

# 空間光通信における複数プロトコルの自動判別方式\*

今井 明 塚本 昌彦 市川 雄二 稗田 薫<sup>†</sup>

シャープ(株) 技術本部 情報技術研究所<sup>‡</sup>

e-mail: {imai, tuka, ichikawa, hieda}@shpcsl.sharp.co.jp

本稿では、複数の光通信プロトコルを備えた機器において、相手局の備える物理層およびデータリンク層のプロトコルを自動的に判別、選択し通信する自動判別方式について論ずる。送信、受信ともに複数のプロトコルを交互に繰り返す手順を用いて、既存のMAC(メディアアクセス制御)ルールに従ったデータ通信が可能となる。さらに、このような考え方の適用事例として、電子手帳やPDAなどで現在広く使われている通信手順とIrDA(Infrared Data Association)の標準である通信手順という二つのLED(Light Emitted Diode)を用いた赤外線通信方式の自動判別を行なうアルゴリズムを提案し、実装したシステムを用いた実験結果について報告する。

## 1 はじめに

計算機の高性能化、低コスト化、軽量化が年々進んでおり、携帯できる情報機器(携帯情報機器)を用いた個人向けの分散計算機環境の実現に向けての研究が進んでいる。特に、PDA(Personal Digital Assistant)と呼ばれる電池駆動の軽量な情報機器は、ユーザが情報の検索、記録、交換などの操作を必要とした時に「いつでも、どこでも」行なうことが可能であるため、このような機器の利用を前提とした分散計算機環境の研究が進んでいる。携帯情報機器においてデータ通信機能は、その記憶容量が比較的少ないという制限を捕うものであることと、分散環境のシステム側と人間側のデータ入出力インタフェースとして利用されることから特に重要である。

このような携帯情報機器のデータ通信機能は、無線通信[1]で実現されることが望ましい。すなわち、移動し通信を行なおうとする

度に有線での結線操作が必要になると、携帯情報機器の最大の長所である可搬性を大きく損なってしまうためである。また、無線通信のメディアとして赤外線(Infrared: IR)[2]を用いた通信は、電波を用いたものに比べて以下の点で特に有望である。

- 送受信器の小型化、軽量化、低コスト化が容易である。
- 電波の場合は国や地域によって使用可能な周波数や出力の制限が異なり、また免許が必要となることがあるが、赤外線の場合は一般に使用範囲の柔軟性がある。
- 電波は壁を越えて伝わるが、赤外線は目に届く範囲にしか到達しないので、セキュリティなどの観点から運用上の管理が容易である。

有線通信の場合は、通常、機器が電氣的な障害を起こさないように物理層プロトコルは同一のもの同士でしか接続されることはなく、異なるプロトコルが単一のケーブルで混在することは稀である。また、物理層プロトコルが異なるために誤って接続すると何らかの障

\*An Automatic Protocol Recognition Scheme for Wireless Communication Using Light Radiation

<sup>†</sup>Akira IMAI, Masahiko TSUKAMOTO, Yuji ICHIKAWA, and Kaoru HIEDA

<sup>‡</sup>Information Technology Research Labs, Corporate R&D Group, SHARP Corporation

害が発生する可能性がある場合などは、それぞれ異なる形状のコネクタを用いることで問題を回避できる。さらに、ユーザが原理的に接続不可能であることをコネクタの形状の違いによって知ることができ、ユーザの混乱を招く恐れも少ない。

それに対して無線通信では、異なるプロトコルが共通のメディアを同時に利用し、互いに妨害し合うような状況が考えられる。電波の場合には、スペクトル拡散技術である DS (Direct Sequence) や FH (Frequency Hopping) を利用して異なるプロトコルの互いの干渉を削減し共存の可能性を高めることが可能であるが、赤外線の場合は、発光デバイスの ON/OFF による符号化を行なうので、スペクトル拡散技術の適用は一般に不可能である。さらに、無線の場合にはコネクタが存在しないことや、ユーザからは二つの機器が物理的に接続されているかどうかを確認する手段がないことなどから、接続不能である場合にその理由を知ることは容易ではないという問題がある。

赤外線を用いた通信プロトコルがいくつか提案されている [3]。IEEE 802.11 IR [4, 5] は、赤外線を用いた LAN を実現するためのプロトコルを定義している。赤外線データ協会 (Infrared Data Association: IrDA) は民間の標準化団体であり、HDLC に基づいた標準通信方式を定めている。シャープやアップルの携帯情報機器の一部は ASK プロトコル [6] を搭載している。卑近な例では、テレビやビデオなどの家電機器のリモコンで用いられるものがある。これらはほぼ同じ波長の赤外線を用いるため、同じ場所で用いると互いに妨害し合う可能性がある。したがって、使用するプロトコルを動的に変更し、互いに妨害し合わないような機能を実現することが特に重要になる。また、複数のプロトコルを解釈できる機器であっても、それらをユーザが選択するのではなく、システムが自動的に識別することにより、ユーザはプロトコル設定など

の複雑な処理から解放される。

本稿では、通信が不能であった場合に再送を行なうプロトコルの特徴を利用し、使用するプロトコルを動的に変更する方法について提案を行なう。

## 2 自動判別方式

### 2.1 プロトコルスタック構成

自動判別とは、他局との通信要求を受けた際に、相手局と通信可能な物理層プロトコルを自動的に識別し選択する機能である。この機能は、以下の理由から、データリンク層で実現されるべきである。

- データリンク層は MAC (メディアアクセス制御) の機能を含む。もし、自動判別機能がそれ以下の層で実現されるとするならば、MAC 規則に違反し、他の通信リンクを妨害する恐れがある。
- それ以上の層は通信に用いるメディアに依存すべきではない。

自動判別機能を含めたデータリンク層の内部構成を考えるために、まず通常データリンク層が提供する機能を分類する。OSI の LAN 標準ではデータリンク層を LLC (論理リンク制御) 副層と MAC 副層に分割している。LLC 副層にはメディアの信頼性に応じたデータ転送機能の異なるクラスがある。また上位層エンティティを多重化するマルチプレクサ機能が提供されている。MAC 副層では送信衝突を考慮したメディアへのアクセス方法を規定する。IrDA のプロトコルスタックでは、IrLAP (Infrared Link Access Protocol) [7] が再送などのデータ転送機能と MAC 機能を実現し、IrLMP (Infrared Link Management Protocol) [8] がマルチプレクサ機能を実現している。これらのことから、既存のデータリンク層で提供される機能として次の三つの機能が抽出できる。

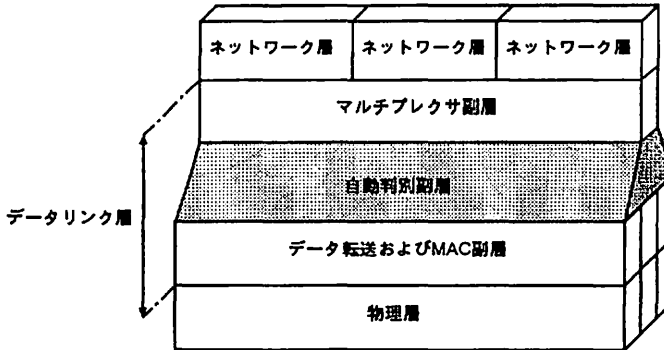


図 1: 自動判別機能を追加したプロトコルスタック

1. マルチプレクサ機能 — 上位の複数エンティティを管理し、パケットを分配する機能。リンクを多重化して複数のアプリケーションで使用する場合に必要となる。
  2. データ転送機能 — データ転送の信頼性を上げるための制御を行なう機能。パケット配送の信頼性が低い場合には、Ack、再送などの手順が使用される。
  3. MAC 機能 — メディアへのアクセスを制御する機能。パケット送信衝突を考慮したアクセス手順が規定される。
- Open — 下位プロトコルを認識し初期化を行なう。下位層がコネクション型の場合にはコネクションを設定する。上位層がコネクションレス型なら、起動時や障害時など自動判別機能の必要時に、コネクション型の場合にはコネクションを設定する前に発呼する必要がある。
  - Close — プロトコルスタックを初期状態に戻す。下位層がコネクション型の場合には最初にコネクションを解放する。
  - Transmit — データを送信する。
  - Receive — データを受信する。

本稿で提案する自動判別副層を追加したプロトコルスタックを図1に示す。本プロトコルスタックにおいて、自動判別副層は、マルチプレクサ機能を提供するマルチプレクサ副層と、データ転送機能およびMAC機能を提供するデータ転送およびMAC副層の間におく。CLNP(ConnectionLess-mode Network Protocol), IP (Internet Protocol), IPX(Internet Packet Exchange Protocol)などのネットワーク層エンティティはマルチプレクサ機能が識別する。自動判別副層は、上位層からの通信要求に対して適切な下位プロトコルを選択し、そのMAC機能とデータ転送機能を用いて通信を行なう。

自動判別副層は、以下のインタフェースを上位層に提供する。

このような自動判別機能を含むプロトコルアーキテクチャによって、メディアに依存したアクセス制御およびデータ転送制御をメディアに応じて切替えることができ、さらに自動判別機能を含むデータリンク層はその複数の上位層に対し、通常のデータリンク層と同様のサービスを提供することが可能となる。

## 2.2 動作原理

本稿で提案する自動判別機能は、識別される複数のプロトコルがそれぞれ、データリンク層において再送を行なったり、またはコネクションを設定したりする機能を利用して実現される。受信部が単一場合には、複数のモードをアプリケーションからの指示により切替えるものと仮定する。

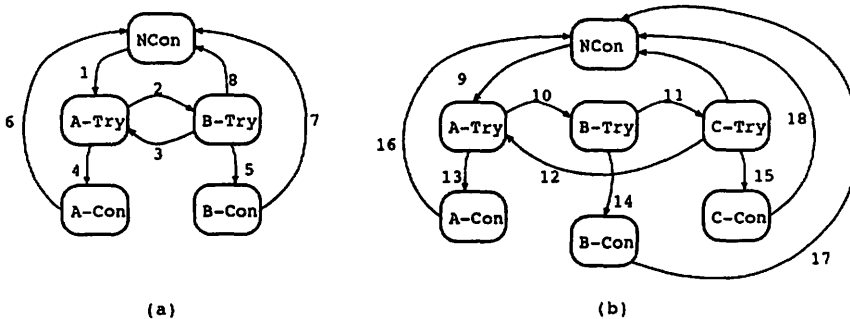


図 2: 自動判別機能を行なう局の状態遷移図

本方式の基本的な動作は以下の通りである。ここで、あるプロトコル A を受信可能であるように設定した場合、「受信器のモードはプロトコル A である」と表現する。同様に、プロトコル A で送信可能であるように設定した場合、「送信器のモードはプロトコル A である」と表現する。

受信側は以下の動作を行なう。

- 受信器のモードを定期的に切替える。具体的には、受信可能なプロトコルが、プロトコル A、プロトコル B、プロトコル C の三つ存在する場合、受信器のモードをプロトコル A で  $t_a$  秒、プロトコル B で  $t_b$  秒、プロトコル C で  $t_c$  秒で切替え、再びプロトコル A で  $t_a$  秒... と繰り返す。
- ある受信器モードで、正常にパケットを受信することができたら受信器のモードをそのモードに固定して、コネクションを設定し、データ交換を行なう。コネクションを解放するまで、受信器のモードの切替えは行なわない。

また、送信側は以下の動作を行なう。

- 送信器のモードを定期的に切替え、それぞれのモードでコネクションの設定を試みる。具体的には、受信可能なプロトコルとして A、B、C の三つのプロトコルが存在する場合、受信器のモードをプロ

トコル A で  $t_a$  秒、プロトコル B で  $t_b$  秒、プロトコル C で  $t_c$  秒で切替えるという動作を繰り返す。

- あるモードで「応答」あるいは「acknowledgement」を受信した場合には、送信器のモードをそのモードに固定して、コネクションを設定し、データ交換を行なう。コネクションを解放するまで、送信器のモードの切替えは行なわない。

図 2 は送信器および受信器の状態遷移図である。ここで、(a) は二つのプロトコルの自動判別、(b) は三つのプロトコルの自動判別を行なう場合である。状態の定義は以下の通りである。

X-Try その局がプロトコル“X”でのコネクションの設定を試みている状態。

X-Con はプロトコル“X”でコネクションが設定されている。

NCon コネクションが設定されておらず、また試みられてもいない。

図 2(a) を用いて、送受信器の動作をより詳細に説明する。送信器側はユーザからデータ送信を要求された場合、状態を“NCon”から“A-Try”に変更する(1)。その後、プロトコル A でのコネクションの確立を予め定められた時間だけ行なう。この間にコネクションが確

立されれば、モードをプロトコル A に固定し (4), “A-Con” 状態で通信を行ない, 通信が完了したらコネクションを解放し, “NCon” 状態へ変更する (6). コネクションが設定できなければ, “B-Try” に変更し (2), 予め定められた時間だけプロトコル B でのコネクションの設定を試みる. プロトコル B でコネクションが設定されれば, モードをプロトコル B に固定して (5), “B-Con” 状態で通信を行ない, 通信が完了したらコネクションを解放し, “NCon” 状態へ変更する (7). コネクションが設定できなければ, 再び “A-Con” 状態に変更して (3) コネクションを試みる. 両プロトコルで, 定められた回数だけ試みてもコネクションが設定されない場合は, ユーザに送信不能を通知し, “NCon” 状態に戻す (7). 受信器側はユーザからデータ受信を要求された場合, 状態を “NCon” から “A-Try” に変更する (1). その後は送信器側と同様に, あるモードでのコネクションが設定されたら, そのモードに固定し, データの受信を行なう.

局が三つ以上のプロトコルを受信可能な場合も, 同様に定期的にモードを切替え (10,11,12), コネクションが設定できた場合にはモードを固定して (13,14,15) 通信を行なう.

### 2.3 切替え時間に対する制約

モード切替え時間の設定に当たっては, 具体的な時間はそれぞれのプロトコルに依存して定められるが, この方法について, プロトコル A とプロトコル B の二つのプロトコルを認識できる場合の設定方法について述べる.

- $T_{c_a}, T_{c_b}$ : コネクションを設定するために最低限必要な時間であり, それぞれのプロトコル毎に定められる.
- $T_{t_a}, T_{t_b}$ : 送信器側の切替え時間. プロトコル A のモードを  $T_a$ , プロトコル B のモードを  $T_b$  それぞれ保持する.
- $T_{r_a}, T_{r_b}$ : 受信器側の切替え時間. プロトコル A のモードを  $T_a$ , プロトコル B

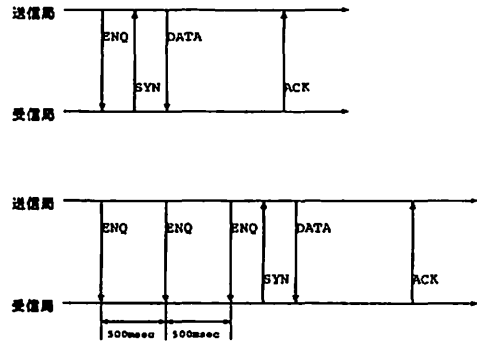


図 3: ASK プロトコルのシーケンス例

のモードを  $T_b$  それぞれ保持する.

この場合, 以下の制約が満たされる必要がある.

$$T_{t_a} > T_{c_a}, \quad T_{r_a} > T_{c_a}, \quad T_{t_b} > T_{c_b}, \\ T_{r_b} > T_{c_b}$$

$$T_{t_a} + T_{t_b} \neq T_{r_a} + T_{r_b}$$

前者の制約は, 接続処理の完了を保証するためであり, 後者の制約は, 送信器側と受信器側が共に二つのプロトコルを自動的に判別する局であった場合に, それぞれ逆のモードに設定し, コネクションが設定できなくなることを防止するためである.

## 3 既存プロトコルへの適用

赤外線を用いた通信プロトコルが種々提案されている. ここでは, 前章で述べた自動判別方式の具体例として, PDA に実装されている ASK プロトコルと, IrDA で提案されている IrLAP の二種のプロトコルを自動的に識別する方法について述べる.

### 3.1 既存プロトコル

ASK プロトコルと IrLAP は前提とする物理層も異なる. ASK プロトコルは, 波長 900~1040 nm の赤外線を用い, 500 kHz の副搬送波を用いた ASK (Amplitude Shift Keying) 変調方式の物理層を用いる. IrLAP は,

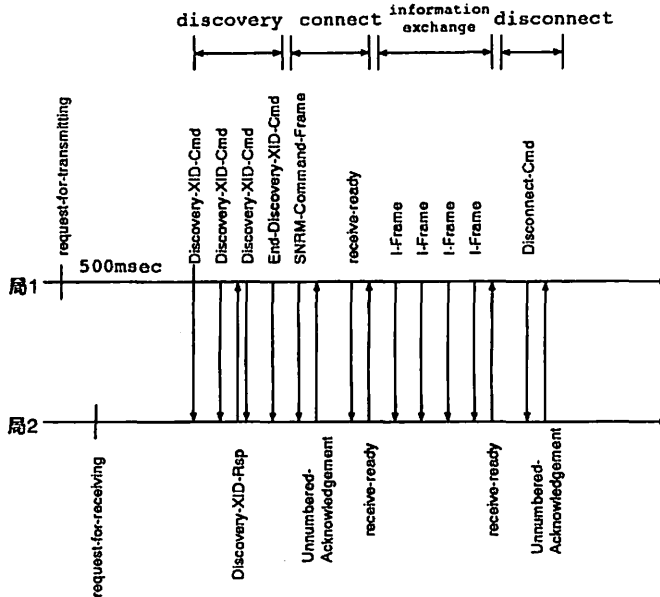


図 4: IrLAP のシーケンス例

波長 850nm のベースバンドの RZ (Return-to-Zero, パルス幅は 3/16) 方式の物理層を用いる [9].

ASK プロトコルのデータリンク層プロトコルを図 3 に示す。これは 1 対 1 の通信専用のプロトコルである。送信局は、最初に ENQ パケットを送信し、SYN パケットの到着を待つ。SYN パケットを受信するとデータパケットを送信するが、SYN パケットが受信できない場合には、SYN パケットが受信できるまで、500 ミリ秒毎に ENQ パケットの再送信を行なう。受信局は、データパケットをエラーなく受信できた場合には CK パケットを送信し、そうでなければ NACK パケットを送信する。送信しようとするデータが、最大データパケット長を越える場合には、データを複数のパケットに分割し、それぞれのデータパケットを、ENQ-SYN-DATA-ACK の列で送信する。各データパケットには、パケット番号と、最終パケットの場合にのみ付随されるフラグがあるため、複数のパケットを送信する場合には、最初のパケットに先立つ ENQ-

SYN の交換でコネクションを設定し、最後のデータパケットと ACK の交換をコネクションの解放手続きとみなすことができる。

一方 IrLAP では、HDLC (High-level Data Link Control) をもとにした 1 対 1 および 1 対 N 型のプロトコルである (図 4 参照)。このプロトコルでは、通常、互いの局アドレス情報を交換する手続きから開始される。一方の局が最初に 500 ミリ秒、トラヒックの有無を検査し、トラヒックがなかった場合のみ Discovery-XID-Command パケットを送信する。もう一方の局は、これに Discovery-XID-Response パケットを送信することで応じる。Discovery-XID-Command 送信後、応答がない場合には、500 ミリ秒おきにこの再送を行なう。局アドレスの交換が終わった後、コネクションを設定し、データを送信し、コネクションを解放する。

ここで重要な点は、いずれのプロトコルにおいても、データ交換に先だって一方の局が相手局に応答を求めるパケットを送信し、一定期間内に応答がない場合には、このパケッ

ト再送する点である。これが、本方式で自動判別を行なう上での条件となっている。

### 3.2 ASK と IrLAP の自動判別方式

本稿で提案する手法を ASK と IrLAP の自動判別に適用した例を説明する。図 5 は送受信器の状態遷移を示したものである。この図の五つの状態は以下の意味を持つ。

**NCon (Not Connected):** コネクションが設定されていない状態。送受信要求も出ていない。

**I-Try (IrLAP Connection Trying):** IrLAP でのコネクションを設定しようとしている状態。Discovery-XID-Command と Discovery-XID-Response の交換を一定期間試行する。送信局の場合この期間は 1000 ミリ秒であり、受信局の場合は 1500 ミリ秒である。

**A-Try (ASK Connection Trying):** ASK でのコネクションを設定しようとしている状態。送信局の場合、ENQ パケットを送信し、500 ミリ秒間 SYN パケットの受信を待つ。受信局の場合、ENQ パケットの受信を 1000 ミリ秒間待つ。

**I-Con (Connected with IrLAP):** IrLAP でコネクションが設定され、通信を行なっている状態。

**A-Con (Connected with ASK):** ASK でコネクションが設定され、通信を行なっている状態。

送信局の動作は以下の通りである。番号は図 5 中の矢印に対応する。

1. ユーザからの送信要求を受けた。
2. IrLAP でのコネクション設定が一定期間 (1000 ミリ秒) 内に完了しなかったため、送受信器のモードを ASK に設定する。

3. ASK でのコネクション設定が一定期間 (500 ミリ秒) 内に完了しなかったため、送受信器のモードを IrDA に設定する。
4. IrLAP でのコネクションが設定されたため、送受信器のモードを IrDA に固定する。
5. ASK でのコネクションが設定されたため、送受信器のモードを ASK に固定する。
6. IrLAP でのデータ交換が完了した。
7. ASK でのデータ交換が完了した。
8. 一定回数 ( $N_1$  回) の試行でコネクションが確立されなかったため、ユーザに通信不能を報告して送信処理を中断する。

受信局の動作は以下の通りである。番号は図 5 中の矢印に対応する。

1. ユーザからの受信要求を受けた。
2. IrLAP でのコネクション設定が一定期間 (1500 ミリ秒) 内に完了しなかったため、送受信器のモードを ASK に設定する。
3. ASK でのコネクション設定が一定期間 (1000 ミリ秒) 内に完了しなかったため、送受信器のモードを IrDA に設定する。
4. IrLAP でのコネクションが設定されたため、送受信器のモードを IrDA に固定する。
5. ASK でのコネクションが設定されたため、送受信器のモードを ASK に固定する。
6. IrLAP でのデータ交換が完了した。
7. ASK でのデータ交換が完了した。
8. 一定回数 ( $N_2$  回) の試行でコネクションが設定されなかったため、ユーザに通信不能を報告して送信処理を中断する。

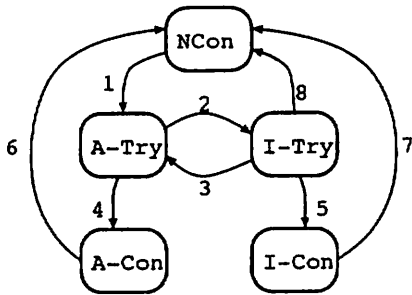


図 5: ASK プロトコルと IrLAP の自動判別における状態遷移図

### 3.3 自動判別機能の拡張 — 優先度の高いプロトコルでの接続の保証 —

送信局、受信局の双方が自動判別機能を有する場合、上記の方法では選択されるプロトコルが不定となる。あるプロトコルが、他のプロトコルより高速であるなどというように、優先度の高いプロトコルが存在する場合には、そのプロトコルでの接続を保証できることが望ましい。ASK と IrDA では最高速がそれぞれ 19.2Kbps と 115.2Kbps であるため、IrDA での接続を保証するために、以下のような拡張を行なう。

ASK の SYN および ENQ パケットにフラグを設定し、IrDA での通信が可能な場合にはそのフラグを立てて送信する。双方がフラグの立ったパケットを交換したら、ASK モードでの設定を放棄し、IrDA モードに固定して、IrLAP での通信を試みる。このようにして自動判別機能を有する局間の接続を IrLAP で行なうことを保証できる。本拡張により状態遷移図は多少変更されるが、紙面の都合で省略する。

## 4 評価

ASK と IrLAP の自動判別方式をパーソナルコンピュータ SHARP PC-6700 上に実装し、評価を行なった。以下に、評価結果を述べる。

### 4.1 実装

前章では、プロトコルを一定時間の満了によって切替えたが、実装の際には、プロトコルに応じて一連の手続き終了によって切替える方が効率が良い場合がある。本実装において、IrLAP では時間満了ではなく手続き修了による切替えを行なった。

実装した方式を簡単に示す。送信側は次の動作を  $240 (= N_1)$  回繰り返す。

1. IrDA モードにする。
2. 600 ミリ秒トラヒック検査後 discovery を実行する。成功すればコネクションを設定し、以後は IrLAP を使用する。
3. ASK モードにする。
4. ENQ を送信。
5. 500 ミリ秒 SYN を待つ。成功すれば以後は ASK プロトコルを使用する。

一方、受信側は次の動作を  $150 (= N_2)$  回繰り返す。

1. IrDA モードにする。
2. 1200 ミリ秒相手局の discovery を待つ。成功すればコネクションを設定し、以後は IrLAP を使用する。
3. ASK モードにする。
4. 500 ミリ秒 ENQ を待つ。成功すれば以後は ASK プロトコルを使用する。

IrLAP での discovery はスロット数 8、スロット間隔 110 ミリ秒で実装した。discovery の手続きで送信側は、XID[00], XID[01], ..., XID[07], XID[FF] を 110 ミリ秒毎に送信する。XID[n] はスロット番号  $n$  の XID フレームを表す。スロット番号 FF は discovery 手続きの修了を表すフレームである。受信側は XID[n] ( $n \neq FF$ ) を受信すると、 $n$  から 7 まで



の乱数を振り ( $a$  とする),  $XID[a]$  を受信直後に  $XID$ -Response を送信する。

以下では実装した機器を DUAL, ASK 機能のみを実装した機器を ASK, IrDA 機能のみを実装した機器を IrDA と呼ぶ。

## 4.2 時間見積り

受信側が既に起動されている場合に、送信側起動後プロトコルを判別し、各プロトコルが動作するまでの時間を見積もる。正しいモードで受信した場合にはエラーは起こらないと仮定する。

**送信側 DUAL, 受信側 ASK の場合** 送信側は、600 ミリ秒間トラヒックを検査した後 discovery を行ない、その後 ENQ を送信する。受信側からの SYN を受信後、通常の ASK プロトコルを起動する。この場合、もう一度 ENQ を送信する。この ENQ は受信側のリトライを促す。さらに次の ENQ から通常の手続きが始まる。受信側は、最初の ENQ を受信後 SYN を送信する。続けて、もう一度 ENQ を受信するので、リトライに入る。

discovery に要する時間は  $110 \times 8 = 880$  ミリ秒となる。通常の手続きが始まるまでにかかる時間 (ミリ秒) は、 $600 + 880 + \alpha + 500 = 1960 + \alpha$  となる。 $\alpha$  は最初の ENQ を送信してから SYN を受信するまでにかかる時間であり約 55 ミリ秒である。

**送信側 ASK, 受信側 DUAL の場合** 送信側は、SYN を受信するまで 500 ミリ秒毎に ENQ を再送する。受信側は、ASK モードで ENQ を受信すると、通常の ASK プロトコルを起動し、次の ENQ を待つ。IrDA モード時はこれらの ENQ は無視される。最初の ENQ を受信し、次の ENQ から通常プロトコルに入る場合が最小の場合で、500 ミリ秒となる。受信側が 1200 ミリ秒の IrDA モードである時に、最大三回の ENQ を聞き逃す可能性があり、次の ENQ を受信して通常の ASK プロ

表 1: 自動判別にかかる時間 (典型例)

送信側	受信側	時間 (s)
DUAL	ASK	2.035
ASK	DUAL	1.485
DUAL	DUAL	1.485

トコルに入る。従って、通常のプロトコルの開始である五回めの ENQ までの時間は 2000 ミリ秒となる。

**送信側 DUAL, 受信側 DUAL の場合** IrLAP の discovery は、受信側が  $XID[00]$ ,  $XID[01]$ , ...,  $XID[07]$  の内、どれか一つを受信すると成功する。 $XID[00]$ ,  $XID[01]$ , ...,  $XID[07]$  は 770 ミリ秒かけて送信される。一方、受信側は ASK モードが 700 ミリ秒なので、必ず、 $XID[00]$ ,  $XID[01]$ , ...,  $XID[07]$  のうちどれか一つは受信する。従って、一回目の discovery は必ず成功する。この場合、自動判別は  $600 + 880 = 1480$  ミリ秒で discovery を終了し、通常の IrLAP のコネクションのプロセスに入る。

**送信側 DUAL, 受信側 IrDA の場合** 送信側 DUAL, 受信側 DUAL の場合と同様。

## 4.3 実験結果

受信、送信の順に起動した際の実験結果として、典型的な場合 (この場合の詳細なログを付録に示す) において自動判別にかかった時間を表 1 に示す。ASK としては SHARP OZ-9520 を使用した。タイマ実装の都合上最大 55 ミリ秒の誤差があるが、ほぼ見積り通りの数字が得られている。

## 5 おわりに

本稿では、まず、空間光通信におけるプロトコルの自動判別機能の重要性とそれを実現するためのプロトコルアーキテクチャについて論じた。そして、自動判別を行なうアルゴ

リズムを提案し、提案方式を ASK プロトコルと IrDA の判別に適用した事例を与えた。さらに、実装したシステムを用いた実験を通じて、実用的な時間内で自動判別が行なえることを示した。

本方式を適用することで、ユーザは複数のプロトコルの中から相手局と通信可能なプロトコルを選択する必要がなくなる。また、移動先の異なるネットワーク環境で、安全かつ適応的にプロトコル判別を行なうことも可能となる。今後のハードウェア技術の進歩によりさらに長距離で安価なユニットが開発されると、赤外線を利用した通信のいつその普及が見込まれ、複数のプロトコルが同一のメディアを共有する状況が増えることが予想される。このため、本稿で提案した自動判別機能の重要性はますます高まってゆくものと考えられる。

## 参考文献

- [1] D. F. Brantz and F. J. Bauchot, "Wireless LAN Design Alternatives", IEEE Network, March/April 1994, pp. 43-53.
- [2] F. R. Gfeller and U. Bapst, "Wireless In-House Data Communication via Diffuse Infrared Radiation", Proc. IEEE, Vol. 67, No. 11, November 1979.
- [3] 塚本昌彦, "変わる電子文具: 可能性秘め赤外線通信", Computer Today, No.63, September 1994, pp. 46-51,
- [4] W. Diepstraten, G. Ennis, P. Belanger, DFWMAC: Distributed Foundation Wireless Medium Access Control, IEEE P802.11-93/190, November 1993.
- [5] A. J. C. Moreira, et al., "Wireless Access Method and Physical Layer Specification", IEEE P802.11-94/95,96,97, May 1994.
- [6] "ザウルス出現! Mark II", ソフトバンク社, April 1994.
- [7] Infrared Data Association, Serial Infrared Link Access Protocol (IrLAP), version 1.0, June 1994.
- [8] Infrared Data Association, Link Management Protocol, version 1.0, August 1994.
- [9] Infrared Data Association, Serial Infrared (SIR) Physical Layer Link Specification version 1.0, April 1994.

## 付録: 自動判別送信側ログ

4.3 節で述べた自動判別の実験における送信側ログ(典型例)を示す。Time 欄は時刻を表す。MODE 欄の '1' は ASK, '2' は IrDA を表す。右端の欄は動作を示す。'switch' はモード切替え動作を表す。ENQ+, SYN+ は IrDA サポートフラグのついたパケットを表す。

### 送信側 DUAL, 受信側 ASK の場合

```
Time: 15:49:51.055, MODE: 2, switch
Time: 15:49:51.660, MODE: 2, send XID[00]
...
Time: 15:49:52.540, MODE: 2, send XID[FF]
Time: 15:49:52.540, MODE: 1, switch
Time: 15:49:52.540, MODE: 1, send ENQ+
Time: 15:49:52.595, MODE: 1, recv SYN
Time: 15:49:52.595, MODE: 1, send ENQ
Time: 15:49:53.090, MODE: 1, send ENQ
Time: 15:49:53.090, MODE: 1, recv SYN
Time: 15:49:53.695, MODE: 1, send Data
Time: 15:49:53.750, MODE: 1, recv Ack
```

### 送信側 ASK, 受信側 DUAL の場合

```
Time: 15:53:49.000, MODE: 1, send ENQ
Time: 15:53:49.495, MODE: 1, send ENQ
Time: 15:53:49.990, MODE: 1, send ENQ
Time: 15:53:50.485, MODE: 1, send ENQ
Time: 15:53:50.485, MODE: 1, recv SYN
Time: 15:53:50.815, MODE: 1, send Data
Time: 15:53:50.870, MODE: 1, recv Ack
```

### 送信側 DUAL, 受信側 DUAL の場合

```
Time: 16:19:44.000, MODE: 2, switch
Time: 16:19:44.605, MODE: 2, send XID[00]
Time: 16:19:44.715, MODE: 2, send XID[01]
Time: 16:19:44.825, MODE: 2, send XID[02]
Time: 16:19:44.935, MODE: 2, send XID[03]
Time: 16:19:44.990, MODE: 2, recv XID[03]
Time: 16:19:45.045, MODE: 2, send XID[04]
Time: 16:19:45.155, MODE: 2, send XID[05]
Time: 16:19:45.265, MODE: 2, send XID[06]
Time: 16:19:45.375, MODE: 2, send XID[07]
Time: 16:19:45.485, MODE: 2, send XID[FF]
Time: 16:19:46.090, MODE: 2, send SNRM CMD
Time: 16:19:46.200, MODE: 2, recv UA RSP
Time: 16:19:46.255, MODE: 2, send RR CMD
Time: 16:19:46.310, MODE: 2, recv RR RSP
Time: 16:19:46.365, MODE: 2, send I FRM
Time: 16:19:46.695, MODE: 2, recv RR RSP
Time: 16:19:46.750, MODE: 2, send DISC CMD
Time: 16:19:46.805, MODE: 2, recv UA RSP
```