

네트워크 개념의 등장과 온라인·시스템의 변혁(Ⅱ)

강철희 · 송주석 / 전자장치개발실

—譯者註—

본 해설기사는 1977년 6월15일 발행된 日本의 COMPUTOPIA 라는 잡지의 別冊으로 나온 「컴퓨터·네트워크 入門」이라는 책에서 발췌, 번역한 것으로 지난 호에 게재된 기사의 제목은 다음과 같다.

I. 데이터 통신 시스템의 목적과 특징

II. 라인·콘트를 절차

III. SDLC 등장의 배경과 특징

지금까지는 데이터통신을 할때, 2 지점 간에 서로의 정보를 정확히 주고 받게하는 각종규칙(라인·콘트를 절차)에 대해 설명해 왔는데, 여기에서는 종래의 방식보다 더욱 수준 높은 라인·콘트를 절차 개발의 필요성과, 일반적으로 고수준 라인·콘트를 절차라고 불리는 것의 내용, 특징 등에 대해 설명하고자 한다.

종래부터 사용되어 온 비동기식 또는 동기식에 의한 기본적인 제어방법은, 모두 데이터 통신 내지는 온라인·시스템의 발전과 함께 개량되어 왔지만, 정확히 말한다면, 데이터 통신시스템의 초기 단계의 것으로, 별로 표준화되어

있지 않은 모양으로 발전되어 왔다고 말할수 있겠다.

그리하여, 데이터 전송 수요량의 폭증과 각종 온라인·시스템의 눈부신 발전은 다양한 기능의 추가와 전송될 정보의 내용까지 복잡하게 하므로써, 지금까지의 문자단위의 전송만으로는 도저히 그 많은 요구를 감당해 내지 못하는 결점 등이 여러개 지적되게 되었으며, 이 문제점들을 해결할 수 있는 새로운 제어방식이 고안되게 되었다.

즉 고수준 라인·콘트를 절차로서, ISO의 HDLC(High Level Data Link Control)라고 불리는 절차가 등장하게 되었다. 이 HDLC의 종래방식과 틀린 점을 열거하면 다음과 같다.

- 1) HDLC는 기본적으로 비트(bit) 단위로 된 포맷(format)이므로, 데이터 전송은 〈그림13〉과 같은 프레임 형식으로 행한다.
- 2) 보다 전송능률을 좋게하기 위하여 전 2 중 전송은 물론 연속전송, 일괄응답도 가능케 한다.
- 3) 보다 신뢰성 높은 전송이 가능하도록, CRC(Cyclic Redundancy Check) 방식에 의한 에러 체크(error check)를 행하고, 프레임에 번호제를 채용하여 도중에 프레임이 없어졌는지 여부를 검사한다.

IBM 이 1974년 시스템 네트워크 체계의 발표와 함께 공표한 SDLC(Synchronous Data Link Control)는 약간 다른 점을 빼다면 기본적으로 국제 표준안으로서 제시되었던 HDLC와 동일한 성질의 것이다. 여기서는 SDLC에 관해서 설명할 것이다.

프래그	어드레스	제어	정보	FCS	프래그
01111110	8비트	8비트	데이터	16비트	01111110

〈그림13〉 HDLC의 프레임 형식

1. 종래의 제어절차의 문제점

종래의 제어방법중에서 특히 폴링·어드레스의 경우, 다음과 같은 결점들을 지적할 수 있다.

- 가. 지난 호의〈그림10, 11〉에 보인바와 같이 각 전송블럭마다 응답을 필요로 하기 때문에, 전 2중통신을 행하기는 상당히 어렵다. 또한, 다수의 블럭을 연속해서 송신하는 경우 효율이 떨어진다.
- 나. 분기 접속했을때 어떤 특정의 단말 장치간에 컴퓨터가 데이터를 전송하고 있는 동안, 다른 단말은 회선이 개방될 때까지 완전히 휴식상태에 있게 되어, 회선의 이용도가 떨어진다.
- 다. 비동기식 또는 동기식 전송의 경우, 에러 검사는 데이터 부분에서만 행하고, 〈표1〉에서 정의한 제어문자중 STX의 앞부분에서는 행하지 않는다. 특히 폴링·어드레스중 이거나 응답하는 제어문자 사이에서 에러가 일어나면 검출할 길이 없다.
- 다. 송신하는 전송블럭에 대한 번호를 체크하지 않기 때문에 데이터 블럭의 중복 또는 탈락을 완전히 방지하기는 곤란하다.
- 마. 단말과 컴퓨터 사이의 정보송출 형식은 폴링, 데이터의 송출, 긍정 또는 부정 응답시 각각 다른 모양을 하기 때문에 시스템에는 항상 회선을 감시하고 탐색해야하는 부담이 따른다.
- 바. 특정한 문자 코드에 의해 회선을 제어하고 있기 때문에, 만일 회선제어에 관해서 새로운 기능을 추가하려고 하면, 지금까지 어느 제어문자라도 사용하지 않은 빌구성의 제어

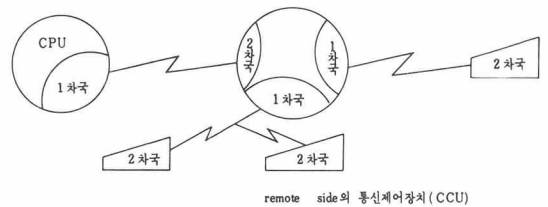
문자를 새로 정의하여야 하는데 이것은 번잡한 일이다.

사. 회선제어 절차에서 한개의 지령을 내리는 경우라도, 회선제어와 단말제어를 다 해야하므로 단말장치의 변경시는 회선제어의 지령부분 부터 새로 짜야만 한다.

2. SDLC의 기능과 특징

SDLC는 동기를 취하는 방법에서 보면 BSC와 같은 동기식 전송에 의한 것으로, 통신회선의 접속이 2점간이나, 분기접속, 루프(Loop)식 교환식 등의 어느경우에도 문제없이 적용될 수 있는 특징을 갖는다.

SDLC는 단말에 대하여 폴링에 의하여 호출한다. SDLC는, 따라서, 처음부터 주국(主局: 1차국이라고도 함)과 종국(從局: 2차국이라고도 함)의 관계로 분류되어 있다. 즉 단말이 접속하는 경우, 멀티 드롭(multi drop: 분기접속)이건, 포인트-투-포인트이건 회선제어의 책임을 1차국이 갖고 있으며, 따라서 1차국은 항상 2차국보다 고도의 판단능력(인텔리전스)을 보유하고 있다. 종래의 중앙제어방식과 틀린 점은 주·종관계가 〈그림14〉와 같이, 컴퓨터와 CCU, CCU와 단말간에 각각 명확히 규정되어 있는 것으로, 이 때문에 CPU의 회선제어 기능이 CCU에 기능분담 되었다고도 볼 수 있다. 그런데 이 경우 2차국에서는 직접 정보교환을 할 수 없다.



〈그림14〉 SDLC의 주국과 종국간의 관계.

앞에서도 언급한 바와 같이 종래의 회선제어절차에 있어서 응답을 주고 받을때의 형식은 지령, 응답, 데이터 전송시 각각 틀리기 때문에 표준화 되어 있다고 말할 수 없는 반면, SDLC에서는 다음과 같은 “프레임(frame)”이라고 불리는 표준화 된 정보형식으로 모든 전송을 행

한다. 즉 지령, 응답, 정보(메세지)가 전부 프레임 단위로 전송된다.

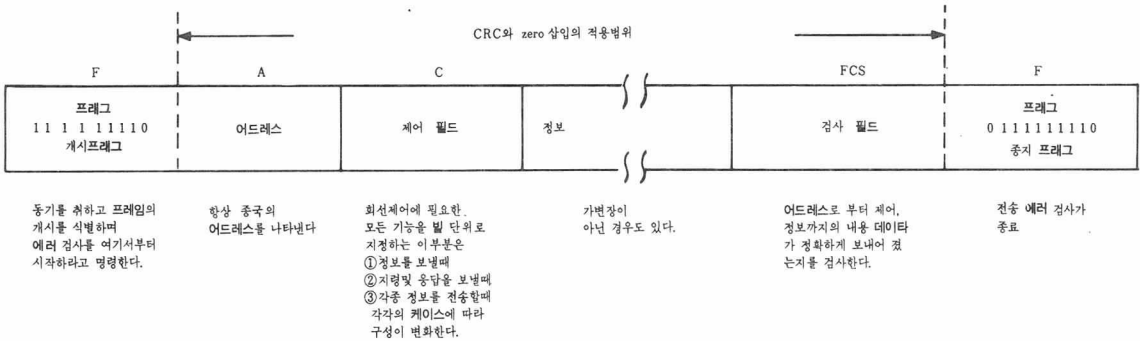
이 SDLC의 프레임은, <그림15>에 보이는 바와 같이 6개의 기능부(필드: field)로 구성된다.

가. 프레임 필드(flag field)

영어의 “깃발”을 의미하는 이 필드의 역할은, BSC의 SYN 코드와 같이 동기를 취하며, 프레임의 시작과 끝을 알리는 역할도 한다. 또한 개시 프래그에 의해 CRC 방식에 의한 프레임의 전송 에러 검사를 시작하며, 종료 프래그에 의해 프레임이 끝이라는 것과 에러 검사를 종료하라는 사실을 알게 된다.

그런데, 프래그를 구성하는 빌구성이 01111110이므로, 이와 같은 빌·패턴이 프래그 이외의 필드에서 나타날 가능성이 있어, 그 경우 중지 프래그를 검출한 것으로 오해하게 된다.

이 때문에, 「0빌삽입법」을 사용하여 오해가 없도록 한다. 즉 프래그·빌 01111110이 프래그 이외의 곳에서 나타나지 않도록 송신측에서 1이 연속하여 5회 나타나면 그 다음이 .1,0에 관계없이 0을 삽입한다. 수신측에서는 이것을 받아 프래그 이외의 곳에서 1이 연속 5회 나타나면, 그 다음 0를 제거시켜 원래의 데이터로 복원시키게 되는데, 이 0는 전송에러검사의 대상에는 포함되지 않는다.



<그림15> SDLC의 프레임 형식

나. 어드레스 필드

개시 프래그 다음에 오는 8빌 구성의 필드로, 항상 2차국의 어드레스를 나타낸다. 1차국에서 2차국으로 전송할 때의 어드레스·필드는 상대 단말국의 식별에 이용되지만, 역 방향의 경우는 2차국 자신의 어드레스를 지정한다. 다시 말해서, 2차국에서 1차국으로의 전송시는 “이 프레임을 보내는 사람은 바로 저입니다.”라고 알리는 것과 같다.

1차국은 프레임 마다 2차국을 지정해 주기 때문에, 2차국은 자기의 어드레스가 들어 있는 프레임을 수신할 때에만 1차국과 접속하면 되도록 되어 있다. 이것을 논 선택 홀드(non-select hold)라고도 부르는데, 종래와 같이 컴퓨터-단말간에 어떤 절차에 의해 몇 블럭씩이나 메세지를 전송할 경우, 계속적으로 접속되

어 있어야 했던 것이 필요없게 된다. 이 방법에 의해서 분기접속의 경우, 하나의 회선이 몇 블럭을 전송하는 시간 동안 하나의 특정 단말에게 점유당하는 일 없이, 동일 회선상의 모든 단말이 신속히 데이터를 주고 받을 수 있게 된다.

다. 제어 필드

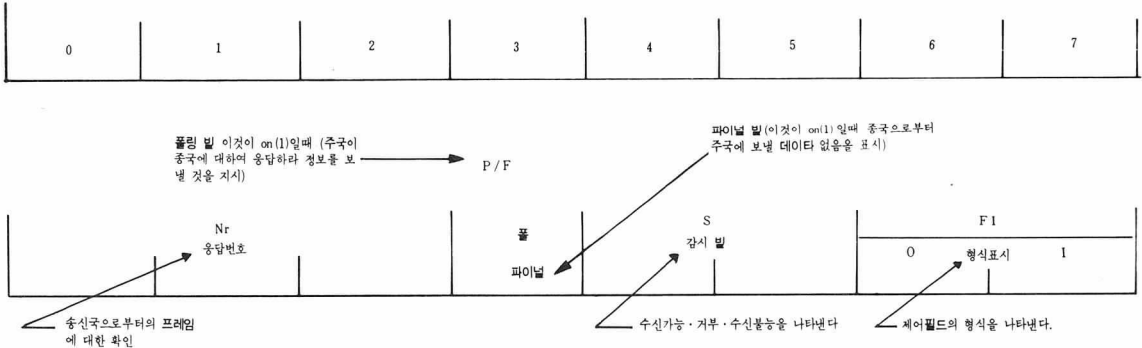
여기서는, 회선제어 기능(즉 호출이나 수신 가능한지, 불가능한지 등)을 전부 빌 단위로 지정해줄 뿐만 아니라 전 2중통신이 가능하게 하기 위해 전송프레임에 연속번호 표시를 해준다.

이 제어 필드에서의 회선제어 기능은 다음과 같이 3개로 크게 나뉘어 진다.

- 1) 데이터 링크를 만들어 주기 위한 제어지령과 응답정보를 보낼때 필요한 감시형식을 정해주는 제어.

- 2) 사용자측의 정보(컴퓨터 처리의 대상이 되는 것, 또는 그 처리결과로서의 정보)를 되보내는데 쓰이는 정보전송(伝送) 형식의 제어.
- 3) 프레임번호의 확인없이 제어지령이나 응

답, 또한 사용자 정보를 되보내는 무순서 형식의 제어.
 제어필드는 <그림16>과 같이 8빌드로 구성되는데 위와 같은 3종의 형식을 제어하기 위해 3 종류의 빌구성을 하고 있다.



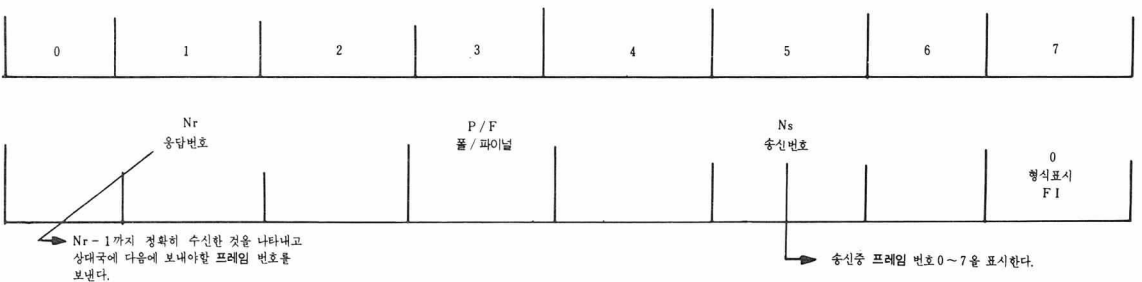
<그림16> 제어 필드의 형식

가) 감시형식의 제어필드

여기서는, P/F(폴/파이널)라고 불리는 빌드와, FI(형식표시: format identifier)라는 2개의 기능이 있다는 것에 주의해야 한다. 특히, 이 2기능은 3종의 제어필드에 모두 공통적으로 쓰인다. FI의 형식표시는 1빌드, 또는 2 빌드로 정의한다. 또 하나의 1빌드, P/F는 송신 또는 수신 제어에 사용되는 것으로, 프레임이 1차국에서 2차국으로 보내어질 때, ON(「1」)이면, 2차국으로부터의 응답이나 정보를 요구하는

뜻이 된다. 이 때문에 이것을 폴링 빌드라고 부른다. 한편, 2차국에서 1차국으로 보낼 때는 이것을 파이널 빌드라고 부르는데 이것이 ON(「1」) 되면 2차국이 송신권을 포기한다는 것을 의미한다. 다시 말해서 2차국이 보낼 데이터가 이제 없다라는 뜻이 된다.

1차국은 결국 중앙에서 폴링 빌드를 ON/OFF 하므로써 2차국과의 데이터 링크를 제어할 수 있다.



<그림17> 정보 전송 필드의 형식.

나) 정보전송형식의 제어 필드(그림17)

이 필드는 사용자정보를 되보낼때에 사용되지만 이러한 경우 송신번호를 표시하는 카운터(counter)와 응답번호를 표시하는 카운터의 2개의 역할을 함으로써 정보의 연속 전송과 전 2중 통신을 가능하게 하는 필드이다.

이 2개의 카운터중에서 Ns(Number Sending)은 송신중의 프레임번호 0~7을 표시한다. 한편 Nr(Numder of Receiving)은 상대측이 다음에 보내야 할 프레임번호를 표시한다. 이와 동시에 상대국이 Nr-1 프레임까지 정확하게 수신했다는 뜻도 포함하고 있다. 거기에서 송신국은 수신국으로 부터의 프레임 Nr 과 자국에서 카운트하고 있는 Ns를 체크하고 있어 그것이 일치하고 있으면 정확히 정보가 전송되었다는 것을 확인한다. 또 이 연속 번호의 표시에 의해서 다음과 같은 모양으로 상대측으로 부터의 응답없이 프레임의 연속전송이 가능하게 된다.

즉 컴퓨터로부터 어떤 개수까지의 프레임을 단말이 정확하게 받아들인 것으로 가정을 하고 이것을 상대측에 알리는 데에는 단말측에서 보내야 할 정보의 프레임의 응답번호를 카운트하고 있는 카운터에 다음에 보내야 할 프레임의 번호를 실어서 보내어 준다. 이것이 편송이라고 불리우고 있는 것으로 이러한 방법에 의해서 상호 필요할 때마다 데이터를 보낼 수 있는 전 2중통신이 가능하게 되는 것이다. 또 이 연속전송에 관해서는 다음에 보이는 바와 같이 따로 난을 만들어 그 구조를 보이고 있으므로 참고하기 바란다.

다) 무순서 형식의 제어필드

무순서 형식의 제어 필드는 주로 회선의 제어기능을 위해서 이용된다.

이 회선 제어기능은 장래의 새로운 기능에 대비해서 확장성이 풍부한 데이터 링크 제어를

할 수 있도록 수식(修飾) 빌을 준비하고 있고 이것을 자유로이 조합하여 사용하도록 되어있다.

비동기식 전송방식이나 BSC의 경우는 각 기능마다 회선제어 문자를 미리 정의하고 있는 데 비해 SDLC에 있어서의 회선제어는 완전히 빌 단위로 하고 있기 때문에 이러한 것이 가능하게 된다.

▶정보필드

이 정보필드는 데이터처리의 대상이 되는 사용자정보를 전송할 때에 사용되는 것으로 제어 지령이나 응답을 되보내는 감시 형식의 필드에는 포함되지 않는다. 정보전송형식 또는 무순서 형식의 제어필드를 갖는 프레임에만 존재하는 것이다. 정보 필드의 길이는 SDLC 방식에서는 아무런 제한도 주고 있지 않다. 그러나 전송정보를 받아 들일 수 있는 버퍼(Buffer)의 크기나 통신회선의 품질로부터 어느 정도의 제한을 받는다.

▶검사필드

정보필드 다음에 오는, 16빌로 구성되는, FCS(Frame Check Sequence) 필드라고도 불리는 것으로, 프레임의 내용이 정확히 전송되었는지 어떤지를 검사하기 위해서 사용된다. 프레임 내의 모든 2진 빌로 표현된 수치를 에러검사를 위하여 CRC 방식에 의해 산술적으로 계산한 값을 이 16빌로 표시한다. 송신국은 계산을 행해서 그 결과인 FCS 값을 보낸다. 수신국은 자국에서 같은 모양의 계산을 행하고 보내어져 온 값과 비교한다.

이 검사방법은, 어드레스로 부터 사용자 정보에 이르는, 프레임의 전체에 걸쳐서 적용되기 때문에, 이것에 의해서 비동기식이나 BSC 등 종래의 회선제어절차에서는 에러 검사의 대상이 되지 않았던 감시정보에 대해서도 완전한 에러 검사가 가능케 되어, 전송에 있어서 신뢰성이 한층 더 높아졌다고 할 수 있다.

연속 프레임 전송의 구조

연속 프레임은 프레임의 분실 또는 중복되는 것을 방지하기 위해서 연속 프레임 체크가 행해진다. 연속적인 프레임을 송신하고 있는 ㉠ 스테이션(Station)은 각각의 연속 프레임 수를 세어서 연속번호를 붙인다. 이 카운트를

Ns라고 부른다. 연속한 프레임을 받고 있는 ㉡ 스테이션에서는 수신한 각각의 정확한 연속 프레임을 카운트한다. 이 수신측의 카운트를 Nr이라고 부른다.

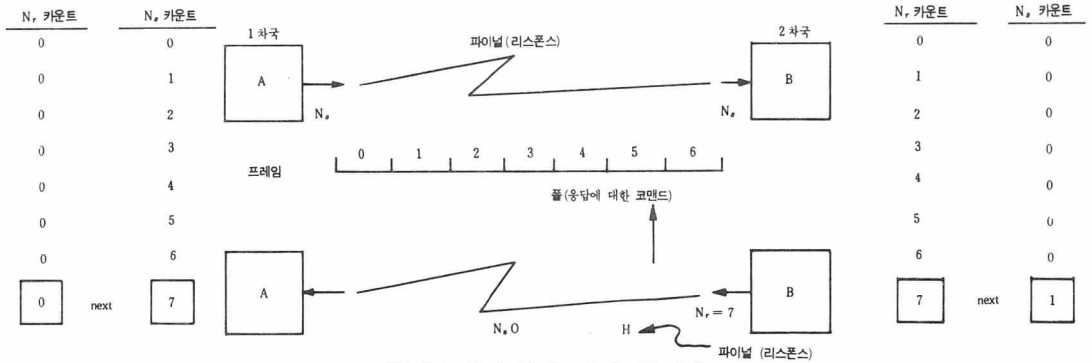
Nr의 카운트증가는 프레임이 체크되어 에러가 없다는 것을 확인한 경우 행하여진다. Nr은 또한 다음에 보내어져야 할 프레임의 카

운트가 된다. 그리고 이것은 다음에 들어올 N_s 카운트와 일치하지 않으면 안된다. 만약 들어올 N_s 가 N_r 과 일치하지 않은 경우 그 프레임은 연속하지 않은 것으로서 제거된다. 그래서 N_r 의 카운트는 진행되지 않는다. 제거된 틀린 프레임은 응용 소프트웨어(Software)에서 받아 들일 것인지 거부할 것인지가 결정된다.

N_r 내지 N_s 에 사용되고 있는 프레임에 번호를 붙이는 방법은 8개까지, 즉 0부터 7까지의 숫자를 사용한다. 7까지는 0으로 부터 연속적으로 계속해서 가고 또 7의 다음은 0이 되어 다시 7까지 계속하여 올라 간다. 만일 전송해야 할 데이터 프레임이 많이 있을 때에는 받는측이 그 N_r 카운트를 보내어진 각 프레임에 대하여 하나하나 송신측으로 보고하지

않고 최고 7개의 프레임을 보낼 수가 있다. 즉 데이터 링크 레벨의 데이터와 리스폰스(response)는 1대1로 대응하지 않아도 좋다. 단 수신측으로 부터 N_r 이 되돌아올 때까지 송신측으로 부터 나온 모든 프레임은 송신측에 의해서 바퍼내에 축적해 놓지 않으면 안된다. 그 이유는 최고 7개까지의 프레임은 전송중에 에러가 있는 경우 그들의 일부 내지는 전부를 재전송하기 위해서 필요하게 되기 때문이다.

양쪽 스테이션의 N_r 및 $N_s=0$ 카운트는 주 스테이션의 판단으로 0으로 이니셜라이즈(initialize) 된다. 다른 한편 카운트는 연속번호가 보내어 질때 혹은 받을 때 마다 증가한다.



〈표 3〉 주요설비 제작 공정표

상기한 바와 같이 7개의 프레임 0~6을 B로부터의 데이터 링크 응답없이 연속하여 A로부터 B로 보낸것으로 가정하자. 만일 에러가 없다면 B스테이션에서는 위 그림과 같이 N_r 의 값을 1로부터 7까지 각 프레임이 수신할 때 마다 갱신하여 간다. A에서 모든 프레임(7개의 프레임)을 다 보냈을때에 B로부터 보내

어진 N_r (상기의 B의 N_r 에 상당)의 값에 의해 A로부터의 전송 (그렇지 않으면 재전송)의 방법은 다르다. 그것을 표시한 것이 다음의 것이다. A에는 B로 보낼 프레임이 「0.1.2.3.4.5.6.7.0'. 1'2'.3'.4'.5'.6'.7'.0'1'...'」라고 하는 순서로 대기하고 있어 그중 0~6까지 B로 보내어진 것으로 가정한다.

B가 응답할때의 N_r 의 값이	스테이션 A는 리퀘스트에 아래의 프레임을 보낸다.
7 (모든 프레임 체크 OK)	7 0' 1' 2' 3' 4' 5' (모든 프레임은 OK 였기 때문에 A에 대기하고 있는 새로운 프레임 7~5'를 보낸다)
6 (프레임6만 버림. 에러 때문)...	6 7 0' 1' 2' 3' 4' (프레임 6만 재전송하고 7~4' 프레임 송신을 계속)
5 (프레임5에 에러, 5와6을 버림)...	5 6 7 0' 1' 2' 3' (프레임 5,6을 재전송하고 7~3' 프레임의 송신을 계속)
4 (프레임4에 에러, 4~6을 버림)...	4 5 6 7 0' 1' 2' (프레임 4,5,6을 재전송하고 7~2' 프레임의 송신을 계속)
3 (프레임3에 에러, 3~6을 버림)...	3 4 5 6 7 0' 1' (프레임 3~6 재전송하고 7 0' 1' 프레임의 송신을 계속)
2 (프레임2에 에러, 2~6을 버림)...	2 3 4 5 6 7 0' (프레임 2~6을 재전송하고 7. 0'의 송신을 계속)
1 (프레임1에 에러, 1~6을 버림)...	1 2 3 4 5 6 7 (프레임 1~6을 재전송하고 7 프레임의 송신을 계속)
0 (프레임 0, 전 프레임을 받지못함)	0 1 2 3 4 5 6 (전 프레임을 재전송)

3. SDLC의 개량점

지금까지 보아온 바와 같이 SDLC는 비동기식이나 BSC와 비교하여 여러개의 개량된 기능을 갖고 있다. 예를 들면 어드레스 필드를 설정함으로써 논 선택 홀드를 가능케 하고 있고 분기접속에서 1단말에 회선을 점유당하는 일이 없도록 했다. 또 제어필드에 있어서 전송 프레임에 연속번호가 설정되는 것에 의해 전 2중통신이 가능하게 되었다.

더욱이 전체적으로 빌 오리엔티드 한 프레임구성으로 표준화된 형식으로 전송을 행하고 있기 때문에 기능의 수정등이 용이하게 되고, 비동기식이나 BSC와 달리, 제어정보 까지도 포함한 프레임 전체가 에러 체크의 대상이 되기 때문에 신뢰성이 향상되는 등, 종래의 회선제어방식에서 문제가 된 여러가지 과제를 해결하게 되었다. 이와같은 SDLC의 장점을 정리하면 다음과 같이 될것이다.

① 모든 전송이 프레임이라고 하는 표준화된 형식으로 행하여 진다.—이것에 의해 설계도입이 간결하게 되어 예외적인 변경이 일어나도 복잡하게 되지 않는다.

② 회선제어의 기능이 완전분리되어 있어 단말의 변경등 시스템을 고쳐야 하는 경우에도 쉽게 고칠 수 있다.—어드레스나 제어, 정보, 검사의 기능부가 각각 독립되어 있고 이것에 의해

서 혼재(混在)의 문제가 해결된다. 이렇게 하여 어드레스, 지령, 응답이라고 하는 회선의 감시기능이 사용자와는 완전히 독립한 특징의 필드에 빌 레벨에서 행하여지기 때문에 융통성과 확장성이 풍부한 데이터 링크제어가 가능하게 되었다.

③ SDLC에 의한 회선제어의 표준화를 꾀하는 것에 의해 동일회선상에 있어서 다른 기능, 이종(異種) 단말의 분기접속이 가능하게 되었다.

④ 사용자정보, 감시기능 모두 빌 지향형이기 때문에 빌 구성의 제약등이 없다.

⑤ 전 2중, 반 2중통신과 논 선택 홀드가 가능케 되었다.—프레임마다 이차국의 어드레스, 프레임번호가 지정되기 때문에 연속전송, 일괄응답이 가능하게 되어 전 2중통신을 원활히 행할 수가 있다. 또한 분기접속에 있어서 동일회선상의 다른 단말과 동시에 데이터전송을 실행할 수가 있다. 즉 논 선택 홀드방식이 취해져 있다.

⑥ 신뢰성이 향상하였다.—이미 보아온 바와 같이 종래의 전송 에러체크가 데이터부만을 대상으로 행하여 진것에 비하여 회선의 제어정보도 포함한 프레임내의 모든 빌이 CRC방식에 의해서 에러체크 된다. 더욱이 이미 기술한 송신, 수신번호의 체크에 의해서 정보의 탈락 및 중복을 감시할 수가 있기 때문에 정보의 전송이 일층 확실하게 되었다.

(계속)

