

5. HDLC

概要 (1)

■ CCITT X.25 (HDLC/LAPB)

- » IBM社のSNAにおける SDLC (Synchronous Data Link Control protocol) を基本に開発
- » HDLC
 - High-speed Data Link Control protocol
 - ITU-T/ISOによるSDLCの国際標準化
- » ADCCP
 - Advanced Data Communication Control Protocol
 - ANSI (米国)によるSDLCの標準化
- » LAPB
 - Link Access Procedure B
 - CCITT X.25 におけるデータリンク層, HDLC と同じ

Copyright ©2000, Suguru Yamaguchi, All right reserved

概要 (2)

- 古いプロトコルだが、現在も広く使われている
 - » ISDNなどの公衆データ通信網で利用
 - » 既存のオンライン網
 - 銀行のオンラインシステムなど
- 物理層プロトコルは X.21 を一般的に使用
- 特徴
 - » bit-oriented
 - bit-stuffing を利用してデータ転送の透過性を確保
 - » connection-oriented
 - データリンクの設定・切断処理が必要
 - 電話型

Copyright ©2000, Suguru Yamaguchi, All right reserved

HDLC frame (1)

- Address: 通信局の判別
- Control: 通信制御のための情報
- Data: データ伝送される情報
- Checksum: CRC-CCITT

8	8	8	≥ 0	16	8
01111110	address	Control	Data	Checksum	01111110

Copyright ©2000, Suguru Yamaguchi, All right reserved

HDLC frame (2)

■ フレームの種類

- » 情報フレーム
- » 監視フレーム
- » 非番号制フレーム

フレームの種類	b1	b2	b3	b4	b5	b6	b7	b8
情報フレーム	0	N(S)			P/F	N(R)		
監視フレーム	1	0	S		P/F	N(R)		
非番号制フレーム	1	1	M		P/F	N(R)		

N(S): 送信フレーム番号
N(R): 受信フレーム番号
P/F: ポール/ファイナルビット
S: 監視機能ビット
M: 修飾機能ビット

Copyright ©2000, Suguru Yamaguchi, All right reserved

HDLC frame (3)

■ ポール/ファイナルビット

- » コマンドの場合は、応答を要求するかどうかを表すポールビット (poll bit) として利用
- » 応答の場合は、最後のフレームかどうかを表すファイナルビット (final bit) として利用

■ 情報フレーム

- » 実際のデータ転送に利用されるフレーム
- » N(S)にはフレーム番号
- » if N(R) = k: (k-1)フレームまで正しく受信されたことを意味する

Copyright ©2000, Suguru Yamaguchi, All right reserved

HDLC frame (4)

■ 監視フレーム

- » データリンク通信中の制御
- » SIによって種類を表す

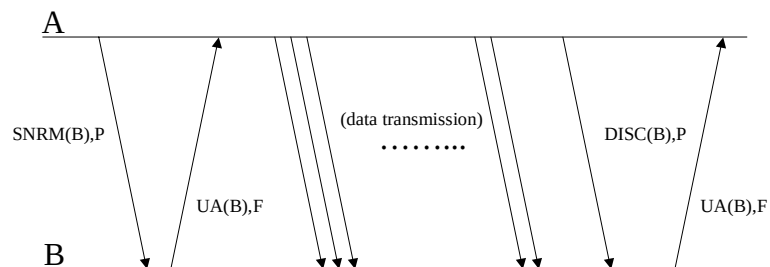
- | | |
|--|---|
| <ul style="list-style-type: none"> • RR <ul style="list-style-type: none"> - N(R)より小さな番号のフレームを正しく受信 - 次のフレーム送信を促す • RNR <ul style="list-style-type: none"> - N(R)より小さな番号のフレームを正しく受信 - 一時的に受信不能(バッファ溢れなど) - RRを送ることで送信再開 | <ul style="list-style-type: none"> • REJ <ul style="list-style-type: none"> - N(R)より大きな番号のフレームを正しく受信していない - Go-back-N で N(R)以降から再送する • SREJ <ul style="list-style-type: none"> - N(R)より小さな番号のフレームを正しく受信 - Selective-Repeatにより N(R)だけを再送 |
|--|---|

Copyright ©2000, Suguru Yamaguchi, All right reserved

HDLC frame (4)

■ 非番号制フレーム

- » データリンクの設定・切断に利用
- » モードがいくつかある



Copyright ©2000, Suguru Yamaguchi, All right reserved

6. LAN技術とメディアアクセス方式

現在の技術

- Ethernet (10, 100, G)
 - » CSMA/CD
- FDDI
 - » Token Ringを改良発展
- ATM
 - » 固定長セルの多重化
 - » VC (Virtual Circuit) とVP (Virtual Path)
- POS
 - » 広帯域・長距離の2点間接続
 - » PPP Over SONET

メディアアクセス方式の概要

■ LAN

- » 複数のホストで単一の物理的伝送媒体を共有
- » どのように伝送媒体を利用するか
- » アクセス制御 (Media Access Control)

■ 種類

- » 制御型 (controlled)
 - 固定割当方式
 - TDMA (Time Division Multiple Access)
 - 要求適応割当方式
 - Token Passing
- » 競合型 (contention)
 - ALOHA, CSMA, CSMA/CD...

Copyright ©2000, Suguru Yamaguchi, All right reserved

TDMA (1)

■ Time Division Multiple Access

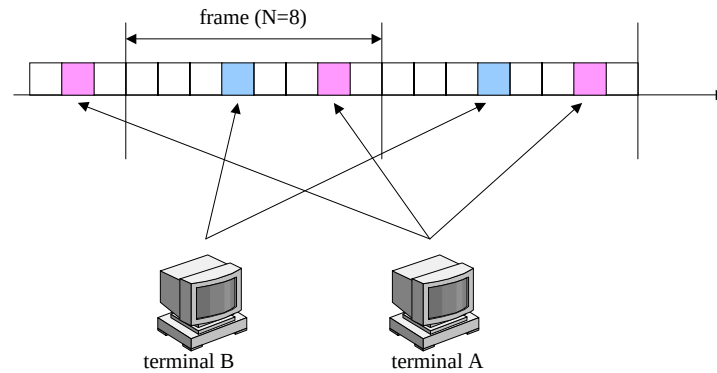
- » 通信路を伝送時間Tで分割 (スロット)
- » N個のスロットで一つのフレームを構成
- » 端末は特定のスロットを利用

■ 例

- » デジタル専用回線多重化装置
- » 1.5Mbps の回線を複数の機器に利用させる
 - 時分割で分割
 - 64Kbpsを単位として分割できる商品多数

Copyright ©2000, Suguru Yamaguchi, All right reserved

TDMA (2)



Copyright ©2000, Suguru Yamaguchi, All right reserved

TDMA (3)

■ スロットの予約と解放

- » 事前に管理者が割り当てる方法は簡単
- » 動的に割り当てる機構は端末側にも特別な機能が必要となる

■ 時刻同期

- » フレームの切り出しは時刻を基準に実施
- » 端末および交換ノードは同期が必要

Copyright ©2000, Suguru Yamaguchi, All right reserved

TDMAからATMへ

- TDMAは広く使われるメディアアクセス方式
 - » 多重化方法としては単純
 - » 性能保証が可能
 - » 現在でもTDMAによる製品は多い
- TDMAではLANトラヒックの収容が難しい
 - » 端末の時刻同期は保証されない
 - » 端末が利用する帯域は事前に予測できない
- ATMの登場
 - » LANトラヒックの収容
 - » TDMA的な多重化効果を狙ったデータリンク

Copyright ©2000, Suguru Yamaguchi, All right reserved

Pure ALOHA (1)

- | | |
|--|--|
| <ul style="list-style-type: none">■ 競合型■ 歴史<ul style="list-style-type: none">» ハワイ大学の ALOHA System で開発» 二つの無線回線<ul style="list-style-type: none">• 端末からホストへ<ul style="list-style-type: none">- uplink• ホストから端末へ<ul style="list-style-type: none">- downlink» 二つの無線回線の共有方式を開発 | <ul style="list-style-type: none">■ 方式<ul style="list-style-type: none">» 各端末は独立に動作» 端末からは送るべきデータが有れば直ちに送出» パケット送信後 RTT (Round Trip Time) まっても ACKが返ってこない場合衝突発生と判断» バックオフ処理<ul style="list-style-type: none">• $[0, K]$ で乱数発生: r• r秒後に再送を実施 |
|--|--|

Copyright ©2000, Suguru Yamaguchi, All right reserved

Pure ALOHA (2)

- 各端末が自律的にメディアにアクセス
 - » 無秩序であるので効率は「当然」悪い
 - » 秩序はシステムの複雑化を引き起こす
- 衝突の多発は性能劣化を引き起こす
 - » 端末数が増える
 - » 通信量が増える
- バックオフ時間
 - » K
 - » 性能に影響を与えるパラメータ

Copyright ©2000, Suguru Yamaguchi, All right reserved

slotted ALOHA

- pure ALOHA を改良
 - » 伝送時間をスロット化
 - 端末からの送信をスロットに従わせる
 - 「秩序」の導入
 - » 回線の無秩序な競合を、スロットの競合に
 - 競合の発生を少なくする
 - 性能の改善
- 問題点
 - » スロットの実現が難しい(時刻同期)
 - » 理論的なモデルにとどまる

Copyright ©2000, Suguru Yamaguchi, All right reserved

CSMA

- Carrier Sense Multiple Access
- 送信前にメディアが使われていないかを調べる
 - » 使われていなければ直ちに送信
 - » 伝送遅延が有るため、衝突は発生せざるをえない
 - » $RTT/2$ の間に他局が送信すると衝突
 - 伝送媒体全体に送信状態が伝播する時間

Copyright ©2000, Suguru Yamaguchi, All right reserved

CSMA

- 搬送波を検出した場合
 - » 1-persistent CSMA
 - 必ず送信
 - » p-persistent CSMA
 - 確率 p で送信
 - » nonpersistent CSMA
 - 衝突発生時の再送アルゴリズムの適用

Copyright ©2000, Suguru Yamaguchi, All right reserved

CSMA/CD

- CSMA with Collision Detection
- 通信中もフレームを監視し、衝突を発見する
 - » 衝突が発生した場合には直ちに送信を中止
 - » 衝突後の不要な回線占有を回避
- 再送アルゴリズム
 - » Ethernet ではbinary backoff 方式を採用
 - » 一度衝突すると backoff timer を2倍にしてい

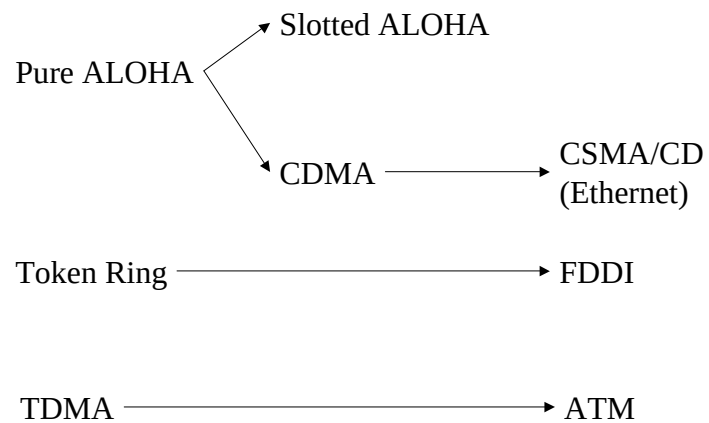
Copyright ©2000, Suguru Yamaguchi, All right reserved

Token Passing

- 802.4 (Token Bus), 802.5 (Token Ring) で採用
- FDDIも Token Passing 方式の一つ(ちょっと複雑)
- 方式
 - » ネットワークに接続されたホスト間でトークン (token) を「順番」に交換
 - » フリートークンを持っているホストが送信できる
 - » token holding timer

Copyright ©2000, Suguru Yamaguchi, All right reserved

歴史的変遷



Copyright ©2000, Suguru Yamaguchi, All right reserved

まとめ

- これまで多数の方式が提案されてきた
- 結局生き残ったのは
 - » 1-persistent CSMA/CD (Ethernet)
 - » Token Ring
- 性能を考えると
 - » CSMA/CD は破綻する
 - » Token Ringは破綻しない

Copyright ©2000, Suguru Yamaguchi, All right reserved