

Rysunek 1: (a) – komunikacja wirtualna, (b) – komunikacja rzeczywista

Część I
Warstwa łącza danych

Warstwa łącza danych realizuje wiele ściśle określonych funkcji. Zalicza się tu obsługę interfejsu do warstwy sieci, określanie sposobu grupowania bitów warstwy fizycznej w ramki, korygowanie błędów transmisji, sterowanie przepływem ramek, itp.

Część II
Usługi świadczone warstwie sieci

Podstawową usługą warstwy łącza danych jest przekazywanie danych z warstwy sieci komputera źródłowego do warstwy sieci komputera docelowego. W komputerze źródłowym znajduje się tzw. jednostka (entity) zwana inaczej procesem, czyli wykonywany na tym komputerze program, wymagający przesłania pewnej liczby bitów danych do komputera docelowego. Ta porcja danych przekazywana jest z warstwy sieci do warstwy łącza danych zgodnie z protokołem warstwy łącza danych, co pokazano na rys. 1 (a). W rzeczywistości bity z warstwy sieci komputera źródłowego docierają do warstwy sieci komputera docelowego zgodnie z rys. 1 (b), ale łatwiej jest rozpatrywać tę transmisję z punktu widzenia protokołu warstwy łącza danych, czyli zgodnie z rys. 1 (a).

Warstwa łącza danych najczęściej świadczy warstwie sieci usługi następującego typu:

- niepotwierdzona usługa bezpołączeniowa,
- potwierdzona usługa bezpołączeniowa,
- usługa zorientowana połączeniowo.

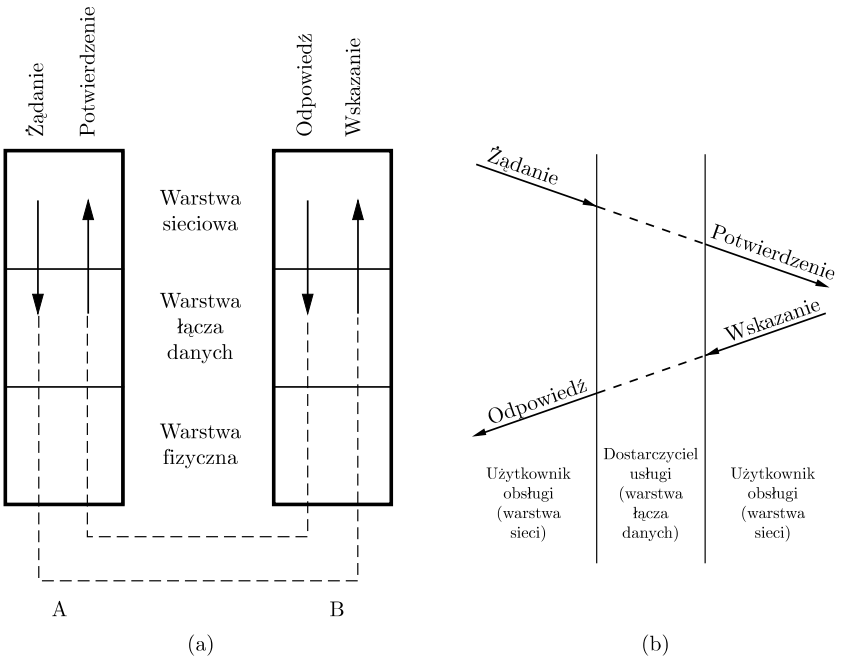
Niepotwierdzona usługa bezpołączeniowa polega na przesyłaniu niezależnych ramek przez komputer źródłowy do komputera docelowego, bez konieczności potwierdzania odbioru tych ramek przez komputer docelowy. Jeśli nastąpi utrata ramki na skutek zakłóceń w linii, warstwa łącza danych nie podejmuje żadnej akcji, by tę ramkę odzyskać. Usługa tego typu ma miejsce wówczas, gdy zakłócenia w kanale transmisyjnym są bardzo małe i gdy zadanie odzyskiwania utraconych z powodu błędów transmisji ramek spoczywa na warstwie wyższej. Jest to też usługa typowa dla ruchu ramek w czasie rzeczywistym, np. podczas rozmowy telefonicznej, przy której dokładne dane przesłane później są gorsze od błędnych danych aktualnych. Takie usługi są świadczone przez warstwy łącza danych wielu lokalnych sieci komputerowych.

Podczas świadczenia potwierdzanej usługi bezpołączeniowej nie używa się stałego połączenia między komputerami źródłowym i docelowym, ale poprawny odbiór każdej wysłanej ramki jest indywidualnie potwierdzany. W tej sytuacji nadawca wie, jakie ramki dotarły do adresata, i jeśli potwierdzenie ramki nie dotrze do nadawcy w odpowiednim przedziale czasu, ramka zostaje wysłana ponownie.

Najbardziej skomplikowaną usługą warstwy łącza danych jest usługa zorientowana połączeniowo. Przy tej usłudze komputery źródłowy i docelowy ustalają połączenie przed rozpoczęciem przekazywania danych. Wszystkie ramki, wysłane zestawionym łączem są numerowane, a warstwa łącza danych gwarantuje, że każda wysłana ramka jest rzeczywiście odebrana. Warstwa łącza danych zapewnia również w przypadku tej usługi, że każda wysłana ramka jest odebrana dokładnie jeden raz i że wszystkie ramki będą odebrane we właściwym porządku. Taki warunek jest nie do pomyślenia w usługach bezpołączeniowych, gdzie utrata potwierdzenia powoduje kilkakrotne wysyłanie i odbiór tych samych ramek. Świadczeniu usługi zorientowanej połączeniowo towarzyszą trzy fazy. W pierwszej fazie następuje ustalanie połączenia, podczas której nadawca i odbiorca ramek inicjalizują zmienne i liczniki, używane do śledzenia procesu odbioru ramek. Druga faza odpowiada właściwemu procesowi transmisji jednej lub więcej ramek, a w fazie trzeciej następuje zwolnienie połączenia, odświeżenie zmiennych, zwolnienie buforów i innych zasobów, potrzebnych dla realizacji usługi.

Do komunikacji między warstwą sieci a warstwą łącza danych stosuje się znormalizowane prymitywy usług OSI, czyli prymitywy żądania, wskazania, odpowiedzi i potwierdzenia. Prymitywów żądania używa warstwa sieci w celu poproszenia warstwy łącza danych, by wykonała jakąś czynność, np. ustalenie lub zwolnienie połączenia, czy wysłanie ramki. Prymitywy wskazania służą do poinformowania warstwy sieci, że nastąpiło pewne zdarzenie, np. ramka dotarła do celu czy też jakiś inny komputer życzy sobie ustalenia lub zwolnienia połączenia. Z kolei prymitywy odpowiedzi używane są po stronie odbiorczej warstwy sieci do udzielenia odpowiedzi na otrzymane poprzednio wskazanie. Wreszcie prymitywy potwierdzenia pozwalają stronie żądającej

warstwy łącza danych zdobyć informację, czy żądanie zostało właściwie zaspokojone, a jeśli nie, to dlaczego.



Rysunek 2: Dwa różne spsosoby reprezentacji prymitywów usług

Na rys. 2 pokazano dwa różne sposoby reprezentacji prymitywów usług. Rys. 2 (a) pokazuje strukturę warstwową oraz rzeczywisty przepływ danych. Na rys. 2 (b), na lewo od dwóch linii pionowych znajduje się użytkownik usługi łącza danych po stronie nadawczej. Między liniami pionowymi usytuowany jest dostawca usługi, czyli warstwa łącza danych, a po stronie prawej od linii pionowych zlokalizowano użytkownika usługi łącza danych po stronie odbiorczej. Czas biegnie od dołu do góry, więc zdarzenia na górze rysunku miały miejsce wcześniej, niż na dole. Tego typu rysunków używa się przy opisie sieci komputerowych dla pokazania użytkowników i dostarczycieli usług w różnych warstwach sieci.

Część III
Tworzenie ramek

Aby warstwa łącza danych mogła dostarczać usługi warstwie sieci musi korzystać z usług, oferowanych przez warstwę fizyczną. Jak wiadomo, warstwa fizyczna jedynie przyjmuje strumień bitów, który usiłuje dostarczyć do miejsca przeznaczenia, czyli warstwie fizycznej po stronie odbiorczej. War-

stwa fizyczna nie ma więc możliwości zagwarantowania, że transmitowany za jej pomocą strumień bitów nie zostanie zniekształcony przez zakłócenia w kanale, czyli że dotrze do adresata w postaci bezbłędnej. Tak więc odebrany strumień bitów może zawierać mniejszą, równą lub większą liczbę bitów niż strumień nadany, a wartości bitów odebranych mogą być inne, niż wartości bitów nadanych. Zatem zadaniem warstwy łącza danych jest wykrycie, i w razie potrzeby skorygowanie błędów, powodowanych przez zakłócenia w kanale transmisyjnym. W tym celu warstwa łącza danych dzieli strumień bitów na ramki o skończonej liczbie bitów, i dla każdej ramki oblicza bity sumy kontrolnej, dołączanej do wszystkich ramek. Po dotarciu ramki do miejsca przeznaczenia suma kontrolna jest odtwarzana, zgodnie z algorytmem, stosowanym po stronie nadawczej. Jeśli odtworzona suma kontrolna różni się od sumy kontrolnej, dołączonej do ramki, zostaje wykryty błąd transmisji i warstwa łącza danych zwykle taką ramkę odrzuca, powiadamiając o tym fakcie nadawcę.

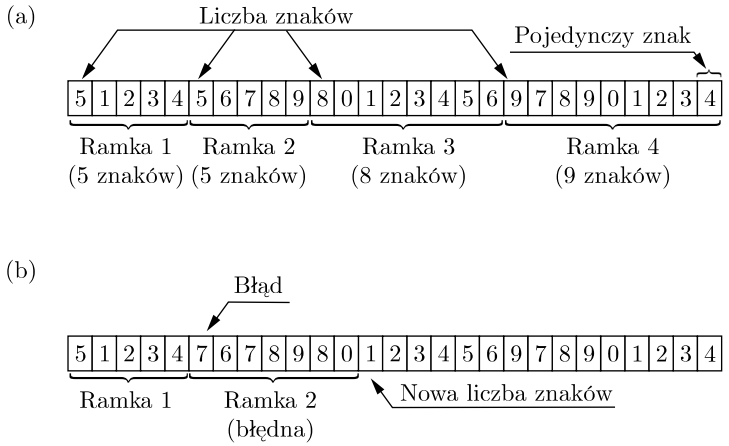
Zasady podziału strumienia bitów na ramki są dość skomplikowane. Na ogół stosuje się przy tym cztery metody:

1. zliczanie znaków,
2. stosowanie specjalnych znaków dla oznaczenia początku i końca ramki,
3. stosowanie specjalnych flag na początku i końcu ramki,
4. stosowanie specjalnych sygnałow na początku i końcu ramki.

Metoda zliczania znaków polega na umieszczeniu na początku każdej ramki jednego znaku, określającego liczbę znaków w ramce. Kiedy warstwa łącza danych po stronie odbiorczej dostrzeże znak początku ramki, wówczas wie, ile znaków będzie w ramce, czyli łatwo określi koniec ramki. Przykład tej techniki dla 4 ramek, zawierających odpowiednio 5, 5, 8 i 9 znaków pokazano na rys. 3. Przy bezbłędnym odbiorze (rys. 3 (a)) wszystko jest w porządku. Jeśli jednak w czasie transmisji zostanie zmieniona wartość znaku, niosącego informację o liczbie znaków w ramce, po stronie odbiorczej proces dzielenia strumienia znaków zostanie zaburzony w taki sposób, że nie wiadomo nawet, gdzie błąd wystąpił (rys. 3 (b)), co uniemożliwia żądanie retransmisji błędnie odebranych ramek. Z tego powodu ta metoda ma dziś tylko znaczenie historyczne i praktycznie nie jest stosowana.

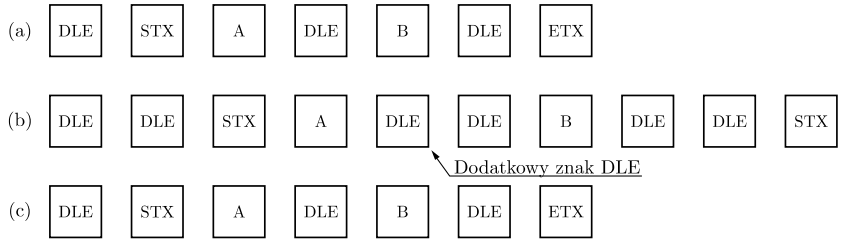
Druga metoda polega na oznaczeniu początku ramki dwoma znakami kodu ASCII: DLE STX (DLE - Data Link Escape, STX - Start of TeXt) i końca ramki znakami DLE ETX (ETX - End of TeXt). Po stronie odbiorczej obecność tych dwóch znaków określa wyraźnie granicę ramki. Jeśli granice pewnych ramek na skutek błędów zostaną zniekształcone, stacja je pomija i poszukuje ramek właściwie ograniczonych.

Ten system pracuje dobrze przy przesyłaniu danych tekstowych. Przesyłanie danych binarnych, np. skompilowanych programów, powoduje pewne



Rysunek 3: Technika zliczania znaków. (a) – ramki odebrane bezbłędnie, (b) – ramki odebrane z jednym błędem

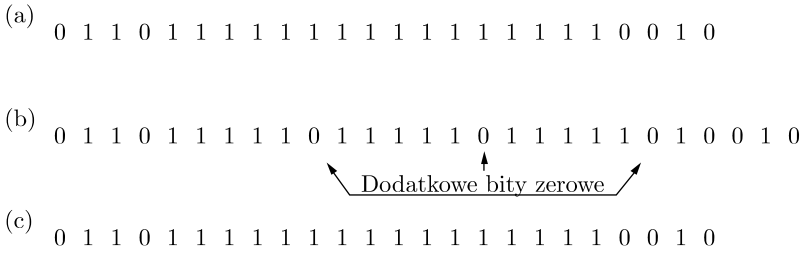
komplikacje, ponieważ w treści ramki mogą pojawić się wówczas konfiguracje bitów, odpowiadające kombinacji znaków DLE STX i DLE ETX. Problem ten pokonuje się w taki sposób, że warstwa łącza danych po stronie nadawczej wstawia dodatkowy znak DLE przed każdym takim znakiem, zawartym w ramce, a po stronie odbiorczej, po wykryciu dwóch znaków DLE, jeden z nich usuwa. Zasadę tej techniki, zwanej też techniką wstawiania znaku (byte stuffing), pokazano na rys. 4.



Rysunek 4: Technika oznaczania początku i końca ramki za pomocą pary znaków sterujących kodu ASCII. (a) – dane warstwy sieci, przeznaczone do nadawania, (b) – te same dane po wstawieniu dodatkowego znaku po stronie nadawczej warstwy łącza danych , (c) – dane przekazane warstwie sieci po stronie odbiorczej

Niedogodność tej metody polega na tym, że jest ona dostosowana do przesyłania danych w 8-bitowym kodzie ASCII, przez co liczba bitów danych jest zawsze wielokrotnością liczby 8. Opracowano więc nową technikę, która nie nakłada ograniczeń na liczbę bitów w ramce (w przewidzianych

granicach) i pozwala na stosowanie dowolnych kodów dla reprezentacji danych. W tej technice każda ramka zaczyna się i kończy kombinacją bitów w postaci 01111110. Kombinacja ta nazywa się flagą. Jeśli nadajnik w warstwie łącza danych zauważy w treści danych obecność 5 kolejnych jedynek, wstawia po nich dodatkowy bit zera. Jest to tzw. technika wstawiania bitu (bit stuffing). Aby stronie odbiorczej warstwy sieci zostały przekazane dane prawidłowe, odbiornik warstwy łącza danych usuwa ze strumienia bitów każde zero, występujące po 5 jedynkach. Tak więc jeśli dane użytkownika zawierają konfigurację bitów w postaci flagi, czyli sekwencję 01111110, jest ona w nadajniku warstwy łącza danych zamieniana na kombinację 011111010, ale w buforze odbiornika warstwy łącza danych ta kombinacja ma postać 01111110 (rys. 5).



Rysunek 5: Technika wstawiania dodatkowego bitu do strumienia bitów danych przy oznaczaniu początku i końca ramki za pomocą flagi. (a) – dane do wysyłania z warstwy sieci, (b) – dane, przekazane przez stronę nadawczą warstwy łącza danych, warstwie fizycznej, (c) – dane, przekazane warstwie sieci po stronie odbiorczej

Ostatnia metoda tworzenia ramek, wykluczająca wstawianie znaku czy też wstawianie bitu, dotyczy takich sposobów kodowania, które pozwalają przedstawić za pomocą sygnału symbole 0, 1 oraz inne jeszcze symbole, stosowane wyłącznie do oznaczania początku i końca ramki. Symbole takie nigdy nie występują w strumieniu danych. Jest to sytuacja typowa dla kodowania stanów logicznych metodą Manchester.

Aktualnie, dla zwiększenia niezawodności używa się kombinacji pierwszej metody z dwiema pozostałymi. Przy odbiorze bada się zawartość pola, zawierającego liczbę bitów w ramce, wyznacza się tym sposobem koniec ramki. Jeśli ogranicznik ramki znajduje się we właściwym miejscu i jeśli suma kontrolna jest poprawna, wówczas ramka uważana jest za ramkę ważną. W przeciwnym przypadku odbiornik ramkę odrzuca i poszukuje następnego ogranicznika początku ramki.

Część IV

Zapobieganie błędom utraty lub powtórzenia ramki

Po rozwiązaniu zagadnienia oznaczania początku i końca ramki powstał problem upewniania się, czy wszystkie ramki we właściwej kolejności dotarły do warstwy sieci po stronie odbiorczej. Rozwiązanie tej kwestii jest istotne dla zapewnienia niezawodnych usług bezpołączeniowych.

Niezawodne dostarczanie ramek do miejsca przeznaczenia wymaga zwykle zastosowania kanału zwrotnego, od odbiornika do nadajnika, aby nadajnik wiedział, co dzieje się z ramkami, które wysyła. Zwykle protokół wymaga, by odbiornik przesyłał zwrotnie do nadajnika specjalne ramki sterujące, zawierające informacje o potwierdzeniu lub odrzuceniu nadchodzących ramek. Jeśli nadajnik nie otrzyma potwierdzenia poprawnego odbioru ramki, wówczas musi ją wysłać ponownie. Dla zapewnienia niezawodnej transmisji, nadajnik musi otrzymać od odbiornika pozytywne lub negatywne potwierdzenie odbioru ramki, ponieważ możliwa jest taka awaria sprzętu, że ramka w ogóle nie pojawi się w warstwie fizycznej. Aby tę koncepcję zrealizować, warstwa łącza danych musi posiadać specjalny licznik czasu, uruchamiany równocześnie z rozpoczęciem transmisji ramki. Licznik ten wyłącza się po przekroczeniu czasu, przewidzianego na dotarcie ramki do miejsca przeznaczenia. Jeśli ramka zostanie odebrana poprawnie, potwierdzenie o jej odbiorze dotrze do nadajnika przed wyłączeniem tego licznika. Mimo to, ramka może dotrzeć poprawnie do odbiornika, ale w kanale transmisyjnym może zaginąć potwierdzenie jej odbioru. W takiej sytuacji następuje na ogół retransmisja ramki i pojawia się niebezpieczeństwo, że odbiornik będzie otrzymywał jedną ramkę dwa lub więcej razy, przekazując tę ramkę warstwie sieci więcej niż jeden raz. Aby zapobiec takim zdarzeniom, transmitowanym ramkom przydziela się numery kolejne, i wtedy odbiornik potrafi odróżnić duplikaty ramek od oryginału.

Część V

Sterowanie przepływem ramek

Ważnym zagadnieniem projektowania warstwy łącza danych (występującym również w wyższych warstwach) jest zapobieganie transmisjom ramek z szybkością, przekraczającą możliwości ich odbioru. Sytuacja taka może się łatwo zdarzyć, gdy nadajnik jest realizowany za pomocą szybszego komputera niż odbiornik. Wtedy nawet przy bezbłędnej transmisji odbiornik nie będzie w stanie obsługiwać przychodzących ramek i w pewnym momencie zacznie ramki gubić. Jest oczywiste, że należy temu zapobiegać, stosując sterowanie

przepływem ramek, polegające na dopuszczaniu takiej szybkości transmisji ramek, która pozwala odbiornikowi bezpiecznie je obsłużyć. Wymaga to zwykle stosowania kanału zwrotnego, za pomocą którego odbiornik ostrzega nadajnik, jeśli zaczyna nie nadążać z obsługą przychodzących ramek.

Istnieją różne metody sterowania przepływem, z których większość oparta jest na tej samej zasadzie: protokół zawiera dobrze zdefiniowane reguły, pozwalające nadajnikowi przesłać następną ramkę. Na ogół reguły te zakazują nadawać ramkę bez wyraźnego lub domyślnego zezwolenia odbiornika na transmisję ramki.

Część VI
Sterowanie łączem

Do funkcji warstwy łącza danych należy też zarządzanie łączem. Przy usłudze bezpołączeniowej czynność ta jest minimalna, ale usługi zorientowane połączeniowo wymagają w tym względzie wykonania wielu czynności, jak nawiązanie i zwolnienie połączenia, inicjalizacja numerów sekwencyjnych ramek i ich reinicjalizacja w przypadku błędów, itp. Należy też sterować konfigurowaniem łącza. W najprostszym przypadku warstwę fizyczną tworzą przewody, łączące ze sobą dwa komputery i często kilka maszyn współużytkuje ten sam kanał. Tradycyjnie jeden z komputerów pełni rolę użytkownika pierwszej kategorii (primary), bardziej uprzywilejowanego od użytkowników drugiej kategorii (secondary), jakimi są terminale. Zarządzanie ruchem ramek należy do uprzywilejowanego użytkownika kanału, który wysyła kolejno do terminali zaproszenie do nadawania. Jeśli terminal posiada ramki do wysłania, to je wysyła, a jeśli nie to następny terminal otrzymuje zaproszenie do nadawania (jest to tzw. polling). W innych systemach terminale mogą wysyłać dane do komputera bez specjalnego zaproszenia, a w sieciach lokalnych wszystkie stacje mają takie same uprawnienia. Zawsze jednak nad tym procesem powinna czuwać warstwa łącza danych.

1 Detekcja i korekcja błędów transmisyjnych

Błędy transmisji są aktualnie jednym z najistotniejszych problemów sieci komputerowych, a jego definitywnego rozwiązania nie przewiduje się nawet w ciągu najbliższej dekady.

W sieciach telefonicznych błędy transmisji powodowane są przez wiele zjawisk fizycznych. Jedno z nich, zwane szumem termicznym, spotyka się zawsze. Polega ono na szybkim i chaotycznym ruchu elektronów, poruszających się w przewodach we wszystkich kierunkach. Zjawisko to jest źródłem szerokopasmowego szumu, który jest jedną z przyczyn powstawania błędów.

Poważnym źródłem zakłóceń w łączu transmisji danych jest szum impulsowy. Statystyczna szerokość impulsu zakłócenia wynosi tu około 10 ms, co przy szybkości transmisji 9600 bitś potrafi zniweczyć 96 bitów. Źródłem impulsów zakłócających są przekaźniki i inne urządzenia elektromechaniczne, pracujące w centralach telefonicznych starszego typu. Inne źródła zakłóceń impulsowych to układy zapłonowe samochodów, sieć energetyczna oraz kanały sygnalizacyjno-kontrolne sieci telekomunikacyjnej.

Poważnym źródłem błędów jest zależność od częstotliwości szybkości propagacji, amplitudy i fazy sygnału, przesyłanego kanałem telefonicznym. Zwykle można od przedsiębiorstwa telekomunikacyjnego wydzierżawić specjalne linie, w których te niekorzystne efekty są minimalizowane, ale opłaty eksploatacyjne za takie linie są o wiele wyższe od opłat za zwykłe linie telefoniczne.

Istnieje jeszcze wiele innych źródeł błędów, do których zalicza się nieliniowość wzmaniaczy, przesłuch i zjawisko echa, związane z eksploatacją tzw. tłumików echa. W łączach mikrofalowych występują zaniki, a warunki transmisji zależą bardzo od pogody.

W łączach telefonii cyfrowej błędy powstają w każdym przypadku rozsynchronizowania nadajnika i odbiornika. Brak synchronizacji trwa zwykle kilkadziesiąt mikrosekund, i wtedy dane nie trafiają do właściwego adresata, ponieważ bity, pojawiające się w n-tej szczelinie czasowej trafiają do n-tego użytkownika, niezależnie od tego, czy synchronizacja działa, czy nie.

Poważnym źródłem błędów są stosowane w modemach tzw. skramblery, których zadaniem jest zamiana długich ciągów zer i jedynek, występujących w strumieniu bitów danych, na pseudolosowy ciąg bitów, nie posiadający składowej stałej. W ten sposób modem wytwarza sygnał dobrze dopasowany do charakterystyki częstotliwościowej kanału telefonicznego, ale błędna decyzja demodulatora, który niewłaściwie zdemoduluje tylko jeden element sygnału, wytworzy w strumieniu bitów odebranych, ciąg bitów błędnych, o długości 20 i więcej bitów.

Ogólnie rzecz ujmując, procesy fizyczne będące źródłem szumów, wytwarzają błędy grupowe, obejmujące pewną porcję bitów sąsiadujących ze sobą, tzn. wyizolowane błędy pojedyncze nie występują często. Zjawisko to jest korzystne, ponieważ dane komputerowe przesyła się przeważnie w blokach o pewnej długości. Przyjmując, że długość bloku transmisyjnego wynosi 1000 bitów, że elementowa stopa błędów wynosi 10^{-3} i że w kanale pojawiają się błędy grupowe, obejmujące 100 sąsiednich bitów, tylko jeden lub dwa bloki na 100 bloków przesłanych będą obarczone błędem. Jeśli błędy byłyby statystycznie niezależne i występowałyby pojedynczo, wówczas przy powyższych założeniach większość odebranych bloków zawierałaby błędy.

Jeśli średnia stopa błędów wynosi e błędów na znak, a blok zawiera n znaków, to przy założeniu statystycznej niezależności błędów prawdopodobieństwo poprawnej transmisji bloku będzie równe $(1-e)^n$. Prawdopodobieństwo błędnego odebrania bloku wyniesie wtedy $1-(1-e)^n$, co przy założeniu, że $e \ll 1$ daje w przybliżeniu en. Jednak z danych doświadczalnych wyni-

ka, że blokowa stopa błędów wynosi w przybliżeniu $10^{-4} \cdot n^{0,8}$ przy blokach, zawierających 8-bitowe bajty. Stopa ta zależy od długości linii, szybkości transmisji oraz wielu innych czynników, dlatego należy ją traktować jako oszacowanie bardzo zgrubne.

Metoda zwalczania błędów transmisji, przyjęta w sieciach komputerowych, opiera się na stosowaniu nadmiaru informacyjnego. Stosuje się przy tym dwa podejścia. Pierwsze polega na wyposażeniu każdego przesyłanego bloku danych w informację nadmiarową (którą nazywa się redundancją), uzależnioną w taki sposób od danych, że odbiornik potrafi wydedukować, w którym miejscu błędy transmisji powstały i jak je skorygować. Drugi sposób zakłada, że informacja nadmiarowa wystarcza jedynie do stwierdzenia faktu zaistnienia błędów transmisji. Tak więc, w praktyce można stosować albo bezpośrednią korekcję błędów transmisji, albo tylko detekcję błędów, które koryguje się przez żądanie retransmisji odebranej z błędami ramki.

Problemy korygowania błędów pojawiły się najpierw pod koniec lat 40., kiedy zaczęto konstruować pierwsze komputery, których pamięci były bardzo zawodne. Powstał wtedy początek teorii kodów korygujących błędy, która z biegiem lat bardzo się rozwinęła i dopiero w ostatnich dwóch dekadach znalazła bardzo wiele zastosowań. Gdyby tej teorii nie rozwinięto, nie było by dziś takich urządzeń, jak pamięci taśmowe i dyskowe, pamięci typu CD ROM, odtwarzacze płyt kompaktowych, czy cyfrowe kamery video.

Podstawowym problemem teorii kodów jest optymalizacja nadmiaru informacyjnego. Chodzi bowiem o to, by kod korekcyjny korygował lub wykrywał jak największą liczbę błędów, przy jak najmniejszym nadmiarze. Problem ten rozwiązuje się przy zastosowaniu dość skomplikowanego aparatu matematycznego, którego opanowanie wymaga dłuższego czasu. Można jednak stosunkowo prosto wyjaśnić zasadę konstruowania takich kodów.

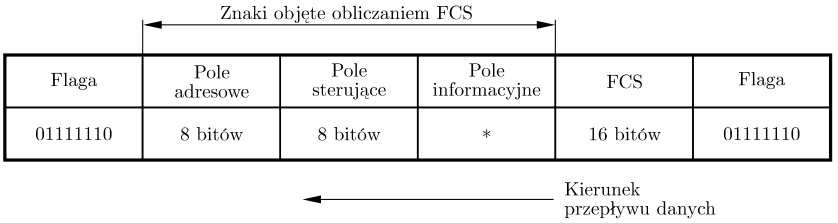
Ramka składa się na ogół z m bitów danych i r bitów nadmiarowych, zwanych też bitami próbnymi lub sumą kontrolną. Tak więc całkowita długość ramki wynosi $n = m + r$ bitów. Taką ramkę nazywa się n-bitowym słowem kodowym. Bity informacyjne w liczbie m mogą mieć dowolną konfigurację, czyli kod korekcyjny o długości n będzie się składał z 2^m słów kodowych. Niech np. $n = 10$ i $m = 2$. W tym przypadku kod będzie posiadał cztery słowa kodowe:

```
0000000000 0111110000 1000001111 1111111111
```

Na początku słowa kodowego znajdują się dwa bity informacji, po których umieszcza się 8 bitów nadmiarowych. Bity nadmiarowe dla tych czterech słów kodowych można wybrać spośród $2^8 = 256$ kombinacji. Należy je dobierać w taki sposób, by dwa słowa kodowe różniły się między sobą jak najbardziej. Miarą tej różnicy jest tzw. odległość Hamminga, czyli liczba miejsc, na których bity słów kodowych mają różne wartości. Łatwo zauważyć, że odległość Hamminga między dwoma słowami kodowymi w rozpatrywanym kodzie wynosi 5. Od tej wartości zależy zdolność detekcyjna i korekcyjna kodu. Aby kod mógł wykryć d błędów, minimalna odległość Hamminga między jego słowami kodowymi powinna wynosić $d+1$, zaś dla korekcji d błędów wymagana jest minimalna odległość $2d+1$. Zatem rozpatrywany kod o 4 słowach pozwala na korekcję dwóch błędów. Algorytm korekcji słowa kodowego z błędami transmisji polega na znalezieniu takiego słowa kodowego, którego odległość Hamminga od słowa odebranego jest najmniejsza i to właśnie słowo kodowe uważane jest za skorygowane odebrane słowo kodowe.

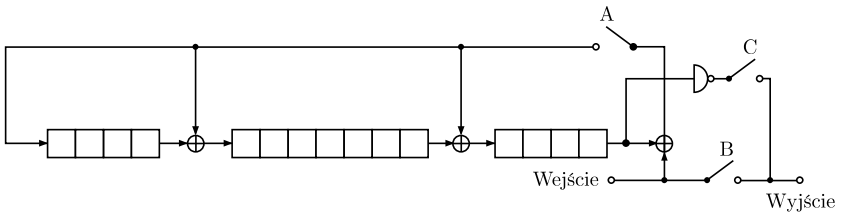
Prezentowany przykład kodu korygującego błędy dobrze ilustruje jedynie zasadę korekcji, natomiast bity nadmiarowe wyznaczone są tu z dużą rozrzutnością, a algorytm korekcji byłby nieefektywny w przypadku dużej liczby bitów informacyjnych. W praktyce n-bitowe słowo kodowe uważa się za wielomian stopnia $n - 1$ o współczynnikach z ciała dwuelementowego (0 i 1), a bity nadmiarowe oblicza się jako resztę z dzielenia wielomianu, odpowiadającego bitom informacyjnym przez specjalny wielomian, stopnia $n - m$, zwany wielomianem generującym, również o współczynnikach 0 i 1, którego wyznaczenie wymaga zaawansowanych wiadomości z algebry. Przy obliczaniu reszty stosuje się arytmetykę modulo 2, dlatego też układ do dzielenia wielomianów jest bardzo prosty i ma postać rejestru przesuwającego ze sprzężeniem zwrotnym zależnym od wielomianu generującego. Takie kody nazywają się kodami cyklicznymi i mają bardzo dobre właściwości detekcyjne i korekcyjne.

Najprostszym kodem detekcyjnym był stosowany od końca lat 40-tych kod z jednym bitem parzystości, równym sumie modulo 2 wszystkich bitów informacyjnych. Taki kod posiada minimalną odległość Hamminga równą 2, ponieważ każdy błąd pojedynczy wytworzy słowo kodowe z niepoprawnym bitem nadmiarowym. Odmianą tego kodu jest kod detekcyjny, stosowany w protokole BSC (rozdz. VII).



Rysunek 6: Struktura ramki w protokole HDLC

Do detekcji błędów transmisji w protokole HDLC, w którym między flagami stosuje się technikę wstawiania bitu, zastosowano kod cykliczny. Format ramki, przewidziany przez ten protokół pokazano na rys. 6, gdzie zaznaczono, które bity biorą udział w obliczaniu bitów nadmiarowych. 16 bitów nadmiarowych, zwanych sekwencją FCS (Frame Checking Sequence), umieszcza się oczywiście przed końcową flagą ramki. Do wytwarzania sekwencji FCS

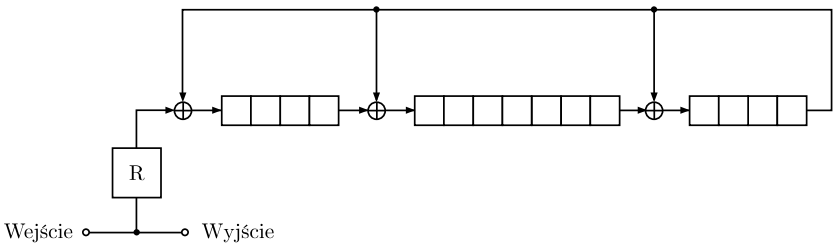


Rysunek 7: Generator sekwencji FCS

w nadajniku używa się 16-bitowego rejestru przesuwającego ze sprzężeniem zwrotnym, odpowiadającym wielomianowi $x^{16} + x^{12} + x^5 + 1$ (rys. 7). Jest to wielomian do generowania skróconych kodów cyklicznych o słowach kodowych, składających się maksymalnie z 32767 bitów, w tym 16 bitów nadmiarowych, zapewiający minimalną odległość Hamminga, równą 4. Wielomian ten nazywa się wielomianem CRC CCITT, a generowane przez niego słowa kodowe pozwalają wykryć do trzech błędów, o dowolnej konfiguracji oraz wszystkie błędy seryjne, rozmieszczone na przestrzeni 16. sąsiadujących ze sobą bitów. Badania symulacyjne potwierdziły dużą skuteczność tego kodu (ponad 97% praktycznie występujących błędów w kanale telefonicznym).

Istnieją dwa sposoby wyznaczania bitów nadmiarowych za pomocą ukła-

du z rys. 7. W pierwszym sposobie stan początkowy rejestru ustawia się na zera. Następnie zamyka się klucze A i B, a na wejście układu podaje się bity informacyjne z zanegowanymi dwoma pierwszymi oktetami. Podczas taktowania rejestru bity informacyje są przesyłane na wyjście i do rejestru, który oblicza na bieżąco FCS. Po przekazaniu na wejście wszystkich bitów informacyjnych otwiera się klucze A i B, a zamyka klucz C. W następnych 16. taktach na wyjście zostaje wysłana zanegowana zawartość rejestru, która stanowi sekwencję FCS. Drugi sposób nie wymaga zanegowania dwóch pierwszych oktetów, lecz ustawienia stanu początkowego rejestru na jedynki.



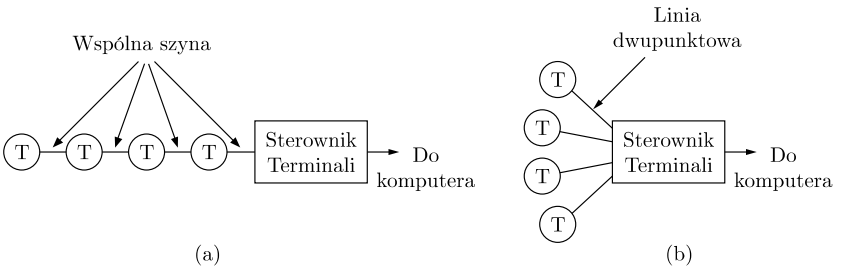
Rysunek 8: Układ do wykrywania błędów transmisji w protokole HDLC

Do sprawdzania poprawności ramki w odbiorniku stosuje się taki sam rejestr, jak w nadajniku, lecz o nieco innej konfiguracji (rys. 8). Sposób pracy rejestru należy dostosować do sposobu działania rejestru w nadajniku. Przy pierwszym sposobie rejestr jest wyzerowany. Na wejście wprowadza się zawartość ramki między flagami z zanegowanymi dwoma pierwszymi oktetami. Sekwencję FCS przed wprowadzeniem do rejestru należy również zanegować. W tym celu stosuje się rejestr opóźniający R, ponieważ odróżnienie FCS od bitów informacyjnych może nastąpić dopiero po odebraniu flagi końcowej. Jeżeli rejestr po wpisaniu do niego sekwencji FCS będzie zawierał same zera, odbiór ramki będzie bezbłędny. Niezerowy stan rejestru wskazuje na wystąpienie błędów transmisji. W drugim sposobie stan początkowy rejestru ustawia się na jedynki i nie neguje się dwóch pierwszych oktetów. Warunek bezbłędnego odbioru odpowiada końcowemu stanowi rejestru 0001110100001111.

Część VII
Obsługa terminali

Zdarza się często, że koszty eksploatacji linii telekomunikacyjnej przewyższają koszty podłączonych do niej urządzeń. Dlatego też, w wielu sieciach komputerowych stosuje się zasadę wspólnego użytkowania jednego kanału

telefonicznego przez wiele terminali. Istota tej koncepcji polega na stosowa-



Rysunek 9: Koncepcja wspólnego użytkowania jednej linii komunikacyjnej przez wiele terminali

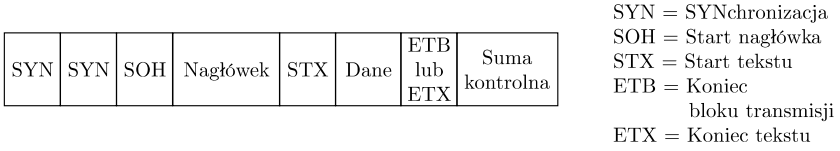
niu specjalnego urządzenia, zwanego sterownikiem (kontrolerem) terminali (rys. 9). Zadaniem sterownika terminali jest umożliwienie używania jednej linii do komputera wszystkim terminalom, które są połączone albo wspólną szyną (rys. 9 (a)), albo też każdy terminal posiada połączenie dwupunktowe ze sterownikiem terminali (rys. 9 (b)). Sterownik terminali ma zatem eliminować występowanie sytuacji, w których wiele terminali równocześnie chciałoby nadawać dane. Najprostszym rozwiązaniem tego problemu jest przyjęcie zasady, że wszystkie terminale oczekują na sygnał ze strony sterownika terminali, zezwalający na nadawanie i że taki sygnał może otrzymać w danej chwili tylko jeden terminal. Tę metodę zapobiegania kolizji podczas uzyskiwania dostępu do łącza nazwać można zaproszeniem do nadawania (nazwa angielska metody – polling, ma co najmniej 11 znaczeń, z czego kilka odnosi się do głosowania, zbierania opinii, a jedno oznacza tę część młotka, która służy do wbijania gwoździ).

Istnieją dwie odmiany tej metody: jedna dla terminali, połączonych wspólną szyną, druga gdy każdy z terminali ma własne połączenie dwupunktowe ze sterownikiem terminali. W pierwszym przypadku sterownik cyklicznie wysyła do każdego terminala wiadomość w postaci zapytania, czy terminal potrzebuje nadawać dane. Wiadomość ta zawiera adres każdego terminala. Każdy terminal reaguje tylko na własny adres, chociaż odbiera wiadomości, przesyłane z kontrolera do wszystkich terminali. Jeśli wybrany do nadawania terminal ma dane do wysłania, to je wysyła, a jeśli nie to wysyła do sterownika specjalną wiadomość, oznaczającą odrzucenie zaproszenia do nadawania. Zwykle sterownik zaprasza do nadawania cyklicznie wszystkie terminale po kolei, ale możliwe też jest w pewnych okolicznościach kilkakrotne zapraszanie do nadawania ważniejszych terminali w ciągu jednego cyklu.

Przy połączeniach półdupleksowych każde zaproszenie do nadawania wymaga odwrócenia kierunków transmisji, tak aby najpierw mógł nadawać sterownik do terminali, a następnie zapraszany terminal do sterownika. Te czynności wymagają też uaktywniania tłumików echa na długich łączach i trwają

kilkaset mikrosekund, co wydłuża znacznie czas trwania cyklu, jeśli nawet niewiele terminali podczas jednego cyklu wysyła dane. W celu wyeliminowania tej niedogodności opracowano inny sposób zapraszania terminali do nadawania, zwany hub polling (hub – centrum, środek, piasta). W tej metodzie sterownik terminali zaprasza do nadawania terminal, znajdujący się najdalej od sterownika. Terminal ten zmienia kierunek transmisji, i jeśli ma dane do przesłania, to je wysyła, jeśli nie, to wysyła zaproszenie do nadawania do terminala sąsiedniego. Z kolei ten terminal podejmuje takie samo działanie i proces się powtarza, aż wezwanie do nadawania dotrze do sterownika. W ten sposób zaoszczędza się na czasie zmiany kierunków transmisji łącza.

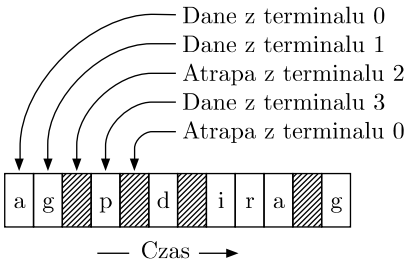
Jeśli połączenie terminali z ich sterownikiem ma strukturę gwiazdy, wówczas cykliczne zapraszanie do nadawania nie musi mieć miejsca. Mimo to, stosuje się i w tym przypadku cykliczne zapraszanie do nadawania, aby sterownik mógł przydzielać kanał terminalom w sposób uporządkowany, ale wiadomość stanowiąca zaproszenie do transmisji nie musi oczywiście zawierać adresów terminali.



Rysunek 10: Format wiadomości protokołu BSC

Do zapraszania zdalnych terminali do nadawania firma IBM opracowała protokół BSC (Binary Synchronous Communication), który przeznaczony jest do obsługi kanałów półdupleksowych. Protokół ten uwzględnia 3 rodzaje kodów: 8-bitowe ASCII i EBCDIC oraz 6-bitowy Transcode firmy IBM. Format wiadomości przy stosowaniu tego protokołu pokazano na rys. 10. Każda wiadomość zaczyna się dwoma znakami SYN, służącymi do podtrzymania synchronizacji, po których następuje znak SOH, oznaczający początek nagłówka. Po znaku SOH przesyłana jest treść nagłówka, co do której protokół nie narzuca żadnych ograniczeń. Po nagłówku wysyłany jest znak STX, sygnalizujący początek danych, a po nim następują znaki danych. Znak ETB (koniec bloku transmisyjnego) lub ETX (koniec tekstu) zaznacza miejsce końca danych. Przy stosowaniu kodu ASCII ustawiany jest bit parzystości, a ostatni znak wiadomości jest znakiem parzystości wzdłużnej (suma modulo 2 wszystkich bitów pierwszych, drugich, itd. każdego znaku, wchodzącego w skład wiadomości).

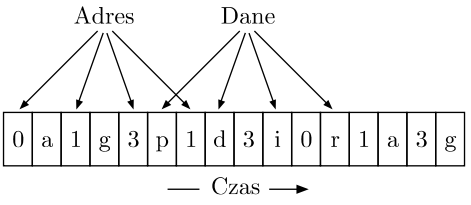
Sterownikami terminali są albo multipleksery, albo koncentratory. Multiplekser z podziałem czasowym (TDM – Time Division Multiplexer) jest urządzeniem, które przyjmuje dane z wielu linii wejściowych w określonej kolejności i kieruje te dane do jednej linii wyjściowej. Ponieważ każdej linii wejściowej przydzielona jest określona szczelina czasowa, nie ma tu potrzeby nadawania numerów linii wejściowych, ale szybkość transmisji linii wyjściowej musi być równa sumie szybkości transmisji linii wejściowych. Na rys. 11 pokazano zasadę działania multipleksera z czterema terminalami. Każdemu terminalowi przydzielono tu 1/4 wyjściowej szczeliny czasowej, niezależnie od jego aktualnej aktywności. Jeśli każdy terminal działa z szybkością 1200 bitś, to linia wyjściowa musi pracować z szybkością 4800 bitś ($4 \cdot 1200$), a w każdym cyklu muszą być przesłane 4 znaki, po jednym na każdy terminal. Jeśli terminal nie ma danych do wysłania, jego szczelina czasowa jest zmarnowana, ponieważ wyjście jest formowane cyklicznie z czterech wejść. W tym



Rysunek 11: Zasada działania multipleksera z czterema terminalami

przypadku szczelinę tę wypełnia specjalny znak jałowy, którego nie można przeskoczyć, bo po stronie odbiorczej sprawdza się dane z każdego terminala, a na początku transmisji komputer i terminal są synchronizowane i znają kolejność szczelin czasowych, przydzielonych każdemu terminalowi. Same dane nie niosą informacji o źródle ich pochodzenia, jeśli więc jakaś szczelina czasowa byłaby przeskoczona, to odbiornik mógłby niewłaściwie interpretować odebrane dane.

Jeśli część podłączonych do multipleksera terminali nie nadaje danych, to szybkość transmisji linii wyjściowej nie jest skutecznie użytkowana. Często więc jest możliwe zastosowanie linii wyjściowej o szybkości transmisji mniejszej niż suma szybkości transmisji linii wejściowych. Mówi się wtedy o tzw. koncentracji, która polega na przesyłaniu wyłącznie danych, bez znaków jałowych, wynikających z bezczynności terminala. Jednak przy tej strategii należy informować stronę odbiorczą o źródle pochodzenia danych. Jednym z możliwych rozwiązań jest przesyłanie z każdego terminala dwóch znaków, z których jeden stanowi numer terminala, a drugi dane. Zasadę tę ilustruje rys. 12. Koncentratory, pracujące na tej zasadzie nazywają się multiplekserami statystycznymi lub multiplekserami ATDM (Asynchronous Time Division Multiplexer).



Rysunek 12: Zasada działania koncetratora z czterema terminalami

Koncentracja stwarza też pewne niebezpieczeństwa. Jeśli bowiem wszystkie terminale zaczną nadawać dane z maksymalną szybkością, to linia wyjściowa nie nadąży z ich przesyłaniem i niektóre dane mogą być stracone. Z tego powodu koncentratory wyposaża się w specjalne bufory, mające na

celu uniknięcie utraty danych. Im większa jest pamięć bufora koncentratora, tym mniejsza jest możliwość utraty danych, ale też i większy koszt tego urządzenia.