VECTOR >

はじめての車載 Ethernet

Ethernet がなぜクルマに必要？期待される役割は

vector.com

目次

発行元：ベクター・ジャパン株式会社

この内容は、アイティメディア株式会社の「MONOist 」に 2019 年 5 月から掲載されたベクター・ジャパン執筆の寄稿記事「はじめての車載イーサネット」への弊社原稿に加筆・修正したものです。

1. Ethernet[1]がなぜクルマに必要？期待される役割は

インターネット経由で誰とでもつながる時代。個人が持つ端末はワイヤレス接続が大半を占めていますが、オフィス等ではいまだに有線によるローカルエリアネットワーク（LAN）が使われています。その LAN の基盤技術の一つとして広く使われている Ethernet（および TCP/IP）が、次世代の車載ネットワーク技術として注目を浴びています。本資料ではその注目の背景・役割・規格動向から、関連するプロトコルの概要まで、幅広く解説していきます。

1.1 なぜクルマに Ethernet なのか？

衝突被害軽減ブレーキの登場から数年が経ち、現在では複数の自動運転支援機能を統合した自動運転レベル 2 に相当する自動運転支援システムの採用など、クルマの高機能化の波が止まりません。これに拍車をかけているのが皆さんも最近よく耳にする言葉「CASE」で [2]、クルマ業界に大きな変革を促す Connected、Autonomous、Shared&Services、Electric の頭文字をとったものです。クルマがより便利になるために外の世界とつながり、自動化とともに電動化が進んでいく、その対応のため、車載ネットワーク・E/E アーキテクチャが大きな変化を遂げようとしています。具体的な影響・変化として、たとえば以下のような点が挙げられます。

- 通信データ量の増大（センサーの高解像度化、セキュリティ対策）
- 分散処理から集中処理（高性能コンピューターの導入）
- 静的な構成から動的な構成（出荷後の機能追加・変更、リソースの共有）
- ネットワークの種類の削減

これはつまり、大量の情報をより速く、それを必要としている相手にだけ（ただし、ネットワークのどこに存在するかは不明、かつ、固定ではない）届ける必要が出てきたということです。将来の完全自動運転車のセンサーから送り出される情報（数 Gbps といわれています）は、現在の車載ネットワークプロトコルの主流である CAN の通信速度（CAN FD でも最大 8Mbps）[3]では、当然さばききれません。そして何より CAN は、バス上に情報をとにかく送り出し、それを必要であれば取り込むという形をとっているため、特定の相手だけを狙って情報を届けるという仕組みがないのです。そしてこの「特定の相手だけに」という点は、情報が多くなればなるほど重要になってきます。

[1] 「イーサネット」「Ethernet」は、国内では富士ゼロックス株式会社の登録商標です。

[2] 2016 年のパリモーターショーにてダイムラー社が戦略の柱として提唱したのが始まりです。順番を入れ替え ACES と言われることもあります。

[3] とはいえ、現在、通信速度 10Mbps 以上を目指した CAN XL という新しい規格の策定も進んでいます。

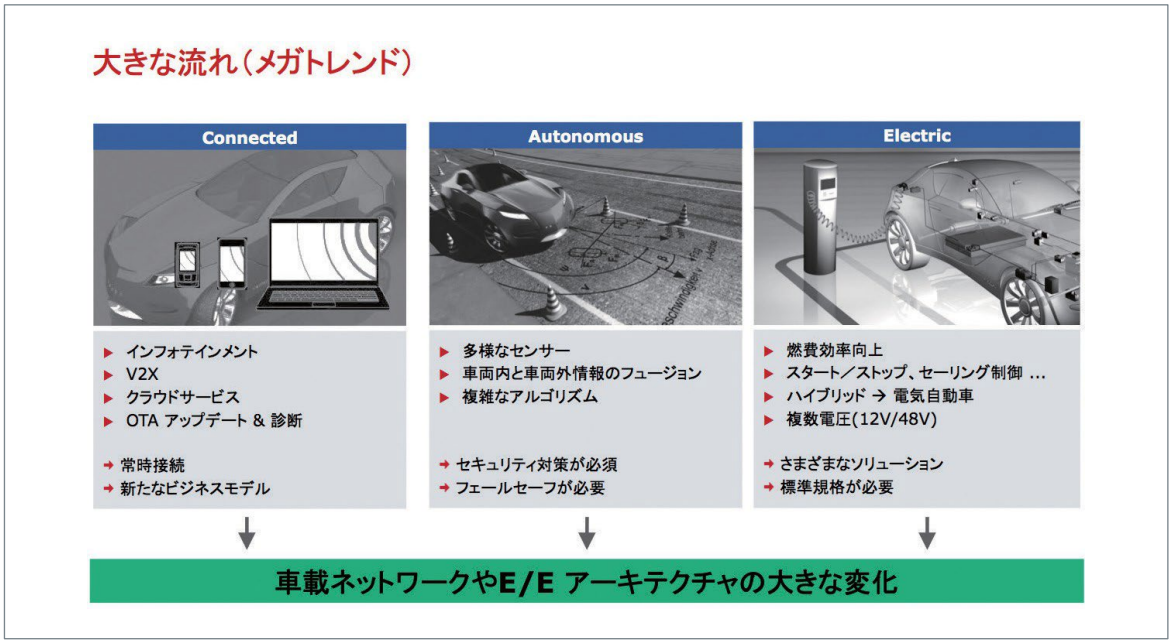

図 1

1.2 Ethernet に期待される役割（ユースケース）

上述の「大量のデータ」を「より速く」、「ネットワーク上のどこかにいる」、「（それを）必要としている相手にだけ届ける」ことは、言うなれば、クルマの中にインターネットに似た仕組みを作るということです。そうであれば、現在のインターネットの基盤技術の一つである Ethernet（および同時に使われることの多い上位プロトコルである TCP/IP）に白羽の矢が立つのは、当然の帰結でしょう。そして、

- バックボーンネットワーク
- 自動運転、ADAS
- インフォテインメント

などの役目を担うために超高速なデータ通信を提供することが、Ethernet には期待されています。以下に、これらのユースケースを支えるべく検討されている E/E アーキテクチャの例を示します。

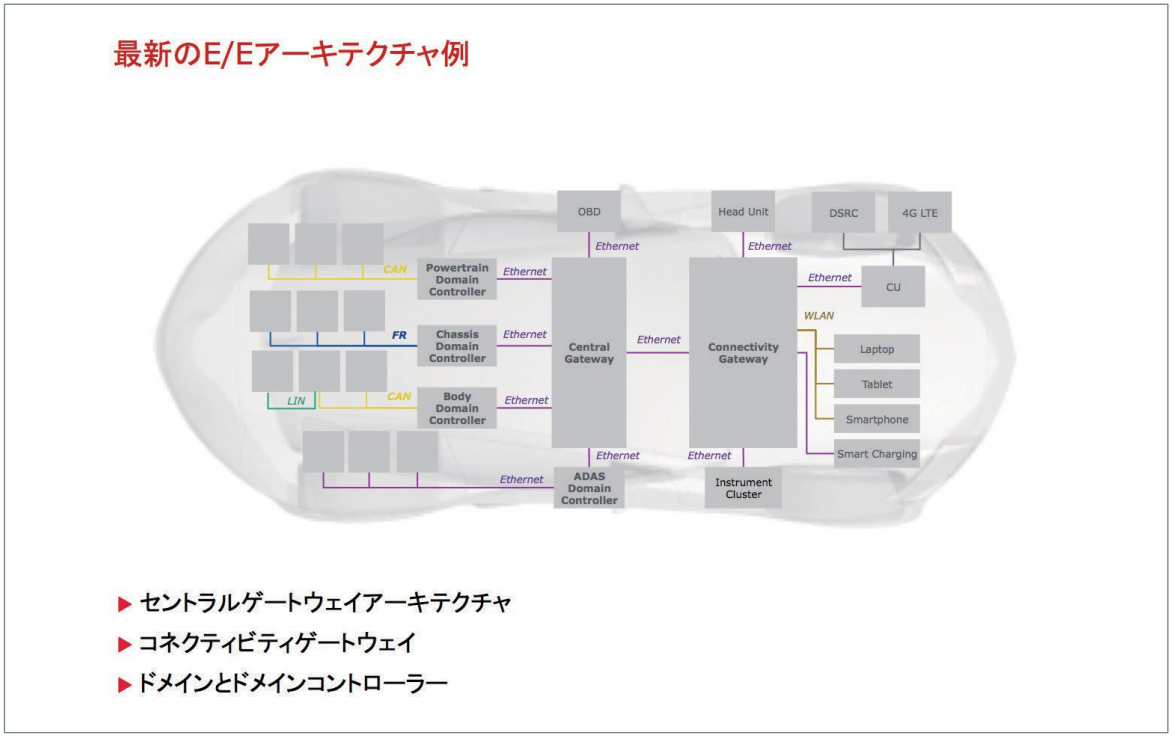

図 2 セントラルゲートウェイから各ドメインコントローラーまでを Ethernet で接続

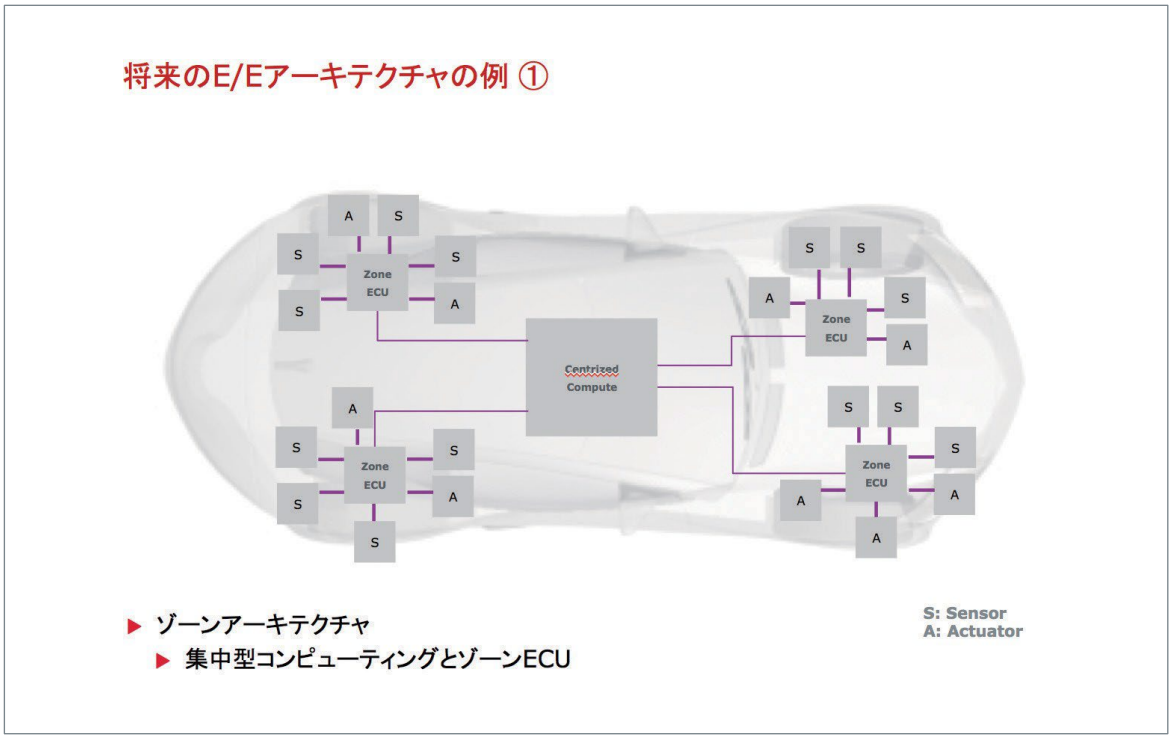

図 3 高性能コンピューターを中心に据え、各ゾーンとの間を高速 Ethernet で接続

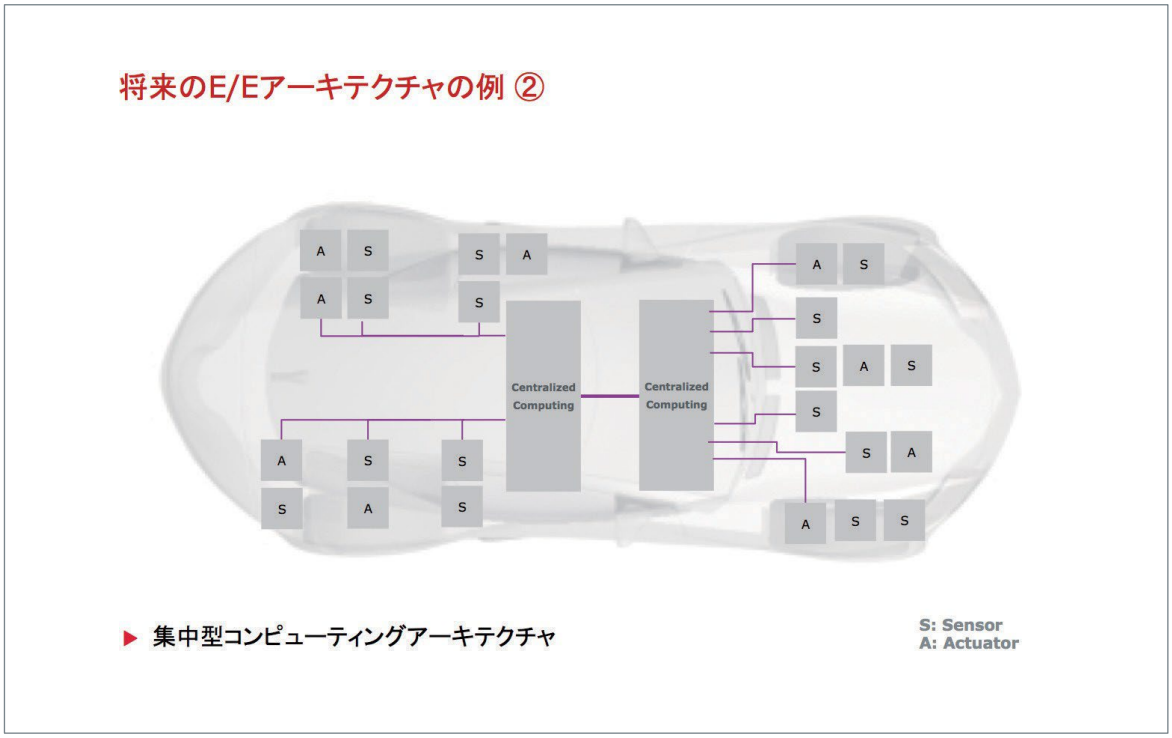

図 4　高性能コンピューターと各センサー・アクチュエータを Ethernet で直接接続

1.3 「Ethernet」と言うけれど

さて、ここまで「Ethernet」という言葉を当然のように使ってきましたが、「Ethernet」とは何なのでしょうか。たとえば、社内 LAN などを指して漫然と「Ethernet」と言ってしまうのは、ネットワークプロトコルに詳しい皆さんにとってみれば、あまり褒められた言い方ではないかもしれません。その理由は、前段落の「Ethernet（および同時に使われることの多い上位プロトコルである TCP/IP）」という一文にあります。本資料がリリースされる頃には、令和という新しい時代が始まり、平成が幕を閉じているはずですが、その平成の始まる少し前、昭和の終わりに IEEE で規格化されたのが IEEE 802.3 CSMA/CD（Ethernet）[4]です。これはバス型接続を同軸ケーブルによって構成したネットワークの規格であり、現在の LAN では、これをベースにスイッチ（スイッチングハブ、レイヤー2 スイッチ）により構成したネットワークに、上位プロトコル TCP/IP を組み合わせて使っているのが一般的です。そのため、いわば、LAN≒Ethernet +TCP/IP を「Ethernet」と呼んでしまっているということが少なからずあると思いますが、本来は一部を指している言葉なのです。

[4] 出典：IEEE 802.3 CSMA/CD（Ethernet）Working Group, March 9, 2006
（https://www.ieee802.org/3/an/public/mar06/iso_2_0306.pdf）の表記より

1.4 OSI 参照モデル [5]と PDU（PCI/SDU）

では「Ethernet」と「（上位プロトコル）TCP/IP」というのは何が違うのでしょうか。それを理解するための考え方として「OSI 参照モデル」というものがあります。これは、OSI（Open System Interconnection）の名が示すように、異なるコンピューターネットワーク・通信システム間での情報のやりとりを実現するための通信機能の構造を示したもので、通信に必要とされる機能を 7 つの階層（レイヤー）に分割したモデルです。階層ごとに担う役割を定めている（上位層は下位層からもたらされる実データを扱う）ので、「各通信プロトコルが、どの階層に相当するか」「どの階層で切り口を合わせるか」を考えやすくなります。以下の図は、それぞれの階層の名称と簡単な説明、それと本資料で取り上げる予定のプロトコルの対応を示します。ただし、あくまで“参照”なので厳密な定義ではなく、この枠に収まらないものも多くある点も、記憶に留めておいていただければと思います。

図 5 OSI 参照モデルの各層と車載アプリケーションで使われるプロトコル

[5] ISO/IEC 7498:1984 として規格化されています（2020 年 7 月時点では ISO/IEC 7498-1:1994）。

図 5 をご覧いただくと、Ethernet は第 1 層フィジカルレイヤーと第 2 層データリンクレイヤーに、IP は第 3 層ネットワークレイヤー、TCP/UDP は第 4 層トランスポートレイヤーに相当していることがわかります。つまり、Ethernet と TCP/IP は担っている役割（階層）が異なるということです。フィジカルレイヤーは物理媒体上で情報をやりとりするための取り決め、たとえば、どのような媒体を使うか、その媒体上でビット(0/1)をどのように表わすのか、どの位のスピードで行うのか、どのような単位でやりとりをするのか等の部分を担っています。データリンクレイヤーではフィジカルレイヤーが定めた単位（ビット等の塊）にどのような意味を持たせ、情報の単位を定め、それとともに、それをやりとりするための必要な識別子等を定める役割を担っています。そして、ネットワークレイヤーはデータリンクレイヤーが運んできた情報を取り出し、状況に応じて異なる通信経路に渡す役割を担っています。さらにその上のトランスポートレイヤーは、ネットワークレイヤーの運んできた情報の統合やエラー検出、それに伴う再送などを行います。このように、階層ごとに役割があり、そこでの通信の際の約束事がありますが、その約束事・通信手順（複数の階層にまたがることもある）のことをプロトコル（Protocol）といいます。先ほどの「上位プロトコル」という言葉の意味は、この OSI 参照モデルにおける「上位」であり下位プロトコルの扱う情報のうち、自プロトコルにとって有用な部分だけを扱う存在ということです。

ここでいう「下位プロトコルの扱う情報のうち、自プロトコルにとって有用な部分」とは、どういうことなのでしょうか。実は、階層化されたプロトコルにおいては、上位から下位に行くにしたがって制御情報が付け足され、少しずつそのサイズが大きくなります（ 逆に言えば、上位に行けば行くほど一度に運べる実データのサイズは小さくなります）。つまり、特定のプロトコルが扱う情報の中には、そのプロトコルでの制御情報と、そのプロトコルが運び上位プロトコルに渡す実情報に分かれているということです。前者を PCI（Protocol Control Information）、後者を SDU（Service Data Unit）と呼び、両者を併せたものを PDU（Protocol Data Unit）と呼びます。これが階層ごとに入れ子になっているということです。

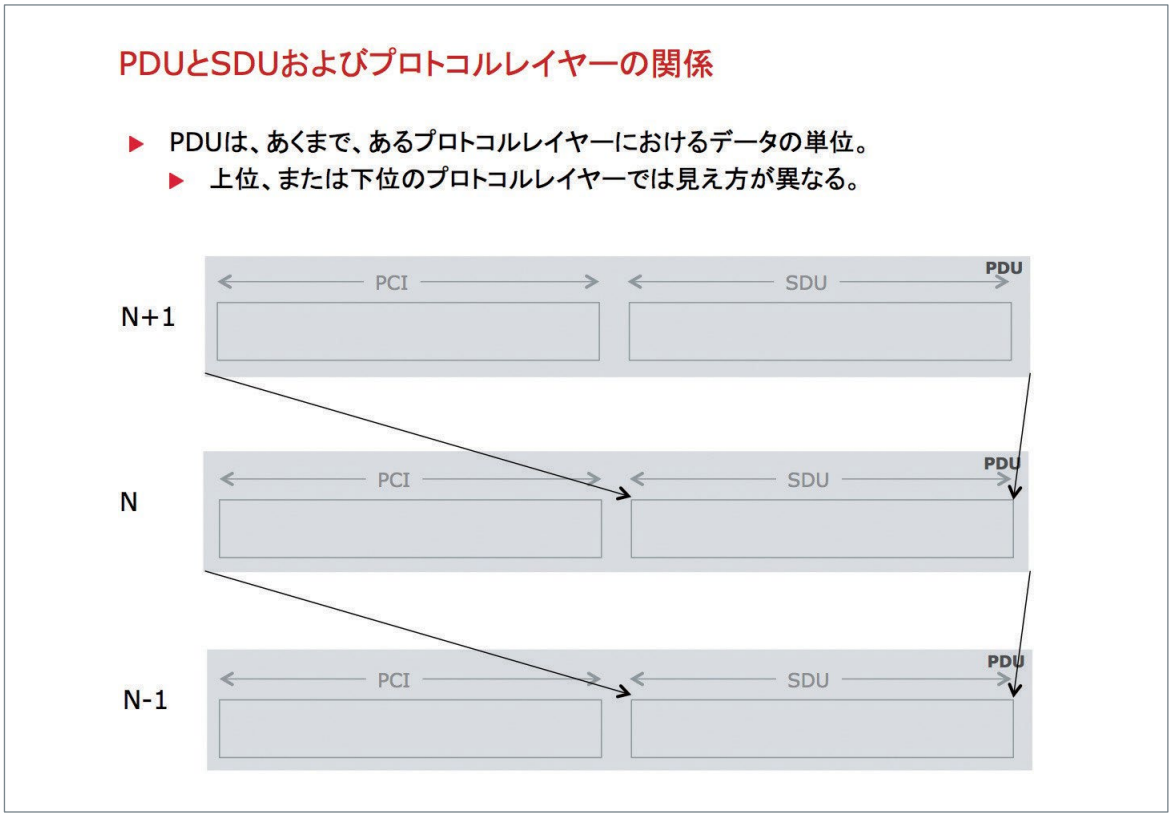

図 6 PDU の構造とプロトコルレイヤー

これを私たちの生活に例えるのであれば、宅配便で送りたい内容物とそれを入れる箱、そして複数の箱を行き先別に入れたバスケット、複数のバスケットを積んだトラックというイメージです（もしくは、マトリョーシカ人形の方がわかりやすいかもしれません）。なお、入れ子構造であるということは、今後本資料の中でも同じ概念を示す言葉が表現を変えて繰り返し出てくること（代表的なのは MAC アドレス、IP アドレス、ポート）と、その言葉が示す情報を直接扱うのは原則その階層においてのみであるということを示しています。これは筆者自身の経験から、Ethernet および TCP/IP を学んでいくときに引っかかりやすいポイントだと思っています。もしかすると、概要の把握という目的であれば「下位プロトコルは上位プロトコルの情報を運ぶためのインフラである」くらいに開き直って理解してもよいかもしれません。

1.5 まとめ

ここまで、Ethernet が車載ネットワーク技術として注目されている背景や、その期待される役割、Ethernet（および TCP/IP）と OSI 参照モデルの関係や PDU の構造を紹介してまいりました。第 2 章以降は、OSI 参照モデルを、一番下のフィジカルレイヤーから順にお話ししていきます。なお、途中の第 3 層（IP）と第 4 層（TCP/UDP）は車載特有というわけではありませんが、SOME/IP・DoIP といったクルマで使われる上位プロトコルが、これらの階層・プロトコルに依存していますので、こちらについては、第 4 章で紹介していきます。

2. 車載 Ethernet のフィジカルレイヤーはどのようになっているのか

前章では Ethernet が車載ネットワーク技術として注目されている背景や、その期待される役割、Ethernet（および TCP/IP）と OSI 参照モデルの関係や PDU の構造を紹介しました。第 2 章では OSI 参照モデルの一番下、Ethernet におけるフィジカルレイヤーについてお話していきます。

2.1 車載ネットワークにおけるトポロジー

従来の車載ネットワークプロトコルと Ethernet（および TCP/IP）が、プロトコルそのもの以外にも、さまざまな面で異なりそうだということは、皆さんもすでに気づかれていると思います。そのうちの一つにネットワークトポロジーの違いというものがあります。ネットワークトポロジーは、誤解を恐れずに言えば、ネットワークのつながり方の形です。つまり、通信をする主体であるそれぞれのノード（車載では ECU）が通信媒体を通じてどのようにつながっているかを示すものです。従来の車載ネットワークプロトコルにおける代表的なトポロジーとして、Bus 型、Ring 型、Active Star 型などが挙げられます。それぞれ特徴があり形も違いますが、「ノード同士が通信媒体を介して直接つながっている」というひとつの共通点があります。これはネットワーク上のどこか一点を計測することで、そのネットワーク上のやりとりがすべて把握できるということを意味しています。一方、現在の Ethernet では、図 8 のようにスイッチを経由したつながり方（スイッチ型トポロジー）が主流になっています。これは図 7 のトポロジーとは異なり、原則 1 対1でつながっているため、どこか一点を見ただけでは、すべてを把握することはできません。また通信媒体上の信号波形からだけでは通信内容を把握することは非常に難しくなっています（理由は後述します）。つまり、従来とは計測の仕方が異なってくるという点を記憶に留めておいていただければと思います。

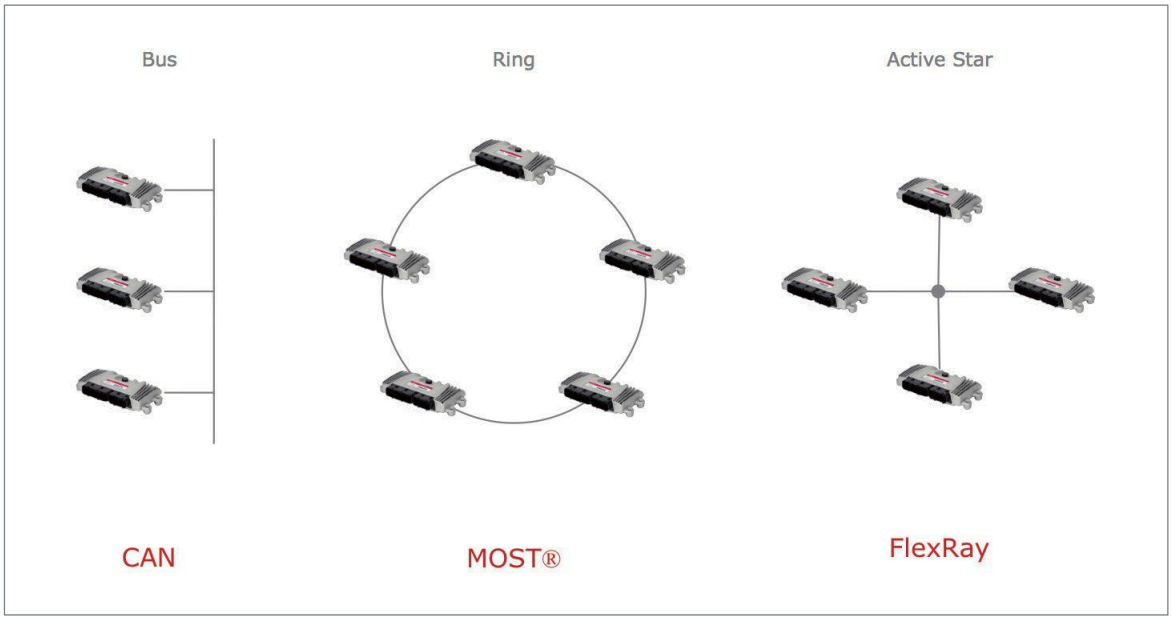

図 7 車載ネットワークプロトコルの代表的なトポロジー

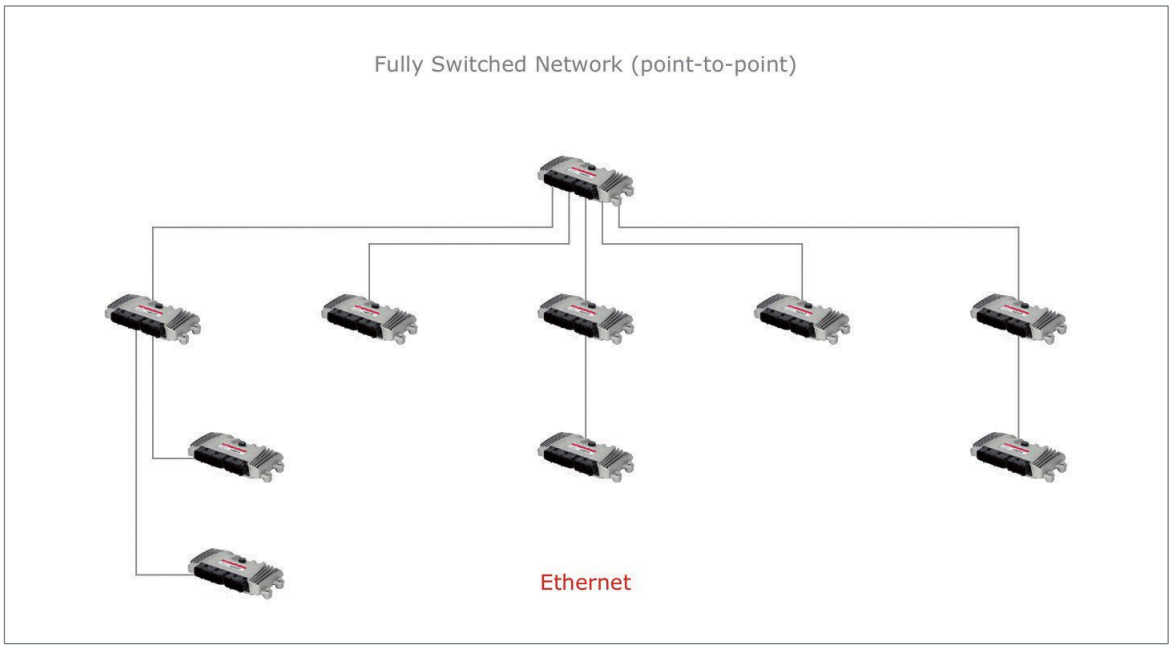

図 8　スイッチ型トポロジー（複数接続を持つものはここではスイッチと考えます）

2.2 車載 Ethernet におけるフィジカルレイヤー

さて、ノード同士をつないでいるフィジカルレイヤーですが、ご存じのとおり、家庭やオフィスの LAN で使われているものには複数あります。同様に「車載（クルマで使われている）」という観点でも複数あり、大きくは「クルマの中で使われるもの」と「クルマと外をつなぐために使われるもの」に分けられます。

- クルマの中で使われるもの
  - 100BASE-T1、1000BASE-T1（車内通信）
- クルマと外をつなぐために使われるもの
  - 100BASE-TX、1000BASE-T（診断通信）
  - ISO 15118（充電通信）

双方ともクルマにおける通信の新しいユースケースに対応するために導入されたものではありますが、前者の方が前回お話した車載ネットワークそのものに対する新たな要求に応えるためのものである、という点が色濃く表れていますので、ここではそちらに重きを置いて説明していきます。

2.3 PHY という存在

それぞれのフィジカルレイヤーのお話をする際、避けて通れないのが PHY[6]（Ethernet PHY）と呼ばれる部品です。これは、ECU の中にあるマイクロコントローラーと通信媒体（物理媒体、銅線、光ファイバー等）との間を取り持つもので、マイクロコントローラーから出てきたパラレル信号を特定の通信媒体上にシリアル信号として送り出す、またはその逆を行う存在です。CAN の場合だと、CAN コントローラーが扱う TXD/RXD を CANH/CANL に変換（ただし、どちらもシリアル信号）する CAN トランシーバーがそれに近い存在です。ただ、PHY の場合、Ethernet の高速通信を支えるために、パラレル信号をそのままシリアル信号にしているわけではなく、符号化・スクランブリング[7]を行っています。この結果、Ethernet の場合、先に述べたとおり、通信波形を見ただけでは通信内容がわからなくなっています。それはさておき、PHY はマイクロコントローラーと通信媒体の間にありますので、それぞれとのインターフェイスを持っており、前者を MII（Media Independent Interface）、後者を MDI（Medium Dependent Interface）と呼びます。MII は通信媒体に依存しないインターフェイスで、送受信それぞれに 4 ビットのパラレル信号を 25Mhz でマイクロコントローラーとやりとりをしており、100Mbps での通信が可能です。バリアントとして 1Gbps の通信が可能な GMII[8]もあり、これは 8 ビットの信号を 125Mhz でやりとりしています。一方、MDI は通信媒体ごとに異なったインターフェイスであり、ペア線（一対、複数対）を用いるのか、光ファイバーを用いるのか等によって違ってきます。もちろん MDI が変われば MII との変換方式も変わりますので、PHY の内部構造も変わってきます。（当たり前ですが）通信媒体ごとに別々の規格・PHY があるということです。では、次にそれらを紹介していきます。

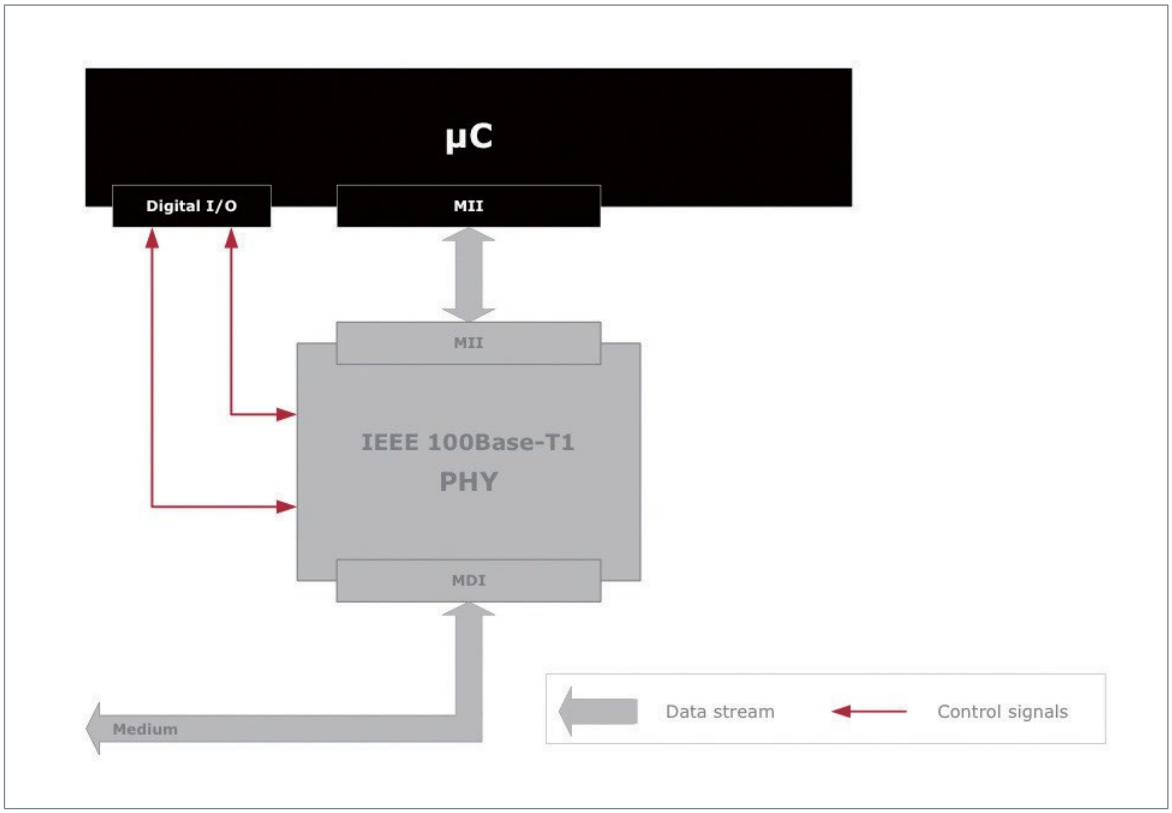

図 9 ECU の内部構成

[6] PHY：フィジカルレイヤー（PHYsical layer）の先頭三文字を取ったものです。

[7] 元の信号をランダムに入れ替え、同じようなパターンが連続しないようにすること。外部に対するノイズを低減することを狙っています。

[8] GMII：G は Giga の略。

2.4 クルマの中で使われるもの

2.4.1 100BASE-T1(IEEE 802.3bw)

これは Broadcom 社が開発した技術 BroadR-Reach を基にした仕様 OABR[9]を、IEEE で規格化したものです。特徴は、1 対のペア線 [10](CAN と同様)で最大 100Mbps の通信を、全二重 [11]で行える点です。CAN が最大 1Mbps(CAN FD では最大 8Mbps)の半二重通信だったことを考えると、飛躍的な速度の向上です。これは、作動信号(BI_DA+、BI_ DA-)の採用のみならず、高度なデータ符号化処理(4B3B、3B2T、PAM3)によりデータを圧縮、通信周波数を下げ、ノイズを抑えることで、実現しています。規格ではペア線のシールドの有無にかかわらず最大 15m[12]までとされていますが、実際の車載環境では狭い場所に配線するための追加のノイズ対策が必要なようで、STP の採用や平行線長に制限をかける等、各社ごとに工夫 [13]をされているようです。

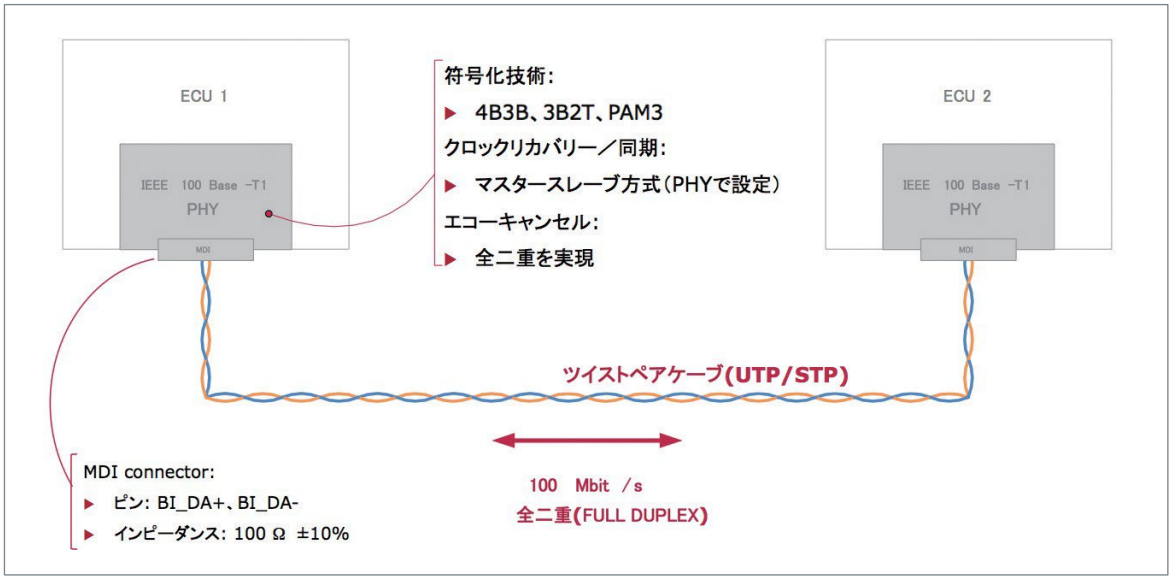

図 10 100BASE-T1

[9] OPEN Alliance BroadR-Reach、OPEN は One-Pair Ether-Net。www.opensig.org/abut/about-open/

[10] IEEE 規格(IEEE 802.3.bw)では"Balanced Twisted Pair Cable"とあり、Unshielded Twist Pai(rUTP)と Shielded Twisted Pai(rOABR では、UTP を念頭に置いています。

[11] 全二重と半二重: ノードから見たときに送受信を一つの媒体で同時に行うことができるものを全二重、送信もしくは受信のいずれかのみ可能なものを半二重といいます。身近なもので例えるとすれば、電話が全二重、トランシーバーは半二重です。

[12] 通信できればよいのであれば 100m くらいまで延ばせるといわれています。

[13] IEEE が主催している Ethernet & IP Automotive Technology Day などでさまざまな研究・事例発表があります。

また符号化は、

> MII からの 4 ビット信号を 3 ビットごとに切り出す 4B3B 変換 [14]
> 切り出した3ビットを三つの状態を持つシンボル（Ternar y Symbol）二つに置き換える 3B2T 変換 [15]
> シンボルを三つの値（-1、0、1）を持つ矩形波として出力する PAM3

の三段階になっています。特に 3B 2 T 変換は、3 ビット（三回のやりとりが必要）を 2 つのシンボル（二回のやりとりで済む）に置き換えることで、通信周波数を 2/3 に削減しています（100Mhz->66.67Mhz）。

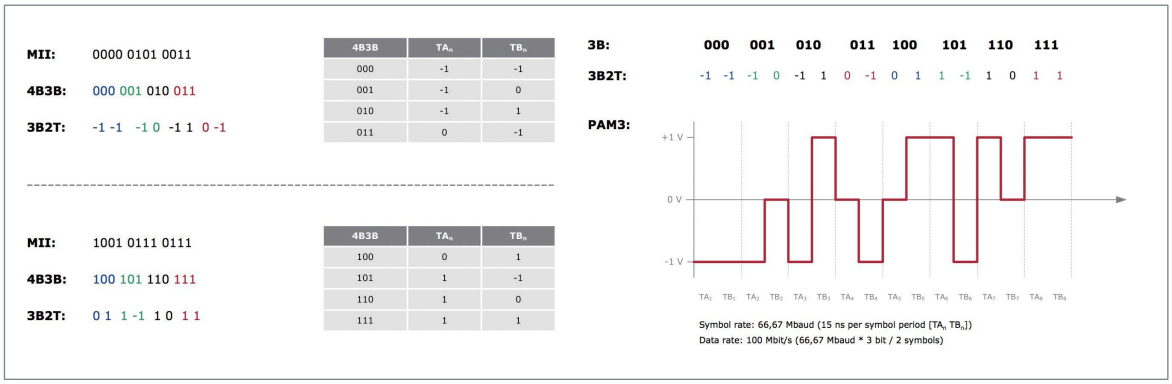

図 11 100BASE-T1 での符号化

なお、双方のノードから PAM3 で出力された電圧はバス上では足し合わされた形で観測されますが、双方の受信処理でそれぞれの出力値を差し引くこと（エコーキャンセル）によって、相手側が何を出しているかを検出する仕組みを持っています。

2.4.2 1000BASE-T1（IEEE 802.3bp）

これは、100BASE-T1 の後に策定された 1Gbps 対応の車載向けフィジカルレイヤーです。100BASE-T1 同様に 1 つのペア線で 1Gbps を達成しています。符号化方法も共通点が多く、100BASE-T 1 の 4B3B 変換の代わりに 80B81B 変換を行いますが、以降の処理は同じ 3B2T と PAM3 を採用しています。80B81B 変換は、GMII の扱う 8 ビットのデータを 10 組まとめたうえで 1 ビット追加する仕組みです。

なお、ここまでお話した二つの規格以外にも、通信媒体としてプラスチック光ファイバーを用いる策定済みの 1000BASE-RH（IEEE 802.3bv）や、規格策定中のものとして CAN の置き換えを見据えているといわれる 10Mbps の 10BASE-T1S（IEEE 802.3cg）[16]、増え続けるデータに対応するためのマルチギガ対応（2.5、5、10Gbps、IEEE 802.3ch）のものもあり、今後の動向から目が離せません。

[14] この後スクランブリングが行われます。

[15] 3 ビットは 23 ですから 8 つの状態を取れます。それを 32、9 つの状態に割り当てるということです。

[16] その後標準化が進み、2020 年 2 月に IEEE から正式に発行されています。

2.5 クルマと外をつなぐために使われるもの

続いて、クルマと外をつなぐものとして採用されているフィジカルレイヤーについて、お話していきます。これらは、ここまで紹介したものと違い、車載を目指して開発されたものではありません。クルマと外をつなごうと考えた際に、すでに存在しているテクノロジーをうまく取り入れたと考えていただけるとよいでしょう。

2.5.1 100BASE-TX/1000BASE-T

これらは診断通信に使われるもので、診断テスターと診断 GW と呼ばれる ECU をつなぎます。いわゆる普通の Ethernet 規格で、皆さんのオフィスや家庭で使われているものと同じです。一般的に診断通信というと、やはり CAN を使ったものが思い浮かびますが、ECU の高機能化に伴うソフトウェアの増加[17]やネットワークの変化に合わせて、より長い期間使えるネットワークテクノロジーとして Ethernet が着目されたという背景があるようです。実際の診断通信には、上位レイヤープロトコルとして、本連載の最後の方で紹介する DoIP（Diagnostic over IP、ISO 13400）を併せて使います。その ISO 13400 は 100BASE-TX を念頭において規定されていますが、現実世界では、より高速の 1000BASE-T も使われているようです。100BASE-TX は比較的古いものなので、ペア線二対で 100Mbps の全二重通信ですが、1000BASE-T は 1 対のペア線ごとに 250Mbps の全二重通信、それを四対使って 1Gbps を実現しています。

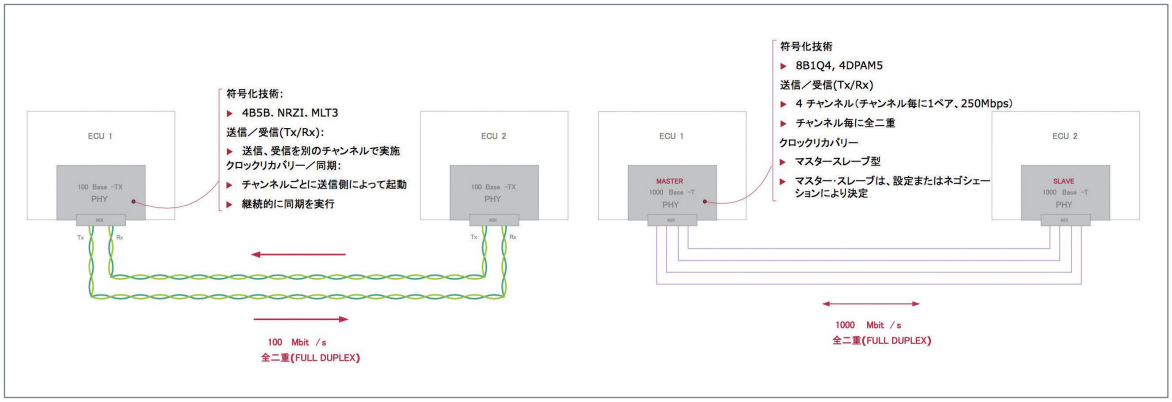

図 12 100BASE-TX と 1000BASE-T

2.5.2 ISO 15118（IEEE 1901）

これは、電気自動車の急速充電に使われるもので、かなり異色です。日本国内では CAN をベースにした CHAdeMO というプロトコルが有名ですが、ISO 15118 ではフィジカルレイヤーに PLC（Power Line Communication、電力線搬送通信）の規格 IEEE 1901 を採用しています。PLC というのは、もしかすると覚えていらっしゃる方もいるかと思いますが、Wi-Fi が普及する少し前に一時期、話題となった「家のコンセントを家庭内 LAN にしてしまう」技術です。つまり、電力を通している電線を通信媒体として使うものです。すでに電力（たとえば、国内だと 100V 15A）という大きなノイズが載っているものなので、通信は OFDM（Orthogonal Frequency Division Multiplex、直交周波数分割多重化）と呼ばれる変調方式を採用しています。なお、本規格の場合、PLC といっても実際の電力線に信号を載せているのではなく、Control Pilot という制御用の別線に載っている矩形波の上にデジタル変調した信号を重畳しています。

[17] CAN では書き換えに時間がかかり過ぎるようになってきました。

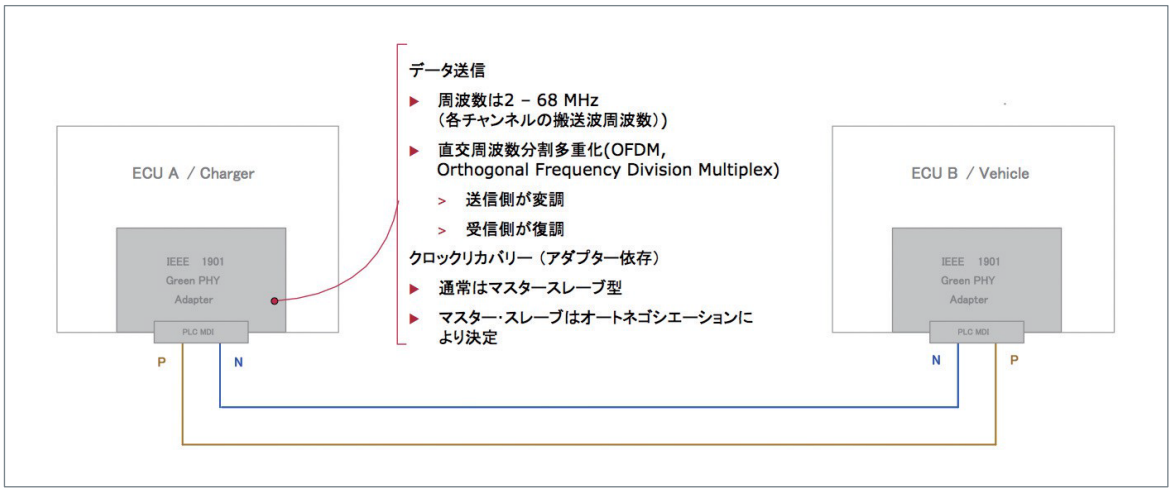

図 13 ISO 15118(IEEE 1901)

2.6 まとめ

第 2 章では、Ethernet の導入に伴うネットワークトポロジーの変化から始まり、現時点で車載(車の中で)使われつつある IEEE 100BASE-T1 をはじめとしたフィジカルレイヤーを簡単に紹介しました。第 3 章では、このフィジカルレイヤーの上、データリンクレイヤー(Ethernet フレーム)とネットワークレイヤー(IP)について書いていきたいと思います。

3. なぜ MAC アドレスと IP アドレスがあるのか？その違いは？

前章では、Ethernet の導入に伴うネットワークトポロジーの変化から始まり、現時点で車載（車の中）で使われつつある IEEE 100BASE-T1 をはじめとしたフィジカルレイヤーを簡単に紹介しました。第 3 章では、このフィジカルレイヤーの上、データリンクレイヤー（Ethernet フレーム）とネットワークレイヤー（IP）について書いていきます。

3.1 Ethernet フレーム

まずは、Ethernet フレームです。これは OSI における第二層データリンクレイヤーにおける PDU です。フィジカルレイヤーを通して送られてきたビット列をまとめて、意味付けをしたものです。構造については、後ほど説明します。

3.2 MAC（Media Access Control）アドレス

Ethernet に接続するインターフェイス（PHY）に割り当てられる固有の 48 ビット（6 バイト [18]）の数値、Ethernet フレームを送り届けるためのアドレス（レイヤー2 アドレス）です。それぞれのインターフェイスを持つノード [19]を識別し、宛先等を示すのに使われます。

6 バイトのうちの先頭バイトの Bit0 と Bit1 は特別な意味を持っています。特に Bit1 が 1[20]の場合、グローバル MAC アドレスという全世界で唯一のもので、インターネットにつながる機器はグローバル MAC アドレスが割り当てられています。グローバル MAC アドレスの先頭 3 バイトは OUI（Organizationally Unique Identifier）と呼ばれ、その MAC アドレスを持つ装置を製造しているベンダーを示す識別子になっています。（図 14 参照）

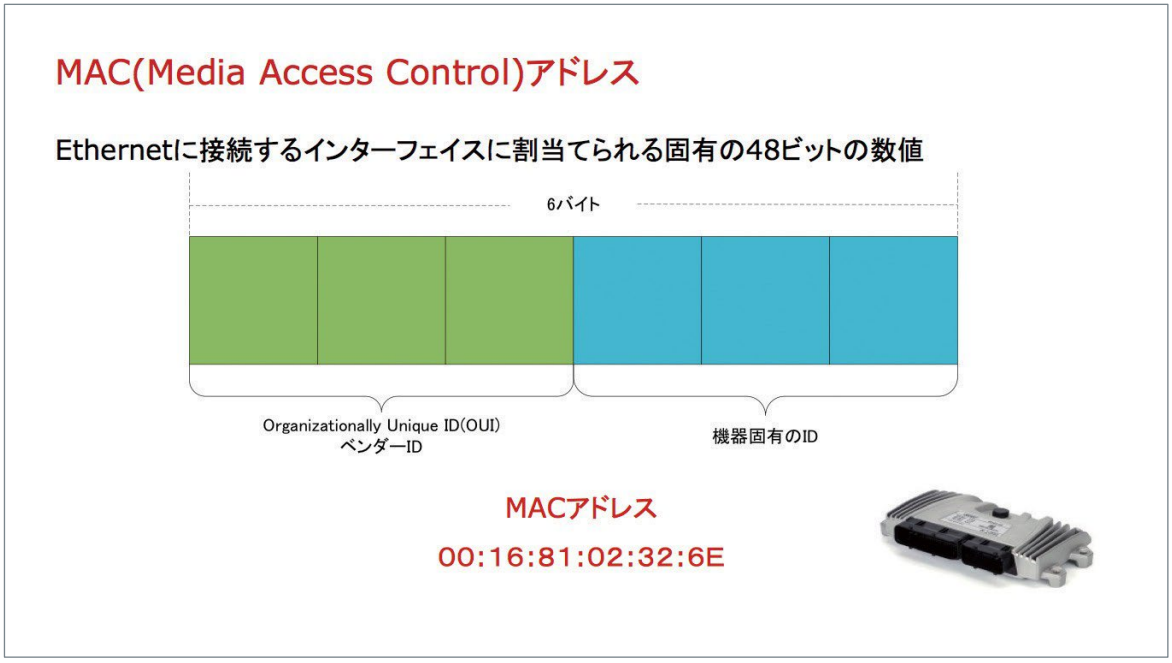

図 14

[18] ネットワークの世界では 8 ビットのことをオクテットといいます。現在では 1 バイト=8 ビットが定着していますが、過去においては必ずしもそうではなかったため、厳密に 8 ビットを示す単位としてオクテットを用いているそうです。

[19] 物理ネットワークノードともいいます。

[20] Bit1 が0の場合はローカル MAC アドレスとなりユーザーが自由に設定できることになっていますが、一般的には使われていません。

3.3 Ethernet フレームの構造

先に書きましたとおり、PDU は PCI と SDU に分けられます。PCI 部分はヘッダーと呼ばれ、宛先を示す DA（Destination Address）、送信元を示す SA（Source Address）、SDU に相当するペイロードの内容を示すタイプの 3 つの部分から成っています。DA を All1 にすると、ネットワーク上のノードすべてに宛てたフレームとなります。この場合のアドレスを、ブロードキャスト（Broadcast）アドレスと呼びます。Ethernet フレームを受信した各ノードは、DA を見て自分宛のフレームであるかどうかを識別し、DA が自分の MAC アドレスまたはブロードキャストアドレスに一致した場合には受信しますが、それ以外の場合には廃棄してしまいます。Ethernet フレームをいろいろな荷物を運ぶためのコンテナに例えると、ヘッダーはコンテ ナの行き先や中身を示すための書式で、コンテナの中身がペイロードです。ここでのポイントは、コンテナの中身（ペイロード）を確認する手間をかけずにコンテナを届けるための仕組みがあるという点です。ヘッダーにあるタイプがそのための情報を提供しています。タイプ値が 1536（0x0600）以上のもののうち、図 15 にある 3 つが今回関連するものです（特に 0x0800 と 0x86DD）。

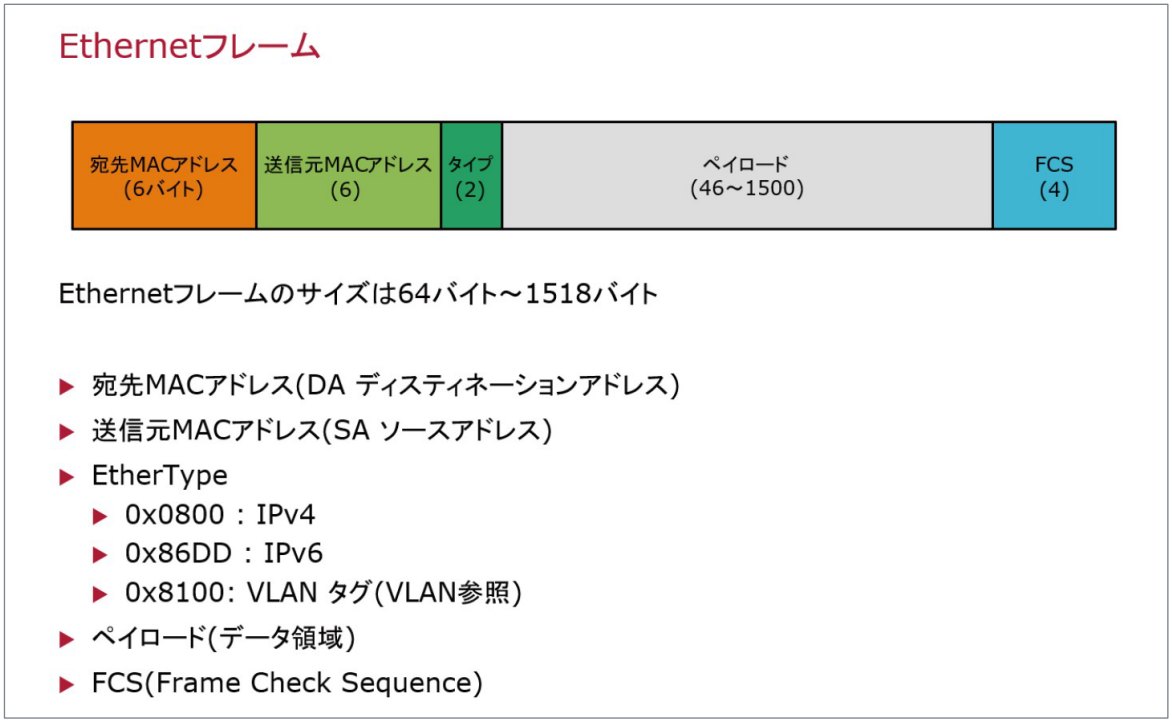

図 15

3.4 スイッチによる転送

さて、前回トポロジーのところで触れましたが、現在の Ethernet ではスイッチを経由してつながるスイッチ型トポロジーです。スイッチ [21]とは複数の Ethernet 接続を持ち、それぞれから送られてくる Ethernet フレームの MAC アドレスを解析し、適切な相手に転送する機能を持っています。スイッチの持つ接続の口のことをポートと呼びます。なお、このように MAC アドレスを基に Ethernet フレーム（レイヤー2PDU）を振り分けるので、ここでのスイッチはレイヤー2 スイッチ [22]と呼ばれることもあります。各ポートにどの MAC アドレスを持つノードがつながっているかを動的に学習していきます。また接続を静的に設定する機能を持つものもあります。

[21] 皆さんのオフィスにもある Ethernet コネクタ（RJ45）が複数接続できるようになっている装置です。

[22] 「スイッチングハブ」と呼ばれることもあります。なお、MAC アドレスを見ることなくすべてのポートに無条件に Ethernet フレームを転送するものを「リピータハブ」と呼びます。

3.5 フラッディング

とはいえ、立ち上がったばかりのスイッチは、どのポートにどの MAC アドレスを持つノードがつながっているかは、わかりません。そのため、次に示すような Ethernet フレームのやりとりを通して、徐々に学習していきます。

1. ポートから Ethernet フレームを受信すると、そのフレームの SA（送信元アドレス）をそのポートにつながるノードの MAC アドレスとして MAC アドレステーブルに登録します。
2. 同時に DA（送信先アドレス）を解析し、そのアドレスを持つポートがあればそこに転送しますが、そうでない（送信先不明の）場合には、受信ポート以外のすべてのポートに転送します。この動きをフラッディングといいます。
3. フラッディングによって転送されたフレームは、各ノードに届きますが、ここで DA が各ノードの MAC アドレスに一致しなかった場合には、そこで破棄されます。一方、一致した場合には受け入れられ、何らかの応答が送り返されます。
4. スイッチは、1 の時と同様に送り返されたフレームを受信したポートに、そのフレームの SA をそこにつながるノードの MAC アドレスとして登録、DA を解析し 1 で登録したポートに転送します。
5. この 1～4 を繰り返し、ポートと MAC アドレスの関係をすべて把握していきます。

この学習の結果により、スムースな Ethernet フレームの転送が実現されているのですが、実はこれには弱点があります。それは、スイッチが扱う MAC アドレステーブルに登録できる数が限られており、また一定時間が経つと MAC アドレステーブルから登録が削除されるという[23]点です。図 16-1 と図 16-2 に示したような小さなネットワークであれば、すべてを登録できるかもしれませんが、ネットワークにつながるスイッチが増えていけば、新たな MAC アドレスを持つ Ethernet フレームが増えますので、そのうち MAC アドレステーブルに登録できなくなり、フラッディングによる無用なトラフィックが発生し、通信の効率が極端に低下してしまいます。つまり、スイッチだけで構成されたネットワークを無制限に大きくすることは現実的ではないということです。

[23] 継続的に通信を行っていれば削除されません。

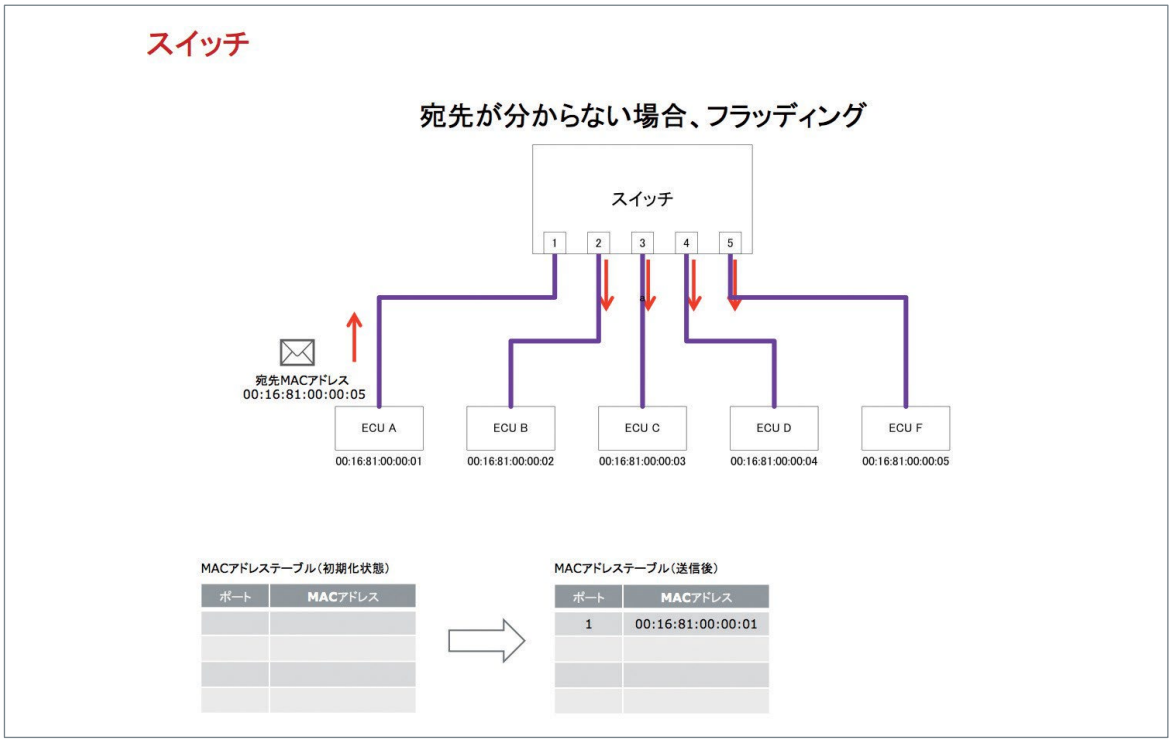

図 16-1

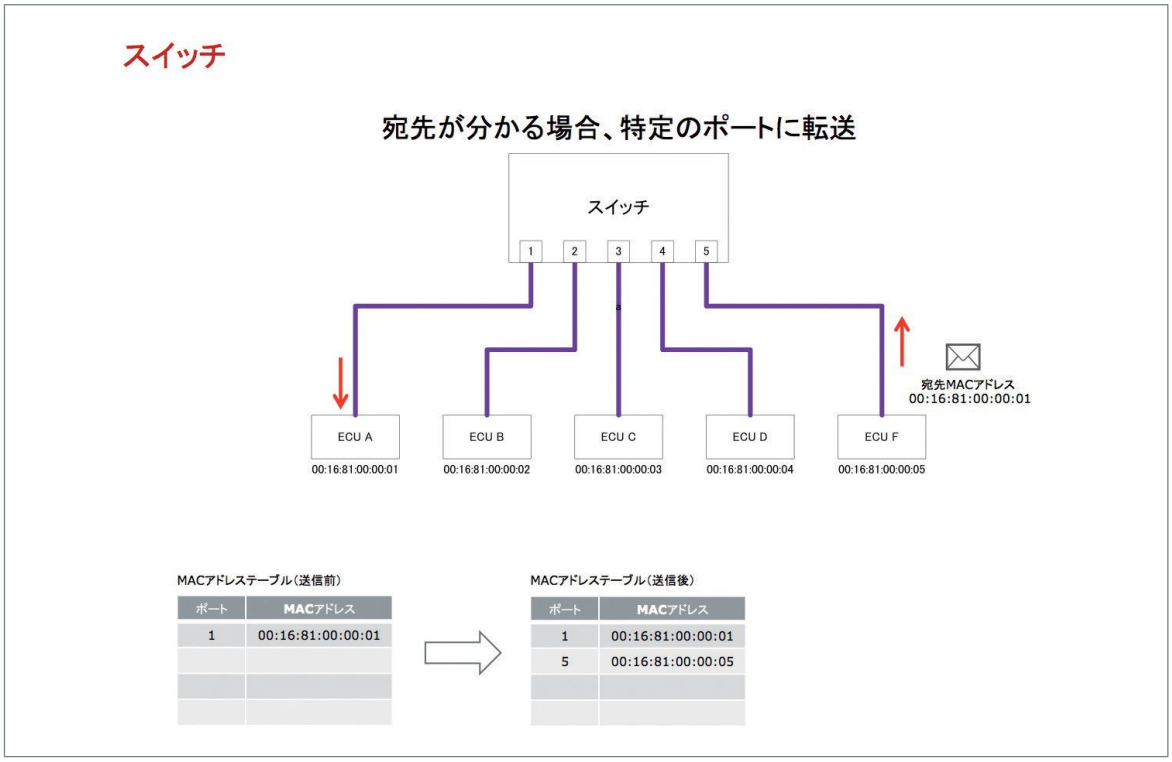

図 16-2

3.6 VLAN（Virtual LAN）

これは、スイッチで構成されたネットワーク内に論理的な壁を立てるための仕組みです。ブロードキャストで送信されたフレームは VL AN の範囲外には届けられません。トラフィック制御やセキュリティの観点で導入されることがあります。1 つのスイッチだけでできているネットワークであればスイッチのポートに VL AN を設定するだけで済みますが、スイッチをまたがる VLAN の場合には VL AN タグを追加することで、その Ethernet フレームがどの VL AN に属するかを、受け取るスイッチに示します。VL AN タグは通常スイッチが扱うので、各ノードでは気にすることはありません。（図 17-1、図 17-2、図 17-3 参照）

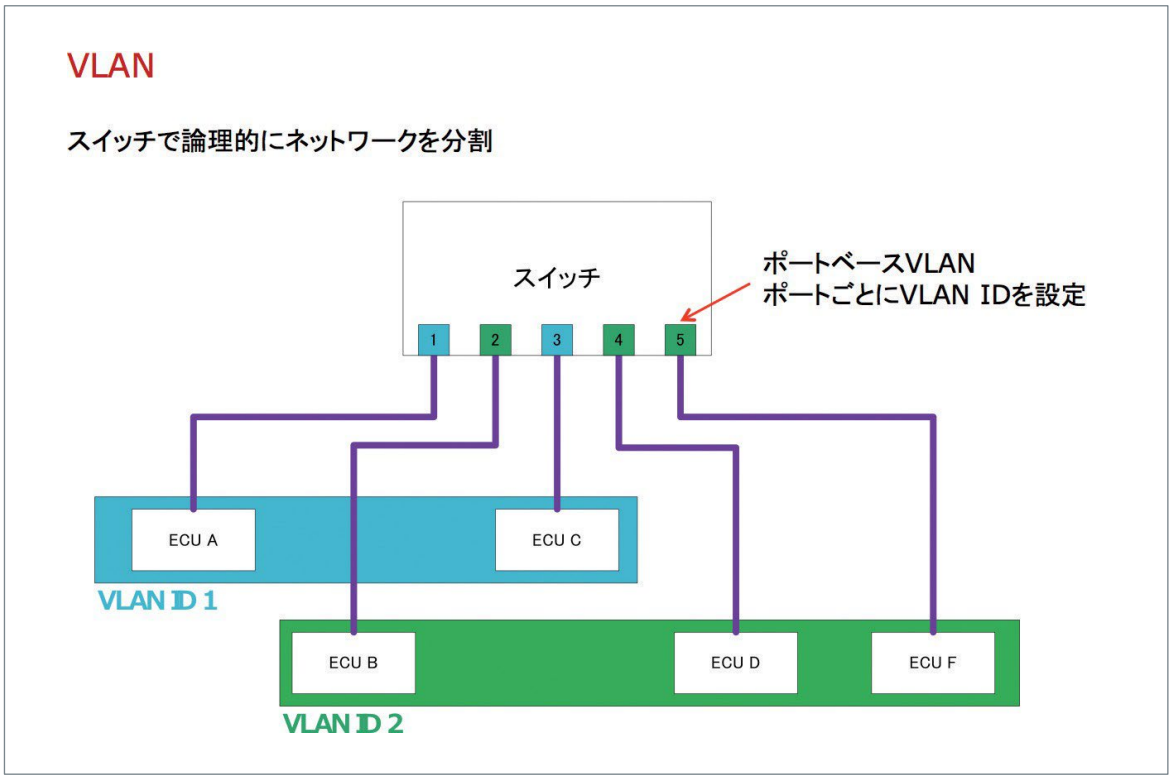

図 17-1

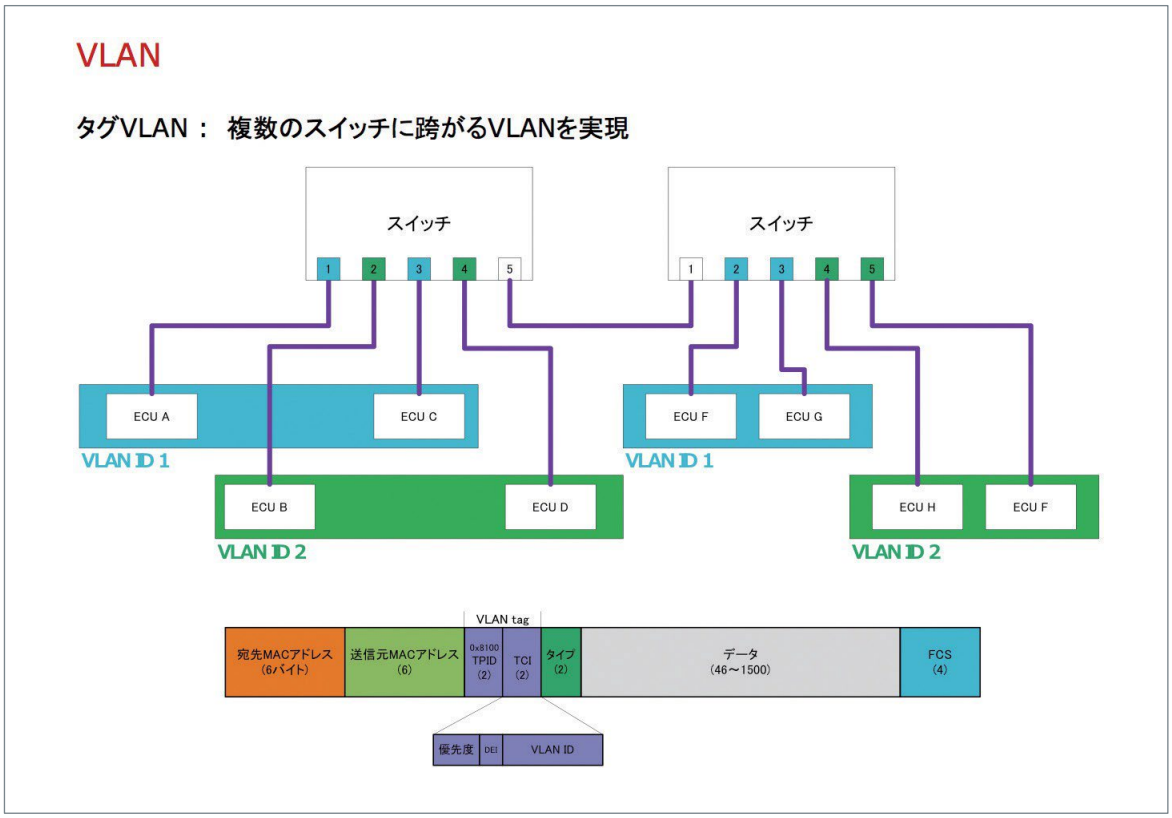

図 17-2

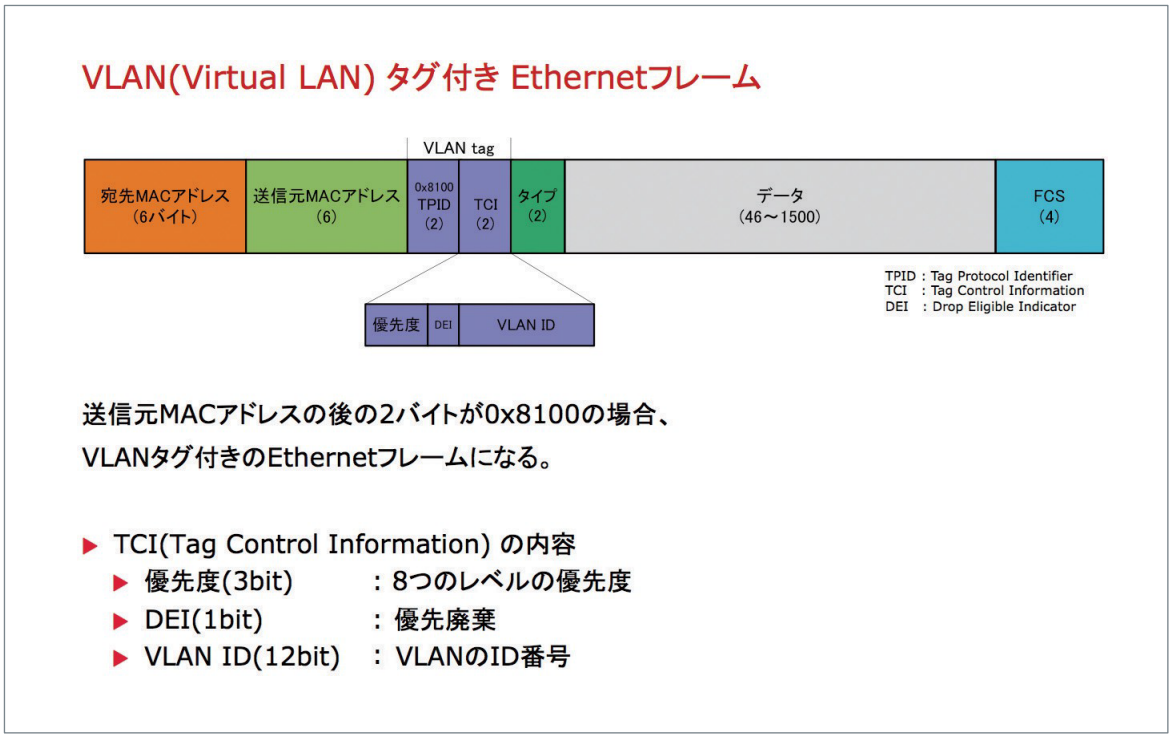

図 17-3

3.7 IP（Internet Protocol）と IP アドレス（v4）

IP は、OSI における第三層ネットワークレイヤーに相当するプロトコルです。データリンクレイヤー[24]に載ってきた SDU を取り出して扱います。ここでの PDU を IP パケットと呼び、それを扱うものはホストと呼ばれます。IP パケットは、Ethernet フレームのタイプ値が 0x0800（IPv4）または 0x86DD（IPv6）の際にペイロードとして送り届けられる情報です。

IP パケットに含まれる IP アドレスは、その IP パケットをどこのホスト[25]に届けるかを示しています。MAC アドレスと同様にグローバルアドレスとローカルアドレスがあり、インターネットにつな がり IP パケットを扱うもの（ホスト）にはグローバルアドレスを持つことになっています。逆にいうと、家庭内、社内、車載（車内）のような閉じた世界ではローカルアドレスを使うのが一般的です。

図 18 に IPv4 アドレスのローカルアドレスを示します。

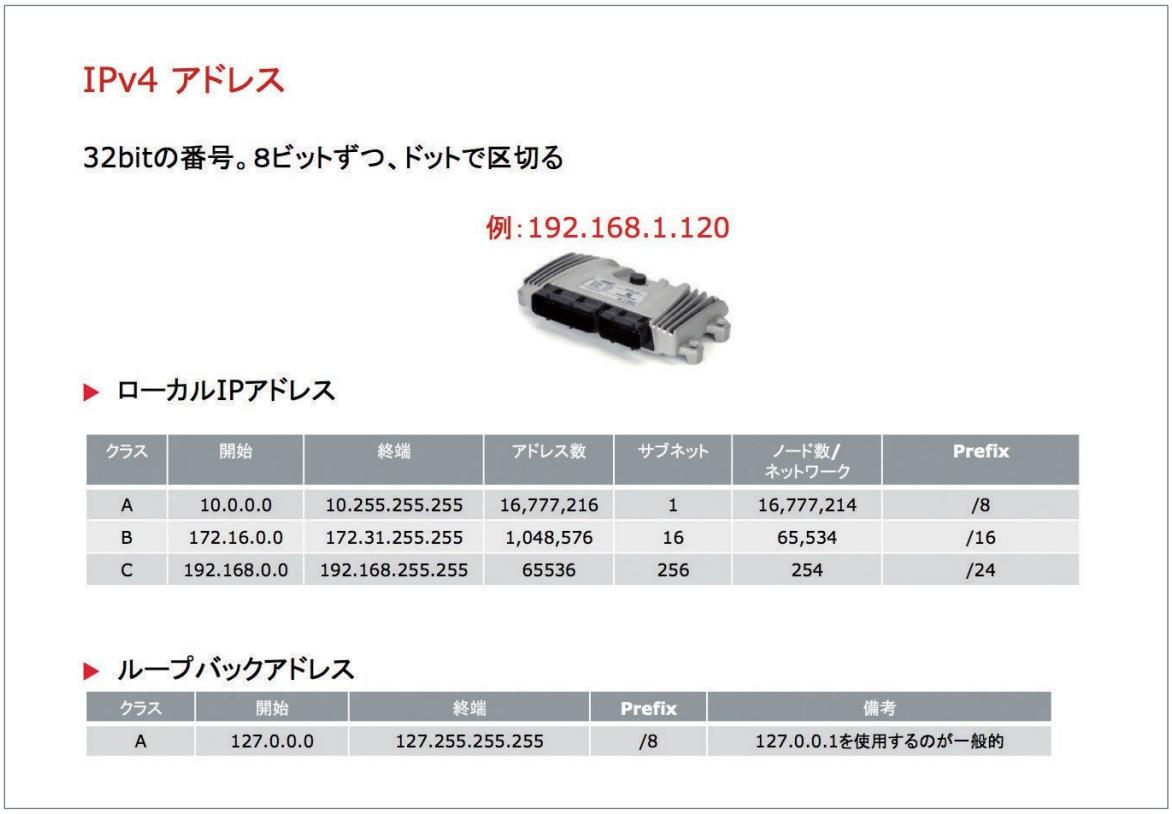

図 18

[24] 本資料の文脈では Ethernet ですが、OSI の考え方に従うと必ずしも Ethernet である必要はありません。

[25] もう少し正しく言うと、IP アドレスが割り当てられるのはネットワークインターフェイスです。

3.8 IP パケットの構造

同じくヘッダー（PCI）とペイロード（SDU）とに分かれています。Ethernet フレーム同様にヘッダー部分には宛先、送信元、内容、ペイロードの種類および送信経路での扱われ方を示す制御情報が入り、ペイロード部分に送り届けたい情報が格納されています。

図 19 は IPv4 パケットのヘッダー構造を示しています。このヘッダーに続いてペイロード（データ）部分が続きます。世の中ではすでに IPv6 が使われていますが、筆者が知る限り、車載では未だに IPv4 が主に使われています。図 19 ではヘッダーは4バイトワード×6 の 24 バイトになっていますが、車載では通常 Options の部分は使われていないため [26]それが除かれ、20 バイトになります（なので図 19 の中の IHL は「通常 5」としています）。

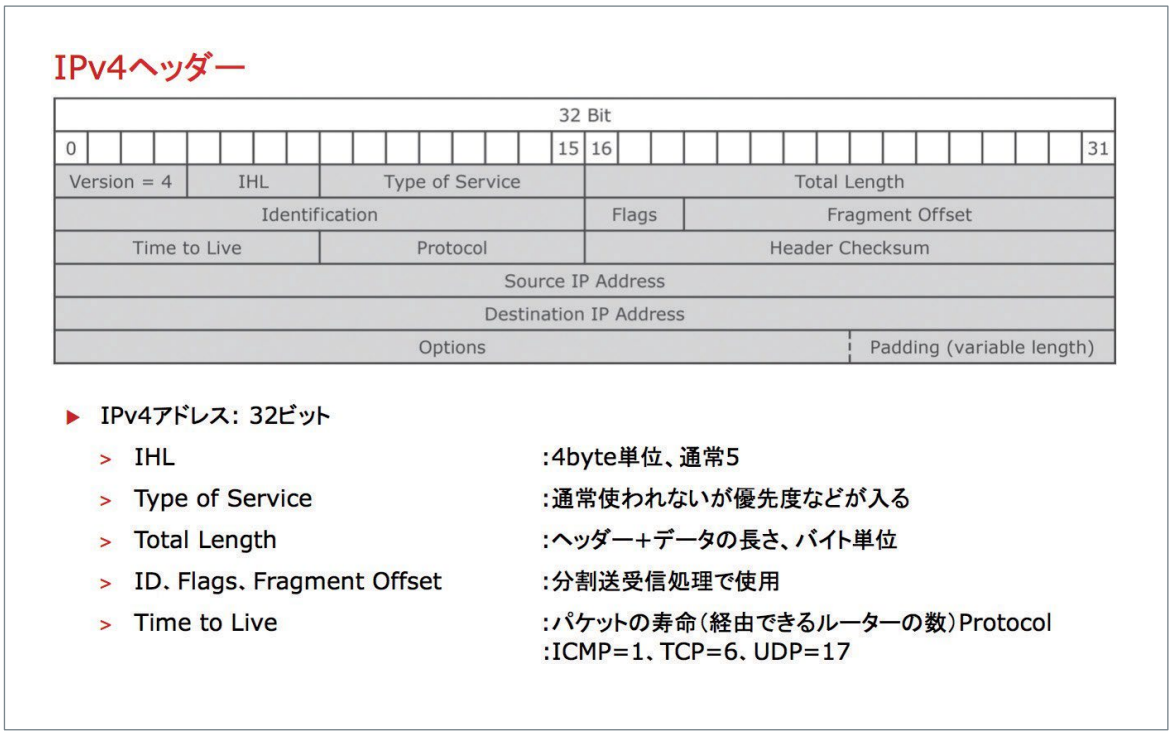

図 19

このヘッダーには複数の情報が入っていますが、今回 Source IP Address と Destination IP Address 以外に重要なのは“、Time to Live”と“Protocol”フィールドです。“Protocol”フィールドは、Ethernet フレームにおけるタイプと同等でこの IP パケットのペイロード部分に何が入っているのかを示しています。なお“Time to Live”については後述します（ルーターの役割で触れます）。

[26] 本資料執筆時点での認識です。

3.9 v4 と v6 の違い

車載では IPv4 がメインですが、v4 と v6 の違いについて、ここで簡単に触れておきます。IP パケットの届け先はインターネットが発達していくにつれて増加し、従来の v4 アドレス（32 ビット=4 バイト、最大 43 億通り [27]）では立ち行かなくなりました。そのため、128 ビット（=16 バイト、最大 3.4*10^38 通り [28]）という広大なアドレス空間を持つ v6 が導入されました（図 20 参照）。アドレスは従来の 10 進数ではなく、16 進数で 2 バイト毎に、(: コロン）で区切って表記するようになっています。この 2 バイト毎の塊をブロック(Block)と呼び、それが 8 つ連なる形です。なお、アドレスに 0（ゼロ、Zero）が連続した場合には、それらをまとめて表現する方法が 2 つあります。ブロックが

0 だけで構成される場合に 1 つの 0 でブロック全体を表す“Reduced Zeros”と、その 0 だけで構成されるブロックが連続した場合にコロンを 2 つ連続で表記することでその区間はすべて 0 であることを表す“Reduced Blocks”です。

アドレス以外には、ヘッダー構造が簡素化された点が大きな違いとして挙げられますが、アドレスが 4 倍の大きさになったためヘッダーサイズは約 2 倍になり、その分ペイロードサイズが小さくなっています。

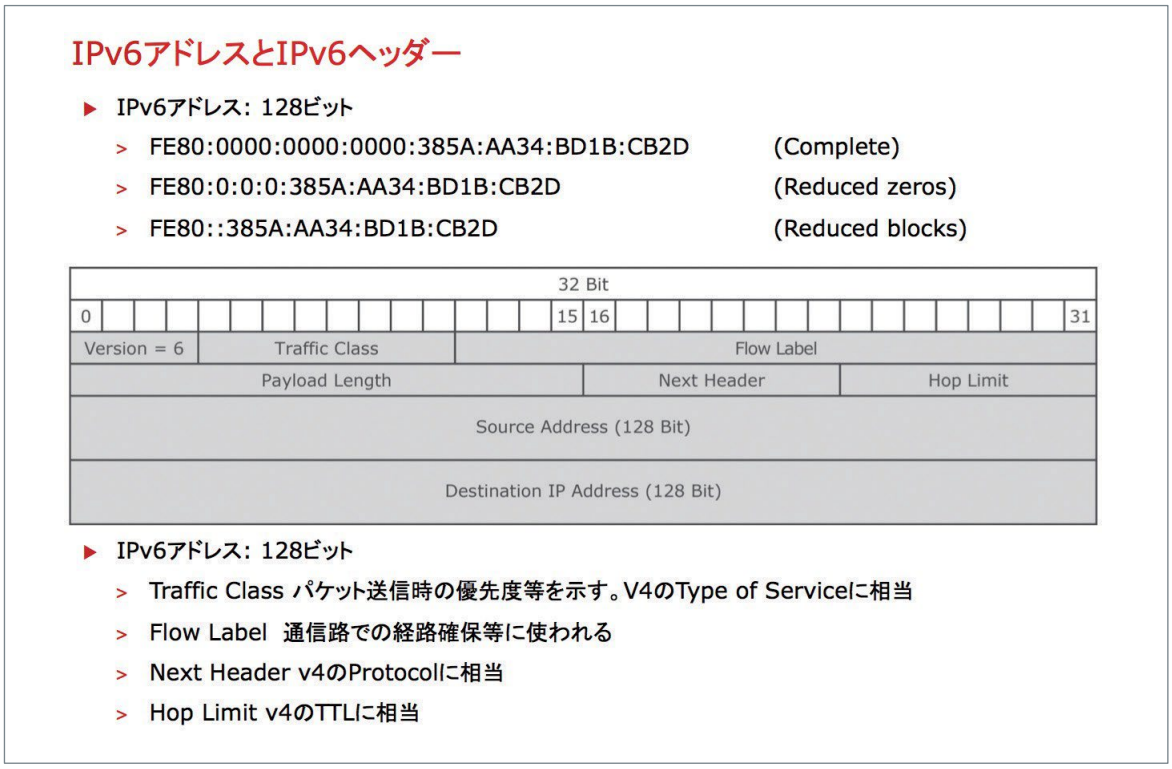

図 20

[27] グローバルアドレスとして割り当て可能な数は、もっと少ないです。

[28] 340 澗（かん）通り。現時点では、実用上ほぼ無限と言っても差し支えないと考えます。

3.10 MAC アドレス、IP アドレスの違い

さて、Ethernet（および TCP/IP）を学ぶ際によく疑問に思われるポイントは、「なぜ MAC アドレスと IP アドレスがあるのか？何が違うのか？」というところです。以下に、その点を説明していきます。

まずお伝えしたいのが、MAC アドレスと IP アドレスは、それを扱っているレイヤーが違うということです。（前者はレイヤー2、後者はレイヤー3）。MAC アドレスは Ethernet フレームの宛先、送信元を示すのに対して、IP アドレスは IP パケットの宛先、送信元を示しています。続いて思い出していただきたいのが、Ethernet フレームは IP パケットにとっての乗り物であるということです（図 21 参照）。Ethernet フレームを列車、IP パケットを乗客に例えると、列車の行き先 [29]を示すのが MAC アドレス、乗客の行き先を示すのが IP アドレスです。

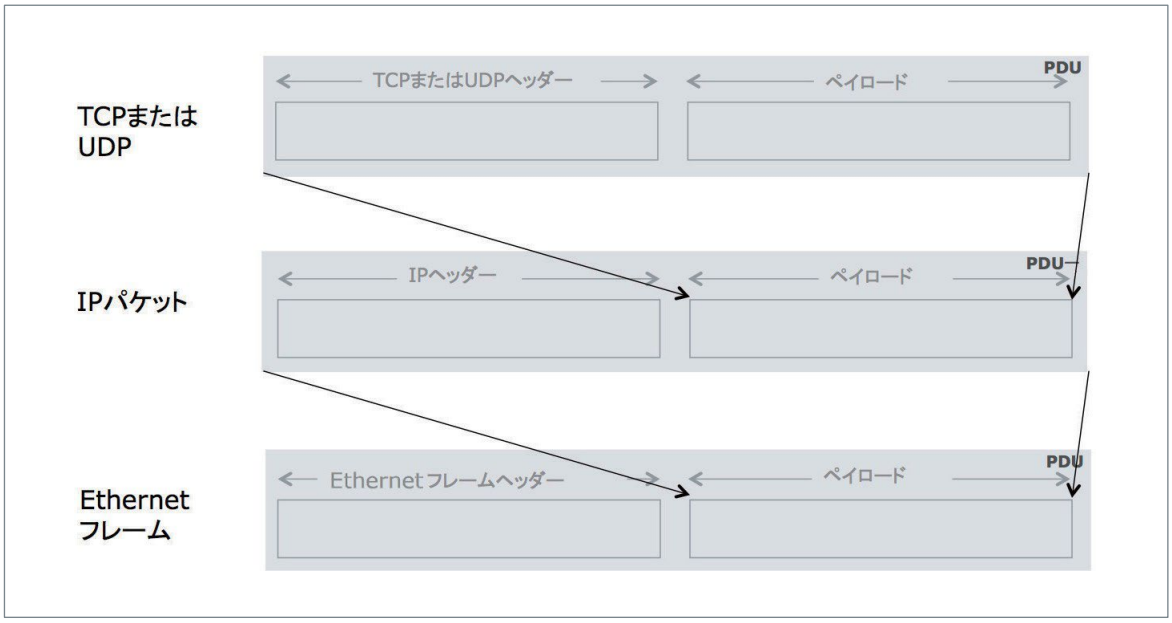

図 21

3.11 ルーターの役割

さて、現実世界で列車に乗って移動する際、鉄道ネットワークがさまざまな路線から成り立っている関係上、列車と乗客の行き先が同じ（路線）であるとは限らず、そうでない場合には乗り換えが必要です。現在のインターネットは、スイッチだけで構成された 1 ネットワークを無数につなぐ形で実現されています。そのネットワーク間をつなぐやりとりを司るもののの一つが、ルーター[30]です。列車での移動に例えるなら、乗換駅兼（次の乗換駅までの）乗換案内役という訳です。さまざまな情報をあらゆる所に届けるための方法が必要になった結果、乗り物である Ethernet ネットワークとその乗り物に乗って目的地に移動するための IP ネットワークという、階層の異なるネットワークが存在することになったのです。

では、ここで MAC アドレスと IP アドレスの関係とルーターの役割を乗換案内に例えて説明してみます（図 22 参照）。例えば、東京の天王洲アイル駅から、名古屋のささしまライブ駅 [31]まで行くとしましょう。Web または乗換案内アプリで経路を検索すると「天王洲アイル駅-モノレール->浜松町駅-山手線->品川駅-新幹線->名古屋駅-あ

[29] もちろん出発地点も表わしています。

[30] IP アドレスに基づいて情報の経路振り分け（ルーティング）をするのがルーターですが、ゲートウェイと呼ばれることもあります。ただし、ゲートウェイの方が広い意味で使われることもあり、かつ、プロトコル変換を伴います。

[31] 天王洲アイル駅、ささしまライブ駅はそれぞれ、ベクター・ジャパンの東京本社、名古屋支社の最寄り駅で、たまたま例として選びました。

おなみ線->ささしまライブ駅」のような経路が出るはずです。ただ、インターネット上の経路は最適な経路が時々刻々と変化していますので、すべての経路が最初から決まっているわけではなく、ルーターがその時に最適であると判断した経路 [32]に従って乗換え（転送）を実現しています。そしてこの例では、各駅がルーターに当たります。したがって乗客（IP パケット）は、まずは「天王洲アイル駅」からモノレールに乗り「浜松町駅」に向かいます。この時に乗るモノレールの行き先が MAC アドレスの DA に相当します。「浜松町駅」に着いた乗客は、自分の IP の宛先アドレスを見せると（実際にはルーターが見て）、次は「品川駅」に山手線を使って行くように指示され山手線に乗り込むわけです。続いて「品川駅」に着いた乗客は・・・という風に繰り返していき、最終的には目的地である「ささしまライブ駅」に到着します。ここでのポイントは、「出発地と目的地を示す IP アドレスは変わらないが、乗換えの際に乗る列車の行き先を示す MAC アドレスは変わっていく」ということです。また IP パケットは無限に乗換え（転送）ができるわけではなく、決められた回数を超えてしまうと乗換えができなくなります。具体的には、IP ヘッダーの説明に出てくる“Time to Live”により、その上限が決められています。このフィールドは、ルーターから別のルーターに転送される際に「1」ずつ減らされていき、その値が「0」になると、それ以上の転送はされなくなります。先ほどの例でいえば、乗換えの回数券のようなものだと思ってください（3 枚あれば行けますが、2 枚だと名古屋駅どまり）。このフィールドは何らかの理由 [33]で目的地に到達できない IP パケットが無限にネットワーク上に存在することを防ぐために存在しています。

MACアドレスとIPアドレス（乗換案内に例えて）

	天王洲アイル		浜松町		品川		名古屋		ささしまライブ	
IPアドレス	DA	SA	DA	SA	DA	SA	DA	SA	DA	SA
	ささしまライブ	天王洲アイル	ささしまライブ	天王洲アイル	ささしまライブ	天王洲アイル	ささしまライブ	天王洲アイル	ささしまライブ	天王洲アイル
MACアドレス	DA	SA	DA	SA	DA	SA	DA	SA	DA	SA
	浜松町	天王洲アイル	品川	浜松町	名古屋	品川	ささしまライブ	名古屋	ささしまライブ	名古屋
Time to Live	3		3		2		1		0	

図 22

3.12 まとめ

第 3 章では、データリンクレイヤーについて、そこで使われる MAC アドレス、Ethernet フレームの構造に続き、その Ethernet フレームに載せられて送られる IP パケットの構造と IP アドレス、および IP アドレスと MAC アドレスの違いについて説明しました。第 4 章では、この IP パケットに載せられて送られる TCP セグメント／UDP データグラムの構造や振る舞いについて紹介していきたいと思います。

[32] 現実世界でも、事故等が発生すると、初めに経路案内で示された経路ではなく、別の経路を取ることがあります。それと同様です。

[33] ルーターの設定ミスや経路上の障害など。

4. IP パケットに載せられて運ばれるもの、UDP/TCP とその上位プロトコル

第 3 章では、データリンクレイヤーについて、そこで使われる MAC アドレス、Ethernet フレームの構造に続き、Ethernet フレームに載せられて送られる、ネットワークレイヤーに相当する、IP パケットの構造と IP アドレス、および IP アドレスと MAC アドレスの違いについて説明しました。第 4 章では、この IP パケットに載せられて送られるトランスポートレイヤー（第 4 層）に相当する TCP セグメント／UDP データグラムの構造や振る舞い、そして車載 Ethernet で用いられる上位プロトコルを紹介していきたいと思います。

4.1 TCP（Transmission Control Protocol）/UDP（User Datagram Protocol）とポート（Port）

まず、おさらいから入りましょう。OSI における第 2 層データリンクレイヤーにおける PDU が、Ethernet フレーム、そのペイロード部分（SDU）に載せられて運ばれるのが第 3 層ネットワークレイヤーにおける PDU、IP パケットでした。これと同じ関係が、IP パケットと TCP セグメント／UDP データグラムの間にはあるわけです。つまり IP パケットのペイロード部分に載せられて運ばれるのが、第 4 層トランスポートレイヤーにおける PDU である TCP セグメントや UDP データグラムです。もちろんこれらも上位層に渡すデータであるペイロード（SDU）と制御情報（PCI）であるヘッダーに分かれています。このヘッダーには、Ethernet フレームや IP パケットのヘッダー部分同様に、宛先と送り元を示す情報が存在しています。その情報を TCP/UDP では、ポート（Port）とよんでいます [34]。前回使った乗換案内の例で改めて説明してみます（図 23 参照）。

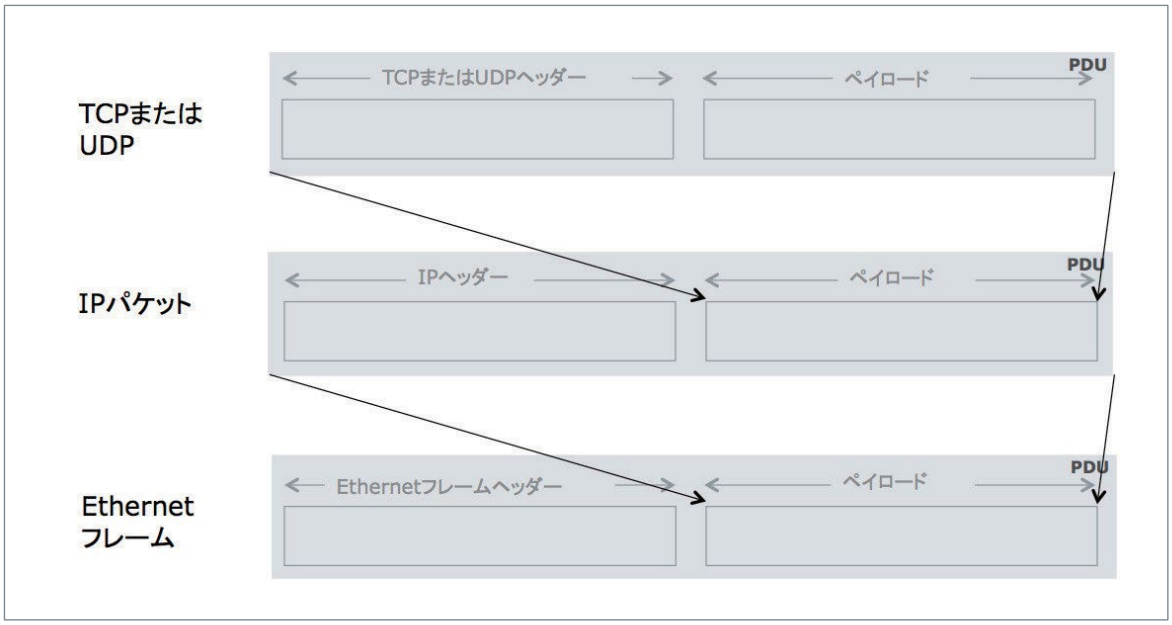

図 23　各プロトコルレイヤーの関係

[34] レイヤー4 アドレスです。レイヤー2 スイッチのポートとは違います。16 ビットの情報で 0-65535 の値をとることができます。そのうち 0-1023 まではウエルノウン（Well known）ポートとよばれ TCP/IP の主要なサービスが使用している予約された番号で、IANA（International Assigned Number Authority、IP アドレスやポート番号、プロトコル番号等の管理を行う組織）が管理しています。この番号は、それらのサービスを提供しているサーバー側で使われます。別の言い方をすると、勝手に使ってはいけないポートです。1024 以降の扱いは 2 種類（Traditional と RFC6056）あり、実装によってどちらを使うかが決まってきます。

まず Ethernet フレームは IP パケットにとっての乗り物、つまり列車。IP パケットはその乗客でした。さらに列車の出発駅・到着駅を示すのが MAC アドレス、乗客の出発駅・到着駅を示すのが IP アドレスです。そして今回のポートは出発駅・到着駅に直結しているマンション TCP 棟と UDP 棟にある部屋番号にあたります。実はこの乗客（IP パケット）、駅の外には出られないのです。出発駅では TCP や UDP という別の荷物担当（上位プロトコル）が、それぞれの担当する棟の部屋（ポート）から荷物を回収し箱（TCP セグメント／UDP データグラム）に詰めて改札口で乗客に渡します。そして最終的に IP アドレスで示された駅に到着したら、荷物を改札で TCP または UDP に渡すと、それぞれの担当の棟のポートで示された部屋まで届けてくれるイメージです [35]。

4.2 コネクションレスコミュニケーションとコネクションオリエンテッドコミュニケーション

続いて、UDP/TCP の PDU 構造や振る舞いを説明したいところですが、その前にこれらのプロトコルを理解するのに重要な 2 つの考え方を紹介します。それが、コネクションレスコミュニケーションとコネクションオリエンテッドコミュニケーションです。これらの特徴は、それぞれ駅のアナウンスと電話における情報の送り方になぞらえることができます。まずはコネクションレスコミュニケーションですが、これは駅のアナウンスだと思ってください。つまり、

- 複数の人に同時に情報を届けることができる
- 情報を受け取るための特別な手続き（コネクション）は不要
- 送り手は、受け手が情報を受け取ったか否かについて興味がない（送りっぱなし）
- 受け手も、聞こえなかったとしても気にしない [36]ということです。

一方、コネクションオリエンテッドコミュニケーションは電話ですので、

- 情報をやりとりするための特別な手続き（コネクション）が必要（電話をかけて相手が出るのを確認する）
- 1 つのコネクションでやりとりすることができる相手は 1 人だけ
- 送り手は受け手がきちんと聞いているかを気にするので、受け手は聞こえたら相づち（アクノリッジ）を返す
- 送り手は受け手から相づちが返ってこないと、次の話はしない（もしくは繰り返す）
- 話が終わったら電話を切る（コネクションを切る）という流れが存在しています。

UDP と TCP は、それぞれコネクションレスコミュニケーション、コネクションオリエンテッドコミュニケーションのプロトコルですので、そのために必要な制御情報や通信時の振る舞いが異なっています。では、それぞれ見ていきましょう。

[35] それぞれの部屋には TCP または UDP の上位プロトコルやアプリケーションがいるわけです。

[36] 重要なアナウンスだと気にしますが普段は聞き流しているはずです。

4.3 UDP の振る舞いとヘッダー構造

繰り返しになりますが、UDP はコネクションレスコミュニケーション、つまり駅のアナウンスなので、「出しっぱなし・送りっぱなし」ということです。したがって制御情報も宛先（デスティネーション）ポート、送り元（ソース）ポート [37]、データ長 [38]、そしてチェックサムの 4 つだけです（図 24 参照）。したがって、エラー処理もシンプルで、受信側におけるチェックサム [39]確認のみです。TCP と比較すると信頼性は劣りますが、オーバーヘッドが小さく、即時性に優れます。なお UDP における PDU（PCI：ヘッダー+SDU：データ）はデータグラムとよばれます [40]（図 25 参照）。

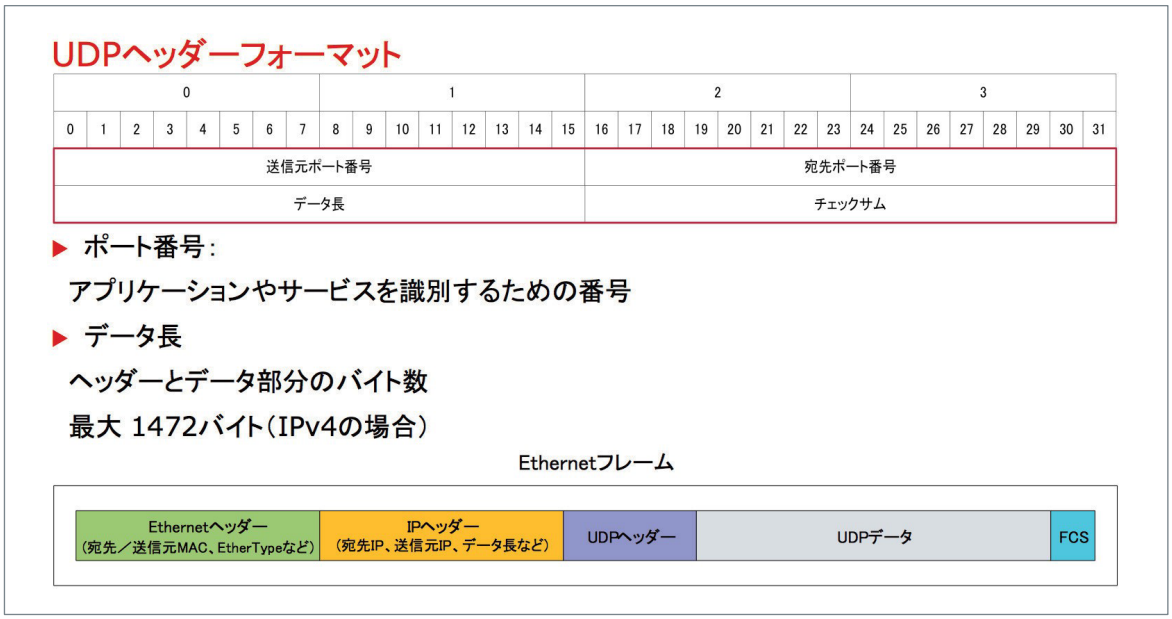

図 24　UDP ヘッダーフォーマットと Ethernet フレーム内での位置

[37] 返信を期待しない処理の場合は、省略する（0 とする）ことができます。

[38] ヘッダーと実データ長の合計、つまりは UDP データグラム全体の長さです。

[39] v4 だと、これすら省略することもできます。

[40] 一般用語である「パケット」を使うこともあります。

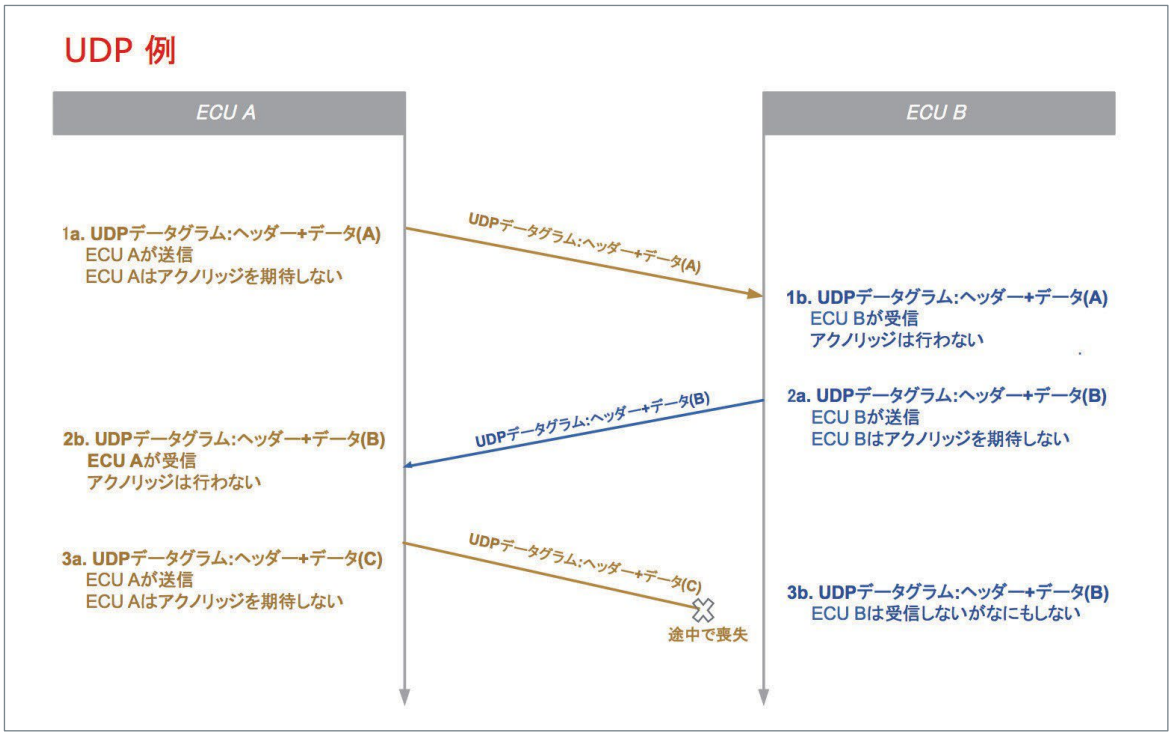

図 25 データ転送(UDP)

4.4 TCP の振る舞いとヘッダー構造

一方、TCP はコネクションオリエンテッドコミュニケーション、つまり電話です。したがって、電話をかける(コネクションを確立する)、話をする(データ転送を行う)、電話を切る(コネクションを解消する)の 3 つのフェーズがあり、かつ、それぞれのやりとりにおいて常に相づち(アクノリッジ、確認応答)を返していきます(図 26 参照)。

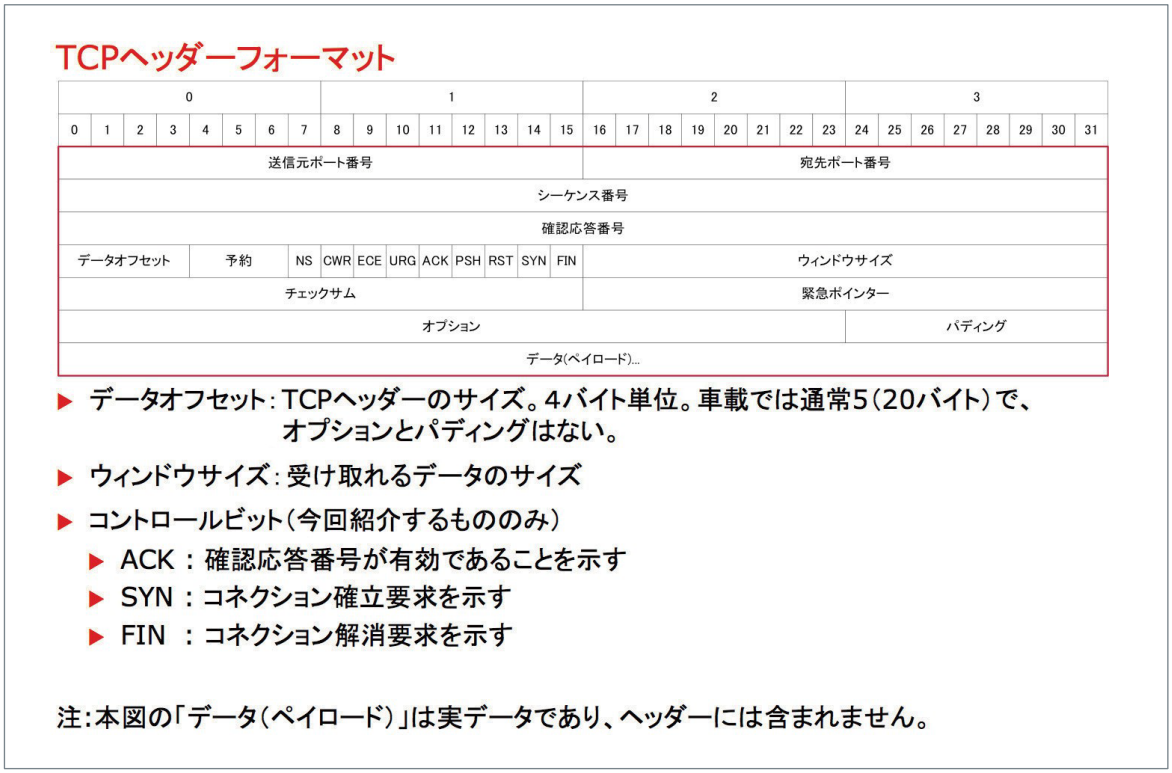

図 26 TCP ヘッダーフォーマット

そのため、制御情報も多くヘッダーも複雑です。今回は、その 3 つのフェーズをシーケンス番号、確認応答（アクノリッジ）番号と ACK、SYN、FIN の 3 つのコントロールビットを使って簡単に説明していきます。

ここでのポイントは

- コネクション確立の際には SYN ビットを ON、コネクション解放の際には FIN ビットを ON にする
- SYN または FIN を受け付けたことを知らせる際（アクノリッジ）には、SYN または FIN と同時に受信したシーケンス番号に 1 を足した値を確認応答番号として送る
- データを受信したことを知らせる際には、データと同時に受信したシーケンス番号に受信したデータのバイト数を足した値を確認応答番号として送る
  （例：シーケンス番号が 970 で 30 バイトのデータを受信したら 1000 が確認応答番号）
- アクノリッジ受信側は上記のルールに合致しているかをチェック。合致していない、または規定の時間内にアクノリッジが返ってこない場合には、異常とみなしてリトライを行う

です。では、以下の例を見ていきましょう（図 27 参照）。

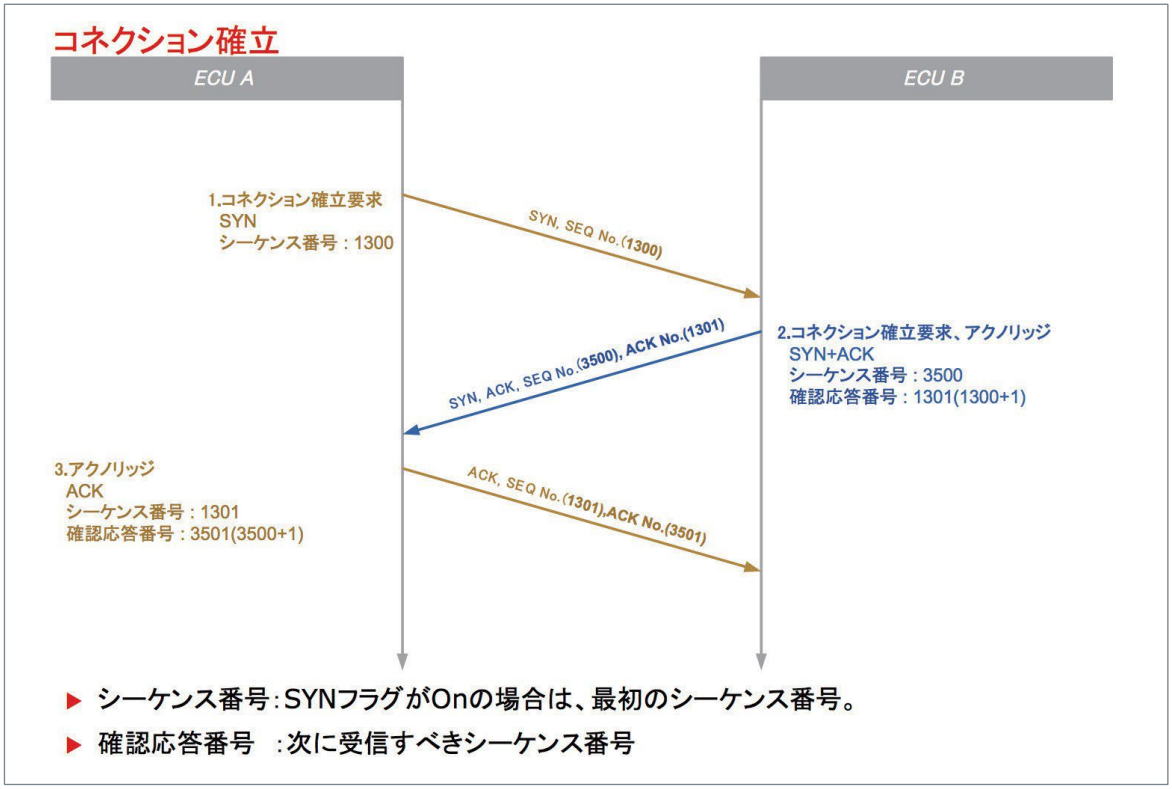

図 27　コネクション確立(TCP)

1. **コネクションの確立**

   コネクションは双方向に確立します。この例では、ECU A[41]がシーケンス番号と SYN をセットした「コネクション確立要求」セグメントを送信します。図 27 内にもあるように、SYN とともに送るシーケンス番号は最初のシーケンス番号で、送る側が無作為に決めた数字です。これを基に以降の通信が進んでいきます。

2. **アクノリッジとコネクション確立要求**

   続いて ECU B では、受け取った「コネクション確立要求」セグメントを受け入れ、それを示す「アクノリッジ（確認応答、ACK）」とそれが何に対する応答であるかを示す確認応答番号（受信シーケンス番号+1）を送るとともに、自らも「コネクション確立要求」を送るべく SYN と（最初の）シーケンス番号をセットした「アクノリッジ+コネクション確立要求」セグメントを送り出します。

3. **アクノリッジ確認とアクノリッジ送信**

   A は受信したセグメントの中に ACK と自らが送ったシーケンス番号（1300）+1 の確認応答番号を確認することで、自らのコネクション確立要求が受け入れられたことを理解します。また SYN がセットされていることから、B からも「コネクション確立要求」があり、それを受け入れたことを示すための「アクノリッジ」を、ACK と確認応答番号（受信したシーケンス番号 3500+1）をセットして送信します。もちろんそれを受信した B も同様に ACK と確認応答番号をチェックし、自らの「コネクション確立要求」が受け入れられたと理解します。

この 3 回のやりとりで双方向のコネクションを確立する方法を「スリーウェイハンドシェイク（Three way hand shaking）」とよびます。
続いて行われるのはデータ転送です（図 28 参照）。

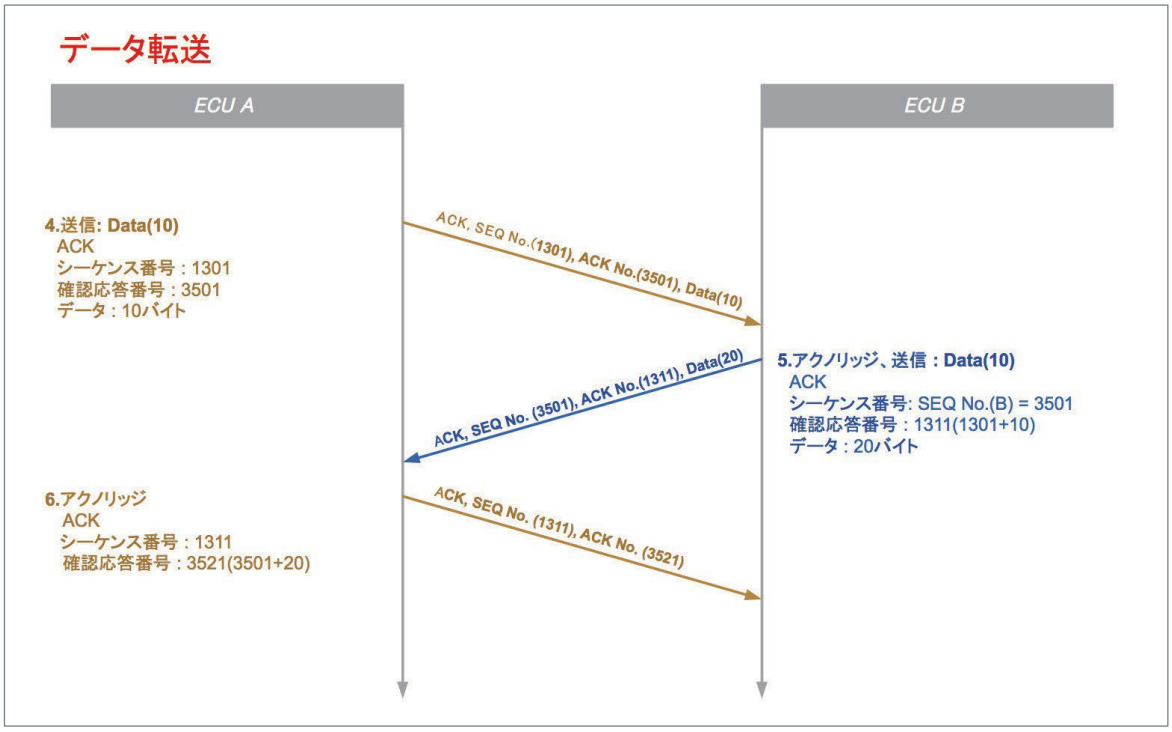

図 28 データ転送(TCP)

[41] TCP/IP の世界での通信は一般的に「クライアント・サーバー通信」で、先に通信を始めるのはクライアント側です。

4. **ECU B へのデータ送信**

   まず ECU A が 10 バイトのデータを送ります。ECU B は 10 バイトのデータを受信しました。

5. **アクノリッジと ECU A へのデータ送信**

   ECU B は、ACK をセットし受信したシーケンス番号（1301）+受信データバイト数（10）=1311 を確認応答番号として「アクノリッジ」を返します。その際に ECU B は併せて 20 バイトのデータを送っています。ECU A は 20 バイトのデータを受信しました。

6. **アクノリッジ**

   ECU A は、送信されてきた ACK と確認応答番号（1311）をチェックし、データが正しく受信されたことを確認します。また B が送ってきたシーケンス番号（3501）+受信データバイト数（20）=3521 を確認応答番号として ACK をセットし「アクノリッジ」を返します。ECU B は、受け取った「アクノリッジ」の応答確認番号が送ったシーケンス番号（3501）+送ったデータバイト数（20）=3521 であることで、データが正常に受信されたことを確認します。

そして最後には、コネクションを解放します（図 29 参照）。

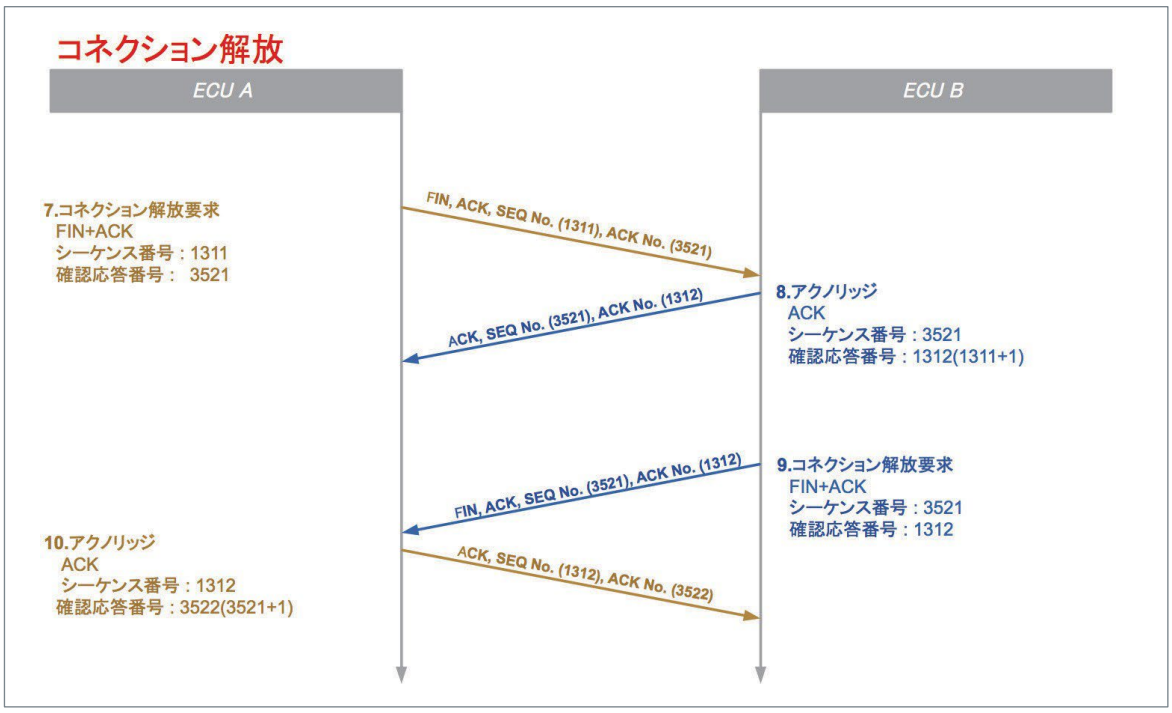

図 29 コネクション解放(TCP)

7. **コネクション解放**

   データ転送が終了し、通信の必要がなくなったため、ECU A が FIN をセットし「コネクション解放要求」を送ります。

8. **アクノリッジ**

   「コネクション解放要求」を受け取った ECU B は、受信したシーケンス番号（13 11）+1=1312 を確認応答番号として「アクノリッジ」を送ります。それを受け取った ECU A は、受信した「アクノリッジ」の応答確認番号が送ったシーケンス番号（1311）+1=1312 であることで、「コネクション解放要求」が受け付けられたことを確認します。

9. **コネクション解放**

10. **アクノリッジ**

    コネクションが双方向でしたので、ECU B から ECU A へコネクション解放も行います。なお、この例では双方 1 往復ずつのやりとりをしていますが、コネクション確立時と同様にスリーウェイで解放することもできます。

ここでは、簡単な例で TCP の動作の一部を紹介しました。実際には TCP はエラー検出時のリトライや、到着したデータの正しい順番への並べ替え [42]をはじめとする高機能かつ信頼性の高いプロトコルです。ただ即時性は、コネクション確立等のオーバーヘッドがあり、UDP と比較した際に劣ります。つまり TCP/UDP ともに一長一短があるため、上位のプロトコル（サービス）が、用途に合わせて使い分けています。

[42] インターネット上では必ずしも同じルートを通るわけではないので、到着順も保証されていません。そのためシーケンス番号を見ることで受信側は正しい順番にデータを並べ替えます。

4.5 車載 Ethernet で用いられる上位プロトコル

最後に、ここまでお話してきた Ethernet や TCP/IP の上で用いられる車載プロトコルのうち、SOME/IP と DoIP の 2 つを簡単に紹介していきたいと思います。

4.5.1 SOME/IP（Scalable service-Oriented MiddlewarE over IP）[43]

SOA（Service Oriented Architecture、サービス志向アーキテクチャ）とよばれる、大規模コンピューターシステムを構築する上での考え方・手法があります。これはネットワーク上に存在する複数の機能（サービス）を複数組み合わせることで、新しいシステムを作っていくもので、さまざまな要求に素早く対応できるのが特徴です。この考え・手法をクルマに取り入れる際に使われるのが、サービス志向通信である、この SOME/IP および SOME/IP-SD（以下、SOME/IP）とよばれる上位レイヤープロトコルです（図 30 参照）。現在のクルマでは、特定の機能（サービス）は ECU に強く紐づけられていて、それを ECU 外から使うのは容易ではないという状況があります。将来的にクルマがスマートフォン化していくにあたっては、新しくダウンロードしたアプリが、クルマのネットワークにあるさまざまな機能を見つけ出し、それを自由に使える仕組みが必要で、このプロトコルは Ethernet・TCP/IP の特徴（大量の情報を高速に、必要に応じて特定または複数の相手に送ることができる）を活かして、それを実現します。

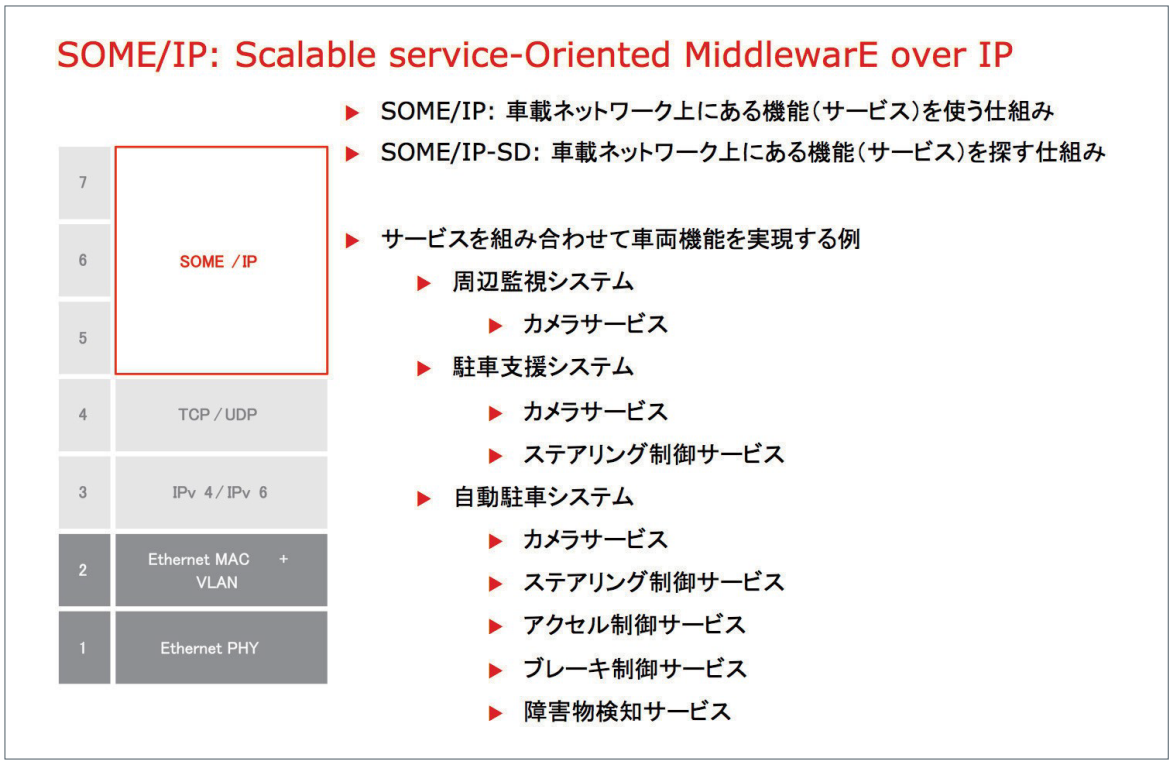

図 30 OSI 参照モデルにおける SOME/IP とその機能

この SOME/IP では、情報のやりとりの代表的な仕組み（メッセージ）として

- リクエストレスポンス：あるサービスが提供しているメソッド（処理、手続き）を呼び出し、その結果を受け取る
- ファイアアンドフォーゲット：あるサービスが提供しているメソッドを呼び出す、結果は求めない
- イベント：特定の事象（イベント）が発生するとサーバー（サービス）からノーティフィケーション（通知）を送る

[43] BMW が提唱したプロトコルで、AUTOSAR 仕様の一部になっています。

> フィールド：サーバーが持つ値を操作（読み出し、書き込み）する、その値が変わった時に通知を送る

があり、提供されるサービスごとにそれらを組み合わせて使っています（図 31-1、図 31-2 参照）。

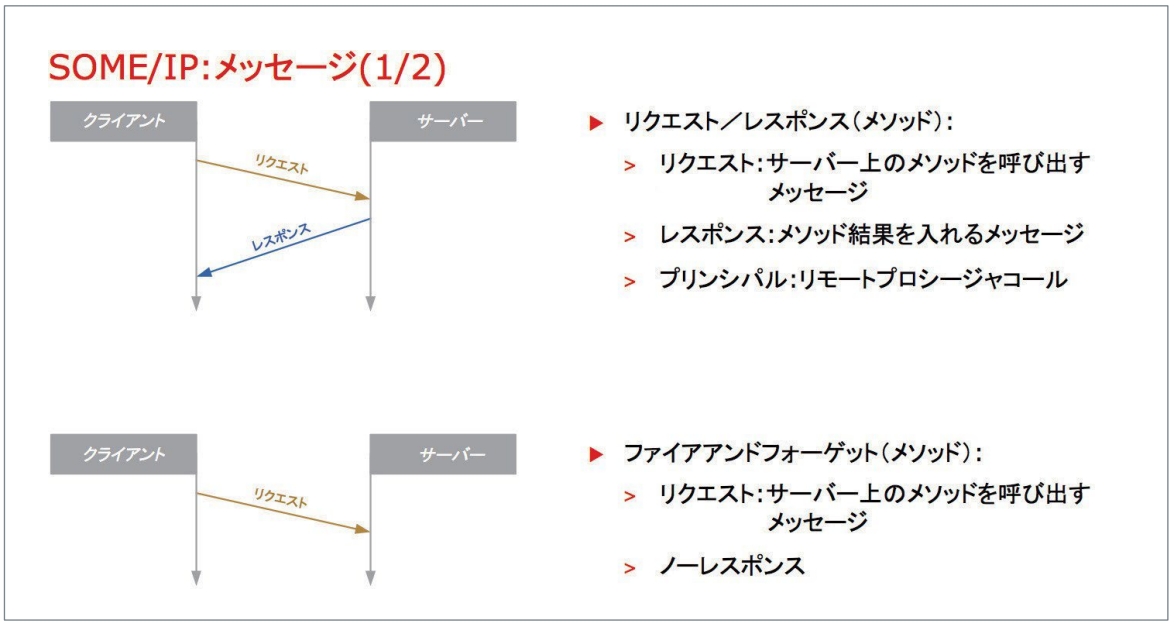

図 31-1 リクエスト／レスポンスとファイアアンドフォーゲット

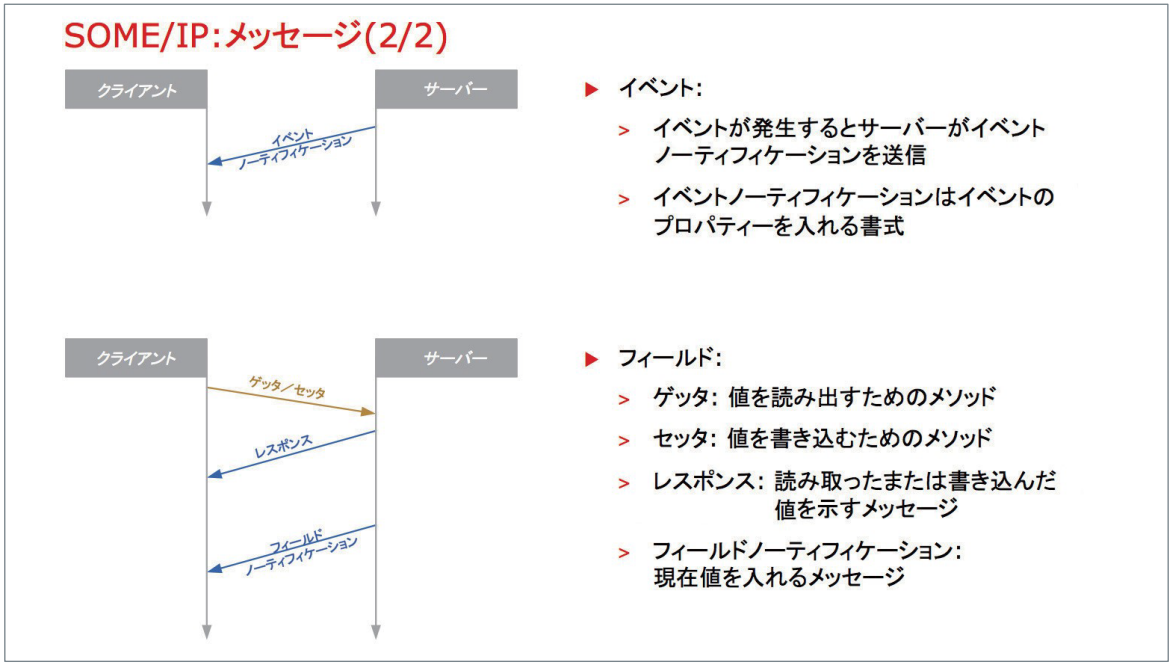

図 31-2 イベントとフィールド

4.5.2 DoIP(Diagnostic over IP)

第 2 章のフィジカルレイヤーで、診断テスターと診断ゲートウェイ（以下、診断 GW）とよばれる ECU をつなぐものとして 100BASE-TX を紹介したのを覚えていらっしゃるでしょうか。その時に診断通信に使われる上位プロトコルとして紹介したのが、この DoIP（Diagnostic over IP、ISO 13400)です（図 32 参照）。

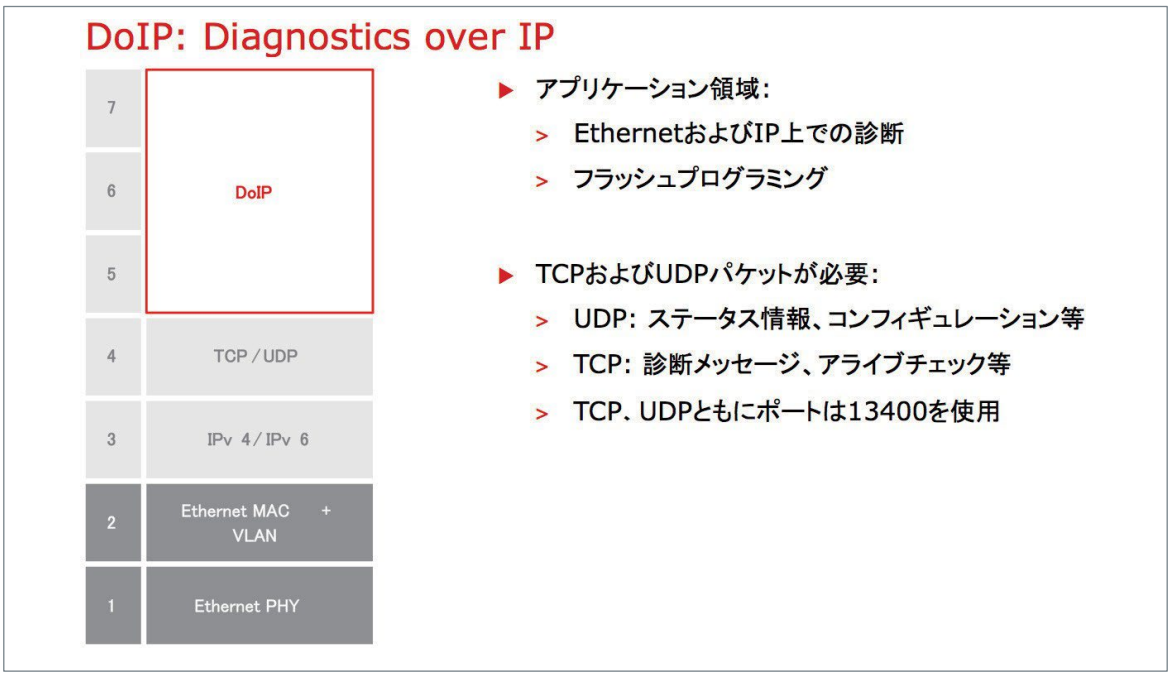

図 32 OSI 参照モデルにおける DoIP と大まかな特徴

このプロトコルは TCP や UDP の中に DoIP メッセージ、そしてその中に診断メッセージを載せてやりとりするための仕組みです。1 番のポイントは、Ethernet・TCP/IP 通信が使われるのは、あくまで診断テスターと診断 GW の間だけであって、診断 GW から先は、たとえば CAN 通信によって従来どおりの診断通信が行われているのが、まだ一般的であるということです（図 33 参照）。つまり診断 GW は、DoIP によって運ばれたきた診断メッセージ（たとえば、UDS の SID と DID に基づく）を CAN メッセージに載せ換えて、最終的な診断対象である ECU とやりとりをしているのです。それでも、従来の通常の CAN による診断通信と比較すると速さは向上しています。

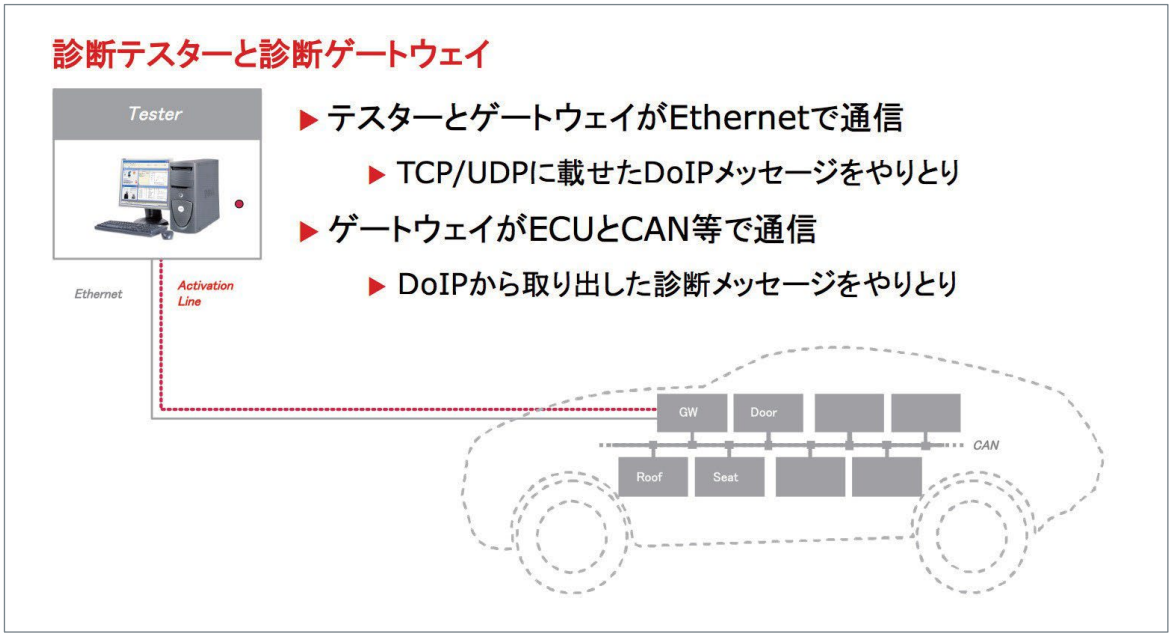

図 33 DoIP メッセージのやりとりの主体、診断テスターと診断ゲートウェイ

4.6 まとめ

第 4 章では、IP ネットワーク上のプロトコル TCP と UDP の PDU（主に PCI、ヘッダー）構造と振る舞い、さらにはその TCP と UDP を使って実現する上位プロトコル SOME/IP と DoIP を紹介しました。これ以外にも、複数のプロトコルが車載 Ethernet 上では使われ始めています。今後、本格的にクルマがスマートフォン化（CASE 対応）していくにつれ、より多くのプロトコルが使われるようになるでしょう。それと同時に、物理層も変化していくことが予想されます。それらの変化に皆様がついていくための基本的な知識を、本資料で少しでも提供できたのであれば幸いです。

5. お問い合わせ窓口

技術関連のお問い合わせは、ベクターサポートまでお寄せください。

✓ サポートリクエスト（Web サイト経由のサポート問い合わせフォーム）
www.vector.com/jp/ja/support-downloads/support/

【お問い合わせ時のお願い】

お問い合わせの際、サポート開始前に製品のバージョン（例：CANape バージョン xx.x など）、シリアルナンバーなどをお伺いいたします。お手数ではございますが、事前にご確認のうえ、お問い合わせくださいますようお願い申し上げます。

【サポート対応について】

夜間／休日に頂いたお問い合わせは翌営業日の対応となります。また、内容によりご回答までにお時間を頂く場合がございます。なお、不具合が発生した際にご使用されていた関連プログラム一式のコピーを、検証のためご提供いただくようお願いする場合がございます。何卒ご理解ご協力のほど、よろしくお願い申し上げます。

✓ **お見積もり／保守契約関連のお問い合わせ**

営業部　TEL：（東京）03-4586-1808　E-mail：sales@jp.vector.com

VECTOR >

MORE INFORMATION
Visit our website for:

> News
> Products
> Demo software
> Support
> Training and workshops
> Contact addresses

vector.com