 echo suppressor를 사용하는 방법과 패킷의 길이를 짧게 만들어(즉, data acquisition 시간을 감소시켜) 전체 종단간 딜레이를 줄이는 방법이 있을 수 있다. 짧은 패킷을 만들어 보내는 방법은 딜레이를 줄일 수 있겠지만 전송해야하는 패킷의 수가 늘어나 채널 용량을 줄인다.

2.3 게이트웨이 구현 요구사항

본 논문에서 연구하는 게이트웨이는 전화선에 직접 맞물림으로써 사설교환기가 보기에 하나의 전화

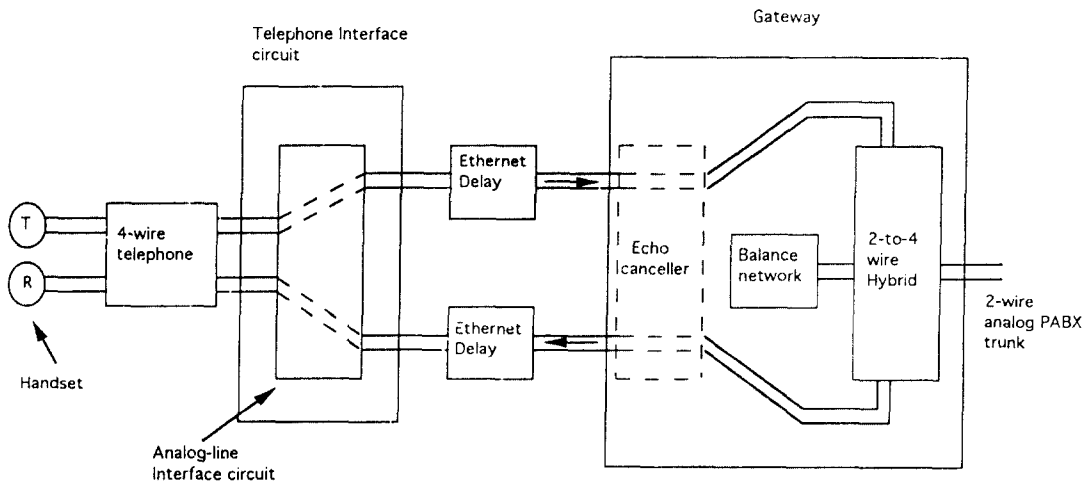


그림 2-3. 패킷망 전화, 게이트웨이 및 PABX 사이의 block diagram

처럼 보이게 한다. 사설교환기와 게이트웨이가 직접 접속되었는 경우에 게이트웨이는 정상적인 전화가 수행하는 여러 signalling을 수행하여야 한다. 그래서 이 장에서는 게이트웨이가 갖추어야 하는 사양을 알아본다.

2.3.1 게이트웨이에서 PABX로 호 설정 요청시

교환기에 신호를 공급하는 경우와 교환기에서 오는 신호를 감지하는 경우로 나누어 진다. 교환기로는 다음과 같은 신호를 공급하여야 한다.

- 점유신호(seizure)
- 발신측 복구신호(clear forward)
- 숫자 정보 신호

위의 3가지 신호를 공급하기 위하여 점유신호와 발신측 복구신호는 DS2249를 이용하고 숫자정보신호는 DTMF 발생기를 이용한다. 또한 교환기에서 오는 다음과 같은 신호를 감지하여 적절히 대응하여야 한다.

- dial tone
- ringback tone
- busy 또는 congestion tone
- howling tone

위의 4가지 신호중 dial tone의 경우에 충분한 지연 시간을 두고 DTMF톤을 발생시킴으로써 해결하고 나머지 신호는 그대로 통과시켜서 패킷 전화사용자가 판단하게끔 한다.

2.3.2 PABX에서 게이트웨이로 호 설정 요청시

교환기에 다음과 같은 신호를 공급하여야 한다.

- 응답신호(answer)
- 착신측 복구신호(clear backward)

위의 2가지 신호는 DS2249를 이용하여 공급한다. 또 추가로 확장된 전화번호를 알려주는 안내방송을 할 수도 있다. 교환기로부터 오는 링신호를 감지해야 하는데 이 분제도 DS2249를 이용하여 해결할 수 있다. 표 2-1에 위에서 언급된 톤들의 스펙이 나타나있다.

Ⅲ. 게이트웨이의 구현

3.1 게이트웨이 프로토콜

패킷망전화와 사설교환기사이를 연동시키는 게이트웨이는 패킷 음성 프로토콜을 수행하면서 동시에 사설교환기와 signalling을 수행하여야 한다. 본 논문에서 구현한 게이트웨이는 전화선이 2개인 것을 대상으로 하였기 때문에 channel number를 두어 이 2개의 전화선을 구별한다. 다음 절에서 게이트웨이가 수행하는 패킷 음성프로토콜과 call packet의 종류, call setup/release 과정, 그리고 channel number 할당에 대해서 설명하였다.

3.1.1 Layered architecture

게이트웨이가 수행하는 패킷음성프로토콜을 OSI seven layer 관점에서 살펴본다.

3.1.1.1 Layered architecture

1) Physical layer

음성통신을 Ethernet 상에서 하므로 Ethernet

표 2-1. Common call progress tone plans

Tone	Precise Tone Plan Frequency	Other Tone Plan Frequency	Level per Frequency (Precise)	interruption Rate(Precise)
Dial Tone	350 + 440Hz	600 + 120Hz 600 + 133Hz	- 13dBm	Continuous
Busy Tone	480 + 620Hz	600 + 120Hz 600 + 133Hz	- 24dBm	60IPM
ATB	Same as busy	Same as busy	Same	120IPM
Ringback Tone	440 + 480Hz	420 + 40Hz	- 19dBm	2s on 4s off
High tone	480Hz	500Hz	- 17dBm	Cont. or Interrupted
Low tone	480 + 620Hz	Same as busy	- 24dBm	Various
Receiver off-Hook Tone(ROH)	1,400 + 2,060 + 2,450 + 2,600Hz	Sam	+ 5VU or + 3to - 6dBm	Cont. or Interrupted

ATB : All Trunks Busy IPM : Interruption Per Minute

physical layer와 동일하며 채널용량은 10Mbps이다.

2) Link layer

LAN 프로토콜의 경우 Link layer는 MAC(Medium Access Control) layer와 LLC(Logical Link Control) layer로 나누어진다. 본 연구에서는 패킷 드라이버가 제공하는 MAC layer만 사용한다. Ethernet type field를 0x8000으로 설정하여 패킷 드라이버가 해당되는 패킷만을 수신하게끔 하였다.

G.764 PVP의 링크 계층에서는 기존의 데이터용 프로토콜과는 달리 재전송을 요구하는 error recovery 과정을 제공하지 않고자 high-level data link control (HDLC)에서 보는 바와 같이 unnumbered informatin(UI) 프레임의 구조를 사용하였다. 음성 프레임의 경우에는 unnumbered information with header check(UIH) frame을, 신호(signaling) 프레임의 경우에는 UI 프레임을 사용하고 있다. 신호 프레임의 경우에는 프레임 전체에 대해 CRC(Cyclic Redundancy Check)를 적용하여 오류를 찾아내는데 비하여, 음성 프레임의 경우에는 프레임 header 필드와 packet header 필드인 8 bytes의 header 필드에 대해서만 CRC를 적용하고(frame header and packet header check sequence) 음성 샘플들에 대해서는 CRC를 적용하지 않는 것이 다르다. 그리고 프레임 header의 주소필드는 영구가상회로(permanent virtual circuit)상에서 동작하기 때문에 미리 정해져 있으며 각 노드에서 정해진 값으로 바뀌어 프레임이 전송된다.

3) Packet layer

상위 계층으로부터 입력된 음성샘플을 패킷으로 만들어 헤더를 붙여 전송한다. 패킷 포맷은 G.764 PVP를 기본으로 하였으며 사용하지 않는 필드는 0으로 하였다. 음성 재생 알고리즘시 사용되는 필드로는 sequence number field만을 사용하였는데 G.764 PVP의 4비트를 8비트로 늘려 사용하였다. Call packet은 상위 Layer로부터 받은 정보를 근거로 하여 적절한 Call packet을 만들어 전송하며 수신한 경우에도 적절한 행동을 취한다.

G.764 PVP는 영구가상회로(permanent virtual circuit)상에 적용하기 위해 만들어지고 음성 패킷과 신호 패킷간에 각각 다른 채널을 통해 패킷을 전송한다. 신호 패킷은 음성 패킷과 다른 논리적 채널을 통해 전송되는 out-of-band signalling방식에 의하여 인

접한 노드들 사이에서 교환되고(link signalling) 송, 수신단에서(end-to-end signalling) 교환된다. 또 signalling패킷과 송신단의 signalling state의 변화를 기록하는 signalling transition 패킷의 송, 수신을 담당한다. 이것은 end-to-end connection management로서 OSI 7-layer의 transport layer의 기능이라고 볼 수 있겠다.

본 연구에서는 call 프레임(frame)과 voice 프레임은 G.764 PVP 프레임을 크게 바꾸지 않고 사용하였다. 음성패킷의 크기는 64바이트와 128바이트의 2가지 경우에 대해서 실험하였다. 그림 3-1과 그림 3-2에 각각의 프레임(frame) format이 나타나있다. 음성

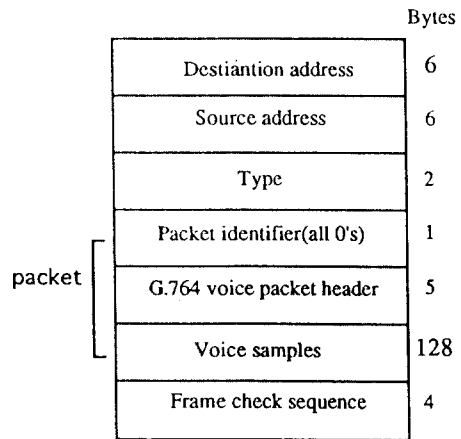


그림 3-1. Voice frame format

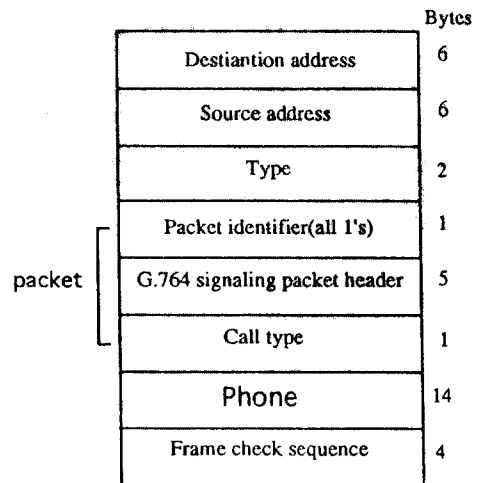


그림 3-2. Call frame format

패킷의 경우에는 원래의 프로토콜과 비슷하며 Call 패킷의 경우에는 Call control type 필드를 1바이트 추가하고 전화번호를 담을 수 있는 필드를 14바이트 추가하였다. 또한 원래의 Time Stamp 필드를 채널 넘버 필드로 사용하여 게이트웨이의 2채널에 대응되도록 하였다. 이에 대한 내용은 다음 절에 자세하게 언급되어있다.

4) Application layer

음성들을 샘플링하고 반대로 수신된 음성샘플들을 재생한다. 또한 PABX에서 오는 링신호를 감지하여 필요한 절차를 수행한다. 외부에서 패킷망전화로 전화를 걸 때 패킷망전화의 어드레스에 해당하는 4자리의 전화번호를 받아 하위 Layer에 알려준다. Packet layer에서 각 상태에 따라 그에 해당하는 톤이 제공된다.

지금까지 구현한 패킷음성프로토콜의 계층별 기능을 살펴보고 다음으로 음성패킷과 call packet의 각 계층에서의 송, 수신 절차에 대해 살펴보겠다.

3.1.1.2 Layer 간의 Interface

1) Link layer procedure

Packet layer entity가 DL_UNIT_DATA_REQUEST를 통해 Link layer entity에게 정보를 전달하면 이 계층의 주체는 Ethernet frame을 만든다음 전송을 한다. 반대로 Link layer entity가 올바른 프레임을 수신하면 Link layer entity는 이 프레임에서 헤더를 제외한 나머지부분을 DL_UNIT_DATA INDICATION primitive를 통해 Packet layer entity에 전달한다.

2) Packet layer procedure

상위계층으로부터 PL_UNIT_DATA_REQUEST primitive를 통해 데이터를 받는다. Packet layer entity는 SSEQ 필드를 1 증가시키고 DL_UNIT_DATA_REQUEST primitive를 통해 Link layer entity에 전달된다. 음성패킷 수신시에는 음성재생알고리즘에 의해 재생한다. 상위 Layer에서 전화선 상태정보를 받은 경우에는 이와 관련된 Call type 필드를 설정한 후 Call packet을 만들어 Link layer에 DL_UNIT_CALL_REQUEST를 통하여 전달된다. 음성패킷의 경우에는 에러 정정을 하지 않지만 Call packet의 경우에는 에러 정정을 한다. 그 과정을 살펴보면 Call packet의 경우 전송실패시 정해진 횟수만큼 재전송을 하며 이후에도 모두 전송이 실패하면 호 설정을

포기하고 상위 Layer에 알린다. 또 전송이 성공한 후에는 타이머를 동작시켜 타이머가 끝날때까지 원하는 응답이 오지 않으면 호 설정을 포기하고 상위 Layer에 알린다. 재전송을 정해진 횟수만큼 했는데도 응답이 오지 않으면 타임아웃 되어 Call이 해제된다. 음성패킷이 계속해서 오지 않을때 타이머를 가동시켜 미리 정해진 횟수 이상 오지 않으면 패킷망전화에 이상이 있다고 판단하여 강제로 Call을 해제시킨다.

4) Upper layer procedure

음성을 샘플링한 후 일정한 크기가 되면 PL_UNIT_DATA_REQUEST를 Packet layer로 보낸다. 전화선의 상태에 따라서 필요한 톤이나 음성을 제공한다.

3.1.2 패킷의 종류

사용되는 패킷의 종류를 다음과 같이 나열하였다. 채널번호 할당과 관련된 Call packet들의 역할은 3.1.4 절에서 따로 설명한다.

1) Call request packet

상대방에게 호 설정 요청시 사용된다. 자신과 상대방의 어드레스가 프레임에 포함되어 있으며 Call type 필드를 Call request로 세팅시킨후 전송한다. 이 때 패킷망 전화에서 PABX로 호 설정 요청시에는 전화번호를 포함하고 있어서 게이트웨이가 DTMF generation을 가능하게 한다.

2) Call accept packet

패킷망전화에서 보낸 call request packet을 받았을 때 게이트웨이는 빈 채널이 있으면 DTMF generation을 한 후 call accept packet을 보낸다. 이 패킷을 보낸 이후에 게이트웨이는 BYPASS 상태로 전환하여 모든 ring back tone따위의 신호와 음성을 통과시킨다.

3) Busy packet

패킷망전화에서 보낸 call request packet을 받았을 때 게이트웨이에 빈 채널이 없으면 busy packet을 보낸다.

4) Call connect packet

PABX에서 패킷망전화로 호 설정 요청을 할 때 패킷망전화에서 보낸 call connect 패킷을 수신한 후

BYPASS 상태로 전환하여 음성을 통과시킨다.

5) Call release packet

게이트웨이에서 이 call packet을 사용하지는 않지만 BYPASS 상태에서 이 패킷을 수신하여 사설교환기와 연결된 전화선을 hook on 시키는데 사용된다.

6) Call release ack packet

BYPASS 상태에서 Call release packet을 수신한 후 이 패킷을 패킷망전화에게 보내어 Call release packet의 재전송을 막는다.

3.1.3 Call setup/release mechanism

이 절에서는 패킷망전화에서 PABX로 호 설정을 요청할 때와 PABX에서 패킷망전화로 호 설정 요청을 할 때로 나누어 호 설정 및 호 해제 절차를 설명한다. 그림 3-3과 그림 3-4에 각각의 절차를 나타내었다.

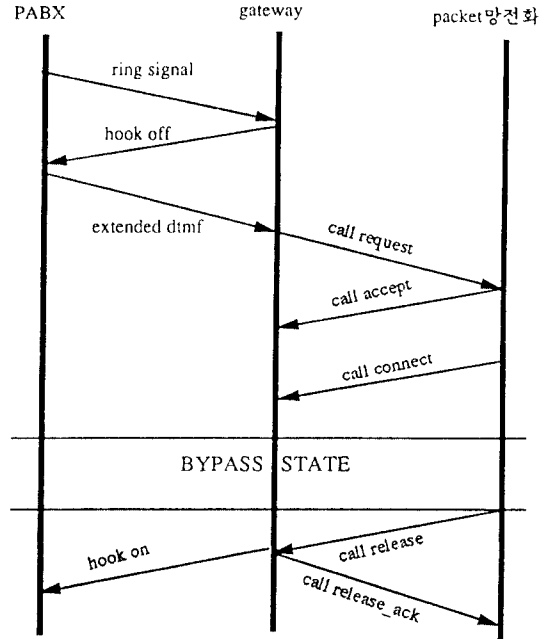


그림 3-4. PABX에서 패킷망전화로의 호 설정 요청시 (call setup release)

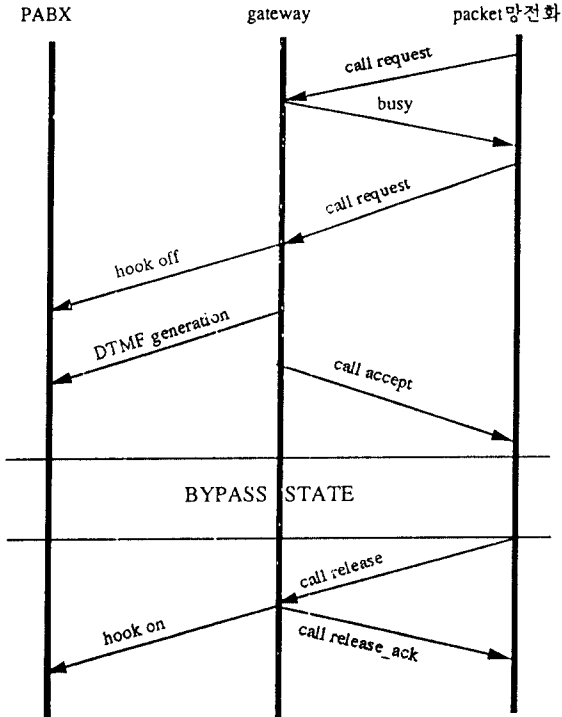


그림 3-3. 패킷망전화에서 PABX로 호 설정 요청시 (call setup release)

먼저 패킷망전화에서 PABX로 호 설정 요청을 하는 경우를 살펴보면 다음과 같다. 먼저 패킷망전화에서 게이트웨이를 통해 외부로나가는 전화번호를 가진 call request packet을 보낸다. 게이트웨이는 이 패킷을 받고 사설교환기와 연결된 2개의 전화선중 빈 전화선을 hook off 시킨다. 해당되는 DTMF generation을 한 후 call accept packet을 보내어 BYPASS 상태로 천이한다. 이 때 ring back tone이나 busy tone도 그대로 통과시켜 패킷망 전화 사용자가 판단하게끔 한다. 만약 패킷망 사용자가 busy tone을 듣거나 대화를 끝낸 후에 수화기를 내려 놓으면 게이트웨이로 call release packet을 전송하여 게이트웨이가 PABX와 연결된 전화선을 hook on 시키고 call release ack packet을 패킷망 전화에게 전송한다.

다음으로 PABX를 통하여 패킷망전화로 호 설정 요청시에는 다음과 같은 절차를 거친다. 먼저 게이트웨이가 PABX와 연결된 전화선의 링 신호를 감지하여 전화선을 hook off시킨다. 다음에 패킷망전화를 구별하는 별도의 4자리의 전화번호를 입력받아 해당되는 패킷망전화로 call request packet을 보낸다. call accept, call connect packet을 받아 BYPASS

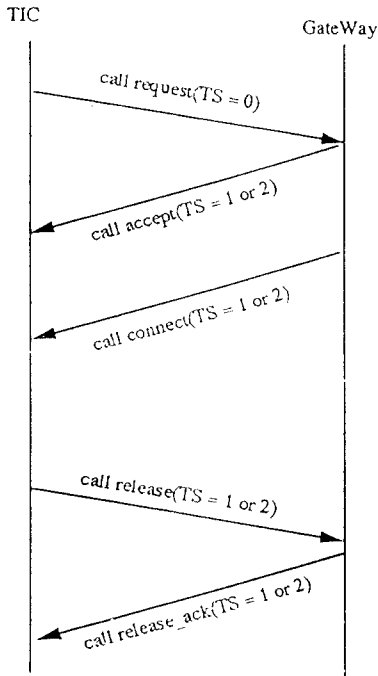


그림 3-5. 패킷망전화에서 PABX로 호 설정 요청시 (channel number)

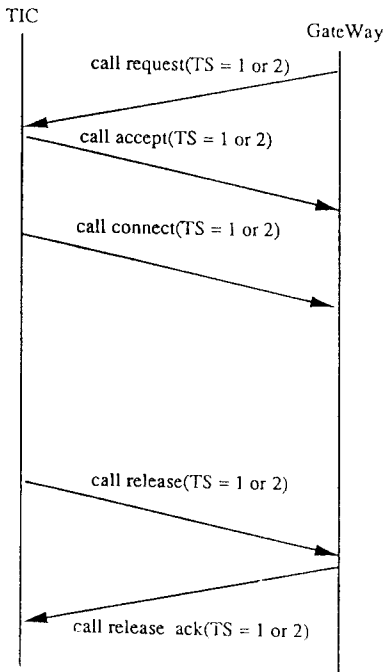


그림 3-6. PABX에서 패킷망전화로의 호 설정 요청시 (channel number)

상태로 가며 호 해제 절차는 위와 동일하다. 이때 타이머를 설정하여 응답이 없을 때에는 전화를 hook on 시킨다.

3.1.4 Channel 번호 할당

게이트웨이가 수용하는 전화선이 2개 이므로 각각에 채널번호를 할당하였다. 그림 3-5와 그림 3-6에 패킷망전화에서 사설교환기로 호 설정 요청을 할 때와 PABX에서 패킷망전화로 호 설정 요청을 할 때의 채널번호 할당에 대해서 나타나있다.

먼저 패킷망전화에서 게이트웨이로 call request packet을 보낼때 채널번호를 0으로 해서 보내면 게이트웨이는 call accept packet에 비어있는 채널번호를 할당하여 전송한다. 그 이후에 패킷전화와 게이트웨이사이에서 주고받는 모든 call packet과 voice packet은 그 채널번호를 가지고서 통신한다.

게이트웨이에서 패킷전화로 call request packet을 보내는 경우에는 call request packet 안에 채널번호를 할당하여 호 해제시까지 사용된다.

3.1.5 전화번호 할당

각각의 패킷망전화에 Ethernet card와 일대일 대응되는 4가지의 전화번호를 부여한다. PABX에서 패킷망전화로 호 설정 요청시에 이 전화번호를 받아 게이트웨이는 패킷망전화와 호를 설정할 수 있다.

3.2 하드웨어의 구조

그림 3-7에 게이트웨이에 대한 블록도가 나타나있다. 게이트웨이 하드웨어는 전화선과의 인터페이스를 담당하는 부분과 음성의 A/D, D/A 변환에 관련된 부분, 전화번호를 받는 부분, DTMF generation 부분, 그리고 PC와의 인터페이스에 관련된 부분으로 나누어진다. 밑에서 이들의 구체적인 구조와 기능에 대해서 살펴보겠다.

3.2.1 전화선과의 인터페이스 부분

이 부분은 사설교환기가 보기에 전화처럼 보이게 하는 부분이다. 실제 전화기가 수행하는 여러가지 일들을 수행하고 미국 Dallas사의 DS2249모듈을 이용하였다. 그림 3-8에 DS2249모듈의 블록도가 나타나있다. 다음에 DS2249가 하는 일들을 설명한다.

- Hook control

일반 전화에서 수화기를 들고 내리는 기능을 수행한다.

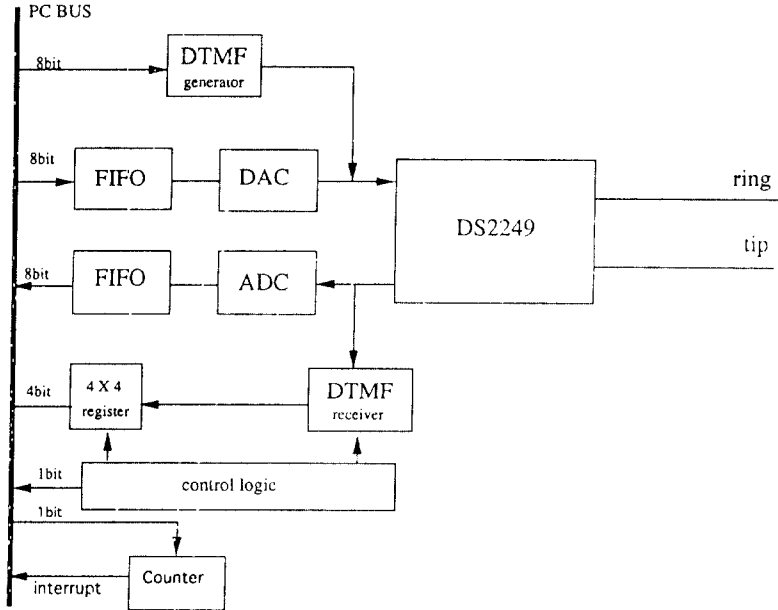


그림 3-7. 게이트웨이 block diagram

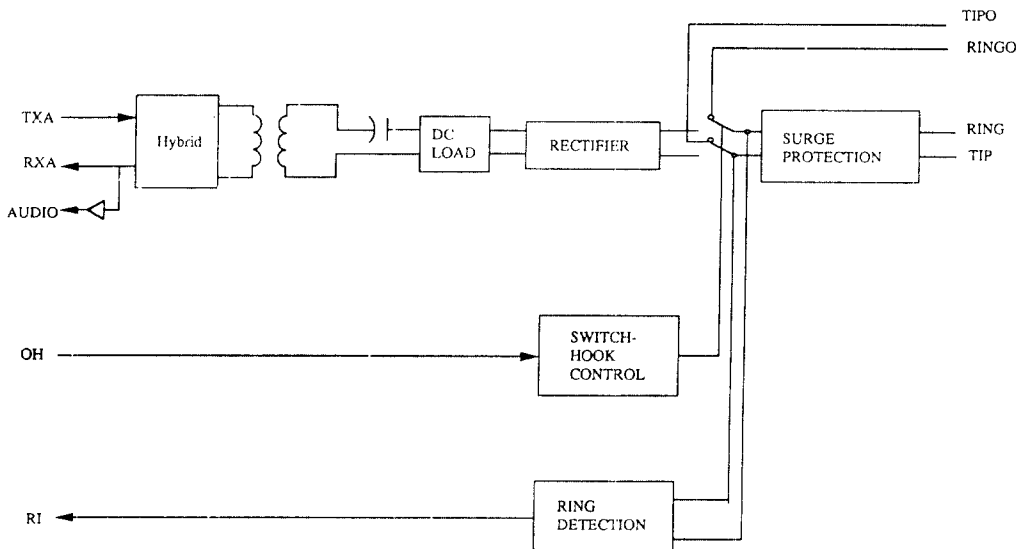


그림 3-8. DS2249 block diagram

- Ring Detection

특별한 주파수와 크기를 가진 링 신호를 감지하여 hook control을 할 수 있게 한다.

- Hybrid 기능

일반적으로 전화선은 송신신호와 수신신호가 함께 혼재되어있다. 이를 전송단과 수신단으로 분리하는 기능을 한다. 이상적인 하이브리드 회로는 전송단에서 보낸 음성신호가 수신단으로 되돌아오지 못하도록 하지만 실제적인 경우엔 그렇지 못해서 에코의 원인이 된다.

3.2.2 Speech Processing

음성신호의 A/D, D/A에 관계된 모든 일들을 처리한다.

1) Filter

Op-amp를 이용한 Sallen nad key low pass filter를 사용하였으며 대역폭은 3.4kHz로 하였다.

2) ADC

음성신호의 디지털 변환은 A/D converter를 사용하였다. 샘플당 비트수는 8비트이고 샘플링은 8kHz로 하였으며 ADC0804를 사용하였다.

3) DAC

8비트의 디지털 음성신호를 아날로그 신호로 바꾸어준다. 또한 안내방송을 미리 저장된 데이터를 이용하여 DAC를 통하여 제공할 수 있다.

4) FIFO Buffer

512바이트용량의 삼성전자에서 제작된 KM75C01 AP-25를 사용하였다. 전송단, 수신단 모두 사용하였는데 전송단, 수신단 모두 8ms 마다 인터럽트가 걸릴 때 데이터를 읽고 쓴다.

3.2.3 Dialing detector

DTMF 전화기의 키패드를 누르면 특정 주파수가 조합된 출력이 나온다. 이런 특정주파수의 조합을 감지해서 그에 해당하는 패킷망전화의 어드레스를 알아낸다. 이를 위해서 DTMF receiver용 IC인 MT8870을 사용하였다. 이는 각 주파수의 조합을 감지해서 이에 해당하는 디지털 코드로 바꾸어준다.

3.2.4 DTMF generation

사실교환기로 호를 설정하기 위해서는 위와는 반대의 과정이 필요하다. 삼성전자에서 나온 IC인 KS58006을 사용하였다. 이 칩은 3.579545MHz의 수정발진자를 함께 사용하는 CMOS IC이다. 또한 DTMF 방식과 PULSE 방식을 동시에 지원하는데 이 논문에서는 PULSE 방식은 고려하지 않았다.

3.2.5 PC와의 인터페이스 및 controller 부분

PC 내의 확장슬롯을 통해서 어드레스 디코더를 이용하여 해당되는 I/O port에 원하는 데이터를 읽거나 쓴다. 입력포트의 내용으로는 게이트웨이의 현상태를 얻기위한 부분과 A/D 된 음성데이터를 얻는 부분 그리고 명령 레지스터의 내용을 얻는 부분으로 이루어져있다. 출력포트의 내용으로는 명령 레지스터에 쓰는 부분과 음성데이터를 DAC에 쓰기 위한부분, 그리고 DTMF generaton을 위해서 쓰는 부분으로 이루어져 있다. 사용된 클럭은 DTMF generation에 이용되는 3.579545MHz 클럭을 제외하고는 PC 내의 확장슬롯에서 제공되는 14.31818MHZ 클럭을 이용해서 ADC, DAC에 8kHz 클럭을 DTMF receiver에 3.5795MHZ 클럭을 카운터로 분주해서 제공하였다.

3.3 소프트웨어의 구조

게이트웨이의 소프트웨어는 그림 3-9와 같은 구조로 되어있다. 패킷 전송은 패킷 드라이버에서 제공하는 send_pkt 함수를 사용한다. 다음 절에서 패킷 드라이버와 수신단, 그리고 state diagram에 대해서 설명하였다.

3.3.1 패킷 드라이버

여러 회사에서 각기 다른 네트워크 카드를 생산함으로써 통신프로그램의 이식성 문제를 야기시킨다. 예를 들면 3COM에서 나온 네트워크 카드위에서 동작하는 통신 프로그램이 Western Digital에서 나온 제품 위에서는 동작하지 않는다. 이런 문제를 해결하기 위하여 여러가지 표준 스펙이 여러 회사에서 나왔는데 대표적인 것으로는,FTP사의 Packet driver specification, 3Com과 MS사의 NDIS(Network Driver Interface Specification), Novell사의 ODI(Open Data-link Interface) 등이 있다. 본 논문에서는 FTP 사의 Packet driver specification에 따라 제작된 드라이버를 이용하여 Ethernet MAC에 해당하는 기능을 대치하였다.

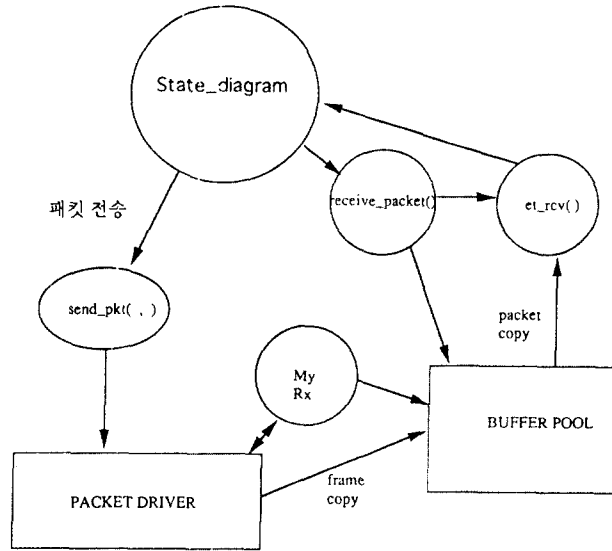


그림 3-9. 게이트웨이 소프트웨어 구조

Next(2byte)
len(2byte)
filled(1byte)
copy_ok(2byte)
dbuf(2byte)

Next field : 다음 RBD 를 가리킨다.

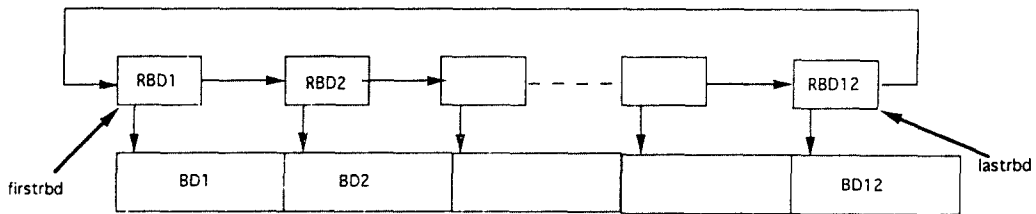
len field : 수신한 프레임의 길이를 나타낸다.

filled field : RBD 가 가리키는 BD 가 비었는지 여부를 나타낸다. 0 -> empty, 1 -> filled

copy_ok field : 수신된 프레임이 몇개의 BD 를 차지하고 있는지를 나타낸다.

dbuf field : BD 위치를 가리킨다.

(a) RBD structure



(b) BUFFER POOL

그림 3-10. RBD structure와 BUFFER POOL

3.3.2 수신단

수신단은 패킷 드라이버가 수행하는 MyRx와 수신된 패킷을 임시 저장하는 buffer pool, 그리고 buffer pool에서 패킷을 꺼내오는 receive packet의 3부분으로 이루어져있다.

먼저 buffer pool단부터 살펴보겠다. buffer pool은 버퍼인 BD(Buffer Data)와 버퍼와 관련된 정보를 가리키는 RBD(Receive Buffer Descriptor)로 이루어져있다. 하나의 버퍼는 256바이트로 이루어져 있다. 그림 3-10에 각각의 format이 나타나있다. RBD(Receive Buffer Descriptor)는 원형 링 형태로 각각 연결되어 있고 first rbd, empty rbd, filled rbd 등의 포인터를 이용하여 동작된다.

다음으로 MyRx단을 살펴보겠다. MyRx는 패킷 드라이버 함수 2번인 access_type을 호출할 때 패킷 드라이버에게 알려주는 프로시저이다. 패킷 드라이버는 해당되는 패킷 수신시 이 MyRx단을 두번 수행한다. 처음 불리어졌을 때 패킷 드라이버는 수신한 패킷의 길이를 알려주어 MyRx가 그 길이에 해당되

는 메모리 위치를 패킷 드라이버에게 알려준다. 패킷 드라이버는 그 위치에 수신한 패킷을 복사한 후 MyRx를 두번째로 부른다. MyRx는 복사가 완료되었음을 확인하고 필요한 포인터와 RBD(Receive Buffer Descriptor)를 수정한 후 다음 패킷이 오는 것을 준비한다. 그림 3-11와 그림 3-12에 MyRx 함수에 관해서 자세하게 묘사되어 있다.

다음으로 receive_packet를 살펴보겠다. receive_packet는 filled_rbd를 이용하여 RBD의 filled란을 검사하여 buffer pool에 수신된 패킷이 있는지를 확인한다. 만약 buffer pool에 패킷이 있으면 et_rcv 함수를 불러서 그 안에서 여러가지 일들을 하게된다. 이 et_rcv 함수가 하는 구체적인 일들은 call 성립전에는 buffer pool에서 응용 buffer로 수신된 패킷을 복사하고 call 성립후의 BYPASS 상태에서는 해당되는 FIFO(First in First out) buffer로 직접 음성 데이터를 복사한다. 그림 3-13과 그림 3-14에 receive_packet 함수에 관해서 자세하게 묘사되어 있다.

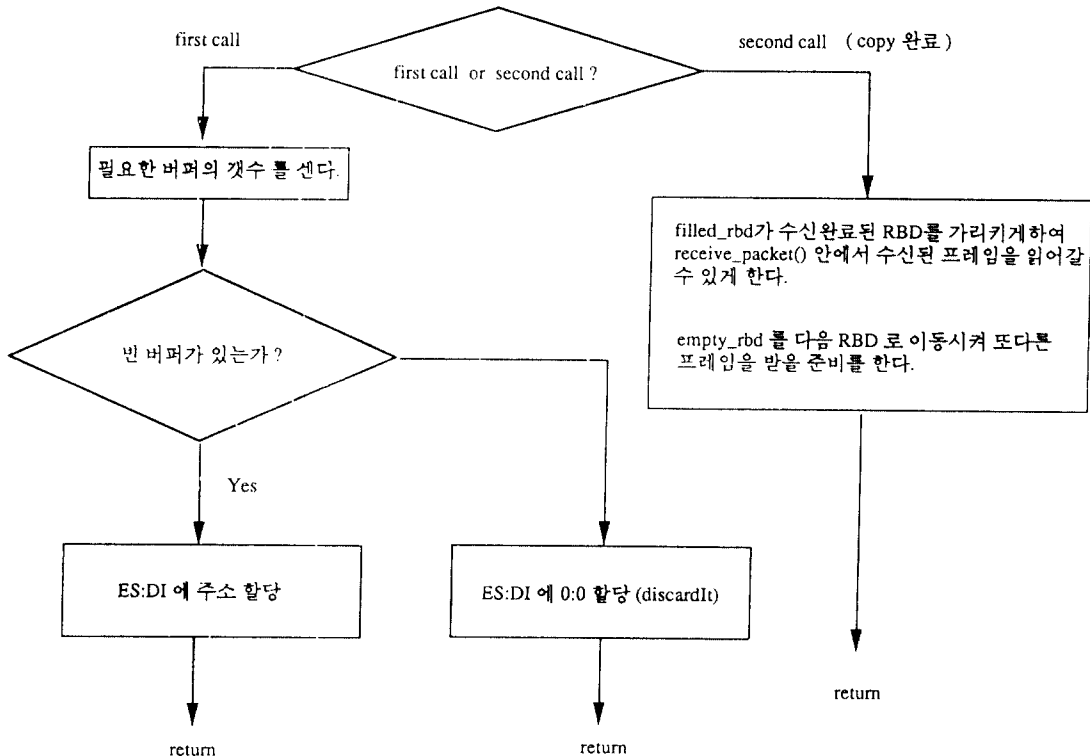


그림 3-11. MyRx의 절차

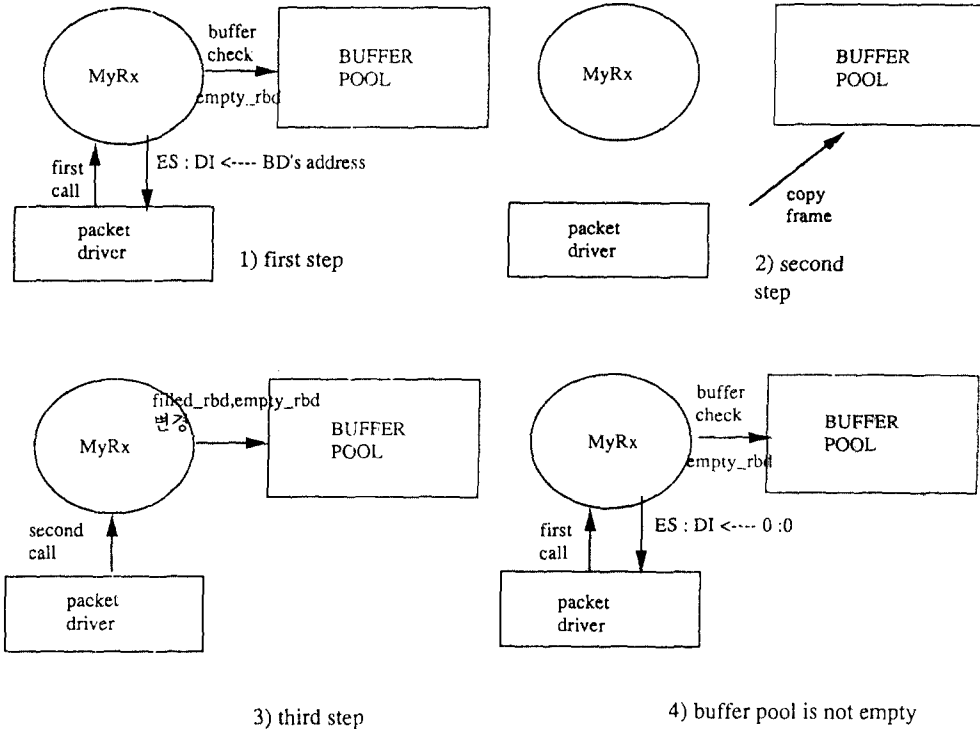


그림 3-12. MyRx 수행시 packet driver와의 상호작용

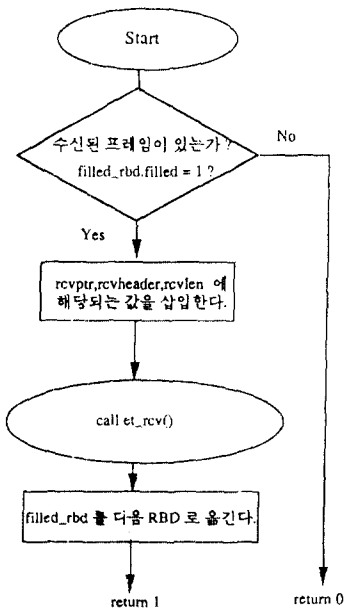


그림 3-13. receive_packet의 절차

3.3.3 State transition diagram

실제 통화시 주고받는 Call 프레임과 그에 따른 상태의 천이는 그림 3-15에 나타나있다. 음성통화시 정의한 상태는 다음과 같다.

1) IDLE state

호 접속이 이루어지지 않은 상태로 초기상태이다. 이 상태에서는 Call request packet을 수신하거나 링 신호가 있을경우 다른 상태로 이동한다. 만약 사설교환기로 나가는 2개 채널이 모두 사용중일 때 Call request packet을 받으면 Busy packet을 보내준다.

2) DIALING state

사설교환기에서 게이트웨이로 호 설정 요청시에만 나타난다. IDLE 상태에서 링 신호가 감지되었을 때 이 상태로 옮겨온다. 이 상태에서는 4자리의 전화번호가 입력될 때까지 기다린다. 이 4자리의 전화번호가 입력되면 그 번호를 읽어 전화번호 테이블에서 해당되는 어드레스를 찾은후에 Call request packet을

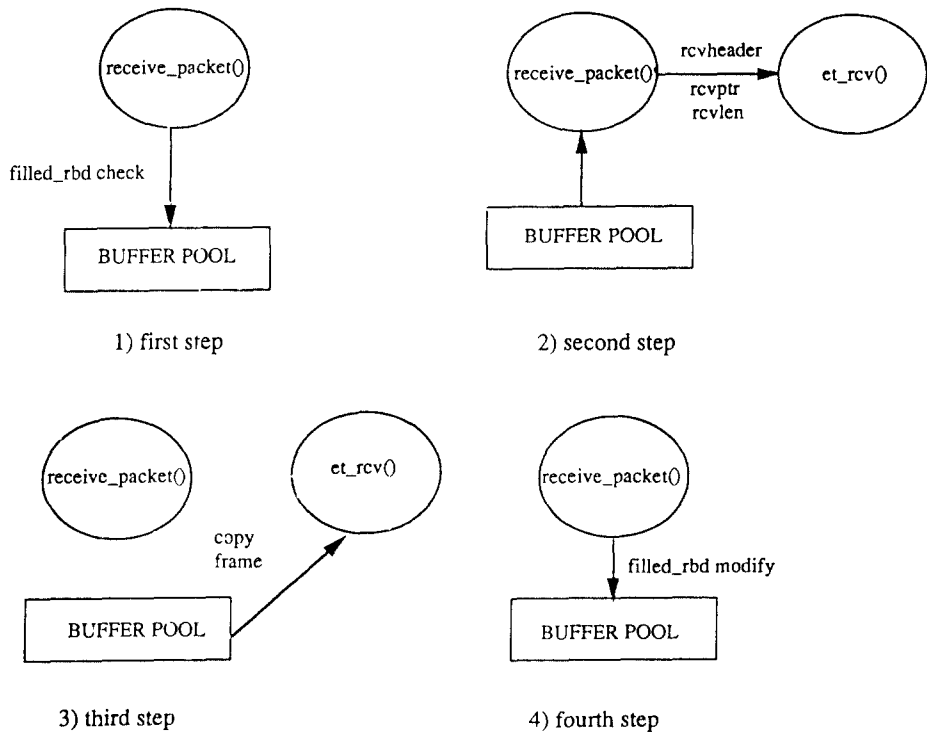


그림 3-14. receive_packet 수행시의 단계

보낸다. 4자리의 전화번호를 입력받기위하여 DTMF receiver chip(MT8870)과 4×4 Fifo chip을 이용한다.

3) WAIT state

사설교환기에서 게이트웨이로 호 설정 요청시에만 나타난다. DIALING 상태에서 Call request packet을 보낸후에 이 상태에서 Call의 Accept를 기다린다. 만약 상대방이 통화중일 경우 Busy packet을 송신하게 되는데, 이 때는 Busy 상태로 전환한다. 또한 Call accept packet을 수신하면 RINGING 상태로 전환한다.

4) RINGING state

사설교환기에서 게이트웨이로 호 설정 요청시에만 나타난다. Call이 Accept된 후에 패킷망전화의 벨을 울려주는 상태에서 사설교환기에 잠시 기다리라는 안내방송을 할 수 있다. Call connect packet을 받으면 BYPASS 상태로 가서 음성패킷을 주고받아 통화를 할 수 있는 상태가 된다. 이 때 타임 아웃되면

STOP 상태로 전환한다.

5) BUSY state

사설교환기에서 게이트웨이로 호 설정 요청시에만 나타난다. Call request packet을 보냈을 때 상대방이 통화중일 경우 Busy packet을 수신하게 된다. 이 상태에서는 사설 교환기로 안내방송을 할 수 있고 전화를 Hook on 한 후 IDLE 상태로 전환한다.

6) STOP state

사설교환기에서 게이트웨이로 호 설정 요청시에만 나타난다. 이 상태에서는 전화를 Hook on 시키고 IDLE 상태로 전환하게 된다.

7) CALL IN state

패킷망전화에서 게이트웨이를 통하여 PABX로 호 설정 요청시에 나타난다. IDLE 상태에서 Call request packet을 받은후에 이 상태로 오게된다. Call request packet안에 있는 전화번호를 저장한 후에 DTMF 상태로 전환시킨다.

8) DTMF state

패킷망전화에서 게이트웨이를 통하여 사설교환기로 호 설정 요청시에 나타난다. 이 상태에서는 DTMF generation을 한 후에 Call accept packet을 보내고 BYPASS 상태로 전환시킨다. 이때 Call release packet을 받으면 하던 일을 중지하고 IDLE 상태로 전환시킨다.

9) BYPASS state

호 설정이 끝나면 이 상태로 와서 패킷망전화와 사설교환기 사이에서 음성신호들을 중계시켜준다. 즉 패킷망전화에서 오는 음성패킷들을 아날로그로 바꾸어 PABX로 돌려주고 PABX에서 오는 아날로그 신호를 패킷으로 바꾸어 패킷망전화로 전송한다. 이때 8ms나 16ms로 interrupt가 걸리면 64바이트나 128바이트씩 패킷으로 만들어 전송한다. 이때의 딜레이는 약 8ms나 16ms 정도이다. 수신단에서는 sequence number를 가지고 음성재생을 한다. 이때 음성패킷이 어느 설정된 값 이상으로 오지 않으면 패킷망전화에 이상이 생겼다고 판단하여 Call을 해제한다. 대화가

끝난후에 패킷망 전화에서 전화를 내려 놓으면 게이트웨이로 call release packet을 보내게 된다. 게이트웨이는 이 패킷을 받으면 패킷망전화로 call release ack packet을 보내고 PABX와 연결된 전화선을 Hook on한 후에 IDLE 상태로 전환시킨다.

IV. 실험 결과

실험은 그림 4-1과 같이 하였다. 본 연구에서는 FDDI Hub에서 나온 Ethernet망위에 패킷음성전화와 사설교환기와 연결된 게이트웨이를 설치하여 실험하였다.

딜레이를 근본적으로 해결하기 위하여 음성패킷을 64바이트씩 보내기로 하였다. 음질을 좋게 하기 위하여 묵음(silence) detection 알고리즘은 사용하지 않았다. 이를 위해 게이트웨이 하드웨어에 카운터를 두어 8ms마다 인터럽트가 걸려 인터럽트 핸들러내에서 패킷 송수신에 필요한 일들을 수행하게끔 하였다. 그 결과 패킷망전화와 일반전화사이에 정상적인 통화가 가능하다는 것을 확인하였다. Call release packet에 의한 호 해제도 정상적으로 수행되었다. 에코는 조금

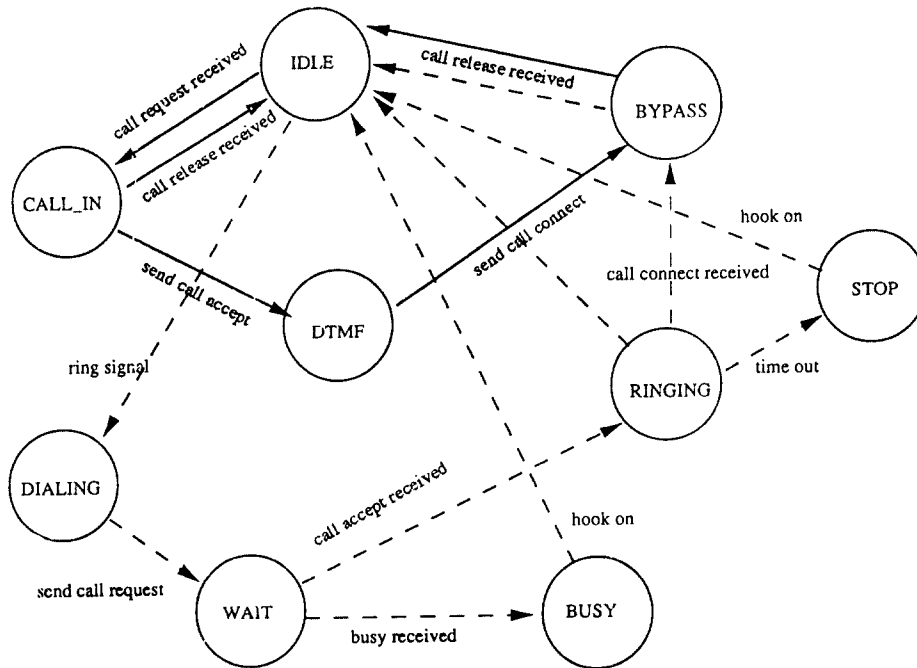


그림 3-15. State diagram

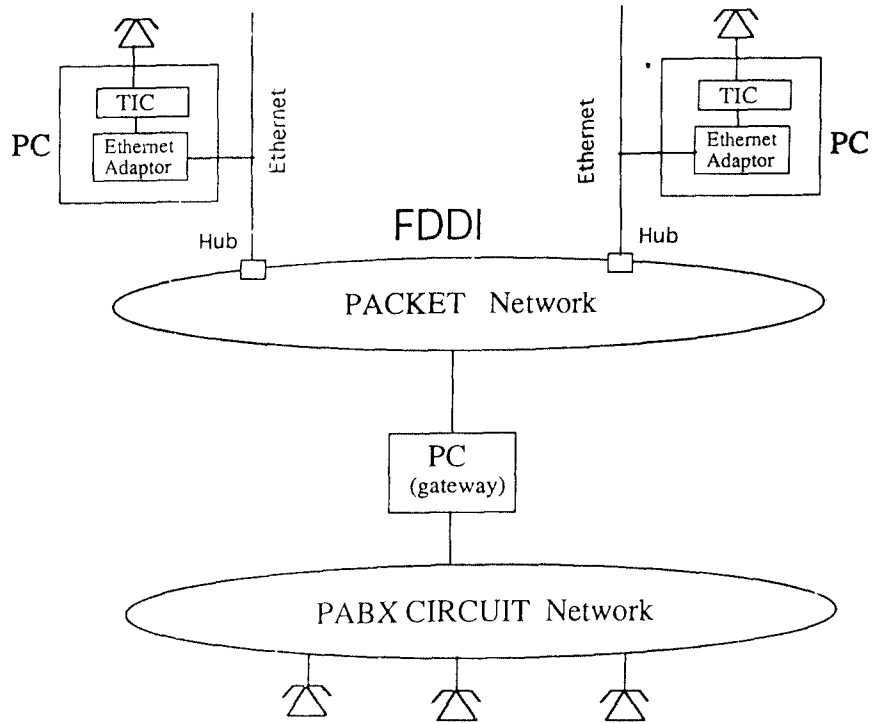


그림 4-1. 전체 실험 개요도

존재하였으나 통화에 지장을 줄 정도는 아니었다. 사설교환기전화와 패킷망전화의 음질을 비교하면, 상업용 사설교환기에서는 거의 느낄수 없는 잡음과 에코가 패킷망전화에서는 상당히 심하였다. 이는 Ethernet을 지날때 겪는 딜레이와 게이트웨이, TIC board에서 발생하는 잡음때문인 것으로 추측된다. 통화는 구내전화와 시내지역까지 실험하였다. 시내지역과 통화했을때 소리가 조금 작아진 것 이외에는 별차이가 없이 정상통화가 가능하였다.

그림 5-1에 나타나 있는바와 같이 게이트웨이와 패킷망전화를 같은 Ethernet위에 물린경우와 그렇지 않은 경우에 대해서도 실험하였다. 현재 실험 네트워크는 FDDI를 단간으로 각 Hub에 Ethernet망이 물려있는 형태로 되어있다. 그래서 Ethernet address만 가지고도 기관내의 어디라도 데이터통신을 할 수 있다. 실험은 패킷망전화와 게이트웨이를 서로 다른 Ethernet망에 두어서 하였다. 이때 음성패킷은 한 Ethernet을 지나 FDDI망을 거쳐 다른 Ethernet망을 지나서 원하는 장소에 도착하게된다. 실험결과를 패킷망전화와 게이트웨이를 같은 Ethernet망위에 물린

경우와 비교해서 별 차이가 없었다. 이는 100Mbps의 속도를 갖는 FDDI 망위에서의 지연시간이 상대적으로 무시해도 될 정도로 작다는 것을 의미한다.

V. 결 론

본 논문에서는 패킷음성통신을 기존의 전화망과 연결시키기위하여 DOS를 사용하는 IBM PC 상에서 게이트웨이를 설계하고 구현하였다. 이번 연구를 통하여 패킷망에서의 음성통신의 범위를 기존 전화망으로 확장시켰다. 패킷망전화는 기존에 연구실에서 개발한 TIC(Telephone Interface Card) board를 사용하였다. 하드웨어는 전화선접속부와 A/D, D/A 변환부, 전송/수신 버퍼와 DTMF 전송단/수신단으로 구성되어있다. 전화선 접속부는 DS2249 모듈을 사용하였으며 음성은 8kHz로 샘플링되며 각 샘플을 8bits로 A/D, D/A 변환되고 전송/수신 버퍼는 FIFO chip을 사용하였다. 외부에서 전화가 오면 전화선을 'Hook on' 시킨후에 DTMF 수신단에서 패킷망전화의 Address를 받는다. DTMF 전송단은 외부로 전화

를 걸때 사용된다.

Ethernet을 구동하는 모듈은 FTP 사의 패킷드라이버 스펙에 맞게 인터페이스를 구현하여 이를 이용하였다. 시스템은 전화를 걸거나, 받는등 입력사건(event)에 따라 디자인된 상태를 천이하며 동작된다. 일단 BYPASS 상태로 가면 하드웨어 인터럽트가 8ms 마다 걸려서 64바이트씩 패킷으로 만들어 전송하고 수신한 음성패킷을 재생한다.

실험 결과 사설교환기와 근거리 통신망사이의 음성 통신기능이 작동함을 보였으며, 이를 이용하면 근거리 통신망의 활용 범위를 음성통신까지로 넓힐 수 있다.

참 고 문 헌

1. T. A. Gonsalves, "Packet Voice communication on an Ethernet local computer network : an experimental study," Communications of the ACM, pp.178-185, 1983.

2. CBM Mishra and R. Madhusudhan, "Voice data integration on FDDI Network implementation and analysis," ICCS, pp.220-228, 1990.
 3. E. Friedman and C. Ziegler, "Packet voice communication over PC-based Local Area Networks," IEEE JSAC., vol.7, no.2, Feb. 1989.
 4. Joong Ma and Inder Gorpai, "A bind voice packet synchronization strategy," RC 13893(#62194), Nov. 1988.
 5. 전홍범, 은종관, 조동호, "패킷음성/데이터 집적단말기의 개발" 전자통신종합학술대회 논문집, pp. 50-56, 1987.
 6. D. K. Melvin, "Coax local network carries both voice and data," Electronic Design, pp.55-66, Jan. 1982.
 7. 이상길, 신병철, 김윤관, "PC용 패킷 음성 프로토콜의 구현," 한국통신학회 논문지 vol.18, no.12, pp.1841-1854, 1993년 11월.



安 鏞 澈(Yong Cheol Ahn) 정희원
 1968년 6월 17일생
 1992년 2월 : 연세대학교 전자공학과 졸업(학사)
 1994년 2월 : 한국과학기술원 전기 및 전자공학과 졸업(석사)
 1994년 3월 ~ 현재 : 삼성전자통신 기술연구소 근무



申 炳 喆(Byung Cheol Shin) 정희원
 1952년 11월 2일생
 1975년 2월 : 서울대학교 전기공학과 졸업
 1977년 2월 : 한국과학기술원 전기 및 전자공학과 석사졸업
 1984년 8월 : 한국과학기술원 전기 및 전자공학과 박사졸업
 1977년 2월 ~ 1980년 2월 : 한국전자통신연구소 연구원
 1987년 3월 ~ 1988년 2월 : SRI International에서 Post Doc. 연수
 1984년 9월 ~ 현재 : 한국과학기술원 전기 및 전자공학과 부교수
 1993년 7월 ~ 현재 : 국내 정보통신전문위원회(SC6) 위원장
 ※주관심분야 : Multimedia 통신, High Speed Network, Wireless Communication, ATM Network 분야

본 논문은 한국과학재단 핵심전문연구과제 지원에 의하여 이루어졌음