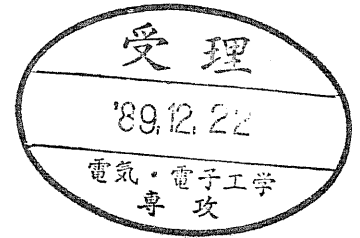


小ゾーン連続形自動車パケット
通信システムに関する研究

酒井清一郎



博士論文

小ゾーン連続形自動車パケット通信システムに関する研究

指導教官 高羽禎雄 教授

東京大学大学院工学系研究科電気工学専攻

No. 77075 酒井清一郎

1989年 12月22日 提出

◎ 目次

項目	頁
1. 序章	
1. 1. 本研究の背景と目的	1
1. 2. 本論文の構成	6
2. システム概念	
2. 1. 序言	9
2. 2. システム説明	
2. 2. 1. システム概要	10
2. 2. 2. システム機能	13
2. 2. 3. システム構成	15
2. 3. 路車間通信仕様	22
2. 4. 通信ゾーン	
2. 4. 1. 通信ゾーン構成	25
2. 4. 2. 通信ゾーン配置間隔	34
2. 5. 結言	44
3. 道路リンクレベル制御	
3. 1. 序言	45
3. 2. 道路リンクレベル制御の機能	
3. 2. 1. 機能概要	47
3. 2. 2. 走行データ通信機能	50
3. 2. 3. 汎用データ通信機能	52

3. 3. データ小分割による伝送方式	
3. 3. 1. 方式説明	5 5
3. 3. 2. シミュレーション評価	5 8
3. 4. 接続制御方式	
3. 4. 1. 方式説明	6 5
3. 4. 2. シミュレーション評価	7 2
3. 5. 情報伝送方式	
3. 5. 1. 方式説明	7 7
3. 5. 2. シミュレーション評価	8 2
3. 6. 路車間通信チャネル割当制御	9 4
3. 7. 路車間通信制御	1 0 3
3. 8. 個別情報リンク内管理	1 0 9
3. 9. 結言	1 1 4
4. グローバルレベル制御	
4. 1. 序言	1 1 5
4. 2. グローバルレベル制御の機能	
4. 2. 1. 走行誘導機能	1 1 7
4. 2. 2. 汎用データ通信機能	1 1 7
4. 3. 個別情報の自立分散型管理	
4. 3. 1. 個別情報の自立分散型管理概念	1 1 9
4. 3. 2. 走行誘導機能の実現	1 2 1
4. 3. 3. 汎用データ通信機能の実現	1 2 3
4. 4. 汎用データ通信のグローバルレベル制御	
4. 4. 1. パケット伝送の基本パターン	1 2 6

4. 4. 2. データパケットのフロー制御	1 2 9
4. 4. 3. リンク旅行時間予測による効率化	1 3 3
4. 4. 4. シミュレーション評価	1 4 0
4. 5. 各装置で管理すべき情報内容	
4. 5. 1. 走行誘導の情報	1 5 8
4. 5. 2. 汎用データ通信の情報	1 5 8
4. 6. 結言	1 6 1
5. 走行誘導への応用	
5. 1. 序言	1 6 2
5. 2. 走行誘導の概念	1 6 3
5. 3. 高速道路分流出点での動的車線規制	
5. 3. 1. アルゴリズム概念	1 6 9
5. 3. 2. 誘導指示作成方法	1 7 2
5. 3. 3. シミュレーション評価	1 7 4
5. 4. 街路の信号交差点での右折車の待合せ制御	
5. 4. 1. アルゴリズム概念	1 8 1
5. 4. 2. 誘導指示作成方法	1 8 4
5. 4. 3. システムへのインプリメント	1 9 1
5. 4. 4. シミュレーション評価	1 9 4
5. 4. 5. 通信ゾーンの配置間隔の検討	1 9 8
5. 5. 結言	2 0 4
6. グローバルシステムへの応用	
6. 1. 序言	2 0 5

6. 2. 提供するグローバルな機能	2 0 6
6. 3. 応用システム例	
6. 3. 1. 双方向データ通信	2 1 2
6. 3. 2. 要求応答型情報提供	2 1 2
6. 3. 3. 放送型情報提供	2 1 4
6. 3. 4. 応用形態の定量的検討	2 1 5
6. 4. 結言	2 1 8
7. スケールモデルでの実現可能性検証実験	
7. 1. 序言	2 1 9
7. 2. 実験システムの装置構成	
7. 2. 1. 全体構成	2 2 1
7. 2. 2. 路車間データ送受信装置	2 2 1
7. 2. 3. 情報処理装置	2 2 5
7. 3. 実験システムの制御プログラム	
7. 3. 1. 車載機用プログラム	2 2 6
7. 3. 2. 地上局用プログラム	2 2 7
7. 4. 交信干渉の基礎実験	2 3 2
7. 5. 制御方式の動作確認と評価	
7. 5. 1. 動作確認実験	2 3 8
7. 5. 2. 性能評価実験	2 4 4
7. 6. ハイブリッドシミュレーション	
7. 6. 1. ハイブリッドシミュレーションの概念	2 4 8
7. 6. 2. シミュレータ装置構成	2 5 0
7. 6. 3. 実験結果	2 5 9

7. 7. 結言

2 6 5

8. 結 び

8. 1. 研究成果

2 6 6

8. 2. 今後の課題

2 6 8

(謝 辞)

2 6 9

(参 考 文 献)

2 7 0

(付 録)

1 . 序 章

1. 1. 本研究の背景と目的

近年, 急激なモータリゼーションに伴って, 特に都市部において交通渋滞などの諸問題が深刻になってきている。これに対して, 現在の交通管制システムでは, 車両感知器等により計測したデータに基づく路側ラジオや可変情報板等による道路交通情報の提供, 及び, 街路における広域信号制御, また, 都市高速道路入口での流入制限[23]等, 地上側において交通流をマクロに捉えた情報を基にして個々の自動車に対しては一方向的な制御を行うといったものが主流であった。

こういった状況の中で, 近年のエレクトロニクス技術の急速な発展を受けて, 交通管制の分野においてもこれをより積極的に導入することにより道路交通における諸問題の解決を図ろうとするいくつかの新たな試みが世界各国で進められてきた。この中の代表的なものとしては, 各自動車に地上側装置との間で双方向のデータ通信を行う装置を搭載し, これにより個々の自動車の情報を地上側に収集することによりそれらをミ

クロに捉えて交通管理を行おうとするものがある。こういったシステムの実例としては、米国のE R G S [37] や西ドイツのA L I [39], 最近では英国のA U T O G U I D E [49] 等がある。これらのシステムでは道路上の主要交差点に誘導無線や赤外線による近接通信を行うためのスポット状の通信ゾーンを設置し、通信装置を搭載した自動車がそこを通過した時に自動車－地上間の双方向デジタル通信を行う。そして、それを通じて自動車の目的地や各道路リンクの旅行時間などといった自動車の経路選択に必要な情報を地上側に収集して動的なナビゲーションを行うものであり、各々にシミュレーションやフィールド実験などを行って効果が確認されている。

わが国においても、通産省によって自動車総合管制技術の大型プロジェクト(C A C S) [38]のシステムが開発されている。これは、E R G S や A L I 等と同様に、路上に誘導無線による極小の通信ゾーンをスポット的に設けて、自動車－地上間の通信によって、そこを自動車が通過したときに目的地や経路等の情報を双方向に伝送することによって、各々の自動車をミク

ロに捉え、経路誘導等の制御を動的に行うものである。

また最近ではこれをベースにして、その通信仕様において伝送情報量の増大などを図った路車間局地ディジタル通信により、付加価値の高い汎用目的の情報を同時に提供しようとする自動車局地通信総合化システム（A R I E S）の研究も盛んである[47]。

また他方では、建設省と民間企業25社の共同で路車間通信システム（R A C S）の開発も行われている[48]。これは、道路上に配置されたビーコンと車載機の間で準マイクロ波による間欠極小ゾーン通信を行うものであり、各ビーコンとシステムセンタ（中央処理装置）を有線ネットワークで結ぶことにより、ナビゲーション機能、道路交通情報や駐車場情報などの各種の情報サービス機能、及びメッセージ通信や運行管理などの個別通信機能を実現することが検討されている。

これらのシステムでは交通管理手法としては、通信ゾーンを道路のリンク単位程度毎にスポット状に設置して、リンク単位の情報提供収集を行うことにより交差点毎の経路誘導を主体に行うことを考えているが、さらなる発展に向けて、欧州のP R O M E T H E U S

[51]や米国のI V H S [52], わが国のS E 2 0 0 0 [53]に見られるように, 将来において高速道路等で自動運転を行うことまでも考えた大型のプロジェクトも着手されている. しかしながら, 完全な自動運転を通常の走行速度下で行うことは各装置の情報処理能力などの点で未だ困難が多いといわざるをえない[59].

これに対して, 本研究ではその制御の細かさが経路誘導と自動運転の中間にあるような新たな交通管理の手法として, 数十m程度の間隔で地上側において, 個々の自動車の走行情報, すなわち位置, 速度, 目的地等の情報を収集し, その情報を基に速度調整, 車線選択, 方向指示などに役立つ情報を提供して道路のリンク内における個々の自動車の走行をミクロに制御する手法を提案し, "走行誘導"と名付けた. さらに, このために設置する路車間の情報のリンクを利用して地上のデジタル通信網と車載の端末を接続して双方向でデータ通信を行い, 自動車内においても交通管理に限定されない汎用目的の情報サービスが任意の地点で受けられるようにする"汎用データ通信"の機能を実現させることを考える. そしてこれらを可能にする

ために、局地デジタル通信にあるような、同一の周波数を使用するスポット状の通信ゾーンを、隣接ゾーン間の干渉が無視できる程度に接近させて連続的に配置し、そこを通過する複数の自動車と地上の間で一つの通信チャネルを使用しながら、双方向でデジタルデータのやり取りを行ってゆく通信システムを提案し、”小ゾーン連続形自動車パケット通信システム”と名付けた[1]。本研究ではこのシステムを理論的に構想し、その概念設計と通信方式の具体的検討を行いその実現可能性を示すと共に、本システムにより実現される走行誘導や汎用通信といった応用面での効果を示し、その有用性を示すことを目的とする。

1. 2. 本論文の構成

以下、第2章では、システムのコンセプトについて説明する。まず、システムの概要とその機能について述べ、次に本システムの概念上の特徴ともいえる路車間の通信仕様及び、通信ゾーンの構成と配置について説明する。

第3章では、本システムの通信制御方式上の特徴ともいえる1つの道路リンクを管理する道路リンクレベルでの通信制御方式について述べる。この章では極小の通信ゾーンの効率的管理のために自動車の走行情報を利用した接続制御方式や情報伝送方式といった通信制御方式を新たに提案し、ミクロモデルを用いた評価シミュレーションを行ってその有効性を示す。そしてこれに基づいて制御方式の具体的設計を行う。

第4章では、さらに上位層の管理を行うグローバルレベル制御について説明する。この際、個別情報の自立分散型管理概念を導入し、リンク旅行時間情報を利用して制御を行うという新たな発想に基づく制御方式を提案し、ミクロモデルによるシミュレーションにより定量的な評価を行ってその有効性を示す。

第 5 章では、交通管理への応用について述べる。ここでは高速道路分流点での動的車線規制と街路の信号交差点における右折車の待ち合わせ制御といった走行誘導のアルゴリズムを新たに提案する。そしてマイクロモデルを用いた評価シミュレーションにより分流点や交差点におけるボトルネックを改善できることを示す。

第 6 章では、本システムのグローバルなデータ通信システムとしての応用例について述べる。ここではまず、提供可能なグローバルな機能について説明し、次にそれらにより実現可能と考えられるいくつかのシステム例を示す。

第 7 章では、本システムの道路リンクレベルの通信制御方式の実現可能性を示すために構築した、モデル実験システムについて説明する。まず本実験システムを用いて隣接ゾーンからの干渉波の影響について検討し、それを基にハードウェア面の制約条件からみた通信ゾーンの可能な配置間隔を調べる。次に、この実験システムに通信制御のプログラムをインプレメントし、制御方式の動作確認とその定量的評価を行う。さらに、

これをソフトウェアによる交通流シミュレータと組み合わせることによるハイブリッドシミュレーションの概念を提案し、実際の自動車の移動に合わせたタイミングにより通信制御機構の定量的評価を行う。

最後に第8章においては、結論をまとめ今後の課題について触れる。

2 . シ ス テ ム 概 念

2. 1. 序 言

本論文で提案する小ゾーン連続形自動車パケット通信システムは自動車の走行誘導や任意の地点での汎用データ通信を行うために、道路上に局地ディジタル通信の設備を相互の干渉が無視できる程度に接近させて連続的に配置するという、これまでない新しい形態のものである。

そこで本章では、まずこのシステムの概念を説明するために、システムの機能及び構成について述べる。そして本システムの概念上の一つの特徴ともいえる通信ゾーンの形状とその構成法について説明する。さらにハードウェア面からみて通信ゾーンがどの程度まで接近させて配置できるかを、隣接した通信ゾーンからの干渉波の影響を基に理論的に検討し、可能な通信ゾーンの配置間隔を求める。

2. 2. システム説明 [1][2][3][15][16][17]

2. 2. 1. システム概要

本システムでは、走行データと汎用データという2種類のデータを扱う。各々の目的を以下に示す。

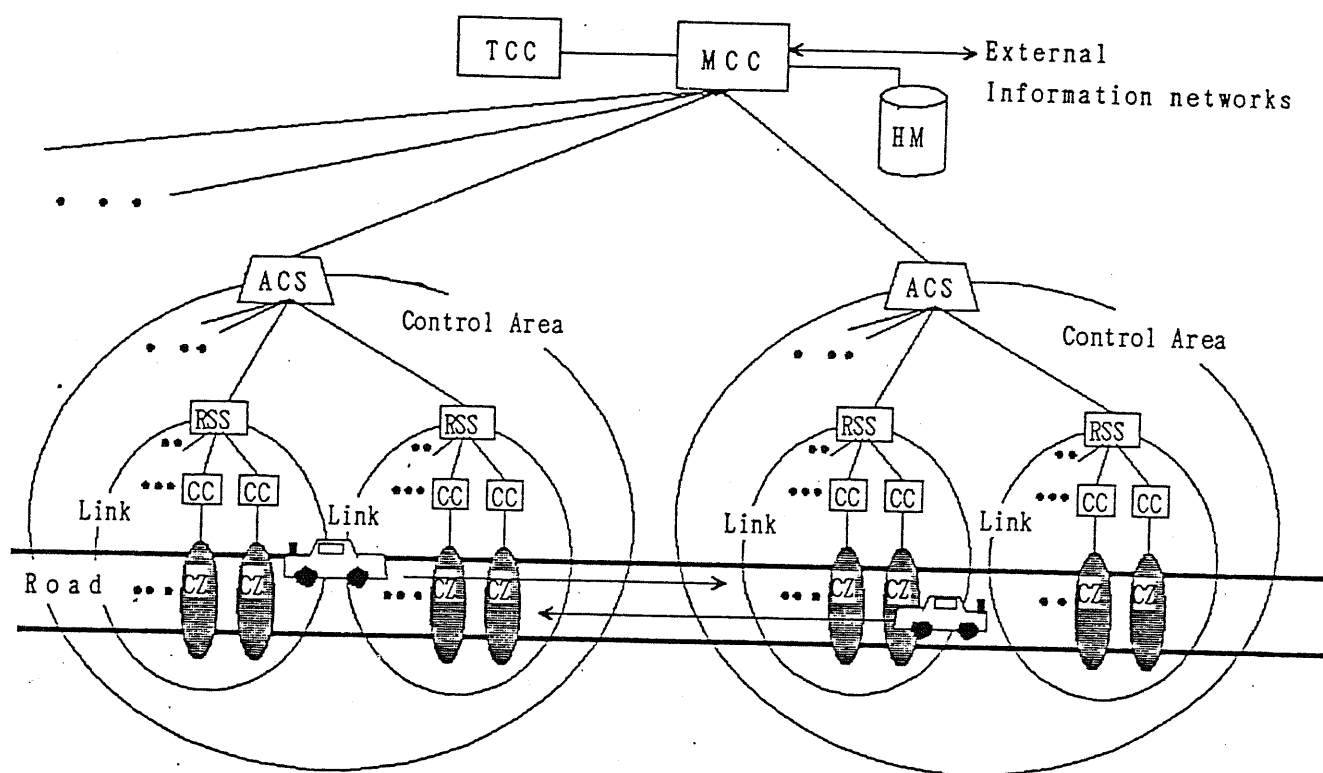
○ 走行データ

各々の道路リンクの内部において自動車の走行誘導を行うために、各通信ゾーン毎に通信されるデータである。従ってこのデータは、情報量は少ないが各通信ゾーン毎に確実に通信される必要がある。

○ 汎用データ

自動車－地上間で上記以外の目的の双方向データ通信サービスを実現するために通信されるデータである。情報量が多く、複数の通信ゾーンにまたがって通信できるが、伝送所要時間が短く、通信のスループットが高いことが望ましい。

本システムの構成概要を図2-2-1に示す。このように、多数の通信ゾーンを効率的に管理するためシ



TCC:Traffic Control Center
 HM:Home Memory
 RSS:Roadside Station
 CZ:Communication Zone

MCC:Mobile Communication Center
 ACS:Area Control Station
 CC:Communication Controller

図 2 - 2 - 1 システムの概要

システムは階層的に構成される。図中の各装置の動作概略は以下の通りである。

各々の通信ゾーンは、通信制御装置 (Communication Controller: C C) によって管理される。各通信制御装置 (C C) は各々の通信ゾーンでの自動車 - 地上間の通信において手順制御を行うとともに誤り制御等を実現する。

路側装置 (Roadside Station: R S S) は一つの道路リンク内にある一群の通信制御装置 (C C) を管理し、各道路リンク内においての走行データ通信と汎用データ通信の制御を行うとともに、走行誘導において自動車へ与える誘導指示を作成する。

いくつかの道路リンクは制御エリア (Control Area) としてまとめられて、各々エリア制御装置 (Area Control Station: A C S) によって管理される。各エリア制御装置 (A C S) は各々の制御エリア内の道路リンクに対する路側装置 (R S S) における走行誘導指示の作成をマクロに管理してゆくとともに、汎用データ packets を各道路リンクの路側制御装置 (R S S) との間で転送したり、複数の道路リンク間にまたがっ

て移動する自動車を追跡して汎用データ通信の継続などを行う。

システム全体は自動車通信中央局 (Mobile Communication Center: M C C) によって管理される。この自動車通信中央局 (M C C) は交通管制センター (Traffic Control Center: T C C) と情報をやりとりしながら道路交通全体を管理してゆく。また、各自動車の位置は制御エリア単位で把握され、この内容はホームメモリ (Home Memory: H M) に登録される。自動車通信中央局 (M C C) はホームメモリ (H M) 中の登録情報をもとに自動車がどのエリアに存在するのかを判断しエリア制御装置 (A C S) との間で汎用データパケットの転送処理を行う。

2. 2. 2. システム機能

ここでは本システムの基本的な機能である走行誘導と汎用通信の各々について説明する。

○ 走行誘導

まず、運転者が車載装置に対して予め目的地を設定

する。自動車が行中、各々の自動車の選択すべき経路が、各道路リンクの入口において次の交差点での右左折直進という形態で指示される。

各々の道路リンク内では自動車が各通信ゾーンを通過する毎に自動車の位置や速度の情報が地上側に伝送される。そして、地上側ではそれを基に車線選択や目標速度に関する指示が作成される。この指示は、各々の車載装置へ伝送される。これを受信した車載装置はその内容を運転者に対してディスプレイに表示して伝達したり、アクチュエータに対する制御を行うことにより、各道路リンク内における自動車の走行がミクロに制御される。

○ 汎用通信

本システムでは、走行誘導を行うために数十m程度の間隔で頻繁に走行データ通信を行うことは先にも述べた。汎用通信はこのために道路上に稠密に設けられた通信ゾーンによる路車間の情報のリンクを利用して自動車－地上間で交通管理以外の汎用目的の情報のやり取りを行うものである。このために、地上側のユー

ザの端末を情報ネットワーク等を介して本システムの地上側装置に接続する。さらに、地上側装置に複数の通信ゾーンにわたっての自動車の追跡や切り換えなどの機能を付加する。一方、車載装置を車内の情報端末と接続する。これにより、地上側ユーザまたは車載側ユーザからの要求により、双方向でデジタルデータのやり取りを行うことができる。

2. 2. 3. システム構成

本節では、本システムの構成について述べる。まず地上側装置について、その基本的な機能である走行誘導、及び、汎用通信の2つに関して、各々を実現するためのシステム構成を説明する。

○ 走行誘導機能

本システムにおいて、走行誘導機能に関する部分の構成を図2-2-2に示す。自動車が一つの道路リンクに進入すると、次の交差点における右左折直進の経路情報はCCを経由してRSSに転送される。この情報は、位置・速度の情報とともにRSSにおいてその

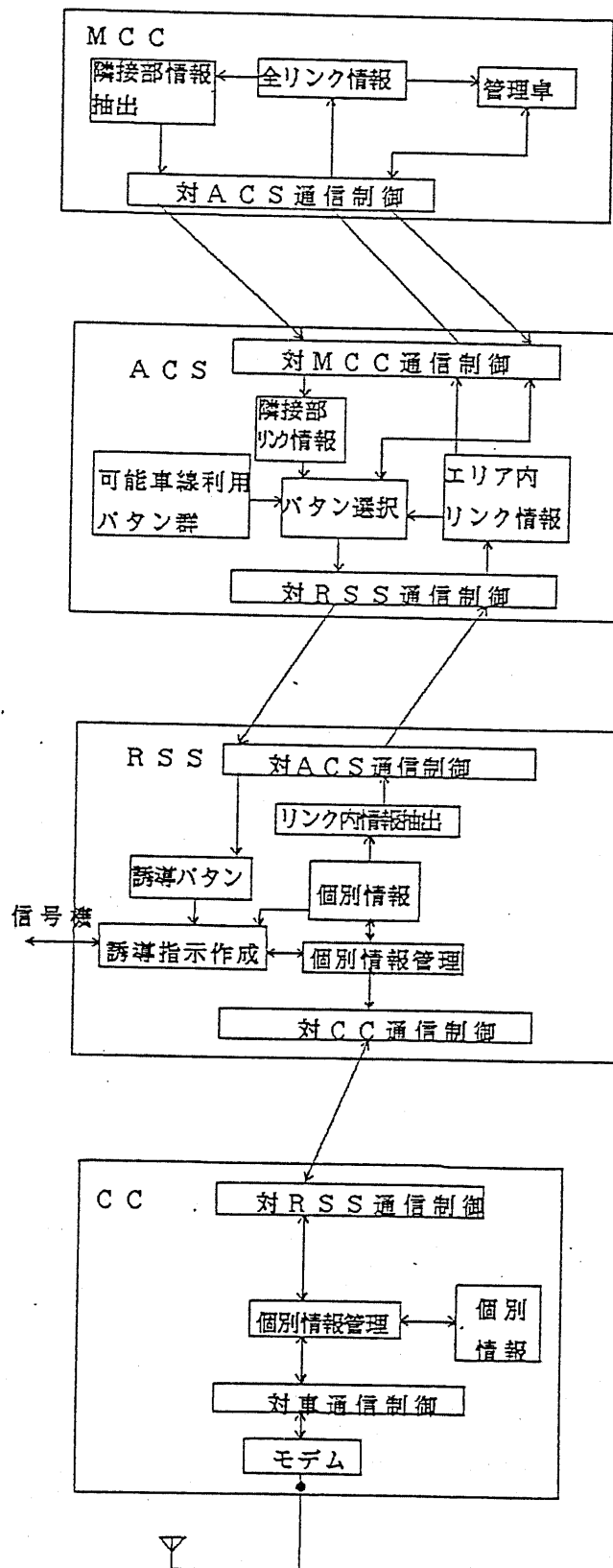


図 2 - 2 - 2 走行誘導機能の構成

自動車の個別情報として管理される。

リンク内の各々の車両は各通信ゾーン毎に、上りの走行データの通信において位置や速度の情報を伝送する。これらの情報はCCを經由してRSSに転送され、当該自動車の個別情報が更新される。RSSではこれに応じて、各リンクで定められている車線利用のパターンに基づき、信号タイミングを考慮した上で、車線選択・速度調整に関する指示を作成する。そしてこれを下りの走行データの通信において、後述の情報伝送方式に従って、CCを經由し自動車に返送する。各自動車はこれに基づいて、車線変更や加減速を行いながら道路上を走行することにより走行誘導が実現される。

一方、地上側においては、各々の道路リンクにおける目的方向別の車両台数等のリンク内情報をACS及びMCCに収集する。この時、ACSではあらかじめ各交差点毎に可能な車線利用パターンのメニューをっており、各交差点毎に、その周辺リンクの情報に基づいて、どの車線利用パターンを用いて車両を捌けばよいかを選択する。但し当該エリアに隣接した他エリアに含まれる部分の道路リンクに関する情報はMCCより

伝達される。選択された車線利用パターンは R S S に伝送されて誘導指示の作成が行われる。

○ 汎用データ通信機能

汎用データ通信機能に関する部分のシステム構成を図 2 - 2 - 3 に示す。汎用データには車載機から地上へ伝送する上り汎用データとその逆の下り汎用データがある。上りデータの伝送は C C, R S S, A C S, M C C 各装置のデータバッファを経由して地上側のユーザへ伝送される。下りデータを伝送するには、地上側において自動車の存在するゾーンを把握する必要がある。ここでは M C C, A C S, R S S の各々において階層的に各装置の担当する領域内での自動車の位置を把握させ自動車を追跡させることとしている。

次に、車載装置について説明する。その構成を図 2 - 2 - 4 に示す。車載装置は大きく分けて、計測部、通信制御部、表示部、及び、アンテナ部から成り立つ。計測部では車両の速度計で走行速度データを計測し、これを通信制御部を通じて地上側に伝送する。地上側からの誘導指示も通信制御部を通じて受信される。こ

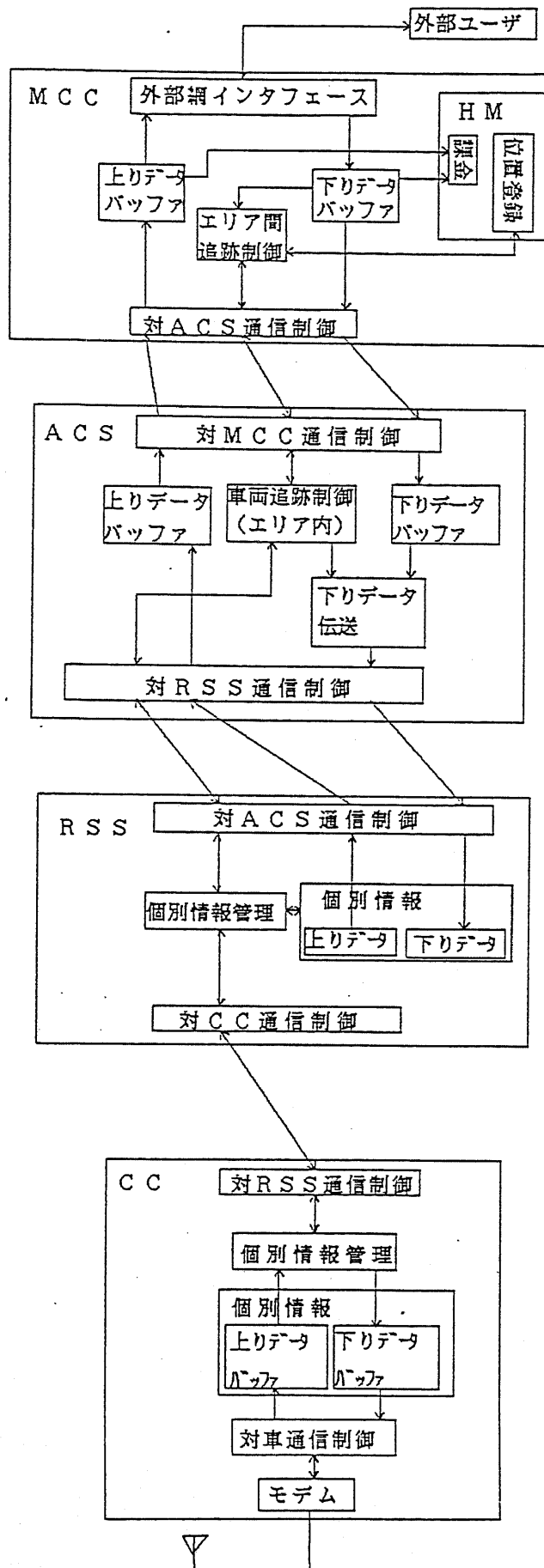


図 2 - 2 - 3 汎用データ通信機能の構成

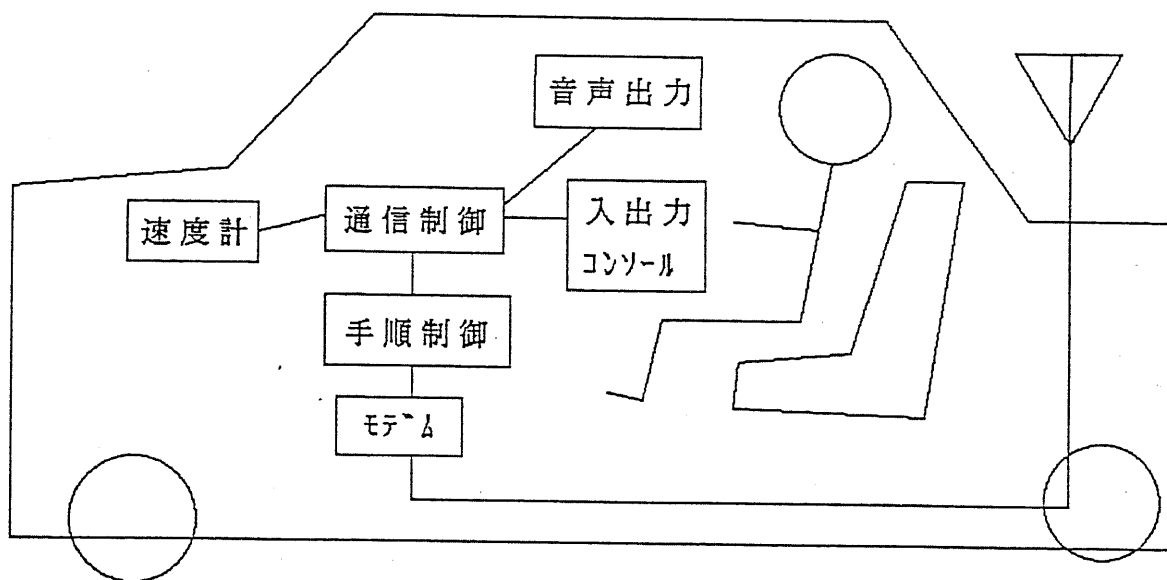


図 2 - 2 - 4 車載装置の構成

れは、表示部へ伝達され、車内のディスプレイ（コンソール）やアクチュエータに出力される。一方、コンソールを汎用目的の利用に対応できるものとする事により、これを通じて地上側のユーザと双方向で汎用目的のデータのやり取りを行う。

2. 3. 路車間通信仕様 [1][3][15][16][17]

一般に、道路上に十 m 程度の極小の通信ゾーンを構成でき、かつ、ゾーンの外への不用な漏洩電界が少なくなるような方式としては、[44]において検討されているように、以下の2通りの方式がある。

① 誘導通信方式

通常、高周波電流の流れている導体からは、放射電界、誘導電界、及び静電界の3種類の電界が発生する。低い周波数帯では誘導電界と静電界が主であり、誘導電界は距離の2乗に、静電界は距離の3乗に反比例して減衰するため遠距離へは到達しない。そこで、それほど高くない周波数帯を用いて放射電界を抑制することにより近距離通信用を行う方式が考えられる。

② 準マイクロ波方式

この方式では直進性の強い準マイクロ波帯の電波を利用する。そして、路側の若干高い位置に道路を見下ろすような形で指向性アンテナを設置し、それよりビーム状に電波を放射して、道路の比較的狭い範囲をカ

バーして、近距離通信を行うものである。

①の誘導通信方式は干渉波妨害が少なく、また、既の実システムとしての多年の実績があり、データ通信方式そのものとしての技術的課題は少ない。

②の準マイクロ波方式は、搬送波周波数が高く広い伝送帯域を取ることができ、高速データ伝送が可能であるので、このような近接検知型の自動車通信システムとして利用が最近盛んになりつつある。

本システムではこれらのいずれの方式を用いてもその特徴を活かすことができるが、実現可能性を実証するため、つくば自動車データ収集提供システム[43]等で実際に用いられて既に技術的に確立した①の誘導通信方式を用いることとする。そして表2-3-1に示したような路車間通信の標準仕様に従って、自動車-地上間で9600bpsの双方向データ通信を行うこととする。

表 2 - 3 - 1 路 車 間 通 信 仕 様

搬 送 周 波 数	車 → 路	2 2 3 . 2 k H z
	路 → 車	3 0 4 . 8 k H z
伝 送 速 度	9 6 0 0 b p s	
変 調 方 式	M S K	
フ レ ム 構 成	H D L C	
誤 り 制 御	検 出	C R C
	訂 正	再 送 信

2. 4. 通信ゾーン [1][3][15][16][17]

2. 4. 1. 通信ゾーン構成

先に述べた自動車総合管制技術 (C A C S) のシステム [38]では, 通信ゾーンの大きさを $2\text{ m} \times 3\text{ m}$ 程度に極小化することによってシステムを実現している.

そして, このシステムでは, 都市の幹線道路のように複数の車線から構成される道路において適用する場合には, 一箇所の通信地点を構成するために複数の通信ゾーンを多少前後にずらして配置し, 道路の全車線を網羅するようにしている.

しかし, このように複数の極小通信ゾーンにより道路の全車線をカバーする方式では,

(1) 自動車の通過位置によっては, 隣接した通信ゾーン同士の間をすり抜ける場合がある.

(2) 自動車が通信ゾーンに存在する時間が短いため, 通信チャネルの使用効率が極端に悪くなる.

等の欠点がある [35][36].

これに対して本システムにおいては、[35][36]に示されているように通信ゾーンを道路の車線全体にまたがるようにしてすり抜けを防ぎ、また、通信ゾーンの長さを車頭間隔と同程度の10mとすることによってチャンネルの使用効率を改善する。

ここでは、このような通信ゾーンを都市の幹線道路において連続的に配置するための、実現可能と考えられる一つの構成案を示すことにする。

通常、誘導通信帯において用いられるアンテナとしては、以下の2通りの方式がある。

① ループアンテナを路面に埋設する。

② フェライトバーアンテナをオーバーヘッドまたはサイドファイヤで使用する。

まず、①のループアンテナを路面に埋設する場合を考えると、全車線にまたがるループアンテナを幹線道路上に連続的に埋設するような工事を行うことは現実的には不可能と言える。

従って、本システムでは②のフェライトバーアンテナを使用して通信ゾーンを構成することを考える。このフェライトバーアンテナを、図2-4-1のように、道路の両側の路側と中央分離帯の3カ所に設置する。そして、図2-4-2に示したように、この3個のアンテナの巻線を等しい信号で励振することによって、合成電界によって目標とする、全車線にまたがる長さ10mの通信ゾーンを構成することとする。

この時、各アンテナからの電界の位相差が問題となり得るが、これについては、本システムで仮定した誘導通信帯では波長が、空中では1km程度であり、ケーブル中でもその1/1.1倍程度である[113]のに対して、ゾーンの長さが10m程度であることを考えると各アンテナからの信号の位相差については殆ど無視できると考えられる。

本システムで採用している路車間の局地デジタル通信における標準仕様に基づく試作機の開発内容を掲載している文献[40]においてはいくつかのハードウェアレベルでの実験結果が報告されている。ここでは通信ゾーンの構成を考える際に、その中で富士通㈱によ

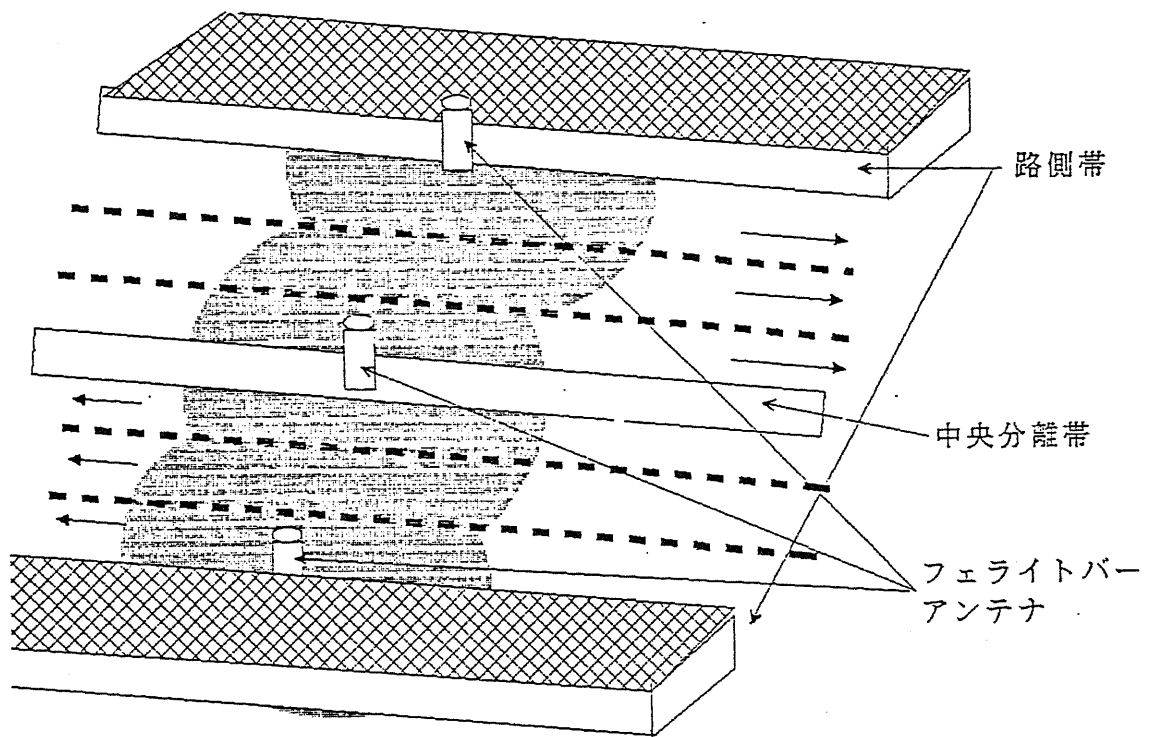


図 2 - 4 - 1 通信ゾーンの構成法

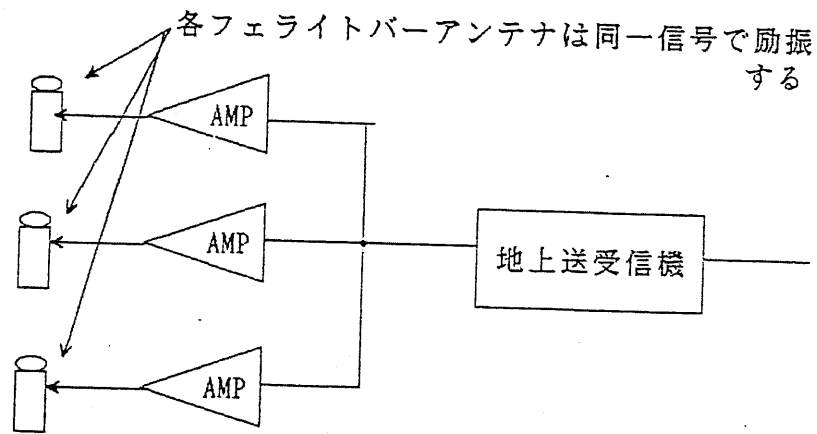


図 2 - 4 - 2 各アンテナの接続法

って報告されているフェライトバーアンテナの特性を参考にした。このアンテナの諸元は表 2 - 4 - 1 に示したようなものである。

文献 [40] 中のアンテナ特性の測定結果よりこのアンテナを自動車のルーフサイドに取り付けて、単独で用いた場合、

I 送信電流 [mA_{p-p}]

r アンテナからの距離 [m]

E 電界強度 [μ V/m]

として比例定数 K を

$$K = 9.265 \times 10^3$$

... (式 2 - 4 - 1)

と定めることにより電界 E [μ V/m] は、

$$E = K \times I \times r^{-2.8}$$

... (式 2 - 4 - 2)

表 2 - 4 - 1 フェライトバーアンテナの諸元

項 目		数 値
形 状	直 径	1 5 ϕ
	長 さ	2 0 0 [mm]
巻 線 巻 数		1 0 0 回
自 己 イ ン ダ ク タ ン ス		1 5 0 [mH]
適 用 周 波 数		2 0 0 ~ 3 0 0 [kHz]

と表されることがわかっている。

このアンテナを図 2 - 4 - 3 に示したように配置して、それぞれに $840 \text{ [mA}_{p-p}]$ の高周波電流を流した場合の電界強度のパターンを図 2 - 4 - 4 に示す。一方、図 2 - 4 - 5 はこれを立体的に表現したものである。但し、ここで道路幅等のデータは [28] を参考とした。

図 2 - 4 - 4 より、 $90 \text{ [dB}\mu\text{V/m]}$ の電界が、道路の車線全体にまたがってほぼ 10 m の長さで形成されているのがわかる。また図 2 - 4 - 5 より通信ゾーンから離れるにしたがって漏洩電界が急速に減衰しているのがわかる。

本システムでは文献 [43] の路車間局地ディジタル通信の標準仕様に従って受信最小レベルを $93 \text{ [dB}\mu\text{V/m]}$ としている。よって、路上機送信時はこれより 3 dB 高い 1190 [mA] の電流を流すことによって目標とする道路の車線全体にまたがる長さ 10 m の通信ゾーンが構成できる。

ここで以上の検討は地上側送信時に関するものであるが、車載機送信時もアンテナの可逆性によって、

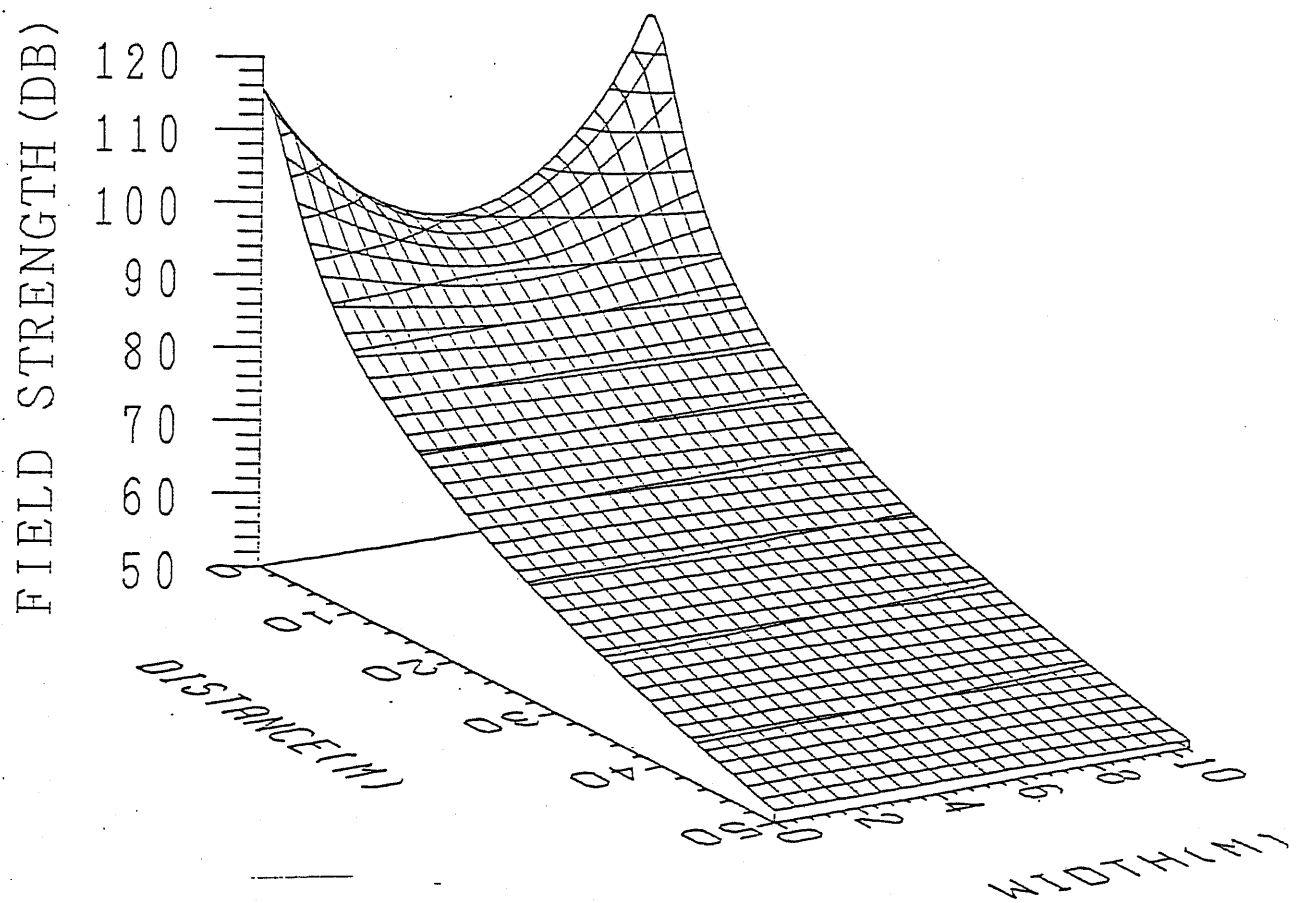


図 2 - 4 - 5 電界のパターン (立体図)

同様なゾーン形状となることはあきらかである。

2. 4. 2. 通信ゾーン配置間隔

本システムで採用した表 2 - 3 - 1 の路車間局地デジタル通信の標準仕様では自動車 - 地上間の誘導無線によってデジタルデータの伝送を行うための通信方式として、位相連続周波数変調方式 (Minimum Shift Keying) を採用している。

一般的に用いられている、MSK の遅延検波回路の構成例を図 2 - 4 - 6 に示す。

MSK において遅延検波を行う場合の誤り率は、雑音を白色ガウス雑音と仮定すると、

符号間の位相変移 $\cdots \Delta \Phi$

$S/N \cdots \gamma$

において

$$P_e = (1/2) \times \{1 - \sin \Delta \Phi \times I_e(\cos \Delta \Phi, \gamma)\}$$

\cdots (式 2 - 4 - 3)

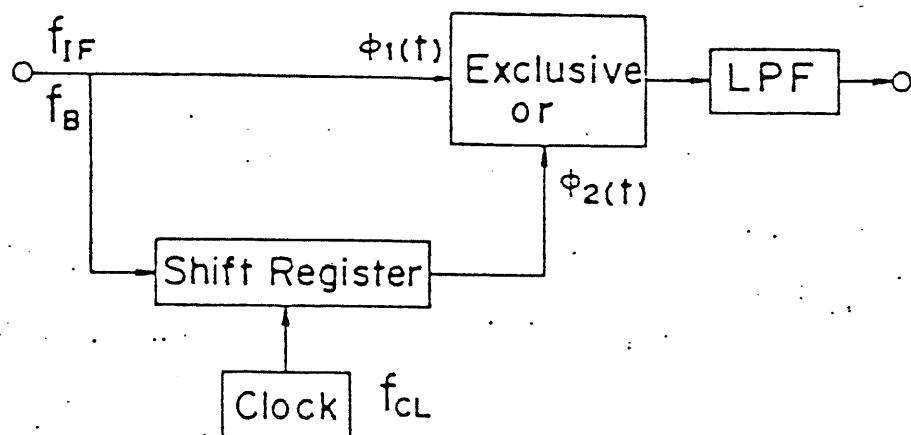


図 2 - 4 - 6 M S K 遅延検波回路の構成例 [62]

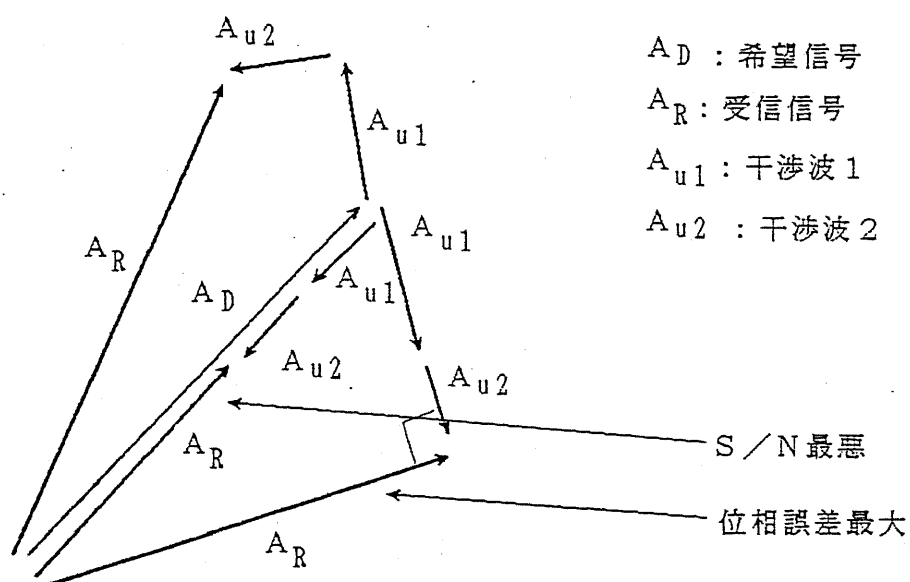


図 2 - 4 - 7 信号のベクトル図

となることが知られている [60][61]. 但し, I_e はライ
スの I_e 関数を表している.

通常, 雑音以外の要因による位相誤差として考えら
れるものには,

シフトレジスタクロックが入力信号と非同期

であるために生じるもの $\dots\dots\dots \Delta \Phi_A$

干渉波によるもの $\dots\dots\dots \Delta \Phi_B$

がある. ここで,

I F 周波数 $\dots\dots\dots f_{IF}$

データ伝送のビットレート $\dots\dots\dots f_B$

シフトレジスタのクロック周波数 $\dots f_{CL}$

とすると, $\Delta \Phi_A$ は,

$$\Delta \Phi_A = (\pi / 2) \times \{ f_{IF} \pm (f_B / 4) \} / f_{CL}$$

$\dots\dots\dots$ (式 2 - 4 - 4)

で表されることが知られている [62].

ここで, 文献 [40]において検討されているものの中で
富士通 (株) のものを例にとると, $\Delta \Phi_A$ の値は,

$$\Delta \Phi_A = 8.3 \times 10^{-3} \pi \sim 1.3 \times 10^{-2} \pi$$

... (式 2 - 4 - 5)

となっている.

干渉波としては, 簡単のため左右の隣接したゾーン
からの2つのみを考える. そして, この2つの D/U
を振幅比に換算し, その平均をとったものを $\gamma_{D/U}$ とす
る.

信号の位相・振幅の関係をベクトル図に表したものが
図 2 - 4 - 7 である. 図より明らかに, 位相誤差が
最悪になるのは,

$$\Delta \Phi_B = 2 \times \sin^{-1} \{ 2 \times \sqrt{(1 / \gamma_{D/U})} \}$$

... (式 2 - 4 - 6)

となる場合である。

また、平均の S/N を γ_0 とすると瞬時的な S/N の最悪値 γ_{MIN} は図 2-4-7 より明らかに、

$$\gamma_{MIN} = \gamma_0 \times \{ 1 - 4 \times (1 / \gamma_{D/U}) \} \dots (式 2-4-7)$$

となる。

以上より干渉波存在下での誤り率 Pe_u は、

$$Pe_u = Pe \{ (\pi / 2) - (\Delta \Phi_A + \Delta \Phi_B), \gamma_{MIN} \} \dots (式 2-4-8)$$

によって求めることができる。これに従って、 S/N をパラメータとして、 D/U と誤り率の関係を求めたものが図 2-4-8 である。

2. 4. 1. でも述べたように、本システムでは最小受信レベルを $93 [dB \mu V/m]$ としている。また、都市雑音のレベルは文献 [40] によれば、新幹線の真下等の特殊な地点を除けば、ほぼ $70 [dB \mu V/m]$ 以下になって

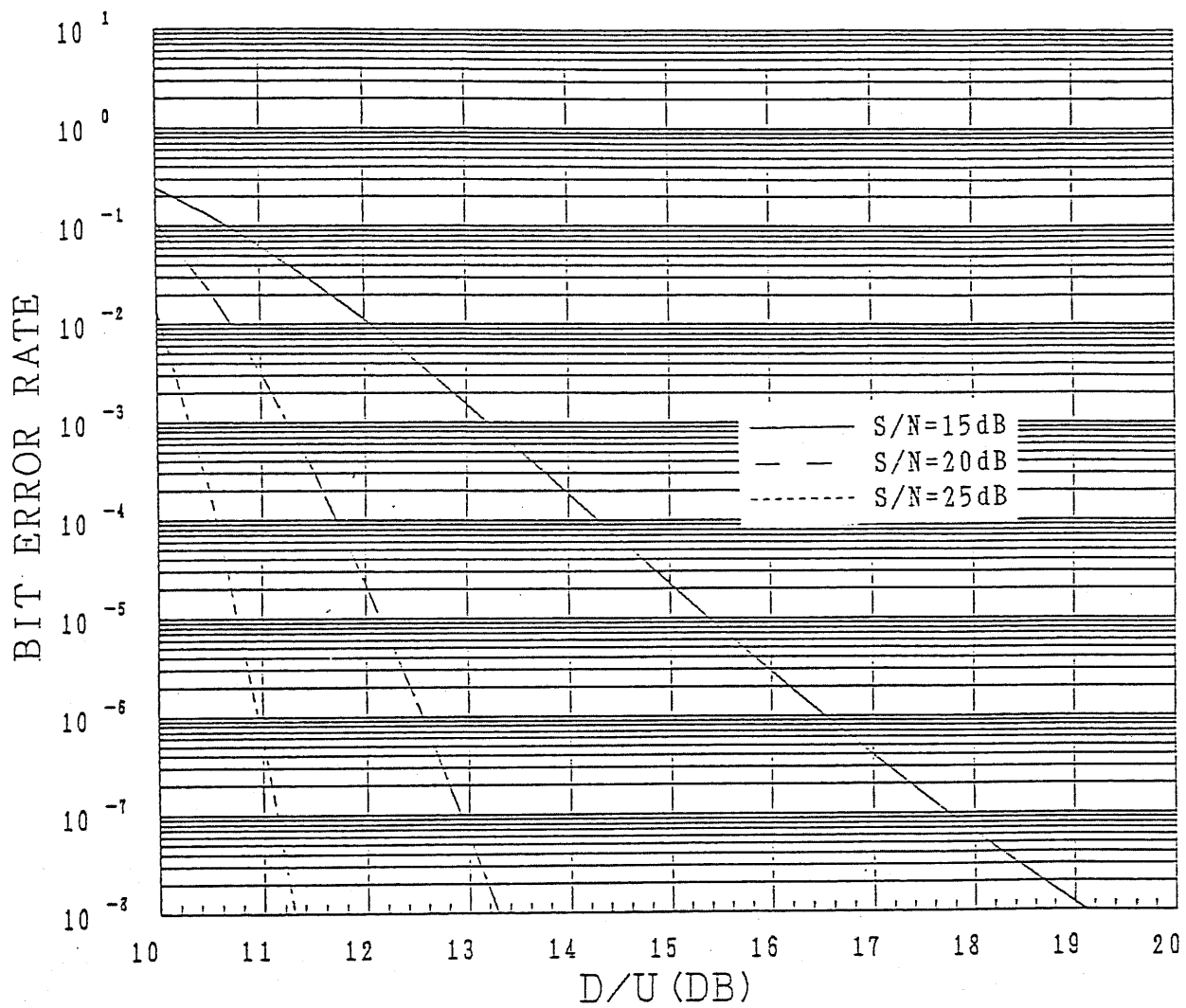


図 2 - 4 - 8 MSK の対干渉波特性

いることがわかる。

これより、本システムにおいては、 S/N は、ほぼ
20 dB以上であると考えられる。本システムでは[40]
における検討内容と同様に許容最大誤り率を 10^{-6} と
した。そして、これを実現するため、図2-4-8よ
り、 D/U の限界値を13 dBとした。

ここでは以上の結果を基に、路上機受信時と車載機
受信時に場合分けして、その各々について、路上アン
テナをどの程度接近させて配置できるかを検討した。

① 路上機受信時

①-A 隣接ゾーンの路上アンテナよりの干渉波

当該通信ゾーンの両側に隣接した通信ゾーンの路上
アンテナからの距離は等しいので、それらからの干渉
波の振幅は等しいと考えられる。従って、図2-4-
4より他の路上アンテナからの距離が15 m離れた時
に D/U が13 dB以下となる。

①-B 隣接ゾーンの車載アンテナからの干渉波

この場合、干渉波を発生している他の通信ゾーンの

自動車のアンテナが、両側の通信ゾーンの中の、干渉を受けている通信ゾーンの路上アンテナにもっとも近い位置にある時に干渉波が最大となる。

図 2 - 4 - 4 より、干渉波を発生する車載アンテナが、干渉を受けている路上アンテナから 15 m 離れたときに D/U が 13 dB で最悪となる。この時は、通信ゾーンの幅が、端部の広がった部分において $2 \times 7 = 14$ m あるので路上アンテナの間隔は $15 + 7 = 22$ m 以上あればよいことがわかる。

② 車載機受信時

② - A 隣接ゾーンの路上アンテナよりの干渉波

この場合は、アンテナの可逆性によって① - B の場合と同じであると考えられる。

② - B 隣接ゾーンの車載アンテナからの干渉波

最悪の状態としては、図 2 - 4 - 9 に示したように、両側の通信ゾーンにある車載アンテナが干渉を受けている通信ゾーンの路上アンテナに最も近付いた状態を考えればよい。

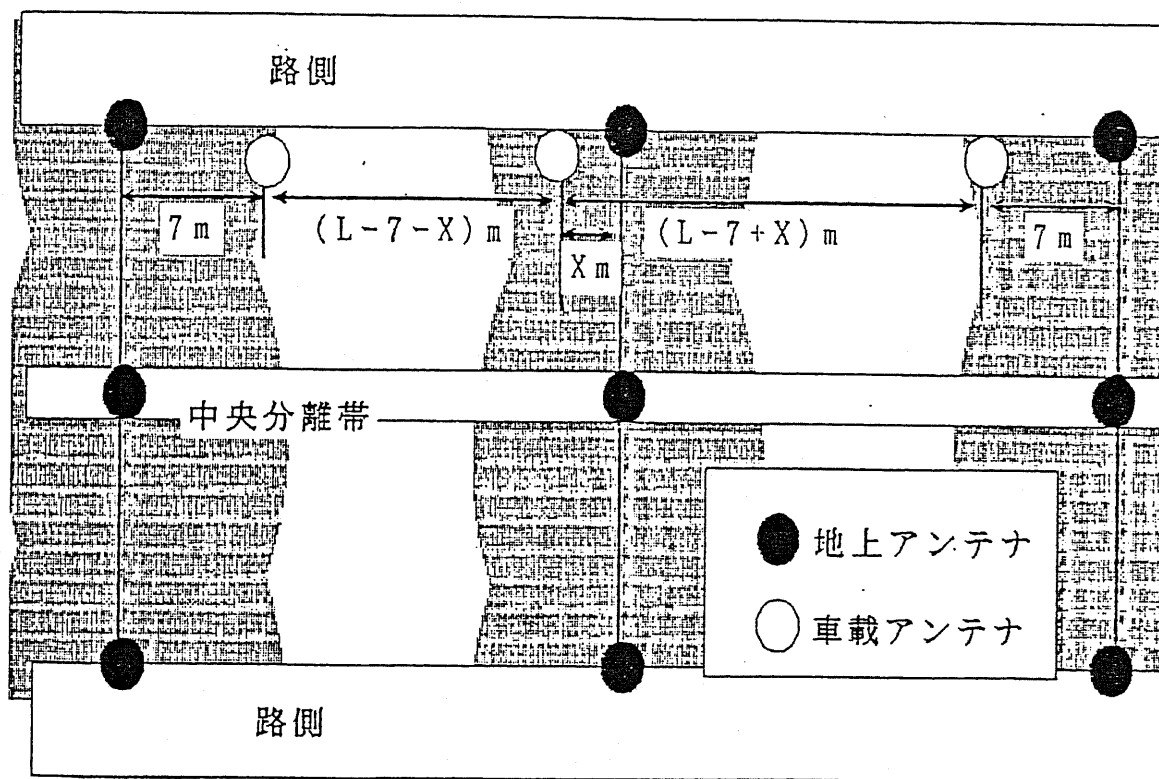


図 2 - 4 - 9 最悪時のアンテナの位置関係

この時、両側の2つの車載アンテナからの干渉波のレベルの平均が、 $93 - 13 = 80$ [dB μ V/m]以内であることが必要となる。これを式で表すと、前出の計算で用いた定数 K と巻線電流 I によって、

$$\begin{aligned} & \{ K \times I \times (L - 7 + X)^{-2.8} \\ & + K \times I \times (L - 7 - X)^{-2.8} \} / 2 \\ & \leq 80 \text{ [dB } \mu \text{ V/m]} \\ & \dots (\text{式 } 2 - 4 - 9) \end{aligned}$$

が $\{-7 \leq X \leq 7\}$ なるすべての X について成り立てばよいことになる。この時、L は隣接した路上アンテナ同士の間隔であり、この不等式を解くと、

$$L \geq 26 \text{ m} \quad \dots (\text{式 } 2 - 4 - 10)$$

となる。

以上①②を総合して路上アンテナは26 m以上離して配置すればよいことがわかった。

2. 5. 結 言

本章では、新たに提案した小ゾーン連続形自動車パケット通信システムの機能及び構成について述べた。そして、既の実績のある誘導通信帯を用いての通信ゾーンの構成法について理論的に検討し、可能と考えられる例を示した。

また、ハードウェア面からみて通信ゾーンがどの程度まで接近させて配置できるかを理論的に求めた。この議論に関しては、後の第7章において、室内においてではあるが、実験を行って妥当性を確認している。

今後は、機能及び通信ゾーンの構成法の両面にわたって現実的な環境条件等に即した検討を行ってゆくことが必要と思われる。

3 . 道 路 リ ン ク レ ベ ル 制 御

3. 1. 序 言

先の第2章では、本研究で新たに提案した小ゾーン連続形自動車パケット通信システムの概念について説明を行った。この中で述べたように、本システムではその構成において、同一の通信チャネルを共用する道路の全車線にまたがる長さ10m程度の通信ゾーンを相互の干渉が無視できる程度に接近させて連続的に配置するというこれまでにない形態を取っている。従って、その通信制御方式においては各自動車ごく短時間のうちに通信ゾーンを次々と移行してゆくといった状況の中で、複数の自動車同士がこれらの通信ゾーンにより提供される通信チャネルを時分割で効率よく共用し、しかも、各自動車の走行誘導のためのデータなどの即所性の要求される内容のデータは要求のあった通信ゾーンにおいて確実に通信できる機能が必要になる。

本研究では、上記のような機能を実現させるための同一の道路リンク内に配置されている通信ゾーンにお

ける通信の管理を、各通信ゾーンの通信制御装置（C
C）と、一つの路側装置（RSS）に行わせることと
し、このための制御を道路リンクレベル制御と名付け
た。

本章では、この道路リンクレベル制御について述べ
る。まず、その機能についてまとめ、次にその中にお
いて新たに提案した、データの分割による伝送方式、
接続制御方式、及び、情報伝送方式について説明する。
そして、その効果をマイクロモデルを用いた評価シミュ
レーションによって示す。更に以降の節では、これら
を盛り込んだ制御方式の具体的設計を行う。

3. 2. 道路リンクレベル制御の機能 [1][3][15][16] [17]

3. 2. 1. 機能概要

本章で述べる道路リンクレベル制御の機能には、道路リンク内の各通信ゾーン毎に確実に自動車を捉えて、車載側からの位置や速度を地上側に伝送し、地上側からの速度調整や車線選択に関する指示を車載側へ伝達する走行データ通信の機能（3. 2. 2. に詳述）、及び、地上側または車載側からの要求に応じていくつかの通信ゾーンにまたがって汎用目的のデータのやり取りを行う汎用データ通信の機能（3. 2. 3. に詳述）が必要になる。

この時、本システムのように、通信ゾーンが小さく、自動車がごく短時間のうちに、通信ゾーンを次々と移行してゆく場合には、通常考えられるように、自動車が通信ゾーンを変わるごとに情報量の多い登録番号を用いて各々の自動車を識別し、さらにその自動車を捉えている地上局を切り換えてゆくのでは、そのための地上側システムの負担が極めて大きくなる。

そこで、本システムではまず、一つの道路リンク単

位程度の通信ゾーン群を一つのRSSでまとめて制御する。そして、図3-2-1に示したように、まとめて制御されているこれらの通信ゾーン群の中で、各々の道路リンク入口にある最初の通信ゾーンにおいて登録番号などによって新たに入ってくる自動車を捉えた際にトークンを与え、以後当該自動車がその通信ゾーン群から出るまでその識別にはこのトークンを用いることによってそのための情報のオーバーヘッドを減少させる。

また、入口の通信ゾーンにおいては登録番号による自動車の識別及び経路誘導、また、トークンの付与のみを行うことにより新たに通信ゾーンに入ってくる自動車を確実に捉える。そして、汎用データの通信はリンク中間部の通信ゾーンにおいてのみ行うこととする。

以降の節で説明する本システムの主要な機能であるところの、走行データ通信、及び、汎用データ通信は、このトークンによる車両の管理に基づいている。

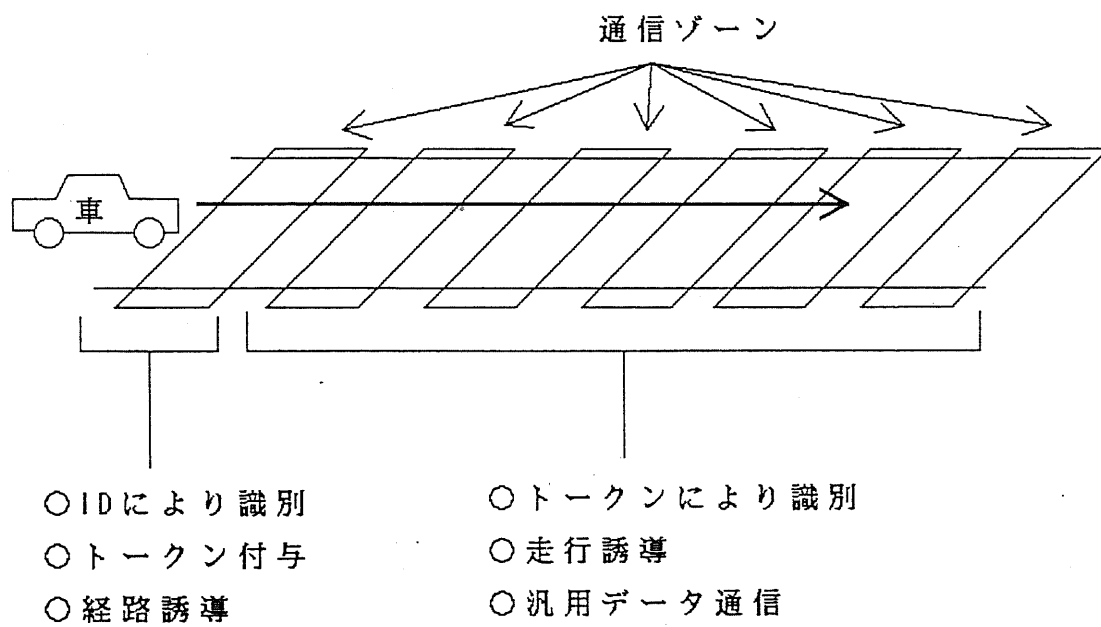


図 3 - 2 - 1 トークンによる車両の識別

3. 2. 2. 走行データ通信機能

走行データ通信機能とは、2. 2. 2. で述べた走行誘導を実現するための走行データを自動車－地上間で通信する機能である。このためには、図3-2-2に示したように以下の動作を行う。

(1) トークンによる車両の識別

自動車が各々の通信ゾーンに進入する毎にトークンによって自動車を識別する。

(2) 上りデータ通信

自動車の速度などの情報を地上のCCに伝送する。

(3) RSSへのデータ転送

CCで受信した各々の自動車からの情報をRSSへ転送する。

(4) 誘導指示作成

自動車からの情報に基づいてRSSにおいて誘導指示を作成する。

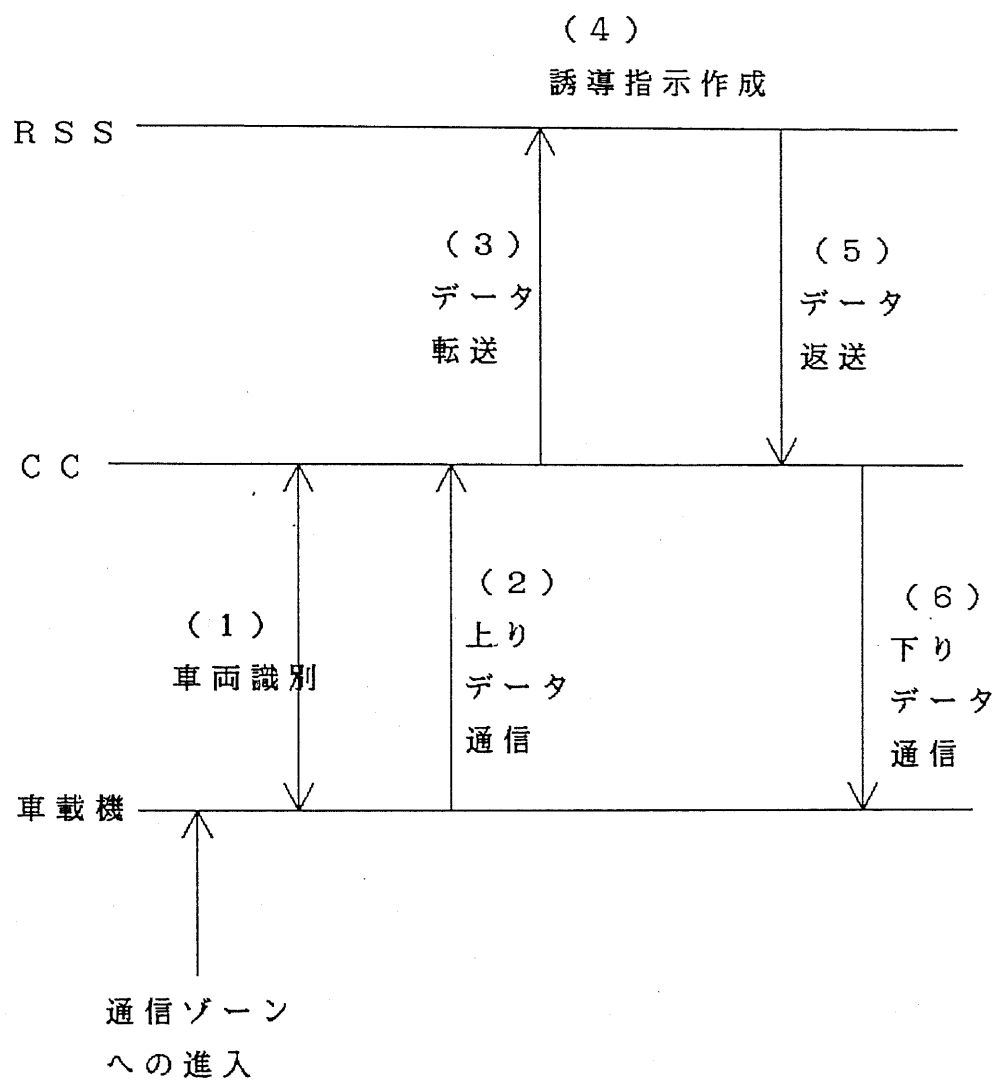


図 3 - 2 - 2 1 つの通信ゾーンにおける走行データ通信

(5) C C へ の デ ー タ 返 送

R S S に お い て 作 成 し た 指 示 を C C へ 返 送 す る.

(6) 下 り デ ー タ 通 信

C C よ り 自 動 車 へ 誘 導 指 示 を 伝 送 す る.

3. 2. 3. 汎 用 デ ー タ 通 信 機 能

汎 用 デ ー タ 通 信 機 能 に お い て は, 2. 2. 2. で 述べたように汎用目的のデータを自動車 - 地上間で伝送する. この際, 以下の動作を行う必要がある.

(1) デ ー タ の 小 分 割

3. 3. に お い て 述 べ る よ う に, デ ー タ を 5 0 0 b i t 程度の小パケットに分割して伝送することにより, 自動車がデータ伝送中に通信ゾーンから出た場合のパケット中断によるロスが少なくなるようにする.

(2) 通 信 の 継 続

い く つ か の 通 信 ゾ ー ン に ま た が っ て デ ー タ を 伝 送 す

る際に、パケットの重複や抜けが無いようにする。

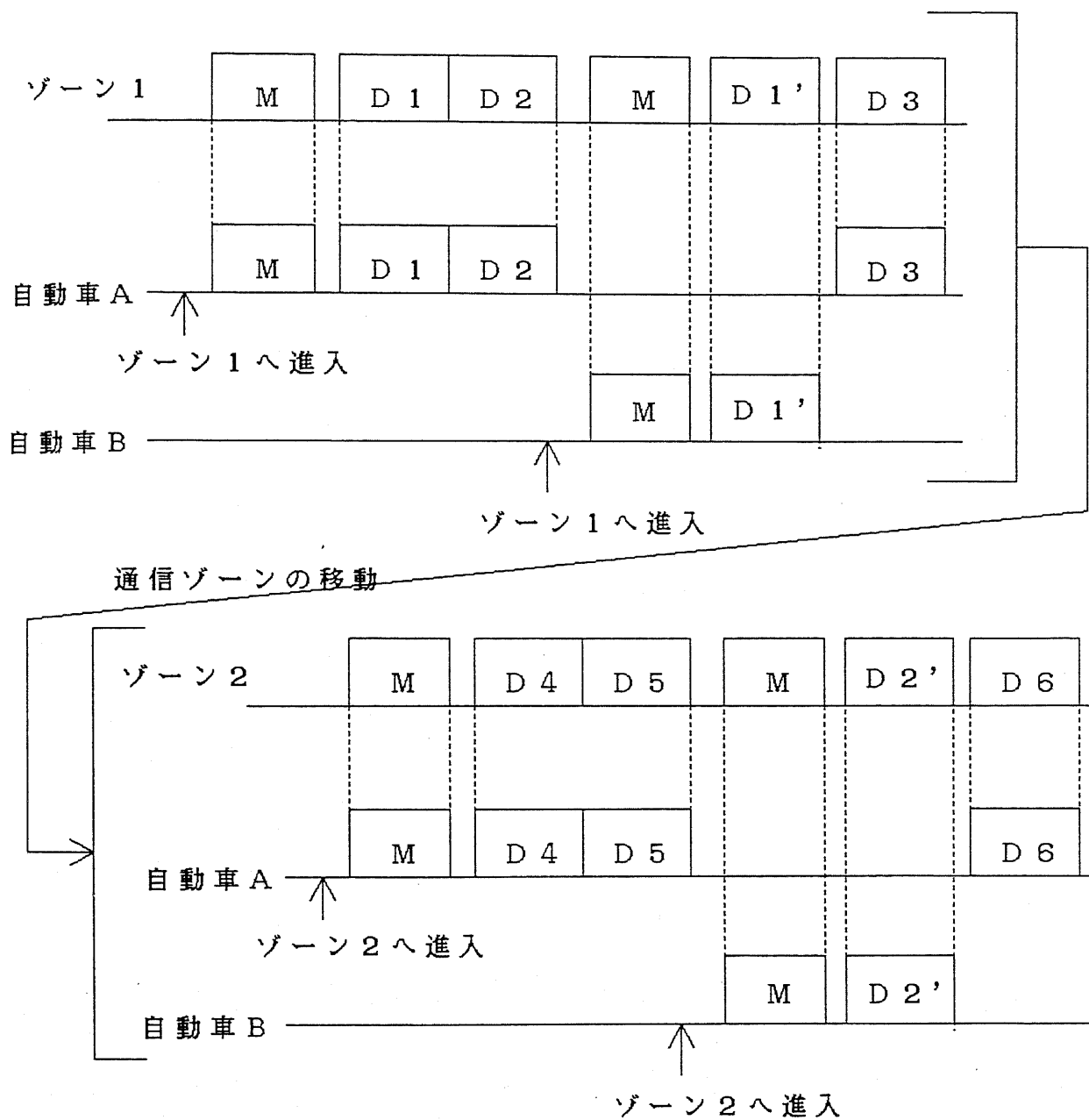
(3) 優先度の考慮

汎用データはいくつかの通信ゾーンにまたがって伝送できるのに対して、走行データでは即所性が強いために各々の通信ゾーン毎に確実に伝送される必要がある。従って、ある自動車が汎用データの通信中であっても、他の自動車が通信ゾーンへ進入したときはこれに割り込んで走行データの通信ができるようにする。

(4) 通信チャネルの共用

複数の自動車が同一の通信ゾーン内に存在し得るのでそれらの間で通信チャネルを共用する必要がある。

以上の動作が行われている様子を図 3 - 2 - 3 に例示する。



M : 走行データ通信

D i : i 番目の汎用データパケット通信

図 3 - 2 - 3 汎用データ通信機能の説明図

3. 3. データ小分割による伝送方式 [1][3][4][10][15][16][17]

3. 3. 1. 方式説明

本システムのように各通信ゾーンの大きさが小さいと、データを一つのパケットで伝送する方式では、データ量が増してくるとパケットの伝送中に自動車は通信ゾーンから出てしまうため、大量のデータは通信できない。

それに対し、本研究ではまず、通信ゾーンがほぼ連続して並べられているという利点を利用して、大量のデータはいくつかの小ブロックに分割しておき、その伝送中に自動車が通信ゾーンから出てしまった場合には、次の通信ゾーンにおいてそのブロックから再送信することにより効率のよい通信が行えることを理論的に示している [9][15]。また、データを一つのパケットで伝送する方式では、先に通信している車が通信を終えるか、ゾーンから出るまで、後続の自動車が通信できないことになる。このため、先に通信している自動車の速度が遅く、後続車が通信ゾーン内において、その自動車を追い越して通信ゾーンから出てしまう場合

には後続の自動車は走行データのように即所性のあるデータの伝送が要求された通信ゾーンにおいて行えないことになる。

そこで、データを分割したブロックの1つ1つを小パケットとして伝送し、何回かのパケット伝送の間に、後続車が通信要求を表す要求信号を発信できるようにすることによって、ある自動車が通信中でも後続の自動車に接続の機会を与えることにする。

その方式を図3-3-1に説明する。まず、地上局は新たに通信ゾーンに進入する自動車からの要求信号の発信可能を示すトリガーパケットを送出する。応答する要求信号のキャリアがないと、再びトリガーを送出する。通信ゾーンに入ってきてこれを受信した自動車はこれに応答して、要求信号を送出する。要求信号を受信した地上局は、この自動車に対してポーリングパケットを送出する。自動車は、これに応じて小分割されたデータパケットを送出する。

データパケットを受信した地上局は再びポーリングパケットを送出する。

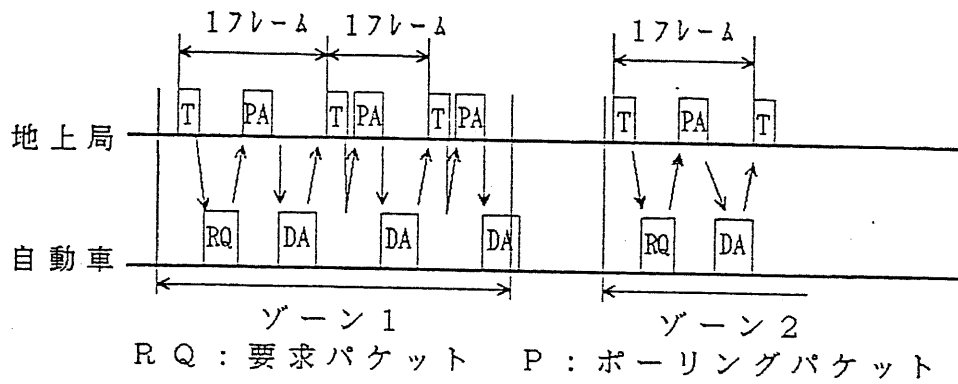


図 3 - 3 - 1 データの小分割とフレーム構成

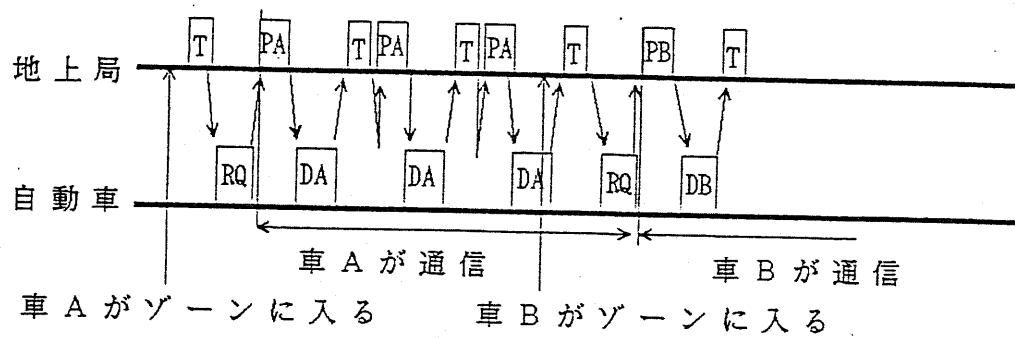


図 3 - 3 - 2 優先度への対応

ここで、何回かのポーリングをまとめて1フレームとしておく。各フレーム毎に、新たに通信ゾーン内に入ってきた自動車に対してトリガーを発生し、応答する要求信号のキャリアが検出されなければ再びポーリングを行うようにする。

このようにすることによって、図3-3-2に示したように先行車が通信中においても、後から通信ゾーンに入ってきた自動車に対して接続の機会を与える。

3. 3. 2. シミュレーション評価

このような方式では、ポーリングパケットや誤り検出コード等のデータパケット伝送制御用の信号のオーバーヘッドが問題となることが考えられる。

その影響を、以下に示した状況を想定したシミュレーションによって検討した。

○ 渋滞流

交通量: $Q = 1200 \text{ (台 / h)} \times 6 \text{ (車線)}$

走行速度: $V_A = 20 \text{ (km / h)}$

○ 飽 和 流

交 通 量: $Q = 2300 \text{ (台 / h)} \times 6 \text{ (車線)}$

走 行 速 度: $V_A = 60 \text{ (km / h)}$

但し, ここでは, スループット S を次のように定義する. T_{ALL} を考慮する全時間とし, その間に伝送されたデータパケットのうちで, 中断なく伝送されたものに含まれる正味のデータビットの伝送に必要な時間を T_{com} とすれば,

$$S = T_{com} / T_{ALL} \quad (\text{式 } 3 - 3 - 1)$$

とする.

また, パケット長を 1 パケット中の正味のデータビットの伝送時間とする.

パケット長とスループットとの関係を, 一時間当たり各車線毎に Q で発生する自動車が, 道路上を一定速度 V_A で移動して行くモデルによって求めた.

この時, 車両の発生分布は文献[30]に従って, 車間距離の分布が, 平均車間距離 $s_m[m]$ が,

$$s_m = 6.903 + 0.1033 \times V A$$

$$+ 0.0024 \times V A^2$$

.....(式 3 - 3 - 2)

車間距離の標準偏差 σ [m]が,

$$\sigma = 1.94 + 0.026 \times V A$$

$$+ 0.00067 \times V A^2$$

.....(式 3 - 3 - 3)

であるような Γ 分布に従うようにした。

また、第2章の内容に基づいて通信ゾーンの長さは
10 m, 通信ゾーンの間隔は16 mとした。

接続制御方式は、一般のパケット交換ネットワーク
で用いられるようなコンテンションアクセス方式と、
隣接通信ゾーンの情報を利用した新方式の両方につい
て検討した。なお、これらの詳細については次節の3.
4. において詳述する。ここで、その平均トリガー間

隔は次節のシミュレーションにおいて良好なスループットが達成できることが示された値であるところの0.06 sとし、信号の検出時間は文献[40]のデータより3 [ms]とした。パケットの衝突以外による通信の誤りはないとする。

ポーリング信号やCRCコード等、データパケットの伝送制御に必要な信号の伝送にかかる時間TCは、1つのデータパケットについて、

○ ポーリング信号

プリアンプル 16 bit

開始フラグ 8 bit

トークン 10 bit

パケット番号 4 bit

CRCコード 16 bit

終了フラグ 8 bit

○ データパケット

プリアンプル 16 bit

開始フラグ 8 bit

C R C コード 16 bit

終了フラグ 8 bit

をトータルして 110 bit となる。従って,

$$TC = 110 [\text{bit}] / 9600 [\text{bps}]$$

$$= 11.4 [\text{ms}]$$

..... (式 3 - 3 - 4)

とした。

その結果を図 3 - 3 - 3, 図 3 - 3 - 4 に示す。図 3 - 3 - 3 は飽和流時, 図 3 - 3 - 4 は渋滞時の結果である。図 3 - 3 - 3 より自動車が比較的良く流れており一つの通信ゾーンに存在する時間が短いときは, データを 0.05 [s]程度の短いパケットに分割して通信したほうが良いことがわかる。また, 図 3 - 3 - 4 より, 渋滞して一つの通信ゾーンに存在する時間が長くなったときでも, 0.05 [s]程度以上のパケット長があれば, それ以上パケット長を長くしても殆ど効率は向上しないことがわかる。

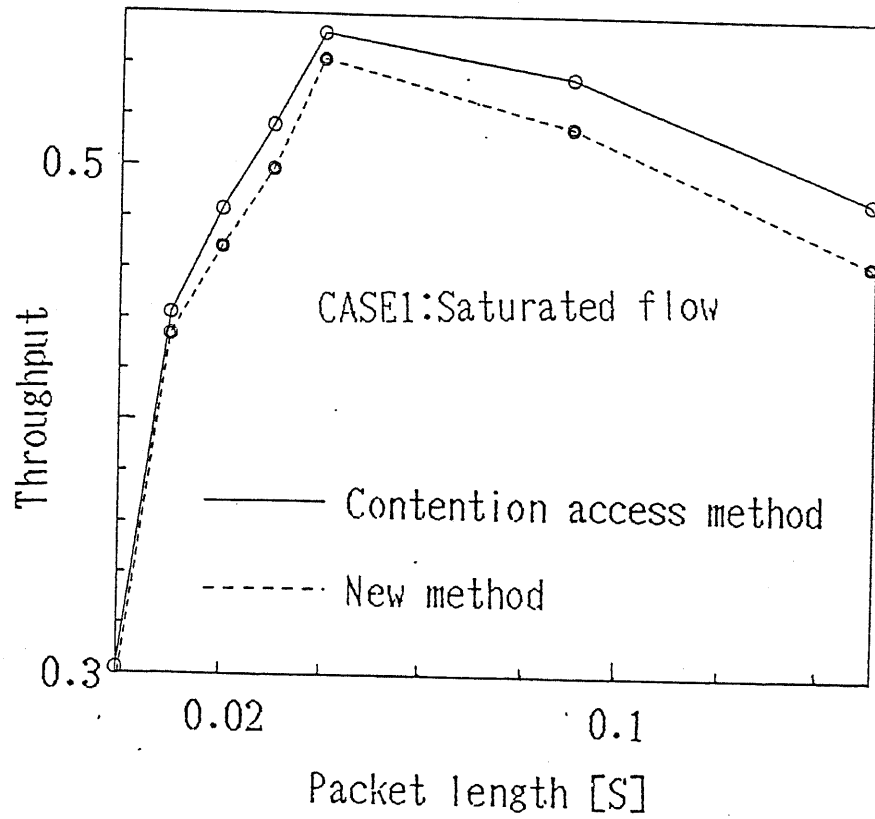


図 3 - 3 - 3 パケット長とスループットの関係 (飽和流時)

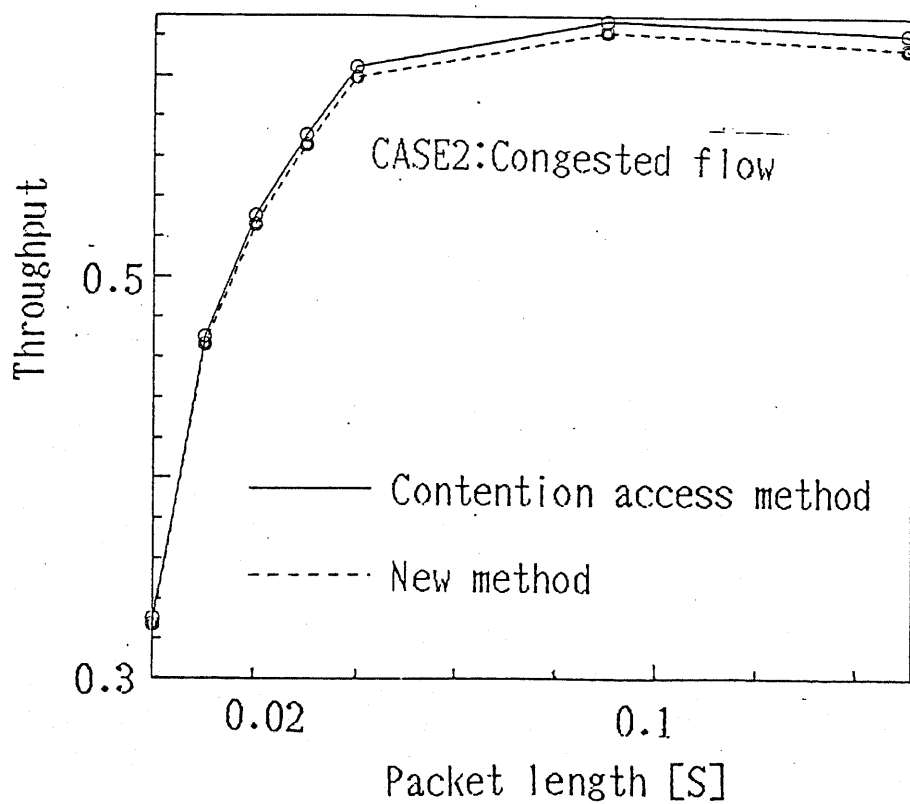


図 3 - 3 - 4 パケット長とスループットの関係 (渋滞流時)

以上より結論として、接続制御方式としては新方式とコンテンション方式のいずれを用いた場合でも1つのデータパケットの大きさを小さく分割することによって効率よく伝送できることがわかる。また、パケット中断の影響が無視できると考えられる渋滞時において、パケット長を長くして伝送制御情報のオーバーヘッドが少なくなるようにしてもほとんど効率が向上しないことから、本方式における伝送制御情報のオーバーヘッドはほとんど無視できると考えられる。

3. 4. 接続制御方式 [1][3][4][10][15][16][17]

3. 4. 1. 方式説明

本システムの接続制御方式には道路リンク入口の通信ゾーンにおけるものと、中間部の通信ゾーンにおけるものがある。これよりその各々について説明する。

○道路リンク入口の通信ゾーンにおける接続制御

3. 2. 1. においても述べたように、小ゾーン連続形自動車パケット通信システムでは最初の通信ゾーンにおいては新しく通信ゾーンに入ってくる自動車を確実に捉えトークンを与えなければならない。ここで本システムでは、最初の通信ゾーンにおいては接続制御方式としては通常のパケット交換ネットワークで用いられるコンテンションアクセス方式によるが、汎用データの通信を行わないことにして確実な接続制御の実現を図る。

その手順を図 3 - 4 - 1 に示す。通信が行われていないとき、地上局はトリガーを一定間隔で送信する。通信ゾーンに新しく入ってきて、これを受信した自動車は、自動車 ID、目的地などの情報を含む要求信号

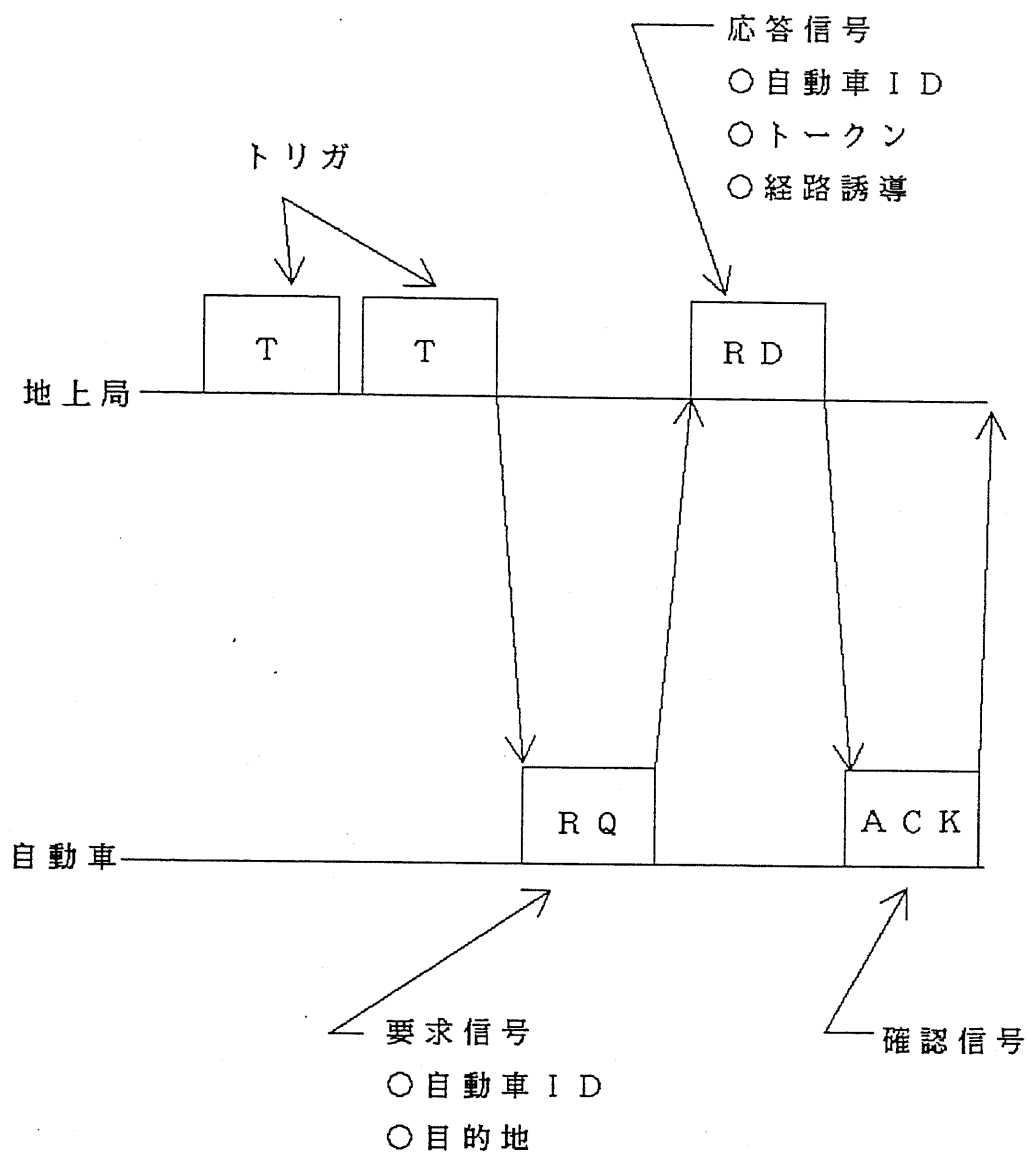


図 3 - 4 - 1 道路リンク入口における接続制御

を送信する。これに対し地上局は自動車 I D, 経路誘導, トークン等の情報を含む応答信号を送信する。自動車はこれを受信することにより通信の完了を確認し, A C K 信号を送出する。これを受信した地上局は通信の完了を確認し, 再びトリガーの送信を再開する。

要求信号が衝突により失われた場合は, 応答信号を送出しないでトリガーの送信を再開する。衝突後の要求信号の再送は適当な確率 P R を持つて行うことにして再衝突を防止する。

本研究では最初の通信ゾーンにおいて新しく通信ゾーンに入ってきた自動車を捉える場合の確実性を理論的計算によって求めた。具体的には, 自動車が通信ゾーンに入ってから通信が完了するまでの時間の確率分布を計算する。この結果と自動車が通信ゾーン内に存在する時間を比較することにより接続の確実性を求めることができる。

この計算の結果, 自動車の速度が 4 5 km/h, 6 0 km/h, 7 5 km/h のいずれの場合においても, ほぼ全ての自動車が 0 . 2 (s) 以内に通信を完了することが分かった [1 5] .

ここで、自動車種々の速度で通信ゾーンを通過したときに、その内に存在する時間は、

45 [km/h] 0.8 [s]

60 [km/h] 0.6 [s]

75 [km/h] 0.48 [s]

である。

従って、最初の通信ゾーンにおいて通常のコンテンションアクセス方式を用いても、汎用データの通信を行わなければ、新しく通信ゾーンに入ってくる自動車を確実に捉えられるといえる。

○道路リンク中間部の通信ゾーンにおける接続制御

通常のパケット交換ネットワークで接続制御方式として用いられるコンテンションアクセスの方式を本システムの道路リンク中間部の通信ゾーンにおいて用いるとすると、図3-4-2に示すようになる。

まず、地上局は任意の自動車に対するトリガーを一定間隔で送信する。通信ゾーンに新たに入ってきた自動車は、地上局のトリガーに応答して要求信号のパケットを送出する。この時、同時に送信された複数の要

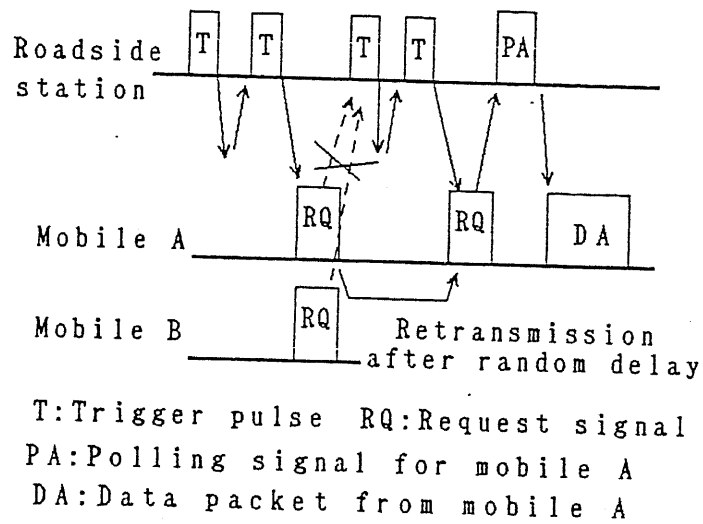


図 3 - 4 - 2 通常のコンテンションアクセス方式

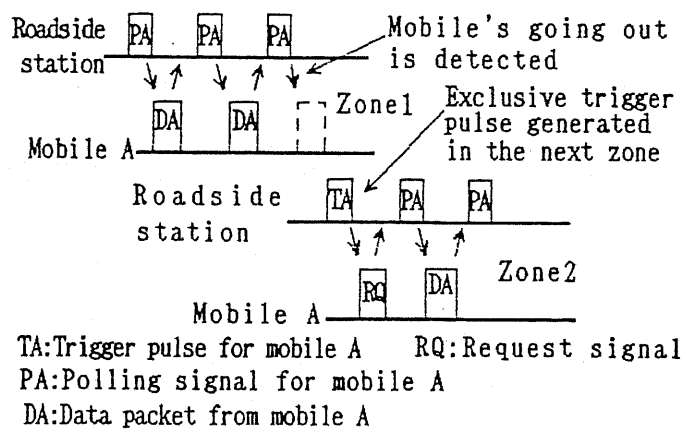


図 3 - 4 - 3 自動車の走行情報を利用した接続制御方式

求信号が衝突を起こして失われた場合は、以後のトリガーからは適当な確率をもって要求信号の送出を行うことによってパケットをランダムに遅延させて再衝突を防ぎながら、接続されるまで要求信号の再送信を繰り返す。

しかし、小ゾーン連続形自動車パケット通信システムにおいては、車が一つの通信ゾーンに存在する時間が短い。そのため、図3-4-2に示すようなコンテションアクセスの方式では、トリガーの間隔がまばらであれば衝突による再送を行っている間に自動車が通信ゾーンから出てしまうということが起こり通信接続の確実性が低下する。また逆に、走行誘導のために各通信ゾーン毎に自動車を確実に捉える必要から、このような接続抜けをなくすためトリガーの間隔を密にして確実な接続制御を行おうとすると、そのための通信チャンネルの負荷が大きくなり汎用のデータ伝送のスループットが低下することが本節におけるシミュレーションによって示される。

これに対して本研究では、走行誘導に用いられる自動車の位置や進行方向などの走行情報を通信接続に利

用する方式を提案する。図3-4-3のように、ある通信ゾーンにおいてポーリングに対する応答がなくなると、通信ゾーンから出たことが検出された車に対して、次に到着する通信ゾーンをこの走行情報より予測し、次の通信ゾーンにおいてその自動車のトークンを含む専用のトリガーを発生して待ち受けて接続することによって要求信号の衝突を無くし、データ通信においてのスループットを低下させることなく確実に接続制御を行うようにする。

但し、複数の自動車が同時に前の通信ゾーンから出てきたときは、次の通信ゾーンにおいてそれらの自動車に対する専用のトリガーをサイクリックに循環させて発生するようにする。

ここで、先にも述べたように、リンク入口部分の通信ゾーンでは自動車の識別や経路誘導等のみを行うこととして、新しくリンクに入ってきた自動車を確実に捉えるが、それ以外の途中の脇道などからリンクに入ってくる自動車に関しては以下のように対処する。地上局からのトリガーの中に、時々、コンテンションアクセス方式の場合と同様な任意の自動車に対すトリガ

一を混入する。そして、途中からリンクに入ってきた自動車は、これに応答して要求信号を発信して接続する。

また、既に接続され追跡中の自動車であっても途中での接続抜け等によって捉えられなくなってしまった場合は、新たに進入した通信ゾーンにおいてトリガーが一巡しても自分のトリガーが送出されない等によって接続抜けと判断し、リンクの途中から入ってきた自動車と同様にして接続する。ここで、これらのための接続制御方式としては通常のコンテンションアクセス方式を用いることとするが、脇道などから入ってくる自動車はごくわずかであると考えられ、また、後のシミュレーション結果からもわかるように接続抜けとなる自動車はほとんど無いので、そのためのシステムの負担はほとんど無視できると考えられるので問題はない。

3. 4. 2. シミュレーション評価

ここで新たに提案した、自動車の走行情報を利用した接続制御方式の効果を、次に示したような状況を想

定したシミュレーションを行って検討した。

○ 飽和流

交通量: $Q = 2300 \text{ (台 / h)} \times 6 \text{ (車線)}$

走行速度: $V_A = 60 \text{ (km / h)}$

ここで、これは両方向とも交通量が最大、つまり接続制御に対する負荷が最も大きくなると考えられる場合である。車両の発生分布は 3. 3. 2. と同様の、一時間当たり各車線毎に Q 台の割合で発生する自動車、道路上を一定速度 V_A で移動して行くモデルとした。また、その他の条件も 3. 3. 2. と同様に、通信ゾーンの長さは 10 m, 通信ゾーンの間隔は 16 m, 信号の検出時間は 3 [ms], パケットの衝突以外による通信の誤りはないとした。また、スループット S の定義も 3. 3. 2. と同様に、 T_{ALL} を考慮する全時間とし、その間に伝送されたデータパケットのうちで、中断なく伝送されたものに含まれる正味のデータビットの伝送に必要な時間を T_{com} とすれば、

$$S = T_{com} / T_{ALL}$$

(式 3 - 4 - 1)

とした。また、ここではパケット長は 0.05 [s]とした。これは、3. 3. 2. の飽和流時のシミュレーションにおいて良好なスループットが達成された値である。

トリガー間隔とスループットの関係を求めたものが、図 3 - 4 - 4 である。図 3 - 4 - 5 は通信ゾーンを通過した全ての自動車のうちで接続されないでゾーンから出たものの割合である。

コンテンション方式と新方式のいずれの場合も、トリガー間隔の広い部分（図 3 - 4 - 4, 図 3 - 4 - 5 の右端部）では多数の接続抜けのために、また、トリガー間隔が狭い部分（図 3 - 4 - 4, 図 3 - 4 - 5 の左端部）では、接続抜けは発生していないがトリガーのオーバーヘッドが大きくなるためにスループットは低くなっている。全体としてトリガーの間隔が 0.06 (s) 付近で良好なスループットを達成できる。

新方式についてみれば、図 3 - 4 - 5 より、良好なスループットが達成される範囲においても接続抜けは

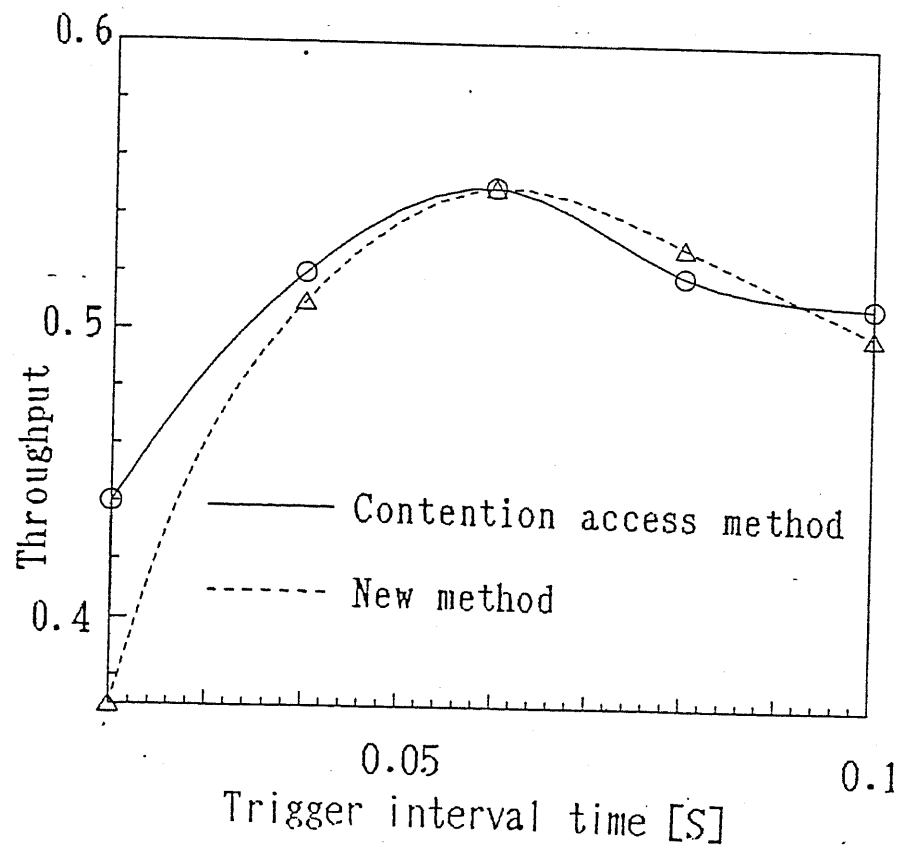


図 3-4-4 トリガー間隔時間とスループットの関係

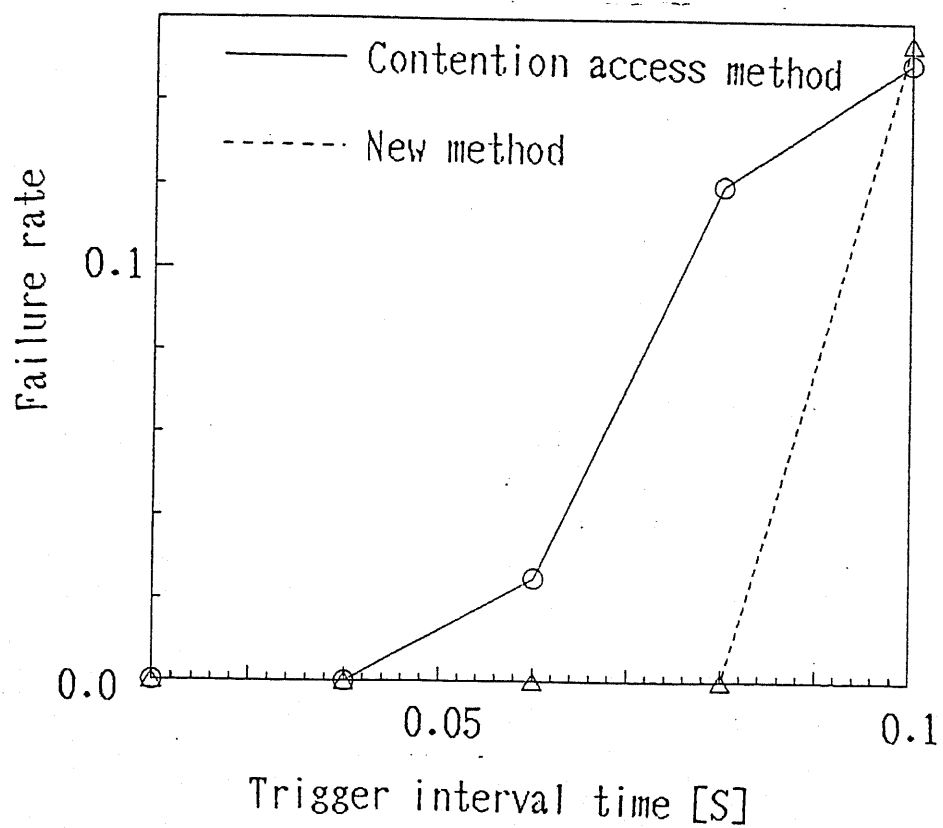


図 3-4-5 トリガー間隔時間と接続抜けの発生率の関係

全く発生しておらず、通信ゾーンに入ってくる自動車は確実に捉えられる。一方、これに対してコンテンション方式は、良好なスループットが達成されるトリガー間隔の範囲においては、数%程度の接続抜けが発生している。この程度の値はスループットに対しては大きな影響は与えず、また、新方式ではトリガーに自動車を識別する情報を含ませるため多少トリガーのオーバーヘッドが大きくなっていることもあり、両者のスループットがほぼ等しくなっていると考えられる。

しかし、本システムでは、先にも述べたように、走行誘導を行うため各通信ゾーン毎に自動車を確実に捉えることが重要である。新方式ではスループットの高い部分でもこれを満たしているが、コンテンション方式では、図3-4-5より、トリガー間隔を0.04 (s) 以下にしなければならず、図3-4-4よりスループットが低下することがわかる。

3. 5. 情報伝送方式[2][5][11][17]

3. 5. 1. 方式説明

3. 2. において述べたように、本システムでは自動車の走行誘導を行ってゆくための走行データと、自動車－地上間で汎用目的のデータ通信を行ってゆくための汎用データという2種類の性質の異なるデータを伝送する。走行データは各通信ゾーン毎に確実に伝送される必要があるものである。汎用データは複数の通信ゾーンにまたがって通信でき、伝送所要時間が短いことが望ましいものである。

しかし、本システムにおいては、通信ゾーンの大きさが小さく各自動車が一つの通信ゾーン内に存在する時間が短い。そのため、通常考えられるように各通信ゾーン毎に、各道路リンクを管理する装置であるRSSと自動車との間で走行データや汎用データのやり取りを完結させてゆこうとすると、自動車からの情報に対するRSSの応答が遅れた場合に情報伝送の失敗や通信効率の低下を引き起こすことが後のシミュレーションにおいて示される。ここではこれに対して、自動車が以後に到着する通信ゾーンをその進行方向等の走行

情報より予測し、その通信ゾーンを管理する通信制御装置（ＣＣ）にＲＳＳよりその自動車に対する情報をあらかじめ転送しそこにバッファリングしておくことにより、ＲＳＳの応答が遅れても確実な情報伝送や効率のよい通信を実現する方式を提案する。以下、走行データと汎用データの伝送のための具体的な手順について説明する。

○ 走行データの伝送方式

走行データの伝送においては、自動車は各通信ゾーンへ進入する毎にＲＳＳに速度等の情報（これを、走行データ（上り）とする）を伝送する。これに対してＲＳＳは走行誘導指示を作成し（これを、走行データ（下り）とする）自動車に返送する。この場合、通常的方式では図３－５－１に示したように、自動車は各通信ゾーン毎にＲＳＳに走行データ（上り）を伝送する（１）（４）。これに対してＲＳＳは走行誘導指示作成の処理を行い（２）（５）、その通信ゾーンのＣＣに対して走行データ（下り）を返送する（３）（６）。ここで、ＣＣが

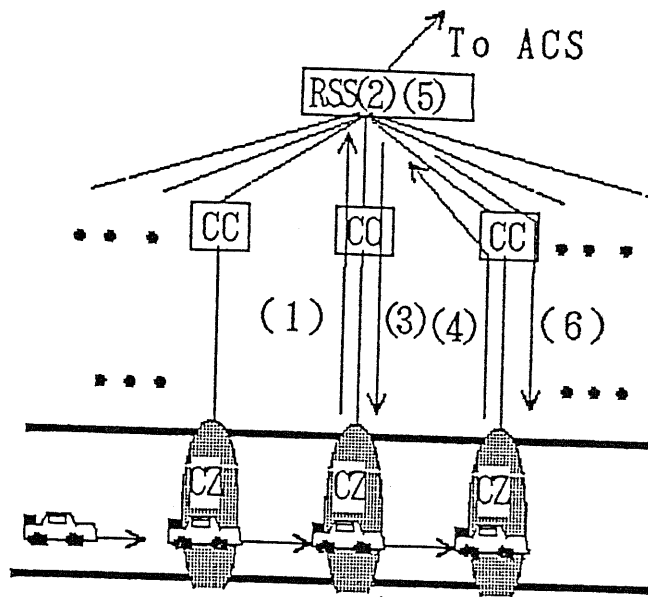


図 3 - 5 - 1 通常の情報伝送方式

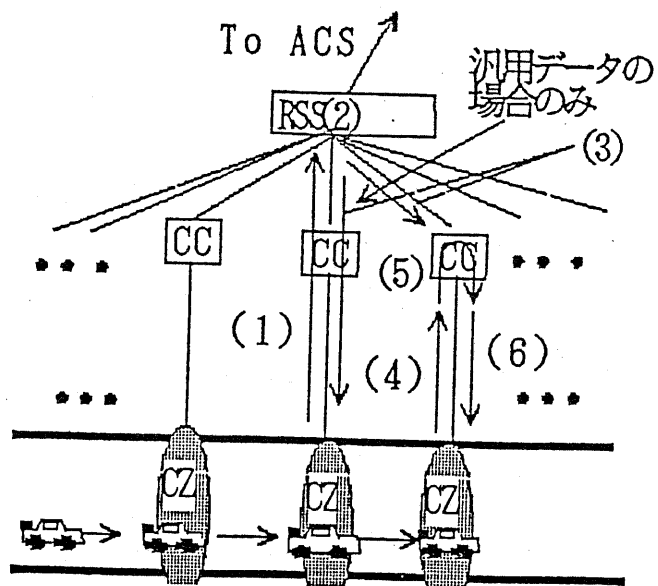


図 3 - 5 - 2 新たに提案した情報伝送方式

走行データ（上り）を受信してから、走行データ（下り）を自動車に対して返送可能になるまでの時間を走行データの応答時間とする。先にも述べたように本システムでは通信ゾーンの大きさが小さいため通常的方式では、この応答時間が長くなるとRSSよりデータが返送される前に自動車がその通信ゾーンから出ることが多くなり、また、これをなくそうとすると走行データの応答時間に対する要求が厳しくなるという欠点がある。

これに対して本研究では図3-5-2のように、ある走行データ（上り）(1)に対して作成された走行データ（下り）(2)は、自動車が次に到着することが予測される通信ゾーンのCCに対して予め転送しておき(3)、そこにバッファリングしておく。自動車がこの通信ゾーン進入して走行データ（上り）(4)を伝送すると、走行データ（下り）をCCより直接伝送する(5)(6)方式を提案した。これにより前述の欠点を解消する。

○汎用データの伝送方式

汎用データの下りに関しては、自動車は地上からの

下り汎用データを受付可能状態にある時は走行データ（上り）の通信の際に各ゾーン毎にRSSに通知しており（要求）、データが存在すれば自動車からのこの要求に応える形でいくつかのゾーンにわたって伝送される。この時、通常考えられる方式では図3-5-1のように、RSSが各通信ゾーン毎に要求に対してデータを返送してゆく。ここでCCが自動車からの要求を受信してからそれに対するデータをRSSより受取りその自動車に対して返送可能になるまでの時間を汎用データの応答時間とする。この応答時間が長くなると一つの通信ゾーンでのデータ伝送可能な時間が短くなりデータをより多数の通信ゾーンにまたがって通信しなければならなくなるので伝送所要時間が増大し、またこれをなくそうとすると、汎用データの応答時間に対する条件が厳しくなる。

これに対して本研究では、図3-5-2のように、自動車からの最初の要求(1)に対して検索された下りデータ(2)は、自動車が以後に到着することが予測される通信ゾーンのCCにも予め転送してバッファリングしておき(3)、以後の要求(4)に対してはこのCCから直

接伝送される(5)(6)方式を提案した。これにより、各通信ゾーン毎にRSSからの応答を待つ必要がなくなり、1つ当りの通信ゾーンで伝送可能データ量が増すために、通信チャネルの利用率の向上や伝送所要時間の減少等の効果が期待できる。

なおここで、汎用データの上に関しては各データパケットに順序番号を付加し、これに基づいてソーティングしながら上位層に転送すればよいので問題はないと考えられる。

3. 5. 2. シミュレーション評価

ここで提案する情報伝送方式の効果を以下の状況を想定した評価シミュレーションにより検討した。

○ 渋滞流

交通量: $Q = 1300 \text{ (台 / h)} \times 6 \text{ (車線)}$

走行速度: $V_A = 10 \text{ (km / h)}$

○ 飽和流

交通量: $Q = 1800 \text{ (台 / h)} \times 6 \text{ (車線)}$

走行速度: $V_A = 36 \text{ (km/h)}$

○自由流

交通量: $Q = 1000 \text{ (台/h)} \times 6 \text{ (車線)}$

走行速度: $V_A = 60 \text{ (km/h)}$

ここでシミュレーションモデルは、3. 3. 2. や 3. 4. 2. と同様に、自動車は道路上を一定の速度で移動するものとした。この時自動車の発生は車頭間隔が指数分布をなすようにし、また、第2章の内容にしたがって通信速度は 9600 [bit/s] を仮定し、通信ゾーンの長さは 10 [m] 、間隔は 16 [m] とした。この時に用いた評価プログラムを付録 A. 1. に示す。

まず、走行データの応答時間とその伝送失敗率との関係を求めた。この時、バックグラウンドで通信されている汎用データの応答時間は 0.1 [s] 、長さは 5 [kbit] とし、データの発生率が 0.1 と 0.3 の場合についてシミュレーションを行った。図 3 - 5 - 3 (データ発生率 = 0.1) と図 3 - 5 - 4 (データ発生率 = 0.3) が渋滞流、図 3 - 5 - 5 (データ発生率 = 0.1) と図 3 -

5 - 6 (データ発生率 = 0.3) が飽和流, そして, 図 3 - 5 - 7 (データ発生率 = 0.1) と図 3 - 5 - 8 (データ発生率 = 0.3) が自由流, 各々の結果である.

いずれの場合にも, 確実な走行データ通信を行うために R S S に対して要求される走行データの応答時間が, 本方式により大幅に緩和されているのがわかる.

次に, 一つの汎用データの長さを 5[kbit]として, その発生率が 0.1 と 0.3 の場合について汎用データの応答時間とその伝送所要時間との関係を求めた. 図 3 - 5 - 9 (データ発生率 = 0.1) と図 3 - 5 - 10 (データ発生率 = 0.3) が渋滞流, 図 3 - 5 - 11 (データ発生率 = 0.1) と図 3 - 5 - 12 (データ発生率 = 0.3) が飽和流, そして, 図 3 - 5 - 13 (データ発生率 = 0.1) と図 3 - 5 - 14 (データ発生率 = 0.3) が自由流, 各々の結果である. この時, 走行データの応答時間は 0.1[s]としてある. いずれの場合でも, 本方式では R S S からの汎用データの応答が遅れても, 通常の場合に見られるような伝送所要時間の急激な増加は見られない.

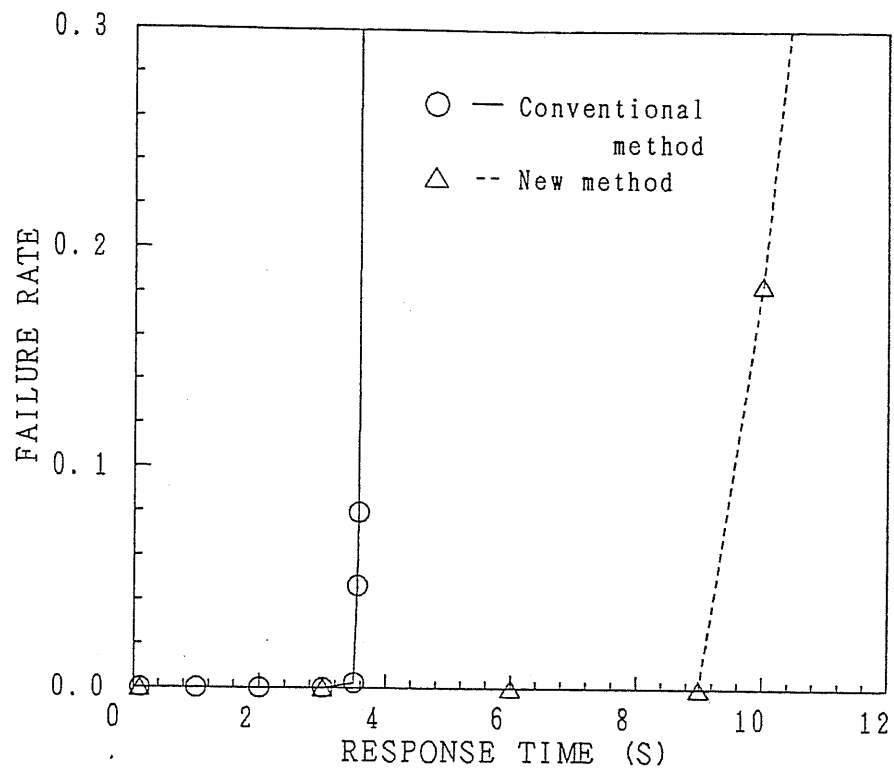


図 3 - 5 - 3 走行データの応答時間と伝送失敗率の関係
(渋滞流時, 汎用データの発生率 0.1)

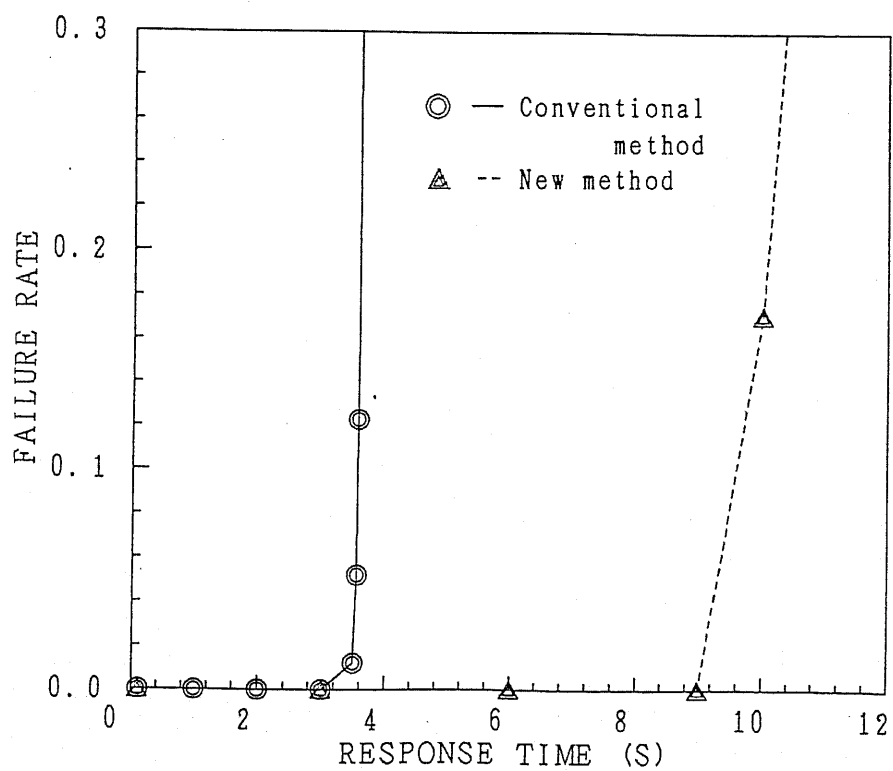


図 3 - 5 - 4 走行データの応答時間と伝送失敗率の関係
(渋滞流時, 汎用データの発生率 0.3)

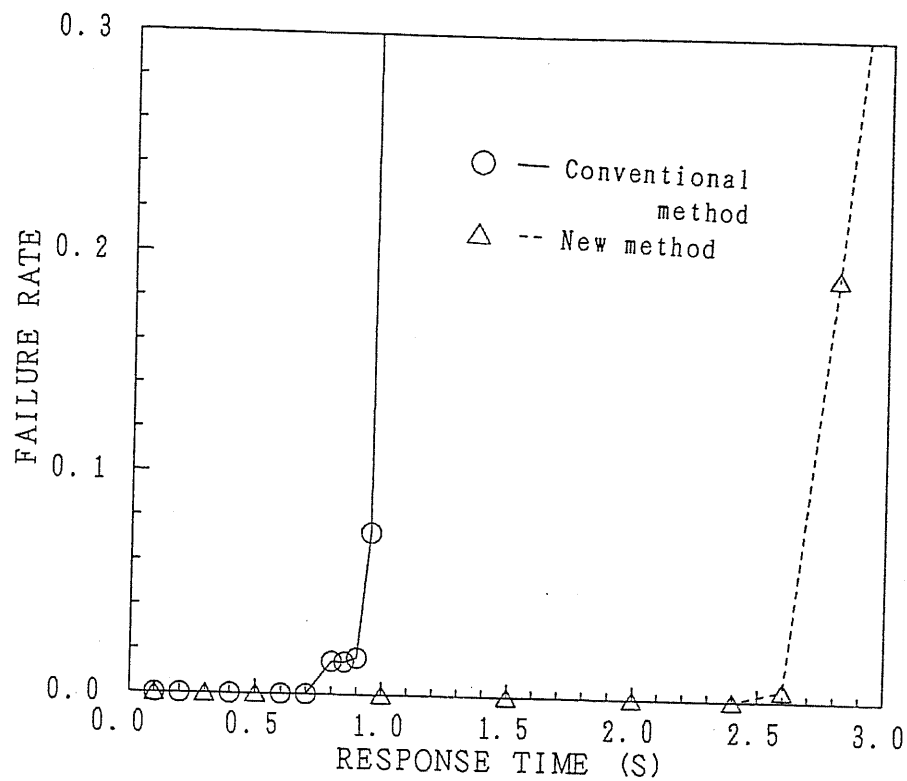


図 3 - 5 - 5 走行データの応答時間と伝送失敗率の関係
(飽和流時, 汎用データの発生率 0.1)

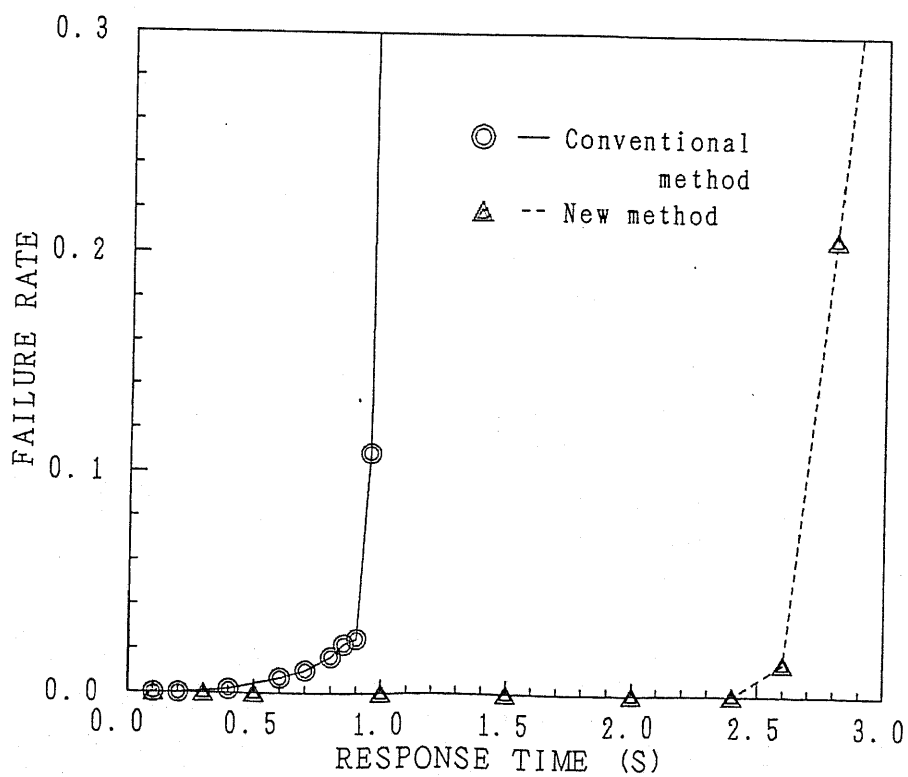


図 3 - 5 - 6 走行データの応答時間と伝送失敗率の関係
(飽和流時, 汎用データの発生率 0.3)

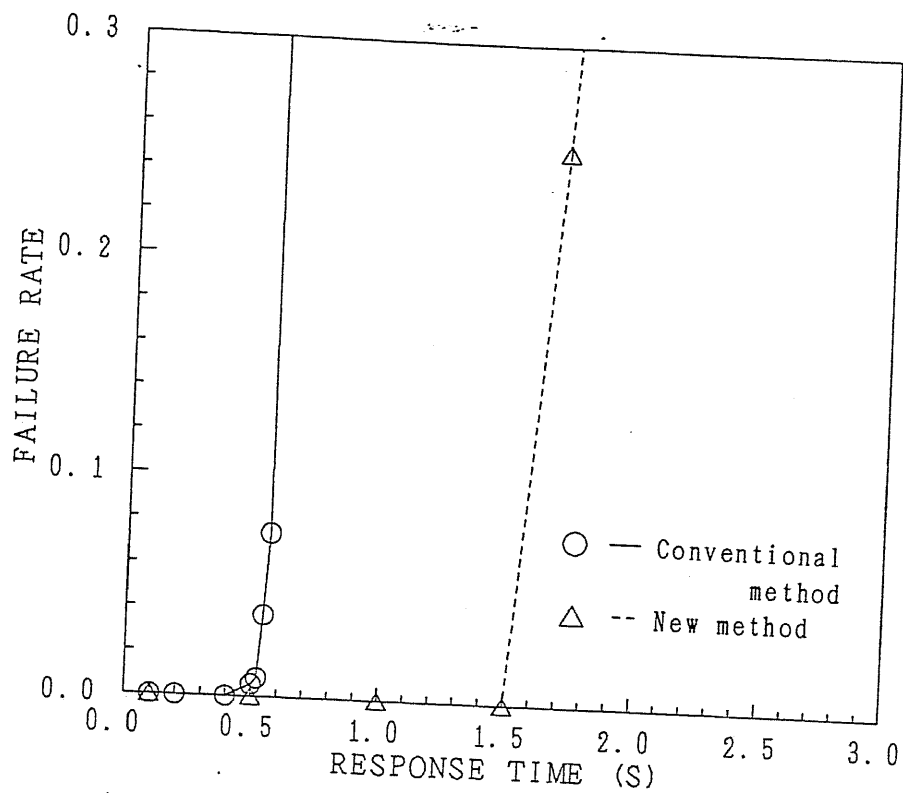


図 3 - 5 - 7 走行データの応答時間と伝送失敗率の関係
(自由流時, 汎用データの発生率 0.1)

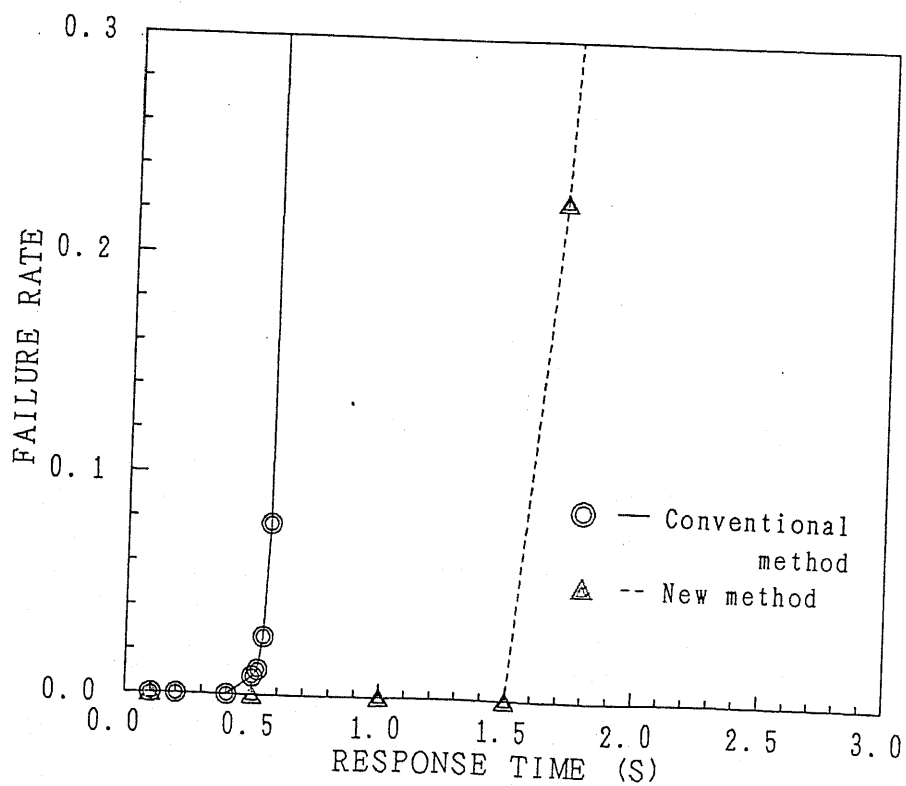


図 3 - 5 - 8 走行データの応答時間と伝送失敗率の関係
(自由流時, 汎用データの発生率 0.3)

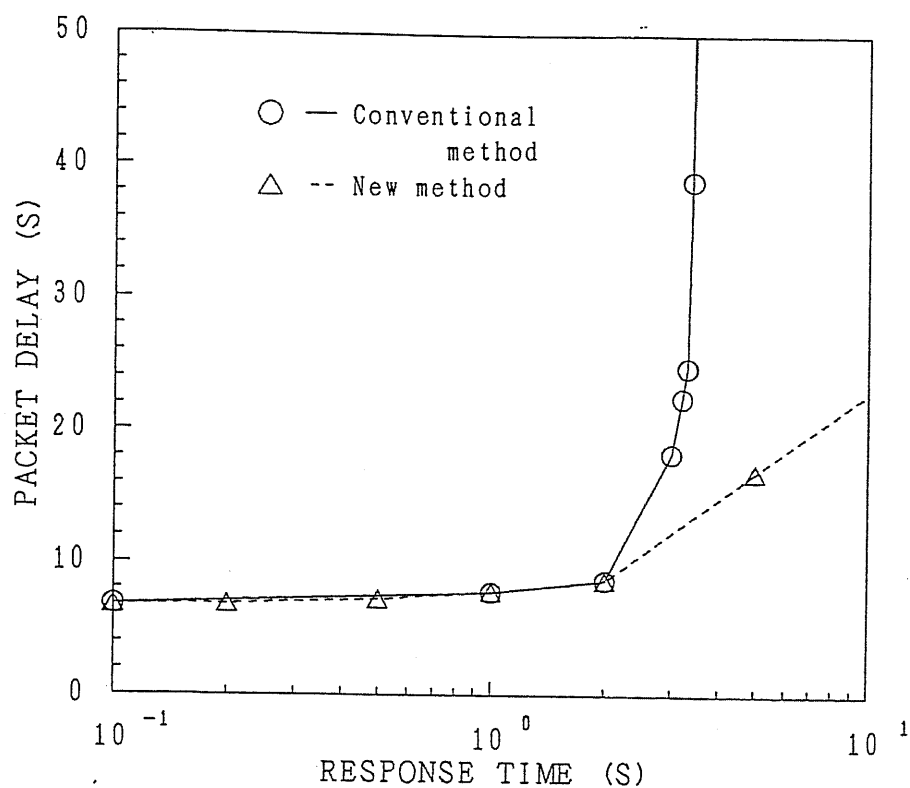


図 3 - 5 - 9 汎用データの応答時間と伝送所要時間の関係
 (渋滞流時, 汎用データの発生率 0.1)

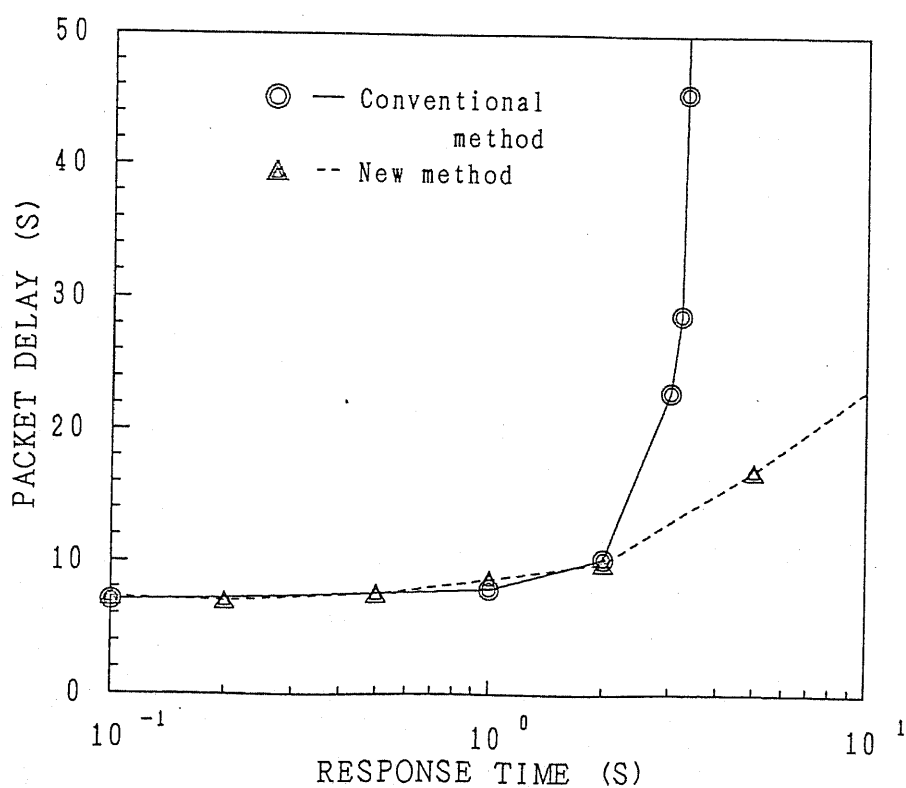


図 3 - 5 - 10 汎用データの応答時間と伝送所要時間の関係
 (渋滞流時, 汎用データの発生率 0.3)

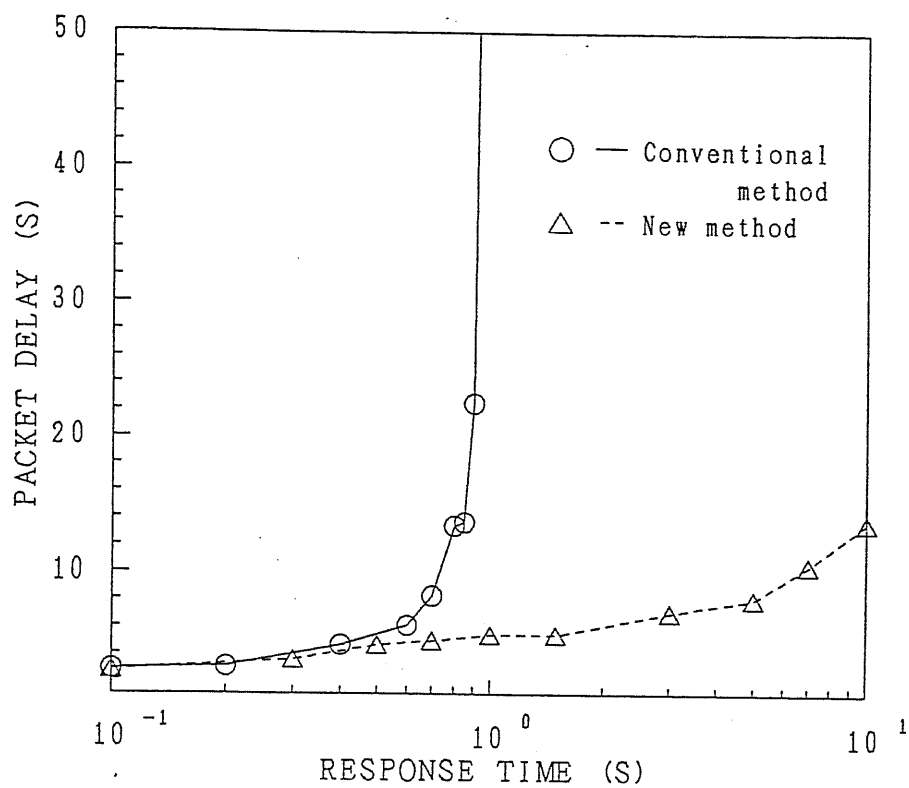


図 3 - 5 - 1 1 汎用データの応答時間と伝送所要時間の関係
(飽和流時, 汎用データの発生率 0.1)

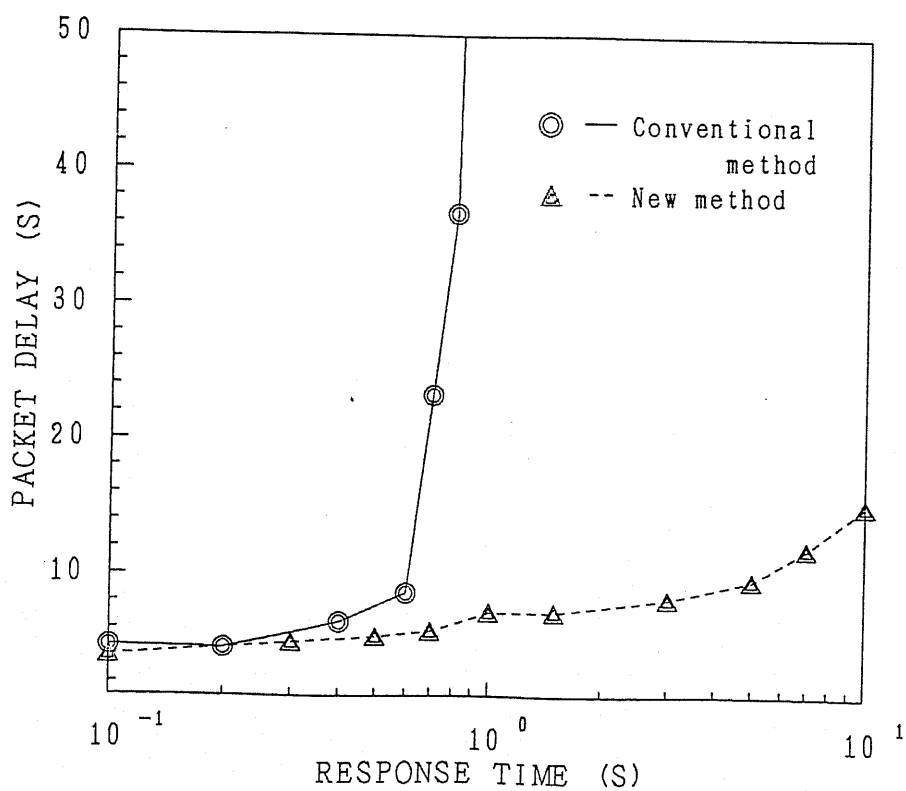


図 3 - 5 - 1 2 汎用データの応答時間と伝送所要時間の関係
(飽和流時, 汎用データの発生率 0.3)

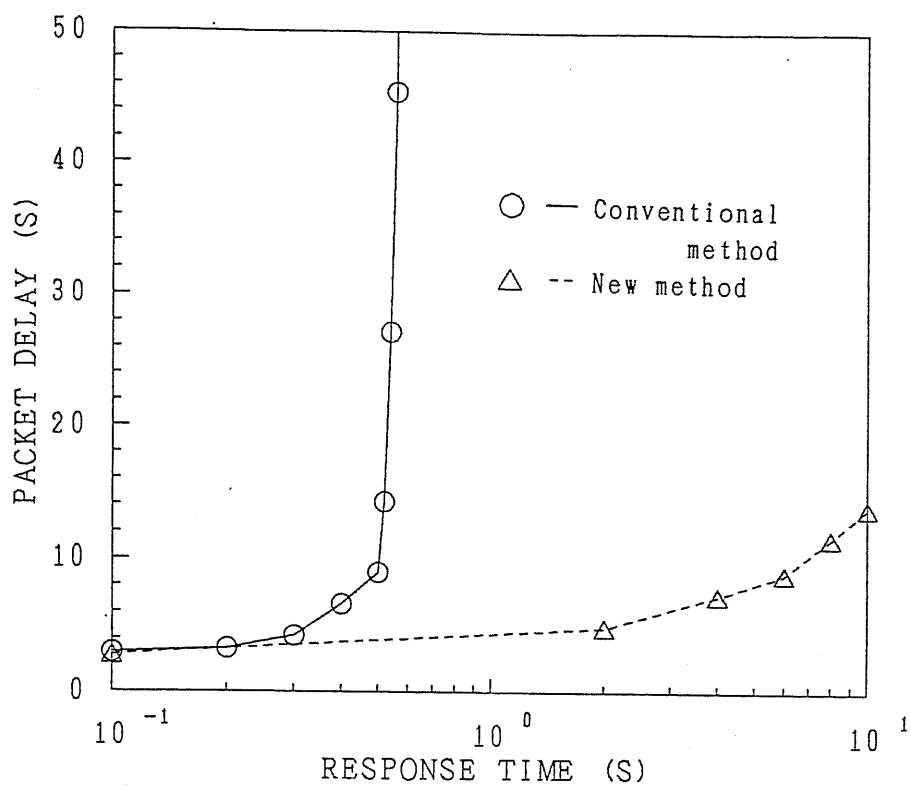


図 3 - 5 - 1 3 汎用データの応答時間と伝送所要時間の関係
(自由流時, 汎用データの発生率 0.1)

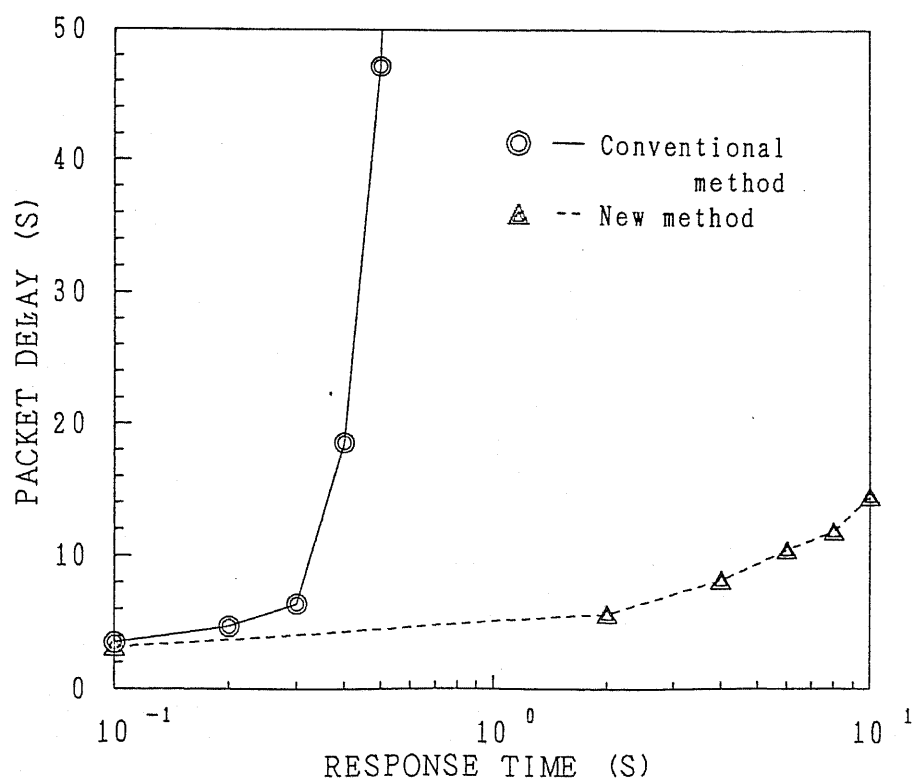


図 3 - 5 - 1 4 汎用データの応答時間と伝送所要時間の関係
(自由流時, 汎用データの発生率 0.3)

次に、通信ゾーンに対する負荷の大きくなる飽和流
時を想定し、通信ゾーンの利用率と汎用データの伝送
所要時間との関係を、汎用データの長さが5[kbit]及び
10[kbit]である場合について求めた。汎用データの長
さが5[kbit]の場合の結果が図3-5-15であり、1
0[kbit]の場合の結果が図3-5-16である。いずれ
の場合においても、汎用データの応答時間が0.4[s]よ
り長くなると通常的方式では伝送可能な情報量が減少
するが、本方式ではそのようなことはないことがわか
る。

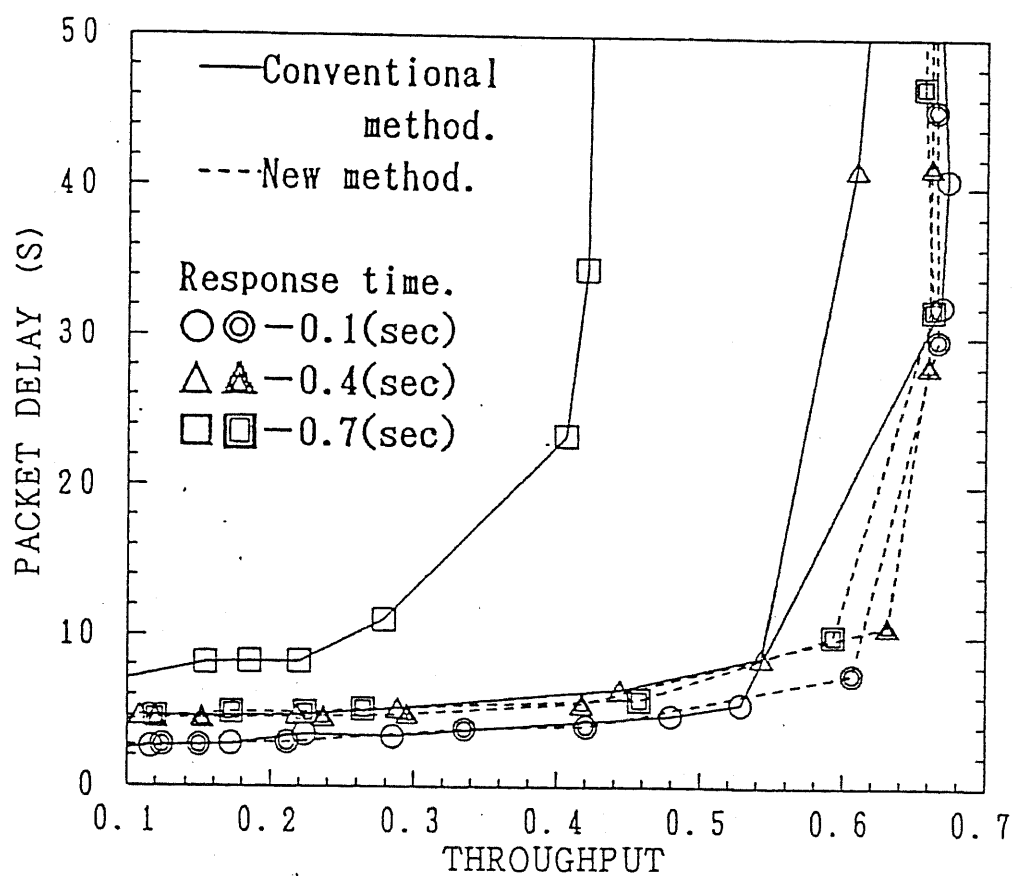


図 3 - 5 - 1 5 スループットと汎用データの伝送所要時間の関係
(汎用データ長 = 5 [kbit])

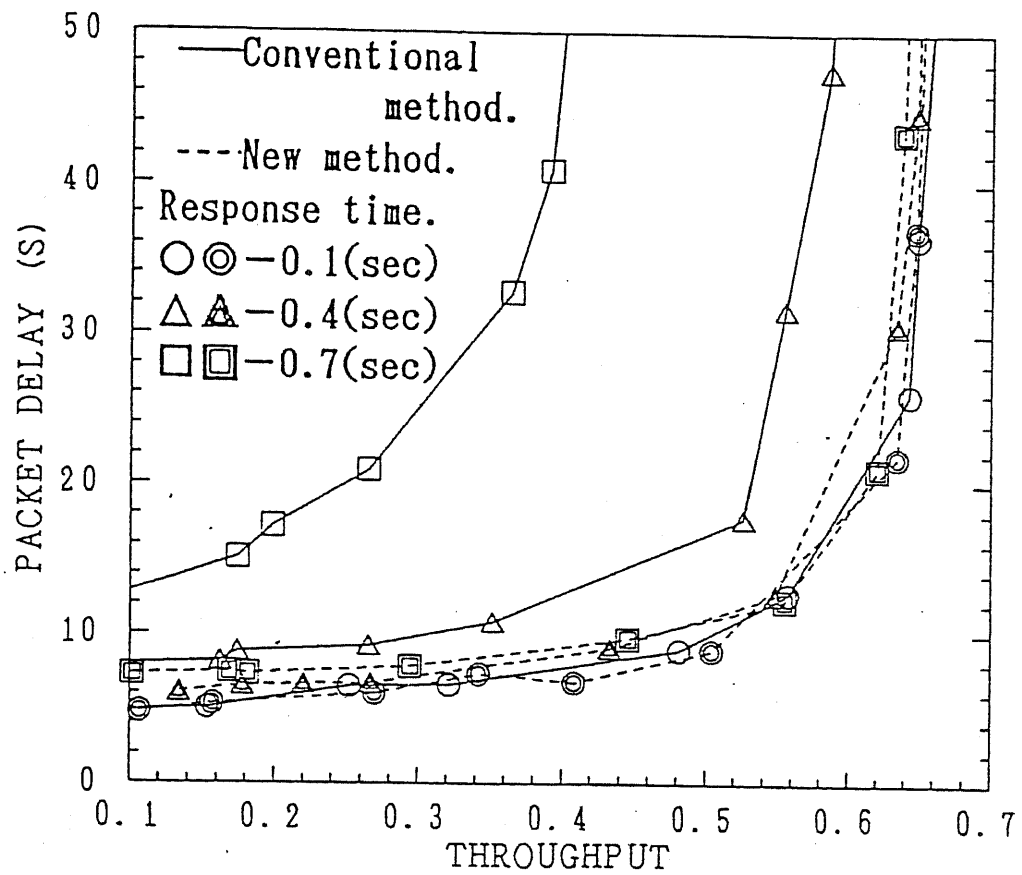


図 3 - 5 - 1 6 スループットと汎用データの伝送所要時間の関係
(汎用データ長 = 10 [kbit])

3. 6. 路車間通信チャネル割当制御[7][8][14][17]

3. 6. ~ 3. 8. では、これまで提案した各制御方式の内容を盛り込んだ道路リンクレベル制御方式の具体的設計について述べる。設計においては、道路リンクレベル制御の機能を路車間通信チャネル割当制御、路車間通信制御、及び、個別情報リンク内管理の3つに分けて行う。まず本節ではその中の路車間通信チャネルの割当制御（以後、チャネル割当制御と略す）について説明する。

小ゾーン連続形自動車パケット通信システムでは、1つの通信チャネルを走行データ通信と汎用データ通信という2種類の通信の間で共用する必要がある。また個々の通信ゾーンにおいて、同時に複数台の自動車が存在し得る。従ってこのとき、通信チャネルを走行データ通信・汎用データ通信に分け、更にそれを各自動車に割り当てる機能必要となる。これがチャネル割当制御機能である。

○ 通信種別の選択

走行データ通信・汎用データ通信へのチャネル割当

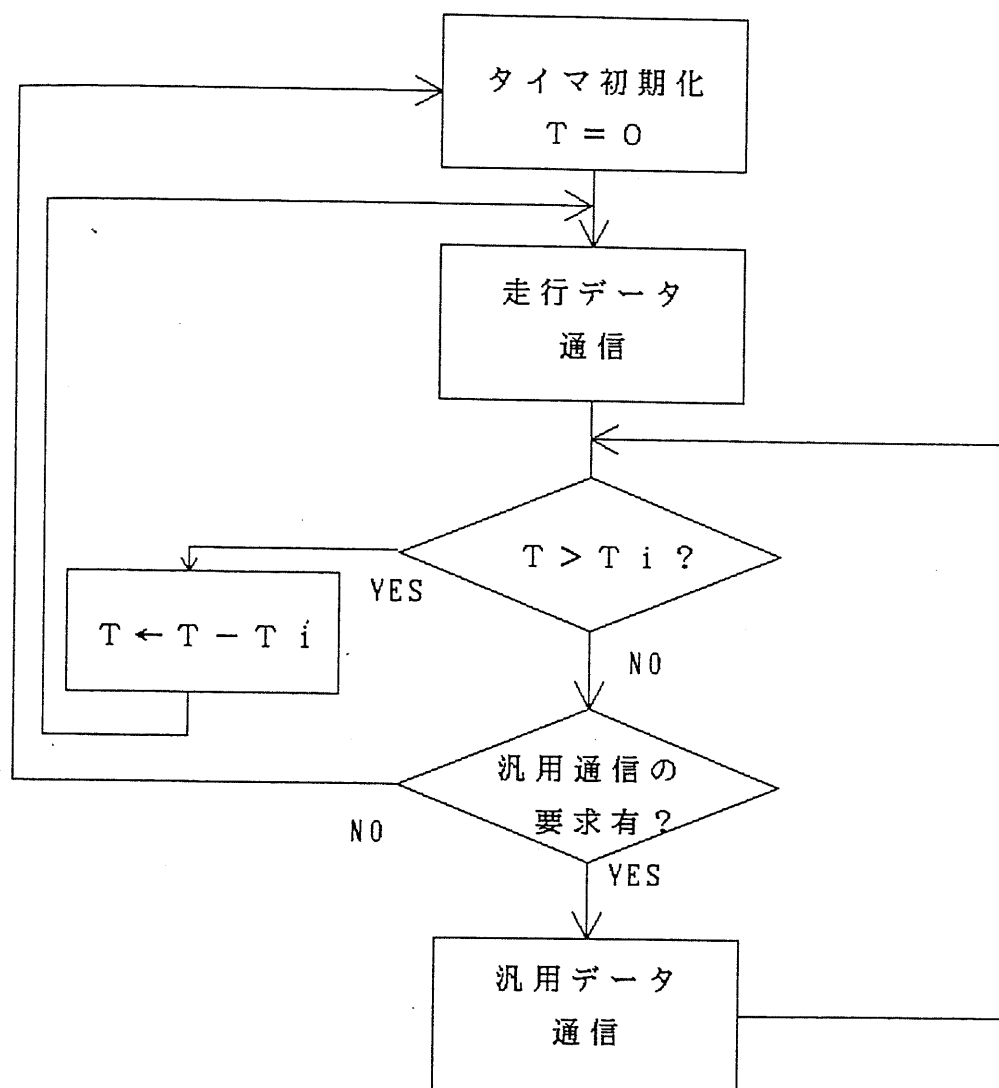
のアルゴリズムを図 3 - 6 - 1 に示す。 3. 4. 2.

での検討結果に従って、確実に自動車を捉えられるように 60 msec に最低一回の割合で走行データ通信に割り当て、さらに、この条件を満たす範囲で汎用データ通信の要求状況に応じて汎用データ通信に割り当てるように制御する。

この動作の例を図 3 - 6 - 2 に示す。まず、全時間を平均トリガ間隔時間 $T_i (= 60 [ms])$ 毎の区間に分割しておく。そして、割当時点で新たな区間に入っていれば、走行データ通信に割り当てる。図 3 - 6 - 2 (a) の区間 3 のように走行データ通信に割り当てられない場合は次の区間 4 でトリガを 2 回送出する。汎用データ通信の要求が一時的になくなる場合は図 3 - 6 - 2 (b) の区間 7 のように汎用データ通信の要求がなくなった時点において、区間を途中で打ち切り、続けて走行データ通信に割り当て、その時点を次の区間 8 の開始時刻とすればよい。

○ 通信対象車両の選択

次は各自動車への割当について述べる。走行データ

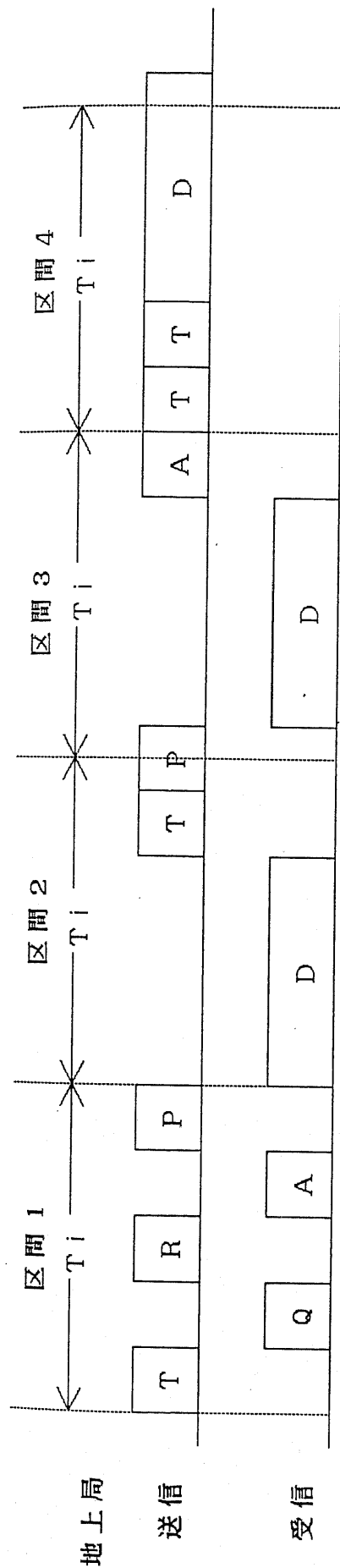


T : タイマ初期化後の経過時間

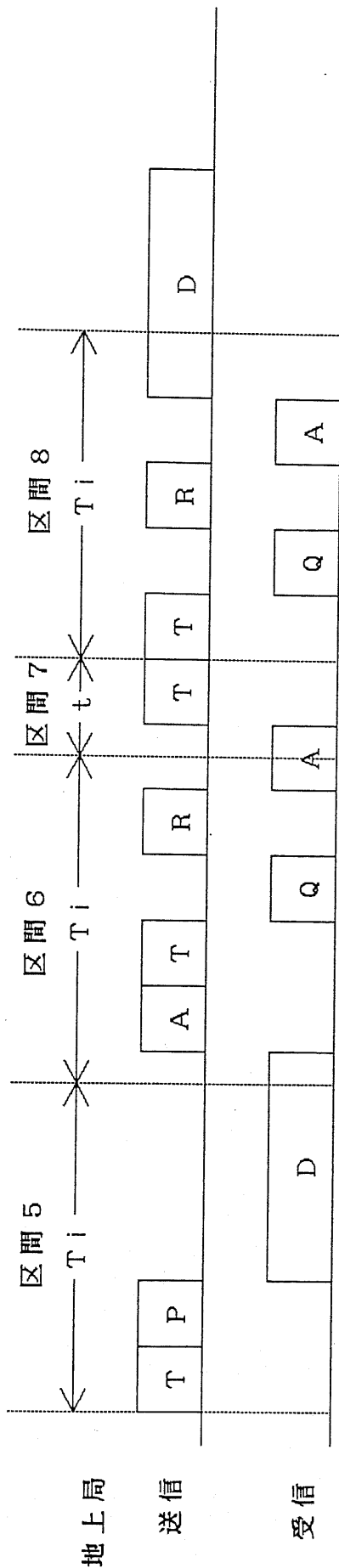
T_i : トリガ間隔時間

(走行データ通信に少なくとも1回チャンネルが割り当てられる
時間間隔)

図3-6-1 走行データ・汎用データ通信への割当アルゴリズム



(a) 汎用データ通信の要求が常にある場合



(b) 汎用データ通信の要求が一時的になくなる場合

T: トリガ, Q: 要求, R: 応答, A: 確認, P: ボーリング, D: 汎用データ
 T_i : 平均トリガ間隔時間, t : 区間 7 の長さ ($t < T_i$)

図 3-6-2 走行データ・汎用データ通信への割当の例

通信チャネルの割当は、まずトークンがすでに割り当てられた自動車の各々に対して、次にトークンがまだ割り当てられずコンテンションアクセスにより走行データ通信を行う自動車に対して、という順番でサイクリックに割り当てることとする。

汎用データ通信チャネルの割当方式として、ここでは新たに各自動車の走行情報を利用した方式を提案した。自動車相互の速度の差が大きいときや伝送要求パケット数が異なるときは、これらの情報を用いて、汎用データ通信チャネルを計画的に割り当てることのできれば、それによりチャネル利用率の改善等が期待できる。

具体的には、通信チャネルを1つのパケットが伝送可能な一定長のスロットに分割し、これを汎用データの通信要求のある自動車に割り当てる。この時、各自動車の速度情報を基に使用可能なスロットを割り出し、はじめは各自動車が使用できる最も先のスロットから、各自動車への割当数が同じになるように順番に割り当ててゆく。そして、これが終了したとき、前のスロットで空いているものがあれば、全体に割り当てたスロ

ットを前にシフトする。

この方式によるチャネル割当の例を、通常考えられるサイクリックに割り当てる方式や伝送済みパケット数が最小の自動車に割り当てる方式と併せて、図3-6-3に示す。ここでは説明を簡単にするため走行データ通信、汎用データ通信とも所要時間は60msecと仮定し、この長さのスロットに時間を区切り、各車に割り当てる場合を考える。この図中において、Dは自動車が通信ゾーンに進入した時に走行データ通信が行われるスロット、数字は汎用データ通信に割り当てられるスロットを、×は自動車が通信ゾーン内に存在しているにもかかわらず伝送するパケットがないスロットを示している。また、図中Nは要求パケット数（車載局が地上局へ伝送しようとしている汎用データパケットの数と、地上局がその車載局へ伝送しようとしている汎用データパケットの数の和）を表している。本方式では、伝送要求パケット数の割に長く通信ゾーンに存在する自動車に対して先にチャネルを割り当てることがないため、通信チャネルのスループットを向上できることが期待できる。

$V = 75 \text{ km/h}, N = 7$

D	1	2	3		4	5	6	
				D				1 2 3 ×

$V = 75 \text{ km/h}, N = 3$

① 走行情報に基づく割当

$V = 75 \text{ km/h}, N = 7$

D	1	2	3		4		5	
				D		1		2 3 × ×

$V = 75 \text{ km/h}, N = 3$

② サイクリックに割当

$V = 75 \text{ km/h}, N = 7$

D	1	2	3					
				D	1	2	3	× × × ×

$V = 75 \text{ km/h}, N = 3$

③ 伝送済みパケット数最小の自動車に割当て

V : 走行速度、 N = 伝送要求パケット数

図 3 - 6 - 3 各自動車への割当の例

次に、この方式の効果をみるために、75 km/hで走行する2台の自動車が見用データ通信を行う場合に各自動車が伝送可能なパケット数を理論的計算により求めた。ここでは、伝送要求パケット数が7以上の自動車Aと、1から7までの値をとる自動車Bの2台が存在する場合について考える。チャネルを2台の自動車に割り当て必要のある場合を全て含むように自動車Aが進入してから自動車Bが進入するまでの時間の差が、AよりBが8スロットに相当する時間後である場合からAよりBが7スロットに相当する時間前である場合についての平均値を計算した。また比較のためサイクリックに割り当てる方式と伝送済みパケット数最小の自動車に割り当てる方式についても計算を行った。自動車Aと自動車Bが使用可能なスロット数の和を表3-6-1に示す。これらより走行情報に基づく方式では他の方式と比較して、通信チャネルにおける見用データ通信のスループットを向上できることがわかる。

表 3 - 6 - 1 各自動車に割り当てられるスロット数の比較

① 走行情報に基づく方式

② サイクリックに割り当てる方式

③ 伝送済みパケット数が最小の自動車に割り当てる方式

	割当 方式	自動車 B の伝送要求パケット数						
		1	2	3	4	5	6	7
自動車 A	①	6.38	6.13	5.75	5.38	5.19	5.00	5.00
	②	6.06	5.69	5.38	5.19	5.06	5.00	5.00
	③	6.00	5.50	5.00	5.00	5.00	5.00	5.00
自動車 B	①	1.00	2.00	3.00	3.88	4.44	4.88	5.00
	②	1.00	2.00	3.00	3.88	4.50	4.88	5.00
	③	1.00	2.00	3.00	3.69	4.25	4.69	5.00
A B の 和	①	7.38	8.13	8.75	9.26	9.63	9.88	10.0
	②	7.06	7.69	8.38	9.07	9.56	9.88	10.0
	③	7.00	7.50	8.00	8.69	9.25	9.69	10.0

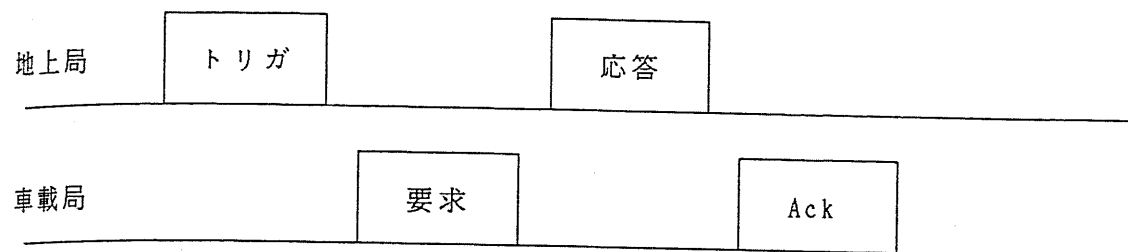
3. 7. 路車間通信制御[7][8][14][17]

路車間通信制御機能は通信制御装置と車載局の間における通信手順に関する制御を行う機能である。この基本的な機能として、ここでは交信シーケンスの制御と、汎用データ通信の伝送パケットの順序制御を考える。

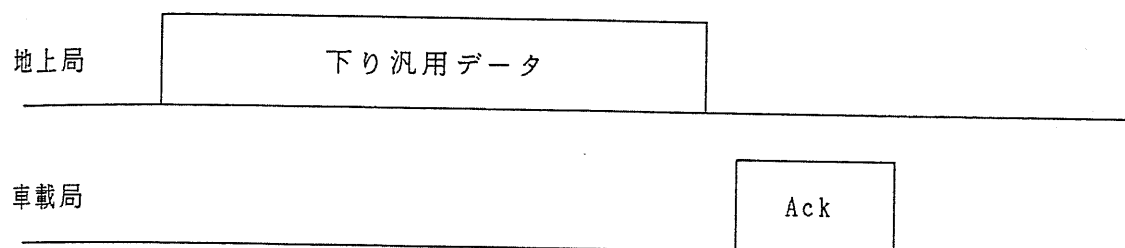
交信シーケンスの制御は通信制御装置と自動車との間の接続制御と伝送済みパケットの確認、及び、誤り制御を行う。汎用データ通信の伝送パケットの順序制御は複数の汎用データパケットを伝送する場合に、パケットの抜けや重複がないようにその伝送順序を制御する機能である。

○交信シーケンスの制御

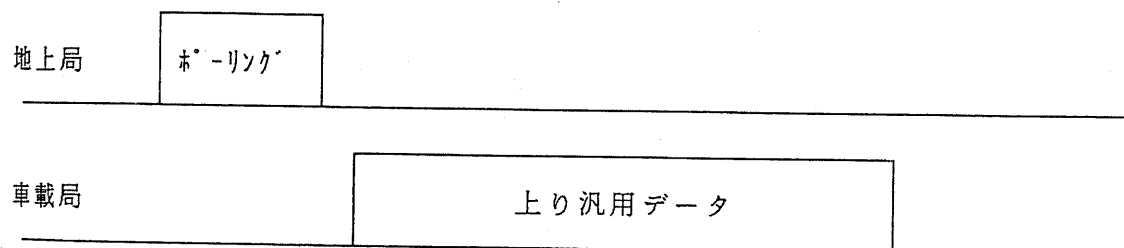
走行データ通信及び汎用データ通信における基本交信シーケンスを図3-7-1に、また、各信号の情報内容を表3-7-1に、各々示す。このように、汎用データ通信の制御情報を走行データ通信の伝送パケットに付加し、汎用データパケットの後に伝送される制御情報のやりとりの一部を走行データ通信の中で行う



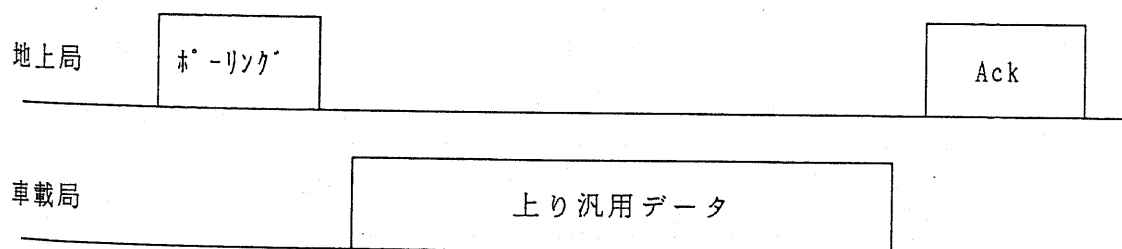
(a) 走行データ通信



(b) 下り汎用データ通信



(c) 上り汎用データ通信 (途中データパケット伝送時)



(d) 上り汎用データ通信 (最終データパケット伝送時)

図 3 - 7 - 1 基本交信シーケンス

表 3 - 7 - 1 各信号に含まれる情報内容

○：常に含まれる情報

△：必要な場合のみ含まれる情報

(a) 走行データ通信の信号

	専用 トリガ	要求	応答	Ack	Nack
信号種別	○	○	○	○	○
トークン	○	○	○		
ゾーン I D			○		
走行データ		○	○		
上り汎用データ通信制御情報		○	△		
下り汎用データ通信制御情報		△			

(b) 汎用データ通信の信号

	上り			下り	
	ホ-リ ソグ	上り データ	Ack	下り データ	Ack
信号種別	○	○	○	○	○
トークン	○	○	○	○	
パケット番号	○	○		○	
汎用データ		○		○	

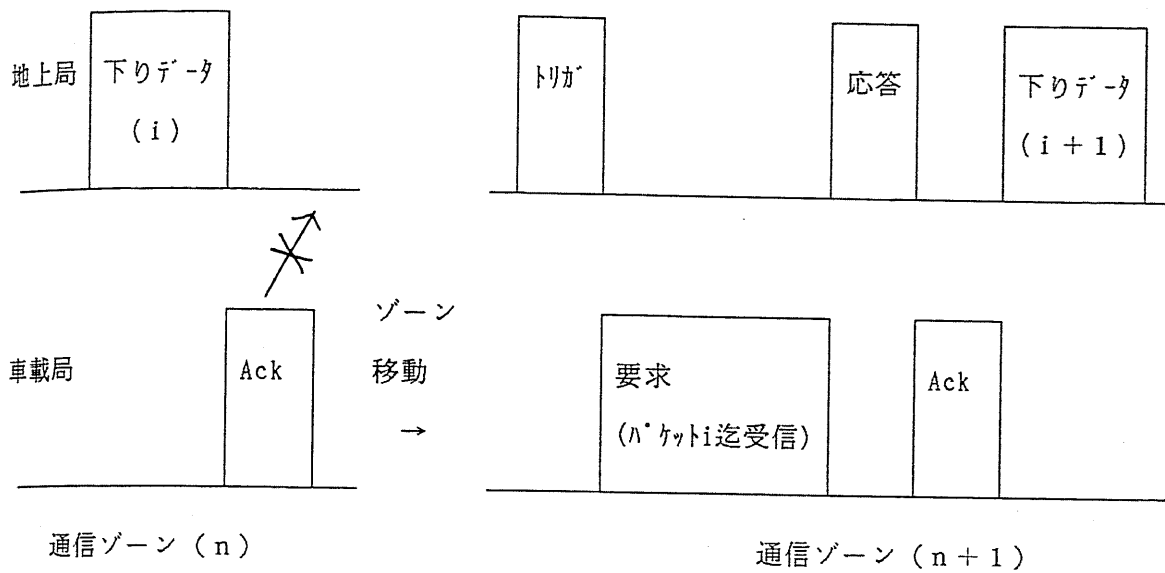
ことにより、交信シーケンスを簡略なものとした。

○ 伝送パケットの順序制御

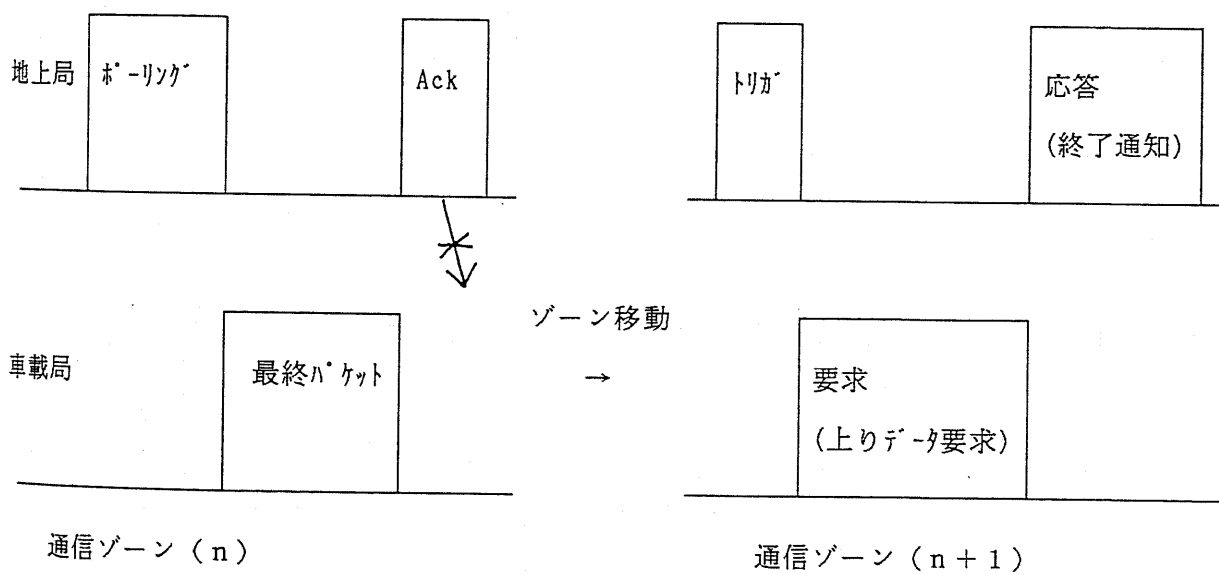
一方、本システムでは汎用データを500bit程度の小パケットに分割して伝送することは先にも述べた。

ここで、各々の汎用データパケットには0番から始まるパケット番号を付加する。これは伝送中のパケットの順序の管理だけでなく、通信相手側の装置に残りの汎用データパケットの数を通知する役割も持っている。下り汎用データ通信に関しては、地上側から送出されたデータパケット中にあるパケット番号とそれに応答して車側より送出される確認信号により伝送パケットの順序が制御される。上り汎用データ通信に関しては、地上局より送出されるポーリングに次に車側が送出すべきパケットの番号が含まれており、これが受信した汎用データパケットに対する確認信号の役割をも併せ持っている。つまり、次のデータパケットの送信を求めるポーリングが肯定応答（ACK）の役割を、同じデータパケットの送信を求めるポーリングが否定応答（NAK）の役割をそれぞれ果たす。

ある自動車が汎用データ通信を行っている途中で通信ゾーンを移動した場合には、次の通信ゾーンで汎用データの前段階として行われる走行データ通信の際に、汎用データ通信制御情報として前の通信ゾーンで伝送が済んだパケットの番号を受信側より通知することにする。こうすることにより、通信ゾーン移動時に送信側と受信側が送受信を確認したパケットの番号が一致しないために生じるパケットの伝送抜けや重複を避けることができる。その様子を図3-7-2に示す。例えば、下り汎用データ通信の確認信号を地上局が受信しない場合には、図3-7-2(a)に示すように次の通信ゾーンでの走行データ通信において、車載局が受信した伝送パケットの数を地上局に通知する。また、上り汎用データ通信でも一連のデータパケットの伝送終了時に地上側より確認信号を送出するが、これを車載局が受信できなかった場合には、図3-7-2(b)に示すように地上局は車載局の上り汎用データパケット伝送要求に対して、前の通信ゾーンで終了したことを通知する。



(a) 下り汎用データ通信



(b) 上り汎用データ通信

図 3 - 7 - 2 確認信号を受信しない場合の処置

3. 8. 個別情報リンク内管理[7][8][14][17]

第2章でも述べたように、本システムでは個々の自動車の走行や通信状況に関して、地上局が各種の制御のために保持すべき情報を個別情報と呼ぶことにしている。そして、これを管理する機能が本節で扱う個別情報管理機能である。これには転送機能、更新機能、廃棄機能がある。

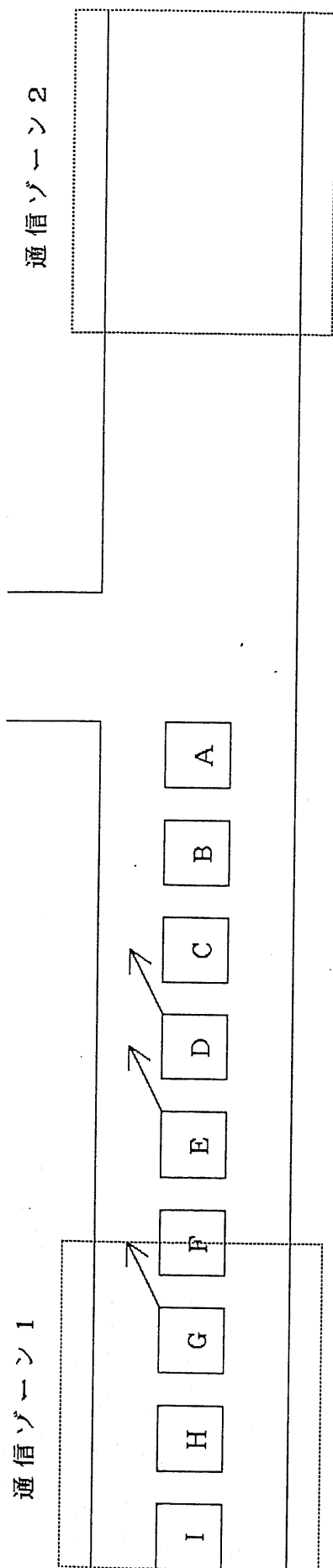
転送機能にはある通信ゾーンにおいて走行データ通信や汎用データ通信が終わったときに必要な情報を通信制御装置から路側装置へ転送する機能と、この情報を受け取った路側装置が次の通信ゾーンの通信制御装置に必要な情報を返送する機能がある。更新機能とは転送された情報に基づいて個別情報を新たに設けたり、更新したりする機能である。廃棄機能とはある自動車が駐車、Uターンなどなんらかの理由でリンクの外に出たと判断される場合において、必要なくなった個別情報を路側装置及び通信制御装置より消去する機能である。以下この廃棄の方式について説明する。

基本的な考え方としては、ある自動車に対して、同一車線を走る後続の自動車が当該自動車よりも先の通

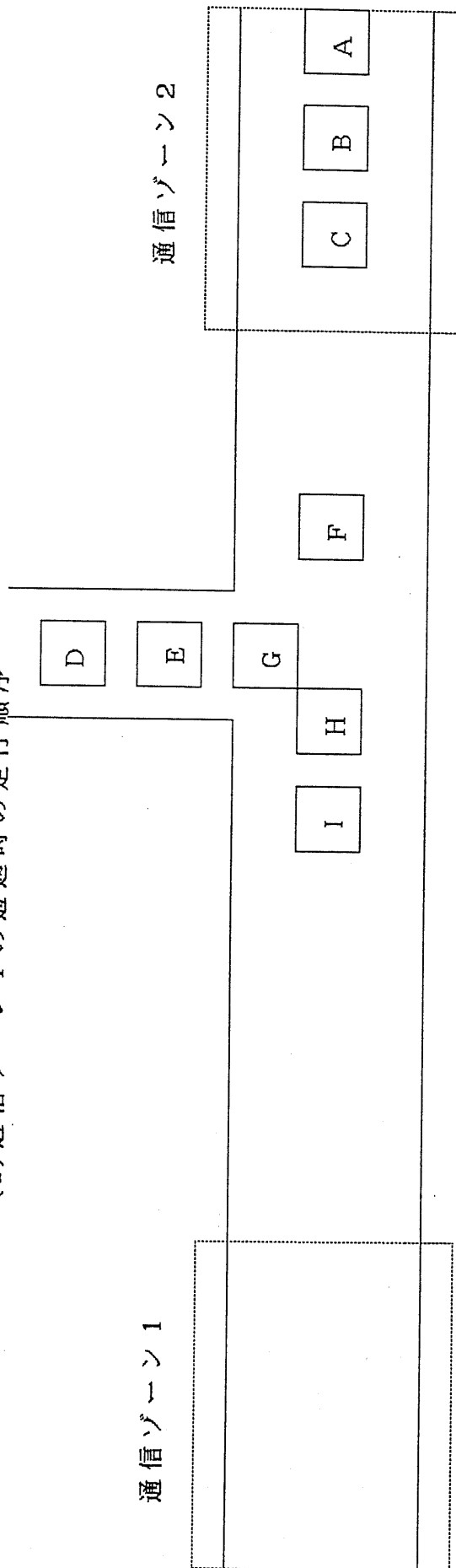
信ゾーンに次々と到着した場合には、その自動車は道路リンクよりいなくなったか駐停車及びUターンしたかのいずれかであり、その自動車の個別情報を廃棄することにする。このためには、廃棄対象車と同じ車線の後続の自動車の、前の通信ゾーンにおける走行データ通信の終了時刻の順序と、現在の通信ゾーンにおける走行データ通信の終了時刻の順序を比較して判断すればよい。通信制御装置は個別情報の廃棄を決定するための情報として走行データ通信が終わった自動車の中で、前のゾーンで最も遅い時刻に走行データ通信を終了した自動車数台の、前のゾーンにおける走行データ通信終了時刻を車線ごとに管理している。ある自動車のトークンを含むトリガを送出したがそれに対して応答が無かったとき、当該車線の記録を参照し、同じ車線の自動車の前の通信ゾーンにおける走行データ通信完了時刻が判定対象車の時刻よりも前のものが一定の台数（ N とする）以上であればその車の個別情報を廃棄することとした。ここで、各自動車が走行データ通信を正常に終了した場合には適宜記録を更新する。

廃棄の対象車の割合をパラメータとして、廃棄待ち

台数の確率分布をシミュレーションにより求めた。シミュレーションのモデルは図3-8-1に示したように、片方向1車線の道路において一定間隔で発生した自動車が通信ゾーンを通過し、それらの中に次の通信ゾーンの手前にあるわき道に入る廃棄対象車がランダムに混在するものとした。ここで廃棄を決めるためのパラメータである被追越し台数 N は2台とした。この結果を図3-8-2に示す。これより廃棄対象車の発生率が数パーセント以下であれば、廃棄待ちの自動車が多数発生することがないことがわかる。



(a) 通信ゾーン1の通過時の走行順序



(b) わき道通過後の走行順序

図3-8-1 廃棄のシミュレーションで想定する道路

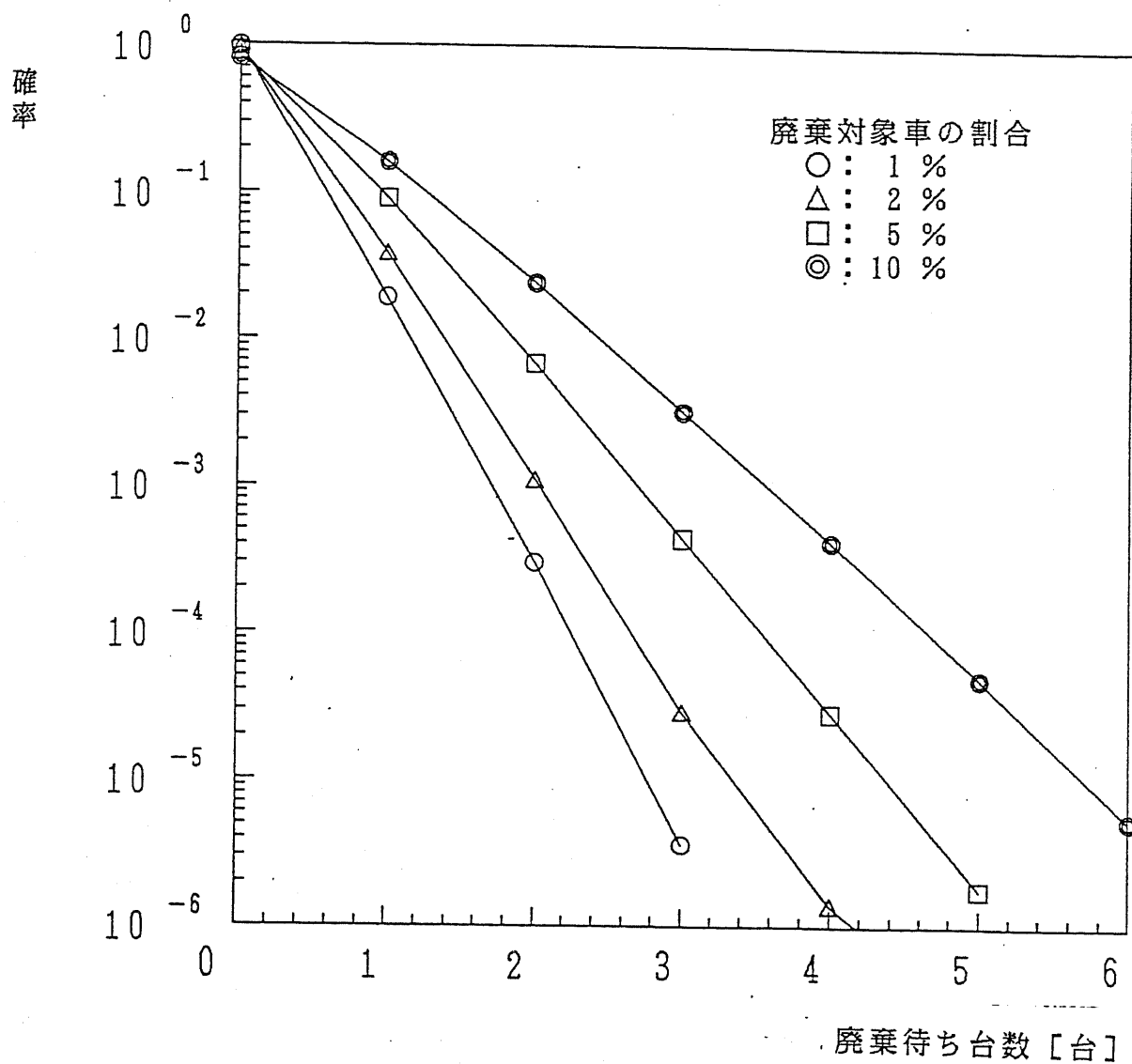


図 3 - 8 - 2 廃棄待ち台数の確率分布

3. 9. 結 言

本章では、小ゾーン連続形自動車パケット通信システムの通信制御方式の中で、特徴的な部分ともいえる、道路上に連続的に配置された極小の通信ゾーンを移動する自動車を、道路のリンク単位で把握するための道路リンクレベル制御方式について述べた。この中では、各種の制御方式の提案とそれらのシミュレーションによる評価を行うとともに、制御方式の具体的設計を行った。

なお、これらの内容は第8章において述べるスケールモデルに制御方式として盛り込まれており、室内実験においてはその動作確認とある程度の性能評価を行っている。従って今後は、都市雑音の影響などのより現実的な面において検討を行うとともに、さらに実フィールドに近い状況を想定した実験などにより、その有効性を評価してゆくことが必要になってくるものと考えられる。

4 . グローバルレベル 制御

4. 1. 序言

先の第3章では、本システムの制御方式の中でもRSS以下の部分で行われている道路リンクレベルの制御について説明した。

ここで実際には、自動車の移動は1つの道路リンク内にはとどまらず複数の道路リンクにまたがったものが通常の態様と考えられる。従って、本システムの制御方式として道路リンクレベルの制御方式に加えて、走行誘導の機能においては隣接した道路リンクの状況を情報として取り込めるようにし、また、汎用データの通信機能においては、データの通信中に自動車が道路リンクを移動した場合も複数の道路リンクにまたがってそれを追跡してデータ通信を継続させるようにする必要がある。本研究ではこのためにRSSより上位層で行う制御をグローバルレベル制御と名付た。

本章ではこのグローバルレベル制御について、まずその機能を説明する。次に、地上側負荷の軽減を図る

ために車載機側のインテリジェンスを積極的に利用した個別情報の自立分散型管理概念を導入し、さらに、道路リンクの旅行時間を予測して情報として利用することにより制御方式の効率化を実現する方式を提案する。そして、それに関してミクロモデルを用いた評価シミュレーションを行ってその有効性を示した。

4. 2. グローバルレベル制御の機能

4. 2. 1. 走行誘導機能

グローバルレベル制御の機能には、走行誘導と汎用データ通信がある。本節では、その中の走行誘導機能について説明する。

グローバルレベルにおける走行誘導機能は、第2章でも述べたように、基本的には各道路リンクへの走行誘導指示作成のボタンを指令するという形態で行われる。各道路リンクを管理するRSSは、各々の道路リンクにおける目的方向別台数などの情報を一定時間毎にACSに転送する。ACSでは各々の道路リンクにおける誘導指示作成のボタンを収納したテーブルを持っており、この情報をインデックスとしてこれを検索することにより各RSSでの誘導指示作成のボタンを決定する。このボタンは各々のRSSへ通知され、各道路リンクにおける誘導指示の作成がマクロに制御される。

4. 2. 2. 汎用データ通信機能

汎用データ通信機能には上りデータ通信と下りデー

タ通信がある。各々の機能は以下に示したようなものである。

○上りデータ通信機能

上りデータ通信においては車載機からのデータパケットを自動車が途中でリンクを移ってもその順序を維持しつつ確実に上位層へ転送する機能が必要になる。

○下りデータ通信機能

下りデータ通信においては、地上側において自動車がどの道路リンクに存在するかを把握していてそこへ下り汎用データを転送し、また、自動車が途中でリンクを移ってもそれを追跡してデータ伝送を継続させる機能が必要となる。

4. 3. 個別情報の自立分散型管理

4. 3. 1. 個別情報の自立分散型管理概念

第2章でも述べたように、本システムの構成においては中央集中型システムにおいて見られる種々の欠点を回避するために各装置に管理機能を階層的に分担させる垂直型機能分散を導入している。しかし、このようにシステムを階層的に構成する場合、地上側で個々の自動車を捉えてゆこうとすると車両が管理区域を移動する毎に装置間でその制御情報の引継を行う必要がある。また、車両移動に伴う追跡制御を全て地上側で行おうとすると、地上側におけるその情報処理の負荷が極端に大きくなるという問題点がある。

一方で、車載機の情報処理装置は年々高性能化している。そこで、ここでは車載機のインテリジェンスを積極的に利用して地上側負荷を軽減するための”個別情報の自立分散型管理概念”を提案する。以下その概念について説明する。

本システムの機能としては個々の自動車をミクロに捉えてきめ細かな制御を行う走行誘導と、地上と各自動車との間で汎用目的の個別通信を行う汎用通信の2

つがある。走行誘導のためには、各 R S S において各自動車の目的地、位置、及び速度等の情報を各自動車個別の情報として管理しておく必要がある。さらに 4. 2. 1. に述べたようにこれをグローバルな機能として実現するためには当該自動車が道路リンクを移動したときに R S S 同士の間でその情報を引き継ぐ必要がある。汎用通信のためには、地上側において各自動車との間のデータパケットのやり取りの状態に関する情報等をデータパケットそのもの含めて各自動車個別の情報として管理する必要がある。さらに 4. 2. 2. に述べたようにこれをグローバルな機能とするためには、走行誘導の場合と同様に、この情報を自動車の移動に合わせて R S S 間で引き継ぐ必要がある。ここでいう個別情報とは、以上のような目的のために、地上側において保持する必要がある、個々の自動車の走行や通信の制御に関する情報を総称したものである。そして個別情報の自立分散型管理とは、車載機のインテリジェンスを利用して、上記のような個々の車の走行・汎用通信の制御のために必要な個別情報の管理を道路リンクより上位のレベルでは車載機に管理させ、道

路リンク移行時の制御を車載機主導型で行うことにより、地上側で自動車をミクロに捉えながらも上位層での負荷を軽減させるというものである。

ここで新たに提案する個別情報の自立分散型管理の概念は本システムでは各機能毎に4. 3. 2. と4. 3. 3. に示したように具体化されている。

4. 3. 2. 走行誘導機能の実現

走行誘導機能における個別情報の自立分散型管理概念の導入の様子を図4-3-1に示す。この場合、図に示したように道路リンクレベルとグローバルレベルの制御をACSとRSSの間において分離する。そして、道路リンクレベルの制御に関しては先に述べたように、通信ゾーンが線的に配置されていることによる特徴を生かして走行情報を利用するなどして各RSSにおいて効率的な制御を行う。一方、道路リンクレベルより上位層のグローバルレベルでは、個々の自動車の目的地やIDといったミクロな情報は車載側で管理し、ACSやMCCでは方向別台数や車線利用パターン等のマクロな情報のみを管理する。

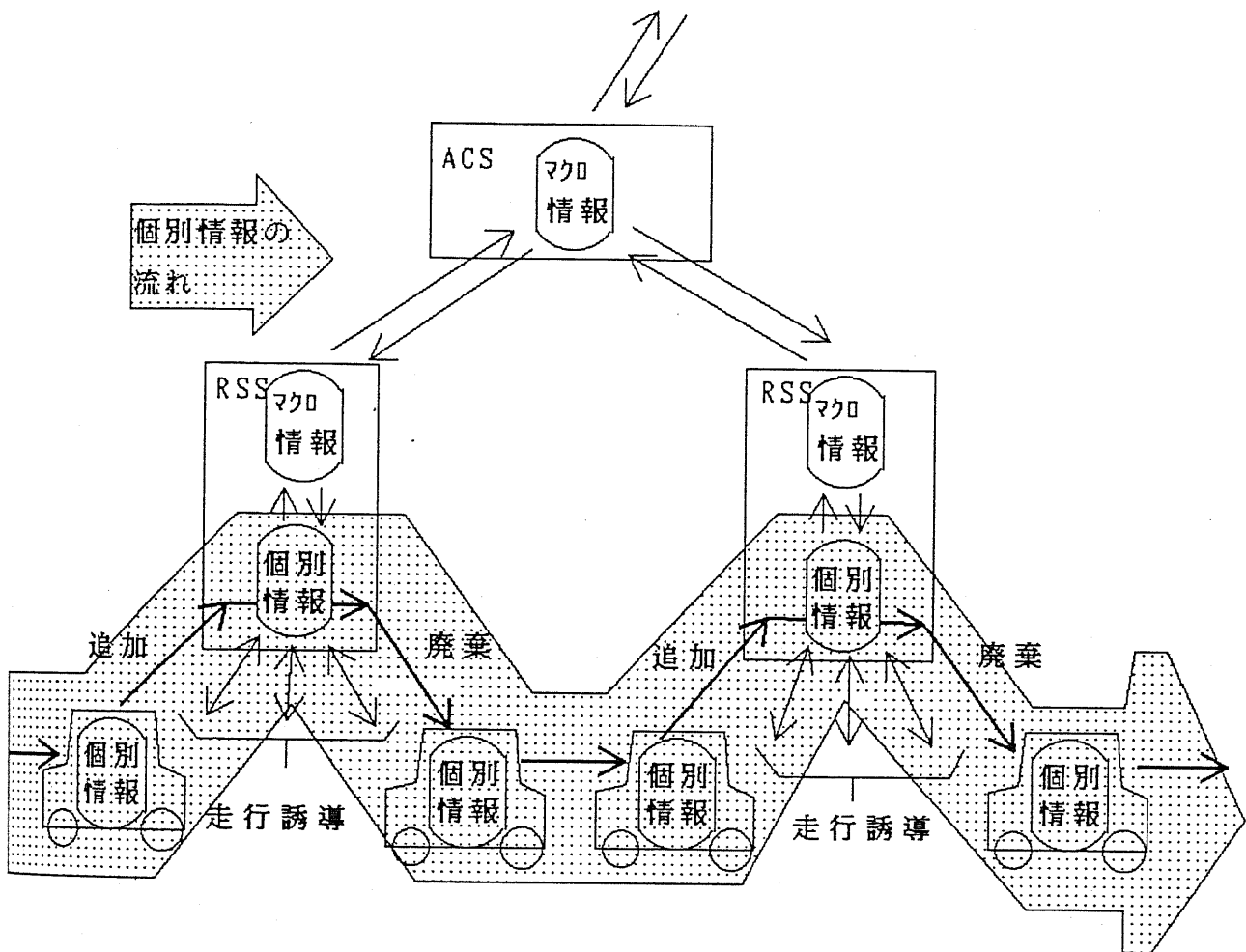


図4-3-1 走行誘導における個別情報の自立分散型管理概念の導入

4. 3. 3. 汎用データ通信機能の実現

汎用データ通信機能においても、4. 3. 2. の走行誘導機能と同様にリンクレベルとグローバルレベルの制御をACSとRSSの間において分離する。そして、道路リンクレベルより上位層では車載側が主体となって制御を行ってゆく。上りデータ通信では図4-3-2に示したように、グローバルレベルでは車載機がパケット番号等のシーケンスの制御情報等を管理して、車載機側主導型で通信を行ってゆく。下りデータ通信では図4-3-3に示したように、最初は地上側より一斉呼び出しにより車載機を起動して、以後はグローバルレベルでは車載機側でシーケンス等の制御情報を管理して、車載機主導型で通信を行ってゆく。

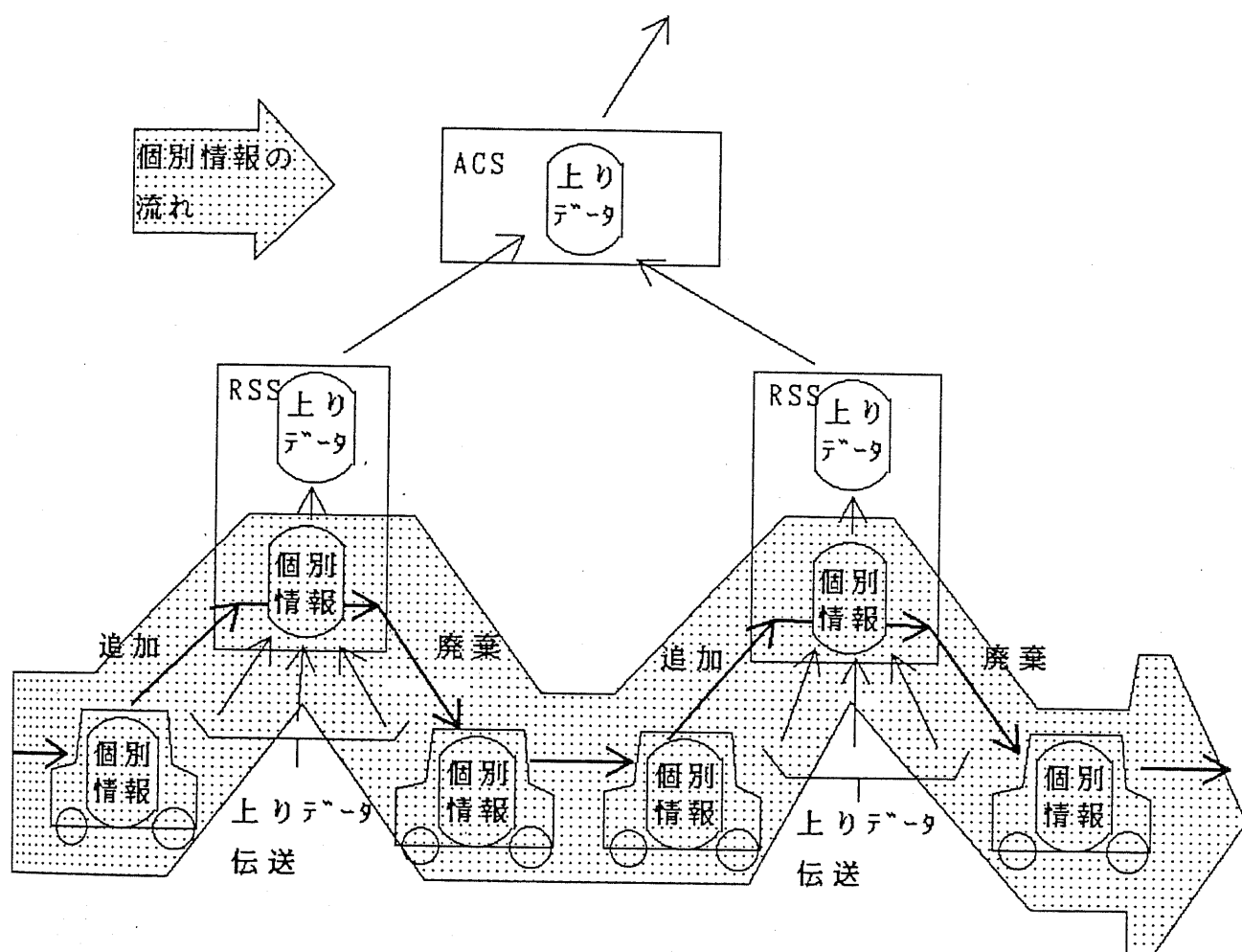


図 4 - 3 - 2 汎用通信（上りデータ伝送）における
個別情報の自立分散型管理概念の導入

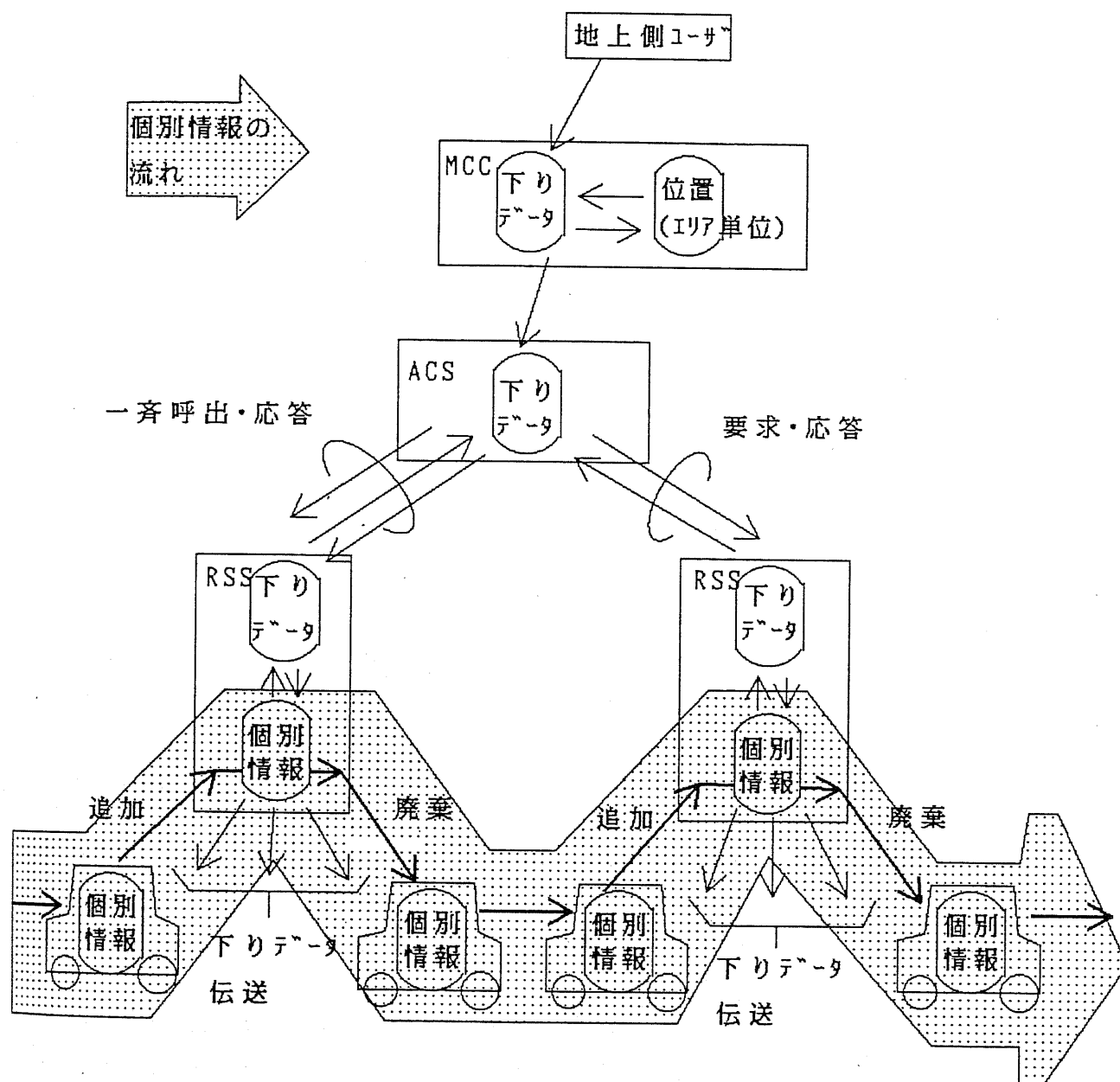


図 4 - 3 - 3 汎用通信（下りデータ伝送）における
個別情報の自立分散型管理概念の導入

4. 4. 汎用データ通信のグローバルレベル制御

4. 4. 1. パケット伝送の基本パターン

本節では、これまで述べたグローバルレベル制御の中で特に汎用データ通信機能にスポットを当てる。汎用データ通信機能には、上りデータ通信と下りデータ通信がある。ここでは、各々におけるパケット伝送の基本パターンについて説明する。

○上りデータ伝送

上り方向へのパケット伝送制御の基本パターンを図4-4-1に示す。この場合に関しては、本システムの装置構成のトポロジーがツリー状なのでルーティングの上での問題はなく、図4-4-1に示したようにRSS、ACS各装置のデータバッファを経由しながら順次上位方向に伝達すればよい。

○下り方向データ伝送

(初期パケット伝送時)

この場合は、図4-4-2(A)に示したように、HMの情報を基にエリアを特定し、エリア内の道路リ

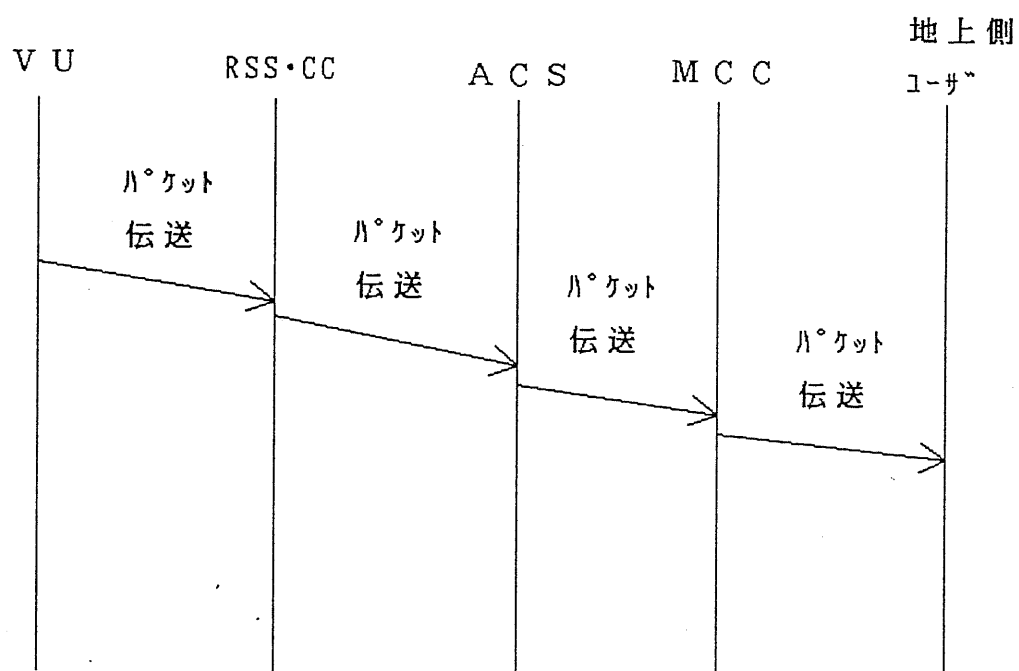
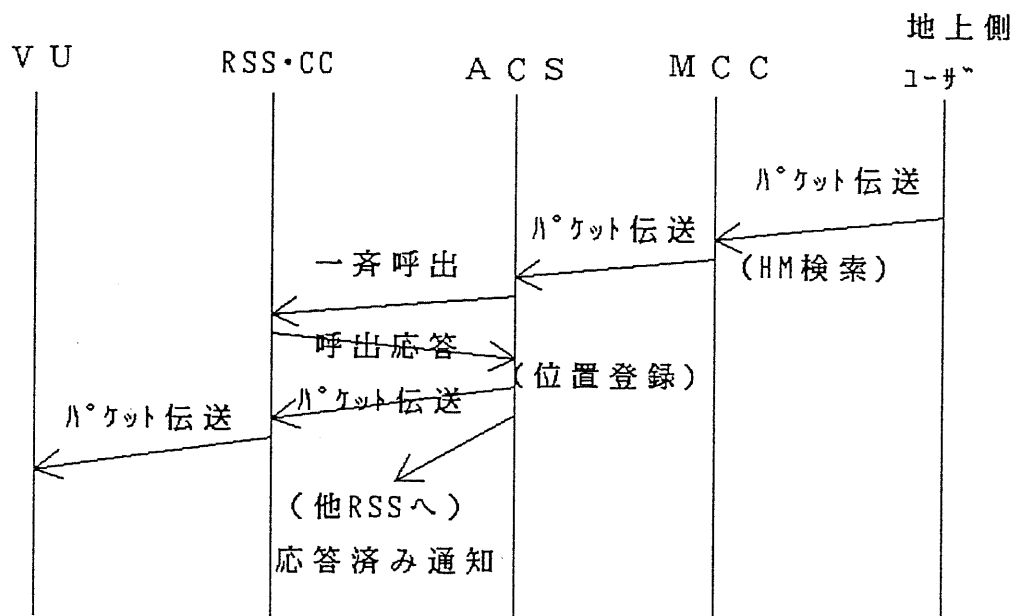
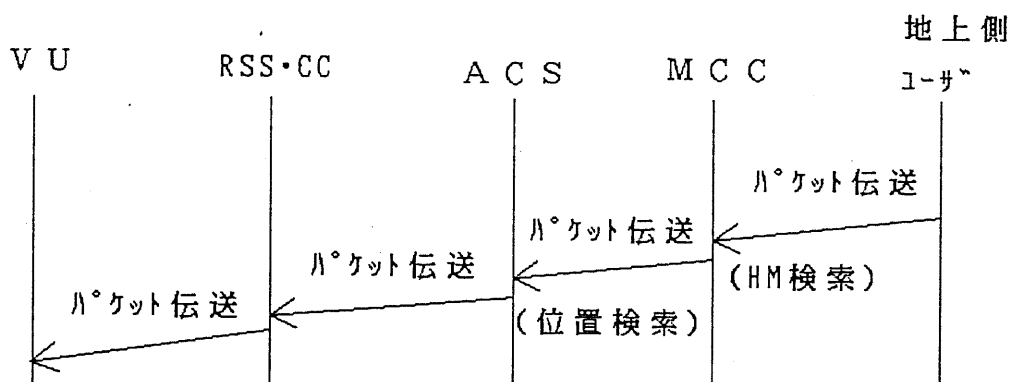


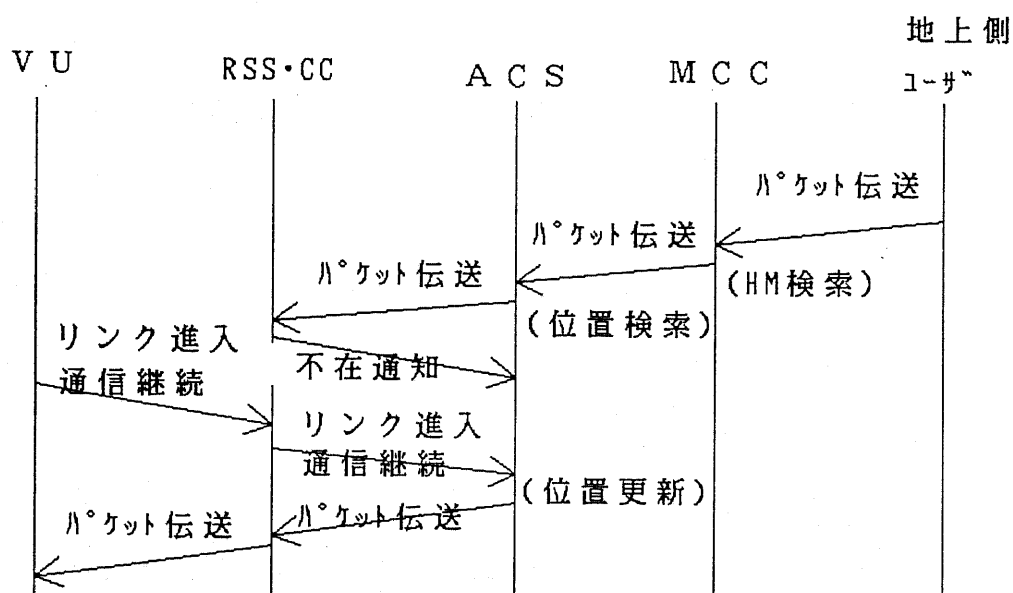
図 4 - 4 - 1 上り方向へのパケット伝送の基本パターン



(A) 初期パケット伝送時



(B) - ① 途中パケット伝送時 (通常時)



(B) - ② 途中パケット伝送時 (通信中リンク移動時)

図 4 - 4 - 2 下り方向へのパケット伝送の基本パターン

リンクに一斉呼び出しを行い車載機を通信状態にする。

(途中パケット伝送時)

途中のデータパケットを伝送中の場合においては、
図4-4-2(B)-①に示したように、通信中は車載機主導型でACSに車両の存在する道路リンクを登録しておく。そして、MCCからの下りデータはこの情報にしたがって伝送される。ここで、その自動車が道路リンクを移動した場合には図4-4-2(B)-②に示したように次の道路リンクへの進入を待って伝送を行う。

4. 4. 2. データパケットのフロー制御

汎用データ通信機能を実現するには、4. 4. 1.の内容に加えて、データパケットのフロー制御が必要になる。以下、上りデータ通信及び下りデータ通信におけるフロー制御方式について述べる。

○上りデータ伝送

この場合においては、図4-4-3に示すように、CCより上位層では通常用いられるマルチバッファに

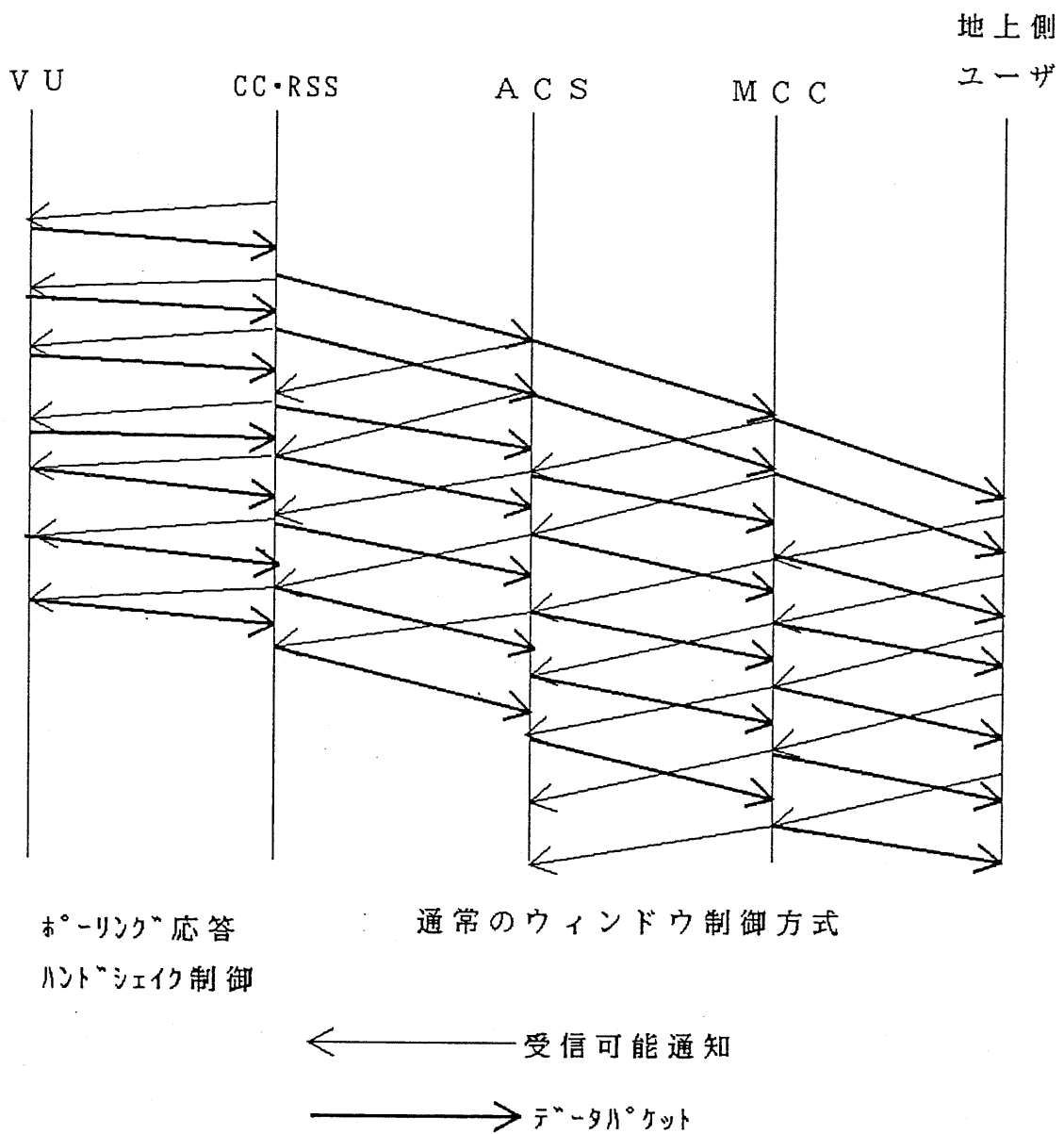
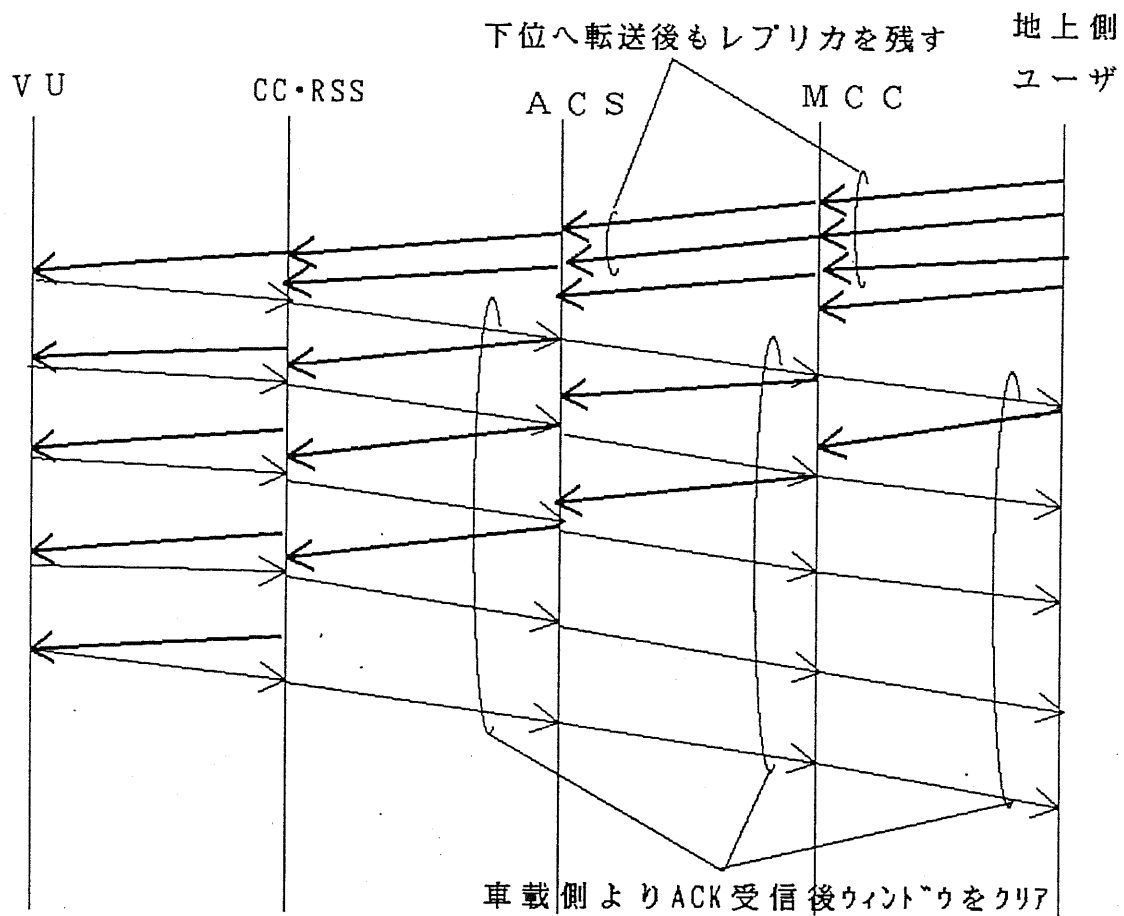


図 4 - 4 - 3 上り方向パケット伝送のフロー制御

よるウィンドウ制御により可能と考えられる。また、
車載機とCCとの間ではポーリング・応答に基づくシ
ングルバッファによるハンドシェイク制御を行う。こ
れは路車間の通信チャネルの割当は地上側が管理する
ことによる。ただしこの図では、CC・RSSにおけ
るバッファ数は1パケット、ACSとMCCにおいて
は2パケットの場合を想定している。各装置はバッ
ファ数分はデータを上位層へ無条件で送れるが、それ以
降は上位層からの受付可能通知に応答する形となる。

○下りデータ伝送

本システムのように階層的にシステムを構成してい
る場合においては、下りデータ伝送のフロー制御とし
て通常のウィンドウ制御を用いると、車両が道路リン
クを移動した場合にはウィンドウ内に残っているデー
タパケットを次の道路リンクのRSSに受け渡さなけ
ればならず、地上側の制御負荷の増大を招く。ここで
は、図4-4-4に示したように、下位層にデータパ
ケットを伝送する際に上位層にもそのレプリカを残し
おき、車両からACKを受信した分に関り上位層のウ



データACK

ハッシュ制御

提案した制御方式

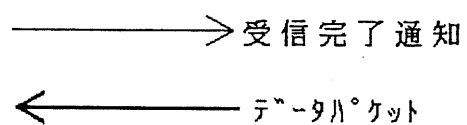


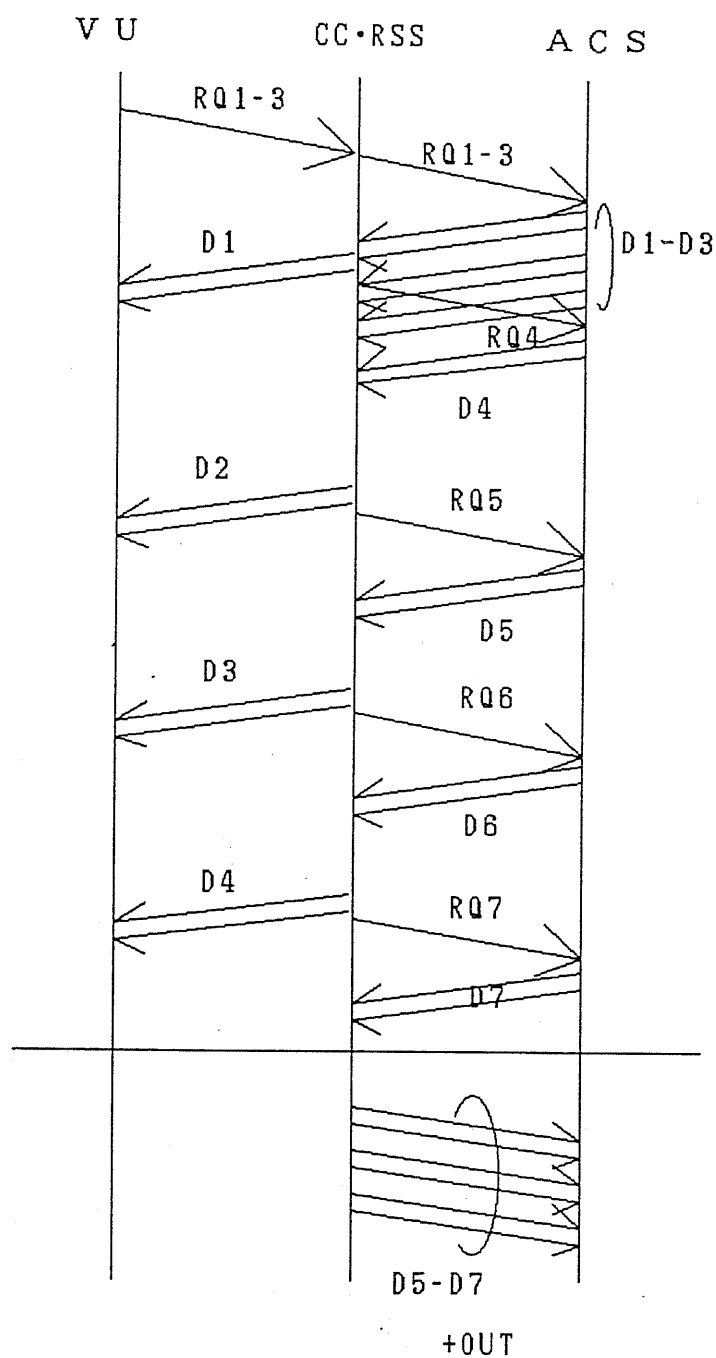
図4-4-4 下り方向パケット伝送のフロー制御

ウィンドウ内のデータをクリアしてゆく方式を提案する。
ただしこの図では、CC・RSSにおけるバッファ数は2パケット、ACSは3パケット、MCCは4パケットの場合を想定している。各装置は車載機からの受信完了通知を受けて当該パケットをクリアすると同時に、次のパケットを下位層へ伝送する。

4. 4. 3. リンク旅行時間予測による効率化

本システムでは地上側の管理負荷を軽減するために個別情報の自立分散型管理の概念を導入することにより、車載機主導型で通信を行ってゆくことは4. 3. にも述べた。ここではさらに自動車から要求のあったデータをACSよりRSSに転送する際にRSSにおいてその自動車のリンク旅行時間を予測し、その自動車が当該リンクにおいて通信可能と予想されるデータパケットのみをRSSに転送することにより、無駄なデータ転送をなくし制御方式の一層の効率化を図る。

図4-4-5は、地上側でデータ通信の管理を行い通常用いられるウィンドウ制御によるフロー制御を行った通常方式、図4-4-6は自立分散型管理概念を



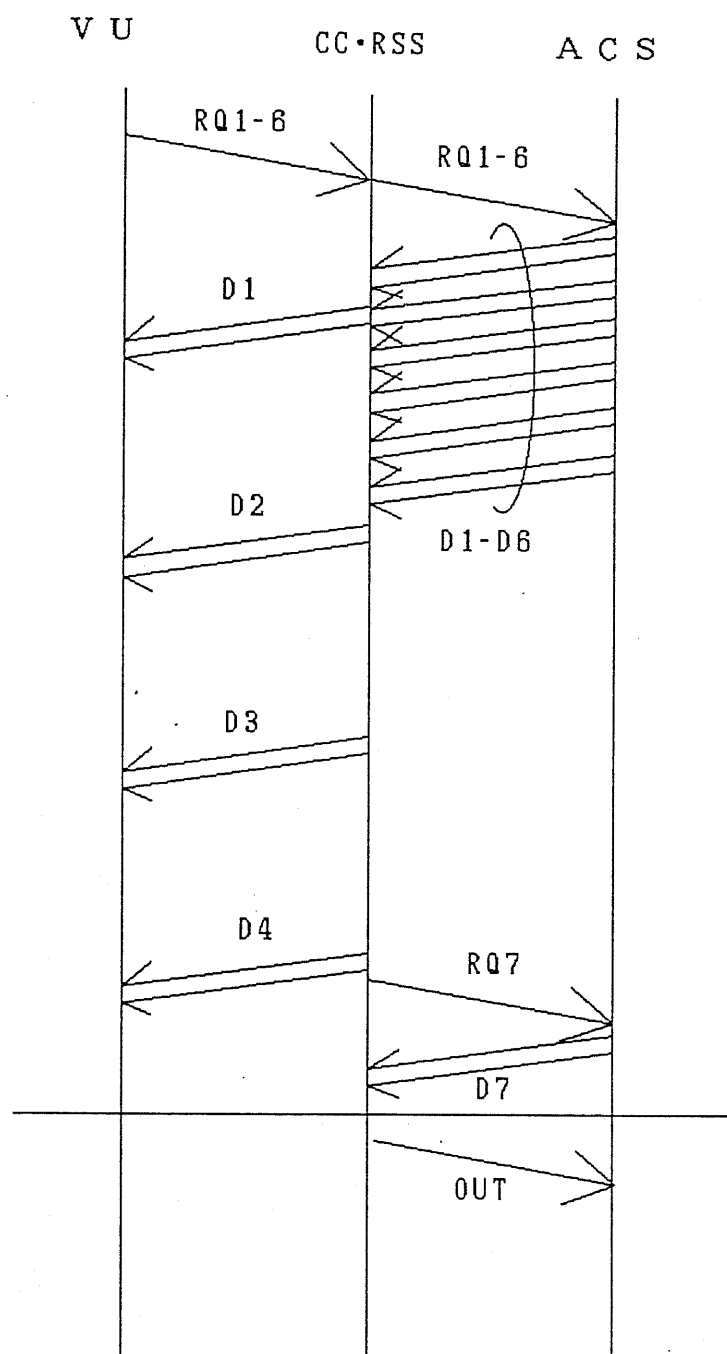
○信号説明

RQ*i* : *i* 番目のデータパケット転送要求

D*i* : *i* 番目のデータパケット

OUT : 自動車の退出通知

図4-4-5 通常の制御方式



○信号説明

RQ i : i 番目のデータバケット転送要求

D i : i 番目のデータバケット

OUT : 自動車の退出通知

図 4 - 4 - 6 新たに提案した制御方式

導入しリンク旅行時間による予測制御を行った新方式、
の各々に関して R S S - A C S 間の信号のやりとりの
例を示したものである。このように通常の方式では地
上側で通信シーケンスの管理を逐次的に行うために R
S S - A C S 間で多量の信号のやり取りをしなければ
ならない。これに対して本方式では車載機側が自立分
散的に管理を行うことに加えて、必要なデータはリン
ク旅行時間の予測により無駄なく一括して転送される
ので伝送信号は少なく済むと考えられる。

○リンク旅行時間の予測とそれに基づく伝送可能デー タ量の見積

ここでは、各自動車は道路リンクに進入したときに、
R S S においてそのリンク旅行時間を予測して、その
道路リンク内において伝送可能と考えられるデータパ
ケットのみを A C S より R S S に転送することを考え
る。しかし、このとき種々の要因によって実際の旅行
時間と予測値との誤差が生じる。従って実際のデータ
伝送量と予測値との間にも誤差が生じることになる。
実際のデータ伝送量の方が少ない場合には A C S より

転送されたデータパケットが無駄になる。予測値の方が少ない場合にはRSSより追加のデータ転送を要求する制御信号を伝送しなければならない。ここでデータパケットの情報量は制御信号の情報量よりも大幅に大きい。従って、ここでは変動する旅行時間の最小値を推定することにして誤差による損失を少なくすることとする。

具体的には当該道路リンクの交通量より自動車の平均走行速度を算出し、道路リンク内を自動車がその速度でノンストップで走行できたとしてリンク旅行時間の最小値の予測を行い、データ転送量を見積ることとする。このときGreenshieldsのモデルによれば、各車線の交通量 Q_{lane} と平均走行速度 V_{avr} の間には、

$$Q_{lane} = V_{avr} K_j (1 - (V_{avr} / V_f))$$

..... (式 4 - 4 - 1)

の関係があることが知られている[25]。ここで K_j は最大密度、 V_f は自由走行速度を表している。これを変形すると、

$$V_{avr} = (V_f / 2) \cdot (1 + \sqrt{1 - 4 Q_{lane} / K_j V_f}) \dots\dots\dots (式 4 - 4 - 2)$$

となる。これよりリンク長を L とするとリンク旅行時間 T_{trip} は,

$$T_{trip} = L / V_{avr} \dots\dots\dots (式 4 - 4 - 3)$$

と求められる。

一方、各通信ゾーンにおいて毎秒毎に伝送されるデータパケット数 N_{ps} は,

$$N_{ps} = B \cdot S / D_p \dots\dots\dots (式 4 - 4 - 4)$$

である。ここで B はデータ伝送速度、 D_p はパケット長を表している。また、 S はスループットを表しており、スケールモデルによる実験により、

$$S = 0.55 - (0.0132 / Q_{lane} \cdot N_{lane})$$

..... (式 4 - 4 - 5)

となることがわかっている [20]. ここで N_{lane} は道路の車線数を表している. また自動車は各通信ゾーンにおいて通信チャネルを複数の車同士で共有することになる. ある自動車が一つの通信ゾーンにいるときに他の車線の自動車で同一の通信ゾーン内にいる平均の台数 N_v は,

$$N_v = (N_{lane} - 1) \cdot K_{avr} \cdot L_{zone}$$

..... (式 4 - 4 - 6)

となる. ここで通信ゾーンの長さを L_{zone} としてある. また平均の密度 K_{avr} は,

$$K_{avr} = (Q_{lane} / V_{avr}) \dots\dots (式 4 - 4 - 7)$$

と考えられる. 以上より, 各ゾーンにおける伝送可能なパケット数の予測値 D_{zone} は,

$$D_{zone} = N_{ps} \cdot (L_{zone} / V_{avr}) / (1 + N_v)$$

..... (式 4 - 4 - 8)

となるので、一つのリンクにおける通信ゾーンの数
 N_{link} とすると、各リンクにおける伝送可能なデータ
 量の推定値 D_{link} は

$$D_{link} = N_{zone} \cdot D_{zone} \dots \dots (式 4 - 4 - 9)$$

となる。本方式では自動車は各道路リンクに進入した
 ときに上記の計算を行い、当該道路リンクでの伝送可
 能パケット数の予測を行い A C S より R S S にデータ
 転送を行うものとする。

4. 4. 4. シミュレーション評価

本方式の効果をミクロモデルを用いた評価シミュレ
 ーションを行って調べた。この際使用したモデルにお
 いて自動車の走行は文献[26]と同様とした。まず交通
 流の巨視的特性に基づいて車間距離 - 希望速度のテ-

ブルを用意する。各車の位置及び速度は各々 5 m 及び 4.5 km/h 単位で表され一秒毎に更新される。車線変更は考慮しないものとし、各車は各々に対する先行車との距離によりテーブルを参照し以後の速度を決定するものとした。

またシミュレーションにおいては信号交差点を出口とした長さ 250 m 往復 6 車線の幹線道路からなる一つの道路リンクを対象とし、流入する自動車は道路リンク入口において指数分布で発生させるものとし、流入交通量は 1 時間 1 車線当り 800 台とした。なおここでは道路リンク出口部分は孤立交差点と考えて道路リンク流入部直前においては交通流は充分ランダムになっているものと仮定している。通信ゾーンの長さは 10 m、配置間隔は 15 m として 10 個の通信ゾーンが設置されているものとする。このとき、道路リンクの出口に設置してある交通信号のサイクルは 60 秒とし、スプリットは 0.5、スイッチングロスはないものとして 1200 秒間シミュレーションを行った。尚、この際用いた評価プログラムを付録 A. 2. に示す。

○ データ転送量

各リンク毎のメッセージの発生率をパラメタとして、メッセージ長が 5 kbit (10 ブロック) から 25 kbit (50 ブロック) の場合について、通常方式と新方式の双方について、ACS - RSS 間におけるデータ転送による負荷をみる。データ転送の総量を求めたものが図 4 - 4 - 7 (メッセージ長 = 5 [kbit]), 図 4 - 4 - 8 (メッセージ長 = 10 [kbit]), 図 4 - 4 - 9 (メッセージ長 = 15 [kbit]), 図 4 - 4 - 10 (メッセージ長 = 20 [kbit]), 及び、図 4 - 4 - 11 (メッセージ長 = 25 [kbit]) である。ここで結果の数値はシミュレーション時間で正規化してある。

新方式と通常方式のいずれの場合においてもデータ伝送の要求トラヒックが大きくなるにつれて通信ゾーンにおける路車間のデータ通信速度による伝送可能量の限界により ACS - RSS 間の転送データ量も飽和している。新方式では通信需要が特に大きい場合においては転送データ量が若干少なくて済むことがわかる。

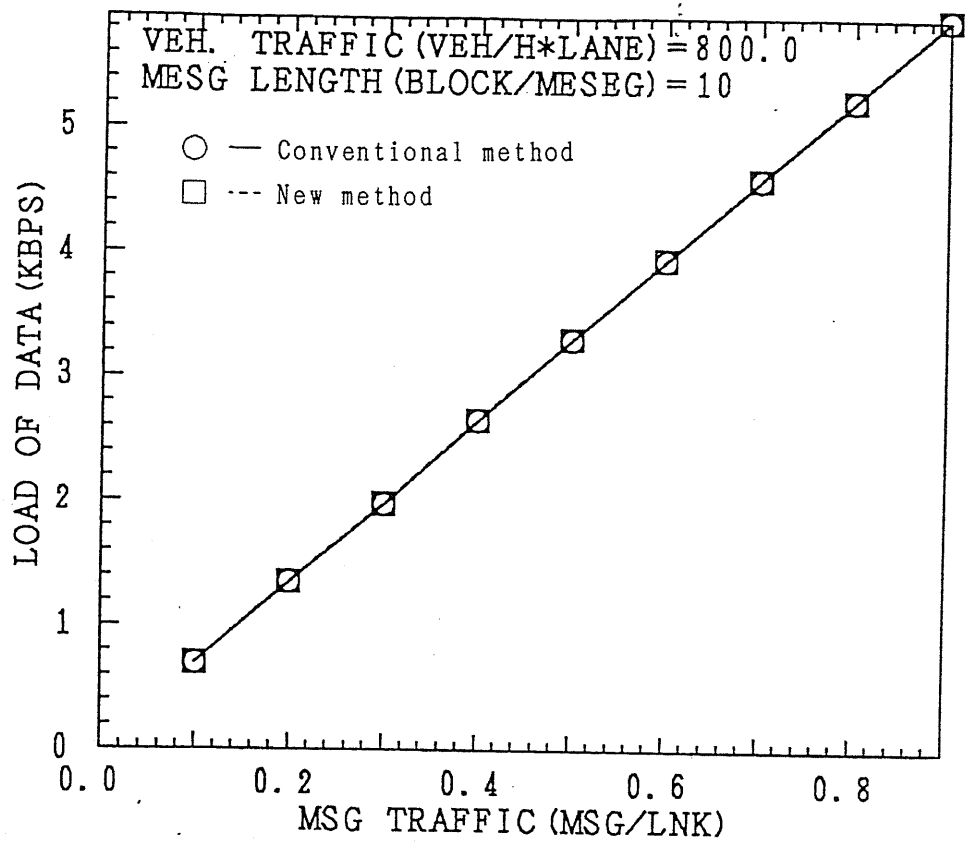


図 4-4-7 メッセージの発生率と平均データ転送量の関係
(メッセージ長 = 5 [kbit])

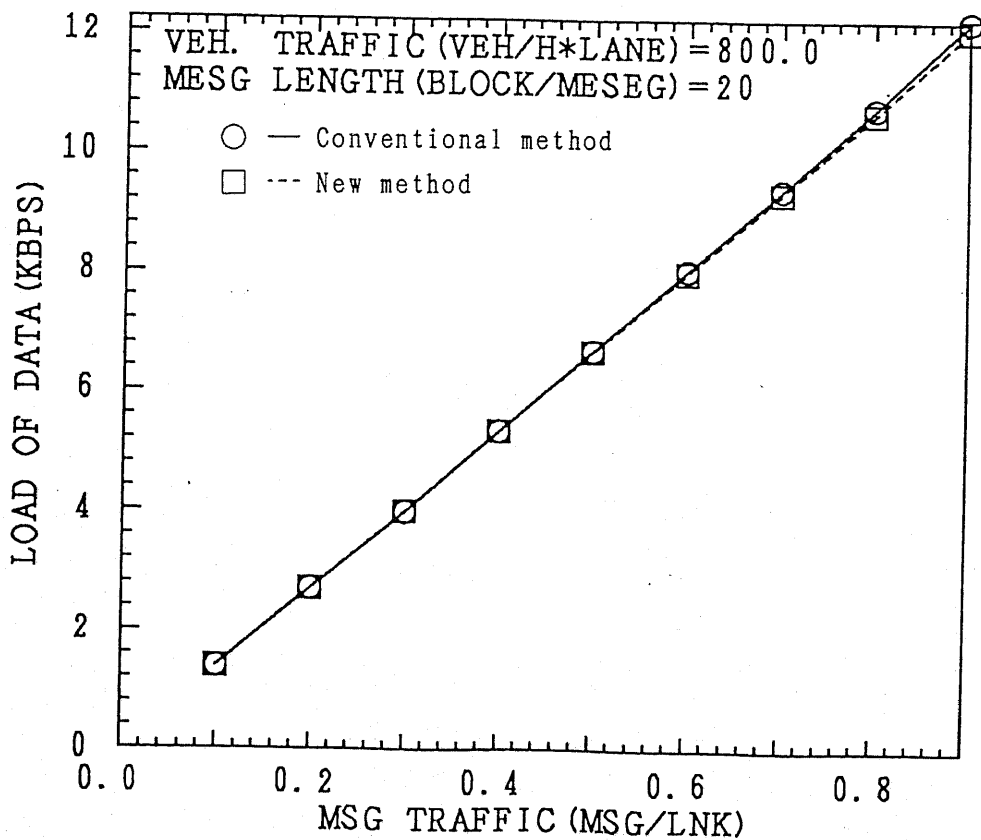


図 4-4-8 メッセージの発生率と平均データ転送量の関係
(メッセージ長 = 10 [kbit])

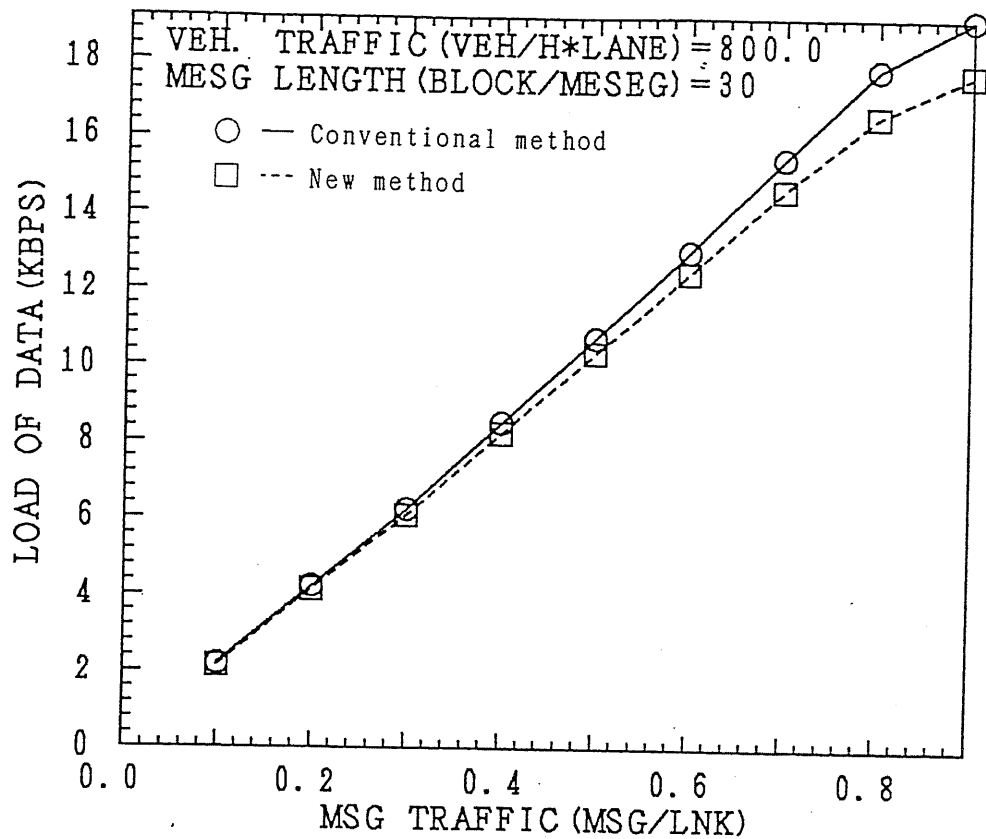


図 4-4-9 メッセージの発生率と平均データ転送量の関係
(メッセージ長 = 15 [kbit])

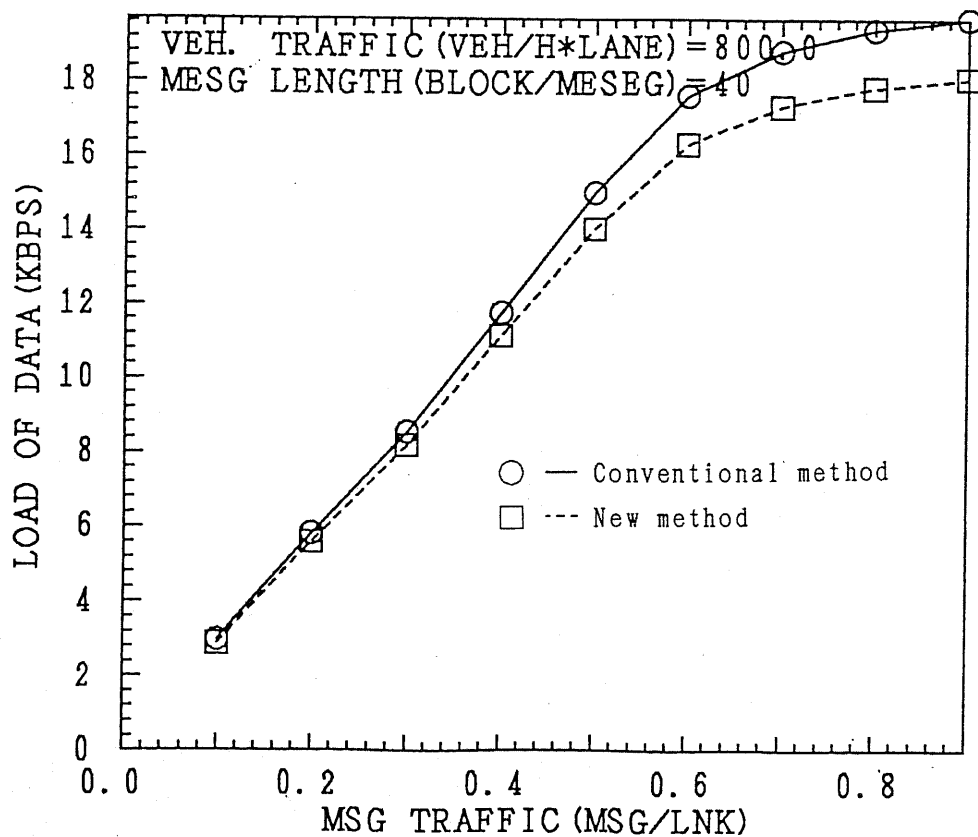


図 4-4-10 メッセージの発生率と平均データ転送量の関係
(メッセージ長 = 20 [kbit])

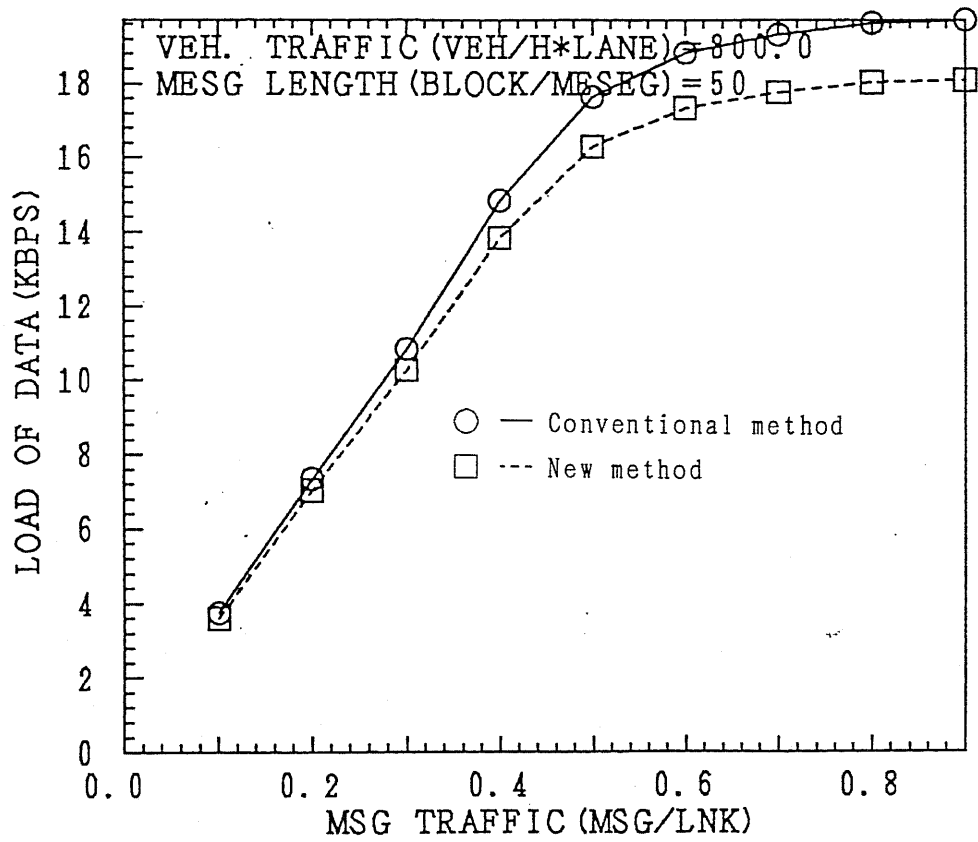


図 4 - 4 - 1 1 メッセージの発生率と平均データ転送量の関係
(メッセージ長 = 2.5 [kbit])

○ 制御信号伝送回数

次に先と同様な状況を想定して、A C S - R S S 間における制御信号転送に伴う負荷の指標として制御信号伝送回数を求めたものが図 4 - 4 - 1 2 (メッセージ長 = 5 [kbit]), 図 4 - 4 - 1 3 (メッセージ長 = 1 0 [kbit]), 図 4 - 4 - 1 4 (メッセージ長 = 1 5 [kbit]), 図 4 - 4 - 1 5 (メッセージ長 = 2 0 [kbit]), 及び, 図 4 - 4 - 1 6 (メッセージ長 = 2 5 [kbit]) である。データ転送量の飽和にともなって制御信号伝送回数も飽和している。新方式ではリンク旅行時間の予測によりデータを一括して転送するので A C S - R S S 間の制御信号伝送回数が通常の方式に比べて約 1 / 3 程度で済むことがわかる。これより本方式によりデータ転送制御の負荷が大幅に低減されることが考えられる。

○ 必要なメモリ

通常の方式では車両に伝送するデータを A C S より逐次的に転送するのに対して、本方式では必要なデータを一括して転送してくるために通常の方式よりもメ

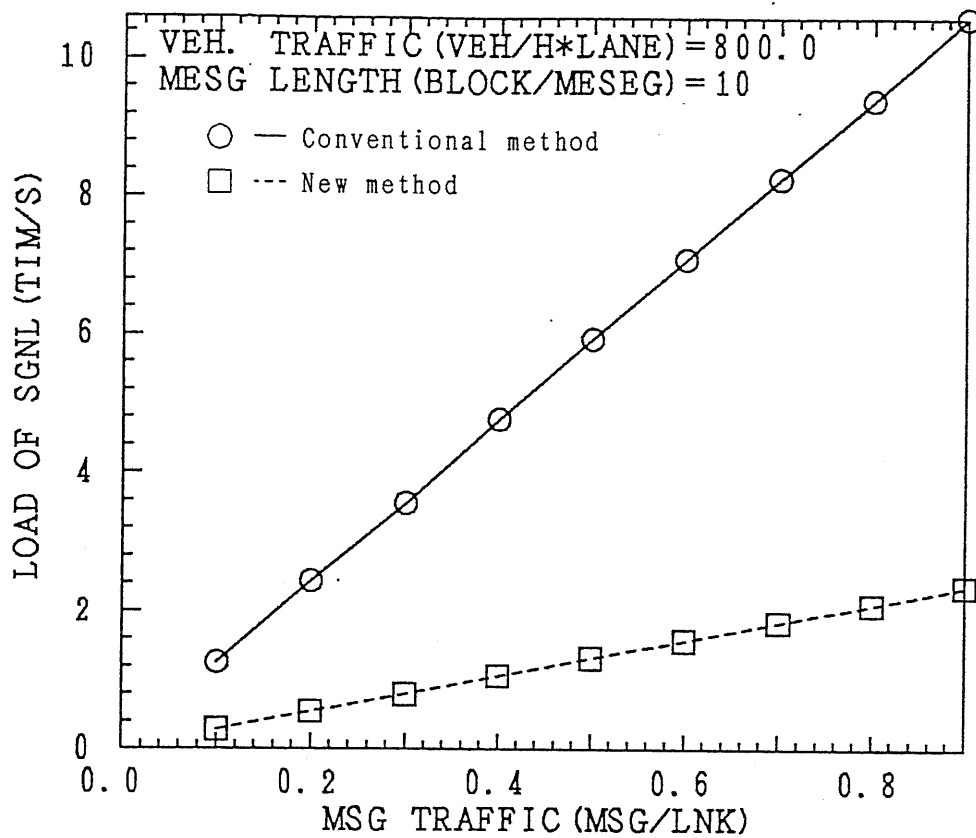


図 4-4-12 . メッセージの発生率と平均制御信号伝送回数の関係
(メッセージ長 = 5 [kbit])

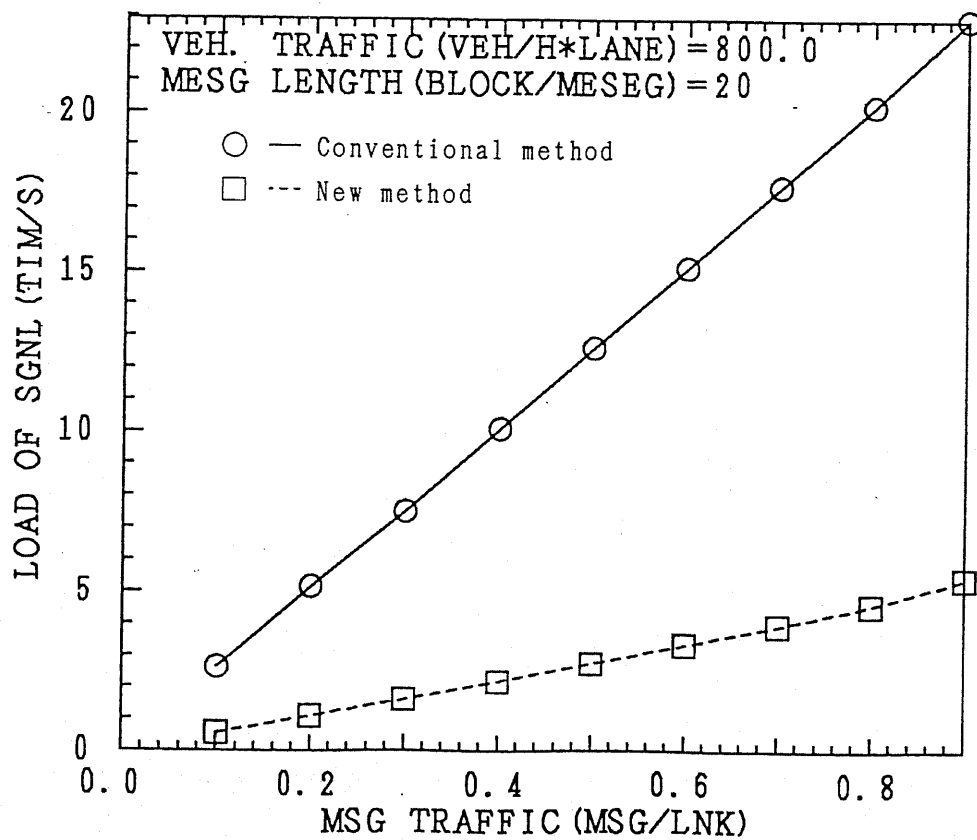


図 4-4-13 . メッセージの発生率と平均制御信号伝送回数の関係
(メッセージ長 = 10 [kbit])

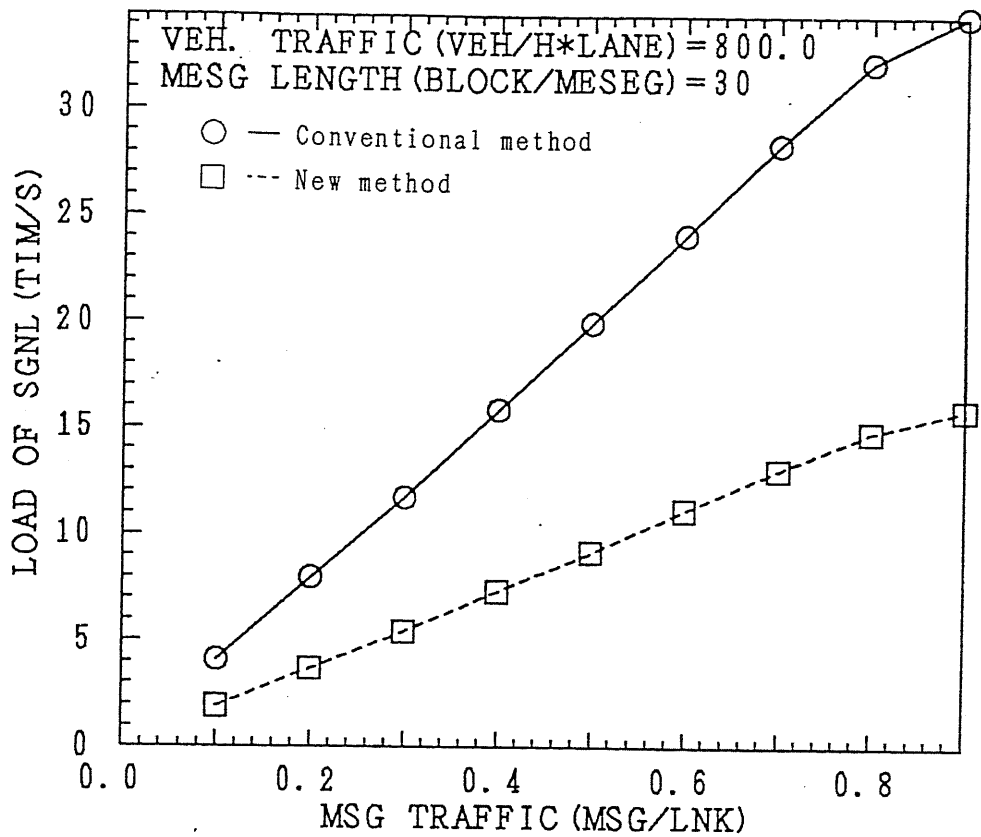


図 4-4-14 メッセージの発生率と平均制御信号伝送回数の関係
(メッセージ長 = 15 [kbit])

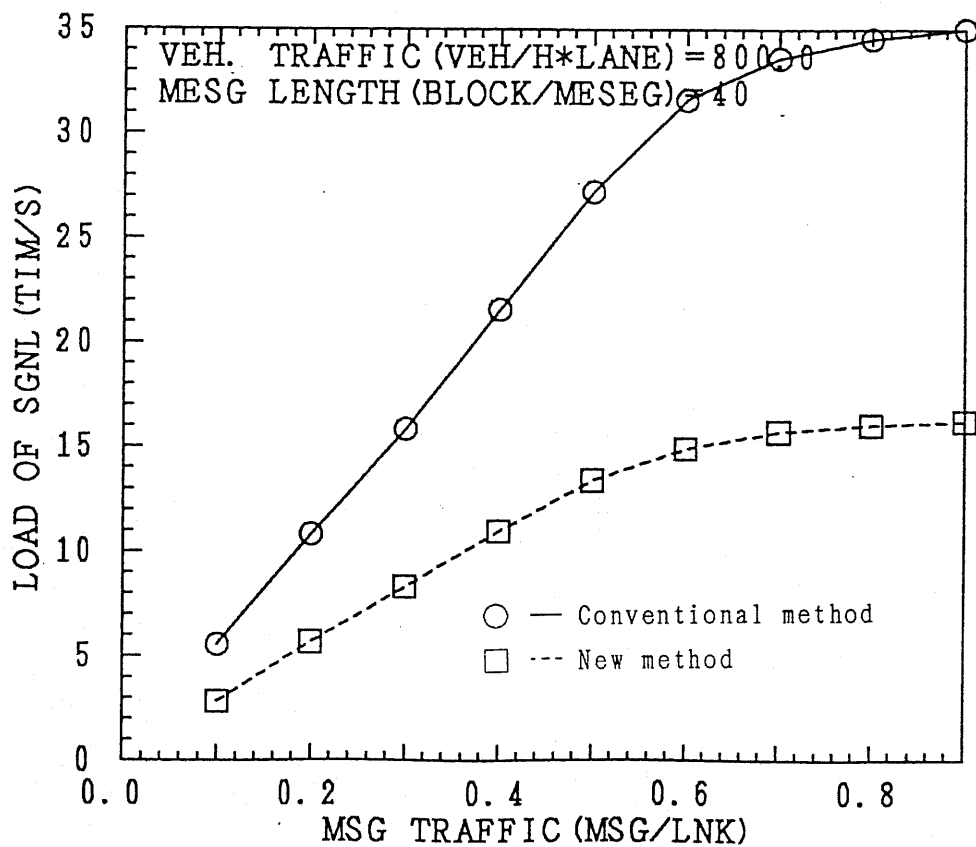


図 4-4-15 メッセージの発生率と平均制御信号伝送回数の関係
(メッセージ長 = 20 [kbit])

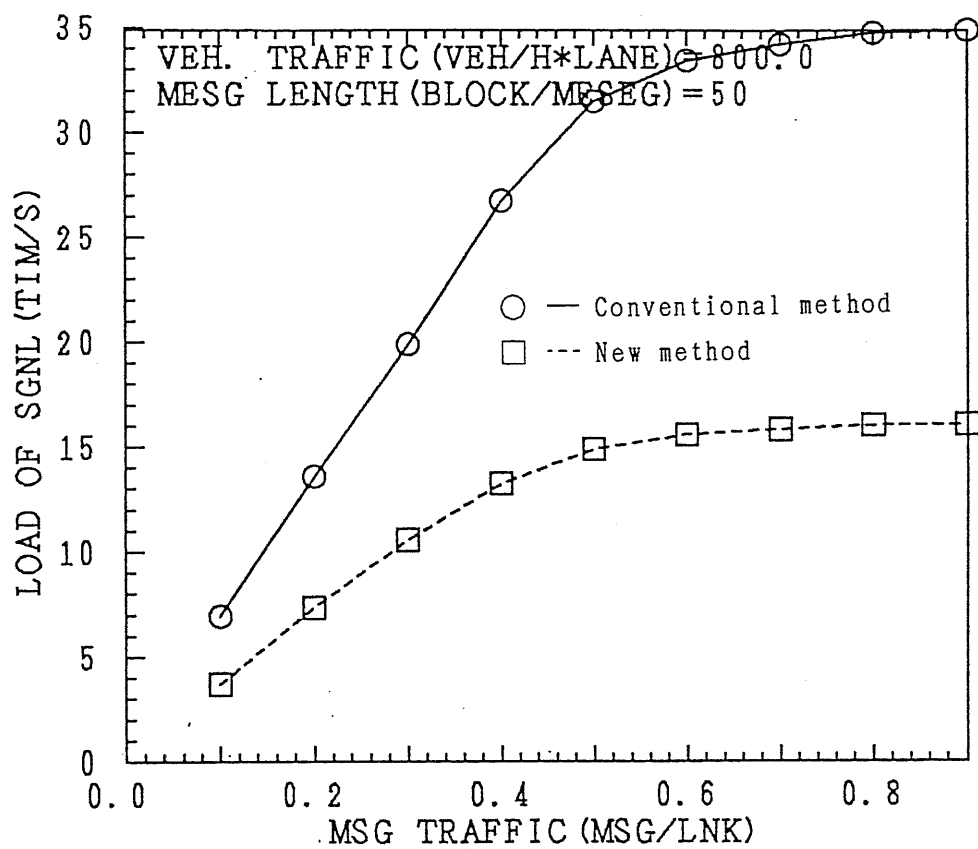


図 4 - 4 - 1 6 メッセージの発生率と平均制御信号伝送回数の関係
(メッセージ長 = 25 [kbit])

メモリコストが増大することが考えられる。ここでは先と同様な状況を想定して、各々の場合について、1つの道路リンクを対象としてその中に存在する自動車の下りデータ通信のために当該道路リンクに対するACSとRSSにおいて必要なデータバッファのためのメモリのバイト数を比較した。1200秒間のシミュレーションの間に必要とした最大メモリ数を求めたものが図4-4-17(メッセージ長=5[kbit]), 図4-4-18(メッセージ長=10[kbit]), 図4-4-19(メッセージ長=15[kbit]), 図4-4-20(メッセージ長=20[kbit]), 及び, 図4-4-21(メッセージ長=25[kbit])である。また, 使用したメモリの平均バイト数を求めたものが図4-4-22(メッセージ長=5[kbit]), 図4-4-23(メッセージ長=10[kbit]), 図4-4-24(メッセージ長=15[kbit]), 図4-4-25(メッセージ長=20[kbit]), 及び, 図4-4-26(メッセージ長=25[kbit])である。これより, 本方式ではデータ伝送の負荷が増大しても必要なメモリ量は通常の方式と比べて1つの道路リンク当り高々100k

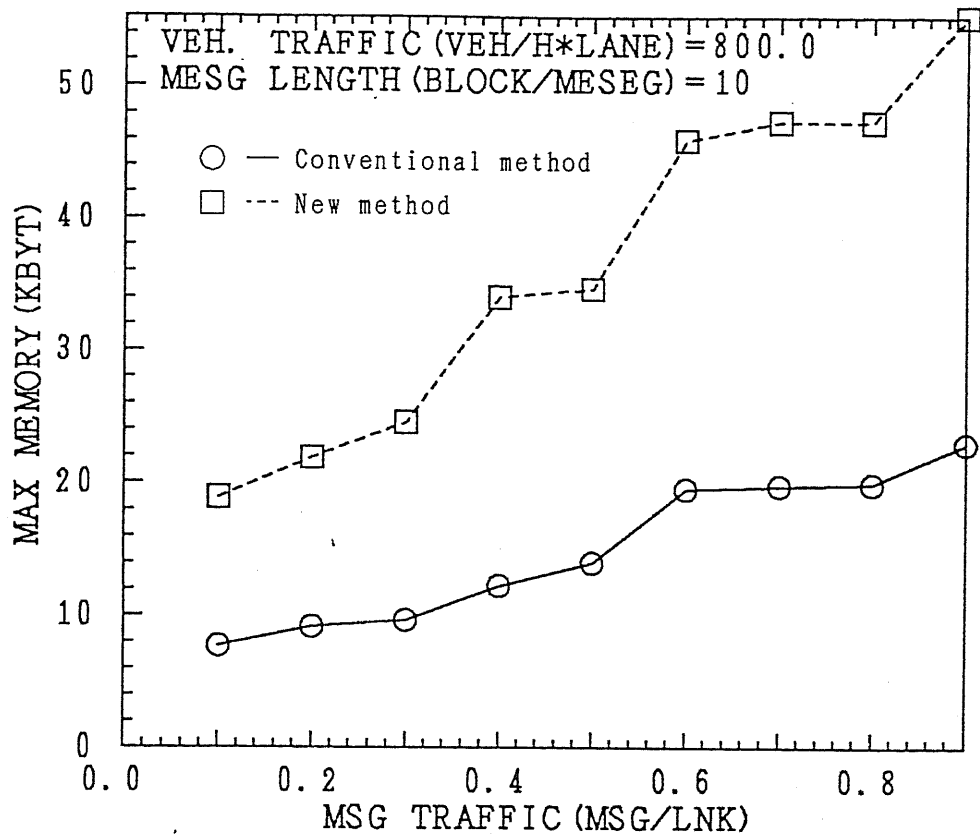


図 4-4-17 メッセージの発生率と最大使用メモリ量の関係
(メッセージ長 = 5 [kbit])

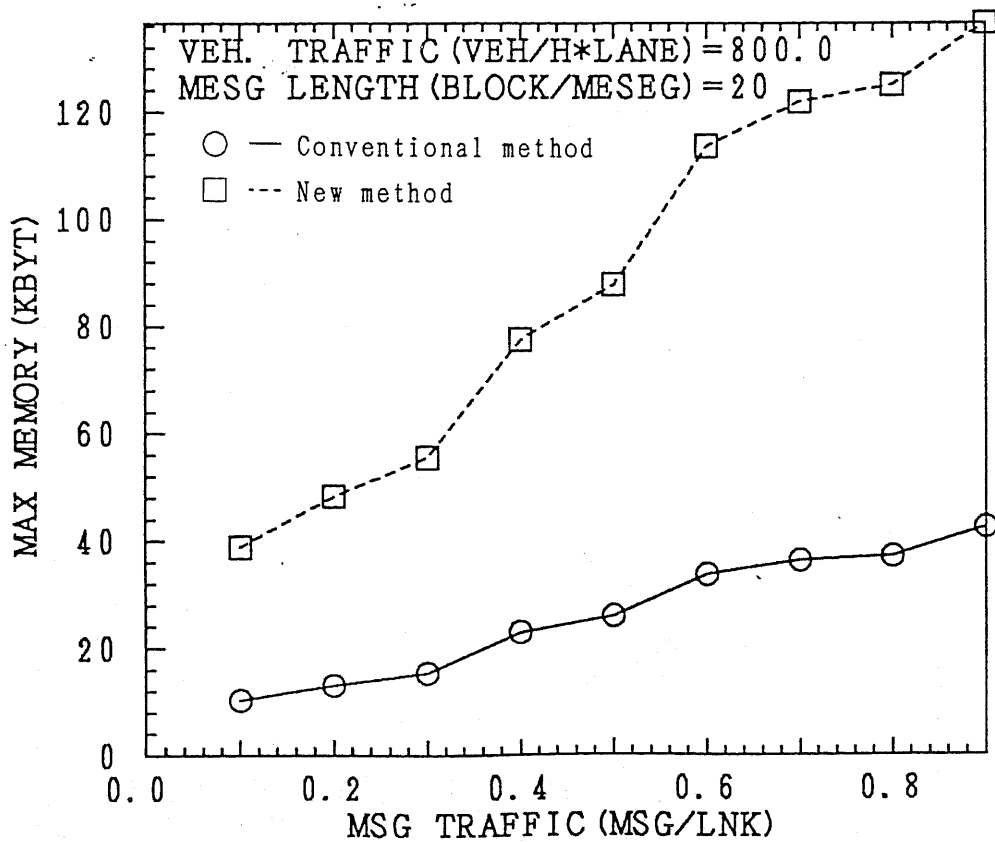


図 4-4-18 メッセージの発生率と最大使用メモリ量の関係
(メッセージ長 = 10 [kbit])

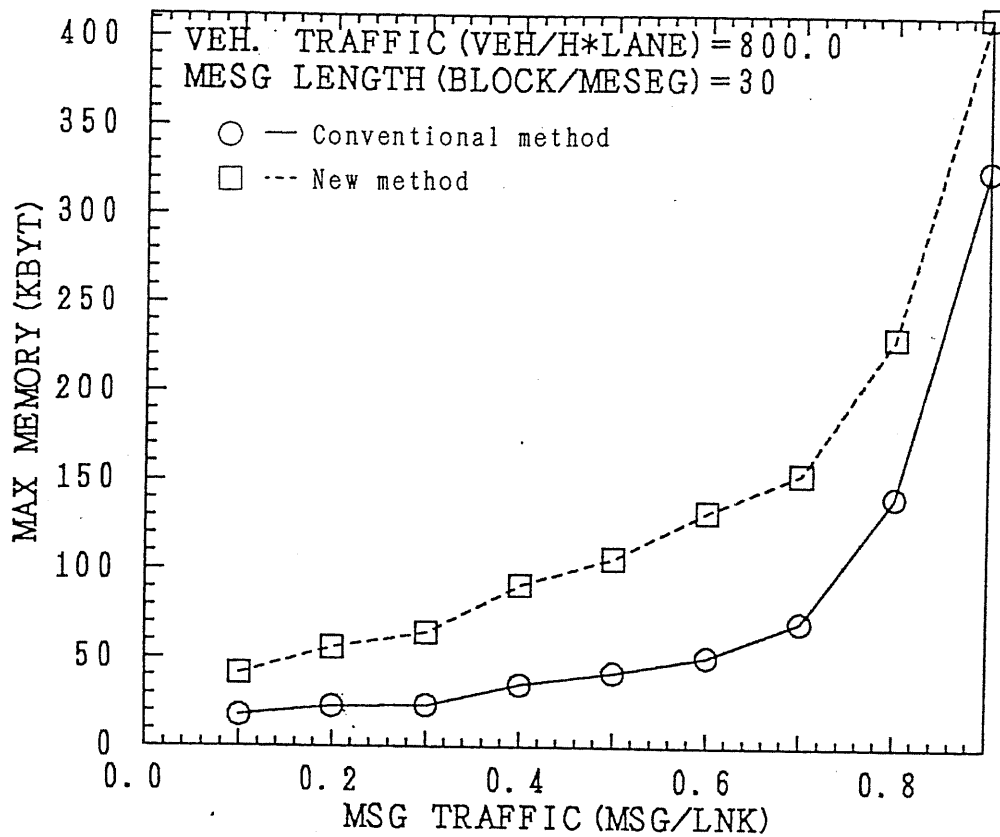


図 4-4-19 メッセージの発生率と最大使用メモリ量の関係
(メッセージ長 = 15 [kbit])

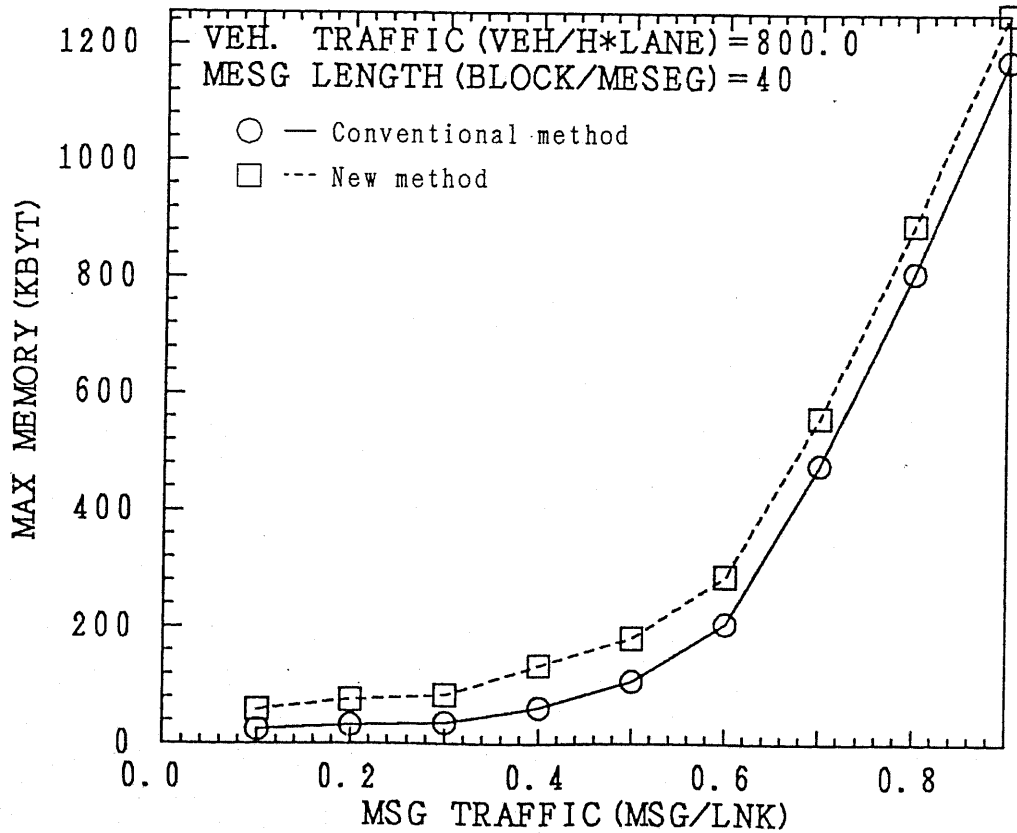


図 4-4-20 メッセージの発生率と最大使用メモリ量の関係
(メッセージ長 = 20 [kbit])

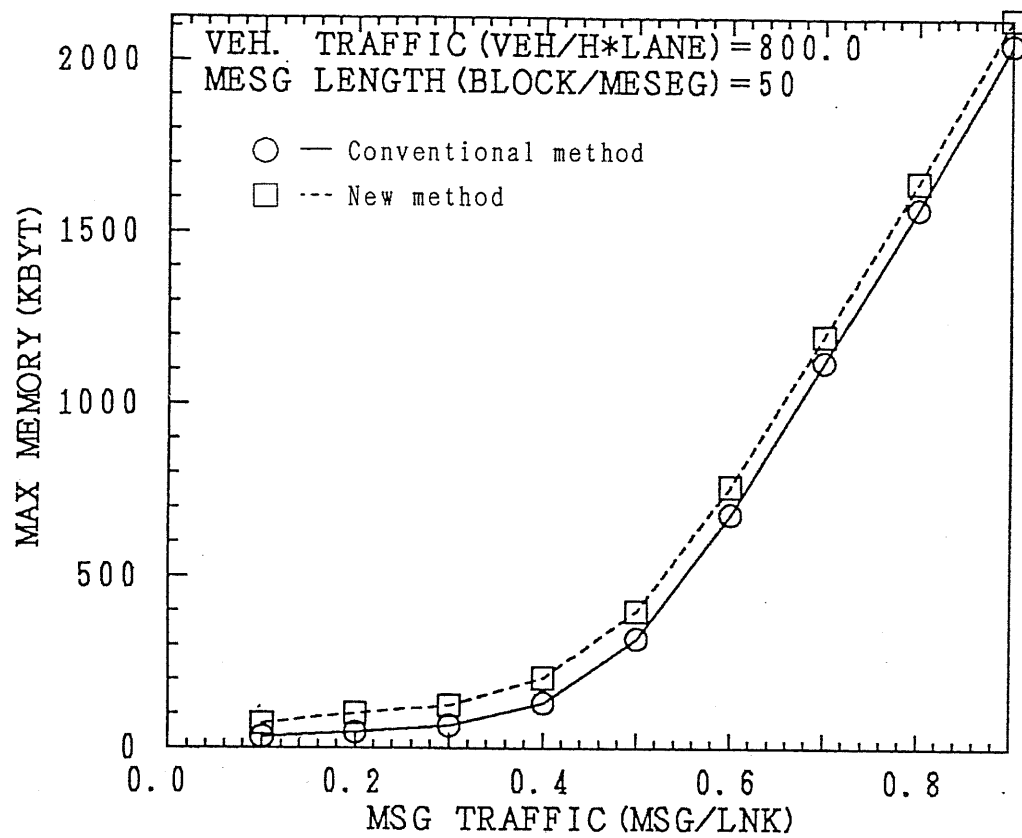


図 4 - 4 - 2 1 メッセージの発生率と最大使用メモリ量の関係
(メッセージ長 = 25 [kbit])

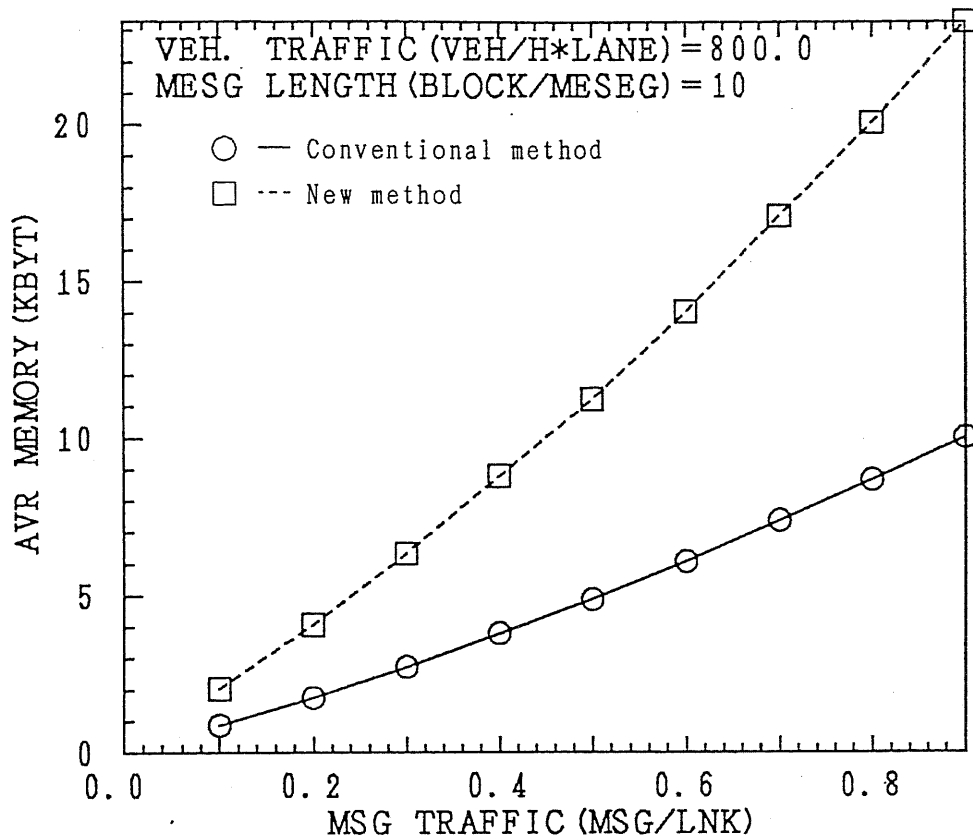


図 4 - 4 - 2 2 メッセージの発生率と平均使用メモリ量の関係
(メッセージ長 = 5 [kbit])

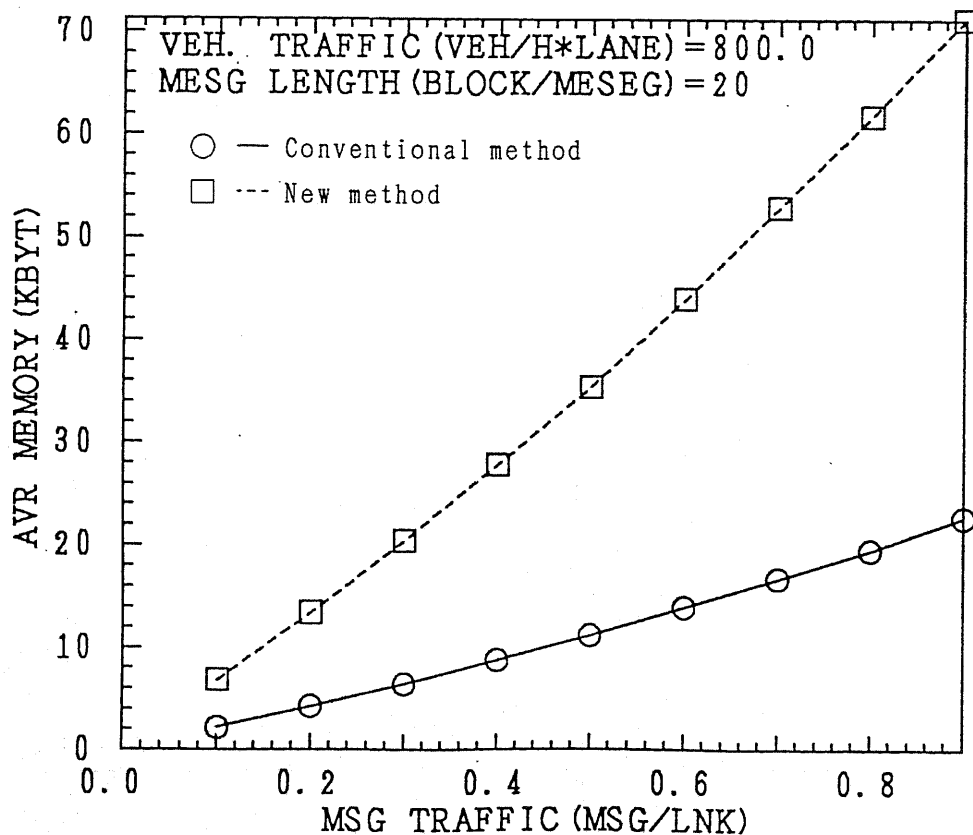


図 4 - 4 - 2 3 メッセージの発生率と平均使用メモリ量の関係
(メッセージ長 = 10 [kbit])

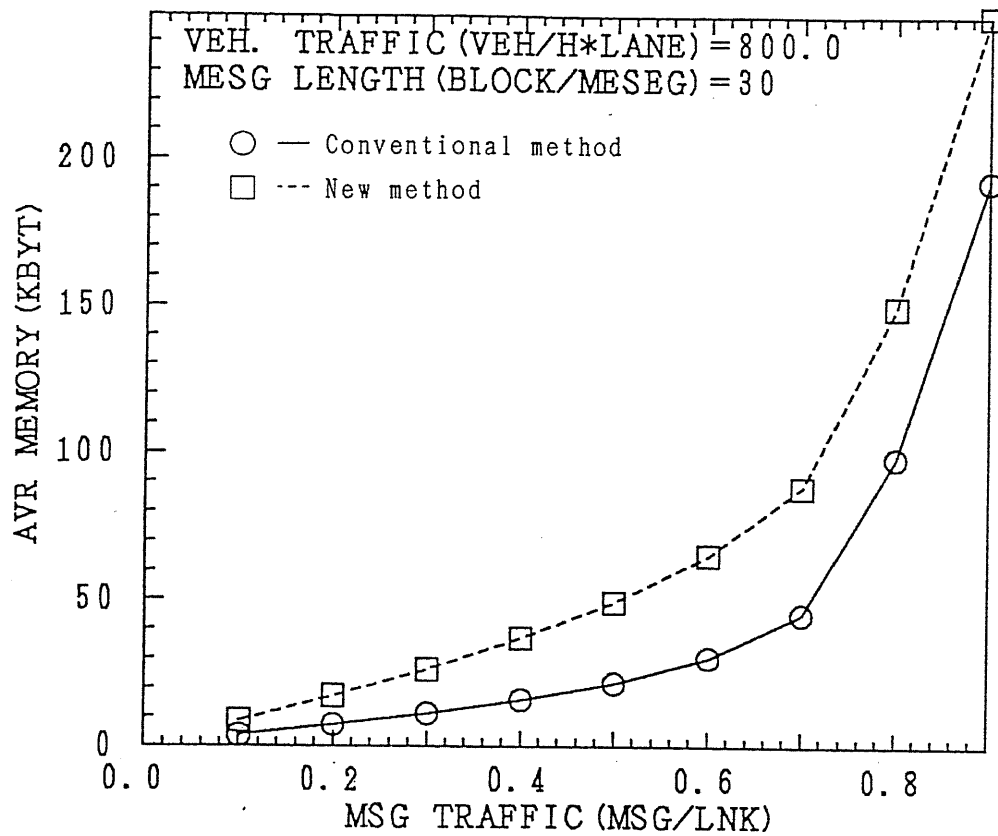


図 4 - 4 - 2 4 メッセージの発生率と平均使用メモリ量の関係
(メッセージ長 = 15 [kbit])

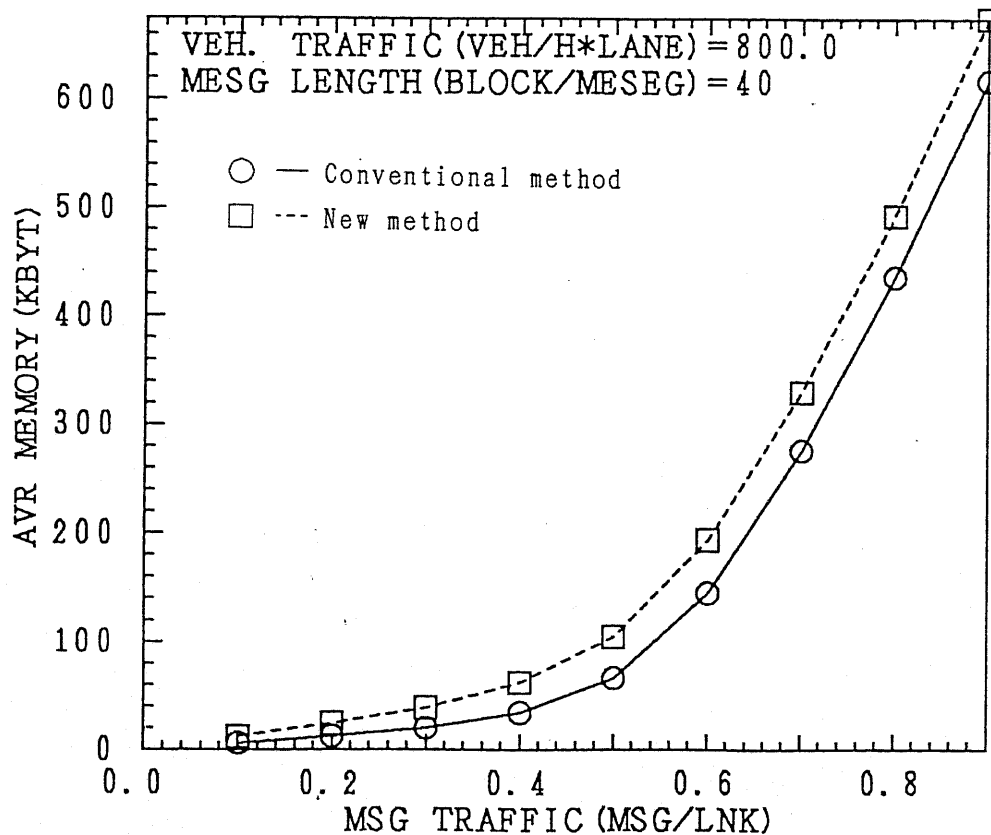


図 4 - 4 - 2 5 メッセージの発生率と平均使用メモリ量の関係
(メッセージ長 = 20 [kbit])

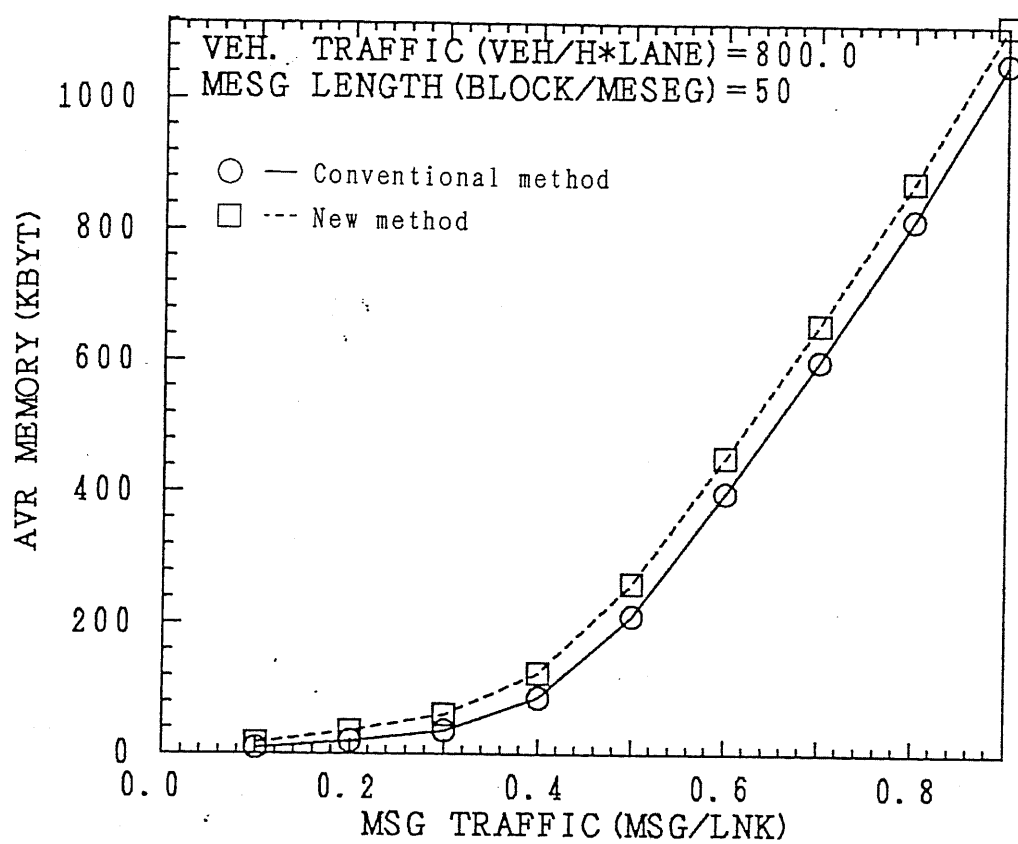


図 4-4-26 メッセージの発生率と平均使用メモリ量の関係
(メッセージ長 = 25 [kbit])

B 程度多くあればよいことがわかる。

これに対して通信コストという点からみれば図 4 - 4 - 1 2 ~ 図 4 - 4 - 1 6 の結果より制御信号の伝送回数にして毎秒 1 0 ~ 2 0 回程度少なくて済むことからデータ量にして繁忙時には少なくとも毎秒 5 0 ~ 1 0 0 オクテットの通信量が節約されることになる。以上より、本方式の導入により、わずかな初期投資で A C S - R S S 間の通信を大幅に効率化できると考えられる。

4. 5. 各装置で管理すべき情報内容

4. 5. 1. 走行誘導の情報

走行誘導のために各装置で管理すべき情報内容を図 4 - 5 - 1 に示す。RSS ではリンク内の目的方向別台数を集計し、ACS に報告する。ACS では各リンクからの情報をもとに、各々のリンクにおける車線の利用パターンを決定し、RSS に対してマクロに指示する。一方、MCC においては全体を管理しており、必要ならば ACS に対して介入を行う。

4. 5. 2. 汎用データ通信の情報

汎用データ通信のために各装置で管理すべき情報を図 4 - 5 - 2 に示す。ACS においては個々の車両の情報はデータ通信を行っている場合のみ動的に設けられる。MCC においては各自動車の存在エリアに関する情報は常時静的に設けられているが、データ通信を行う場合のシーケンス制御等の詳細な情報は動的に設けられる。

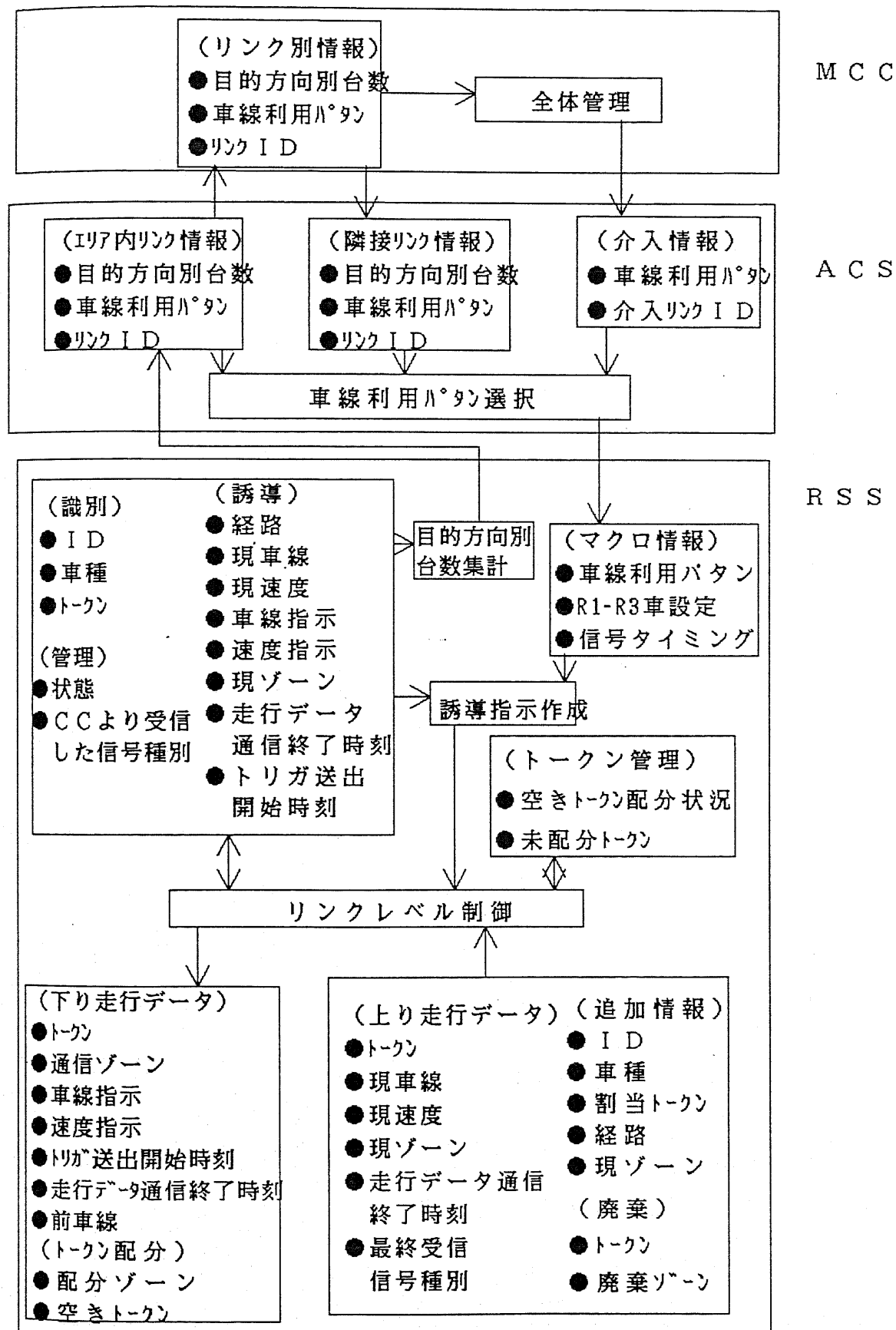


図4-5-1 走行誘導機能のために管理する情報内容

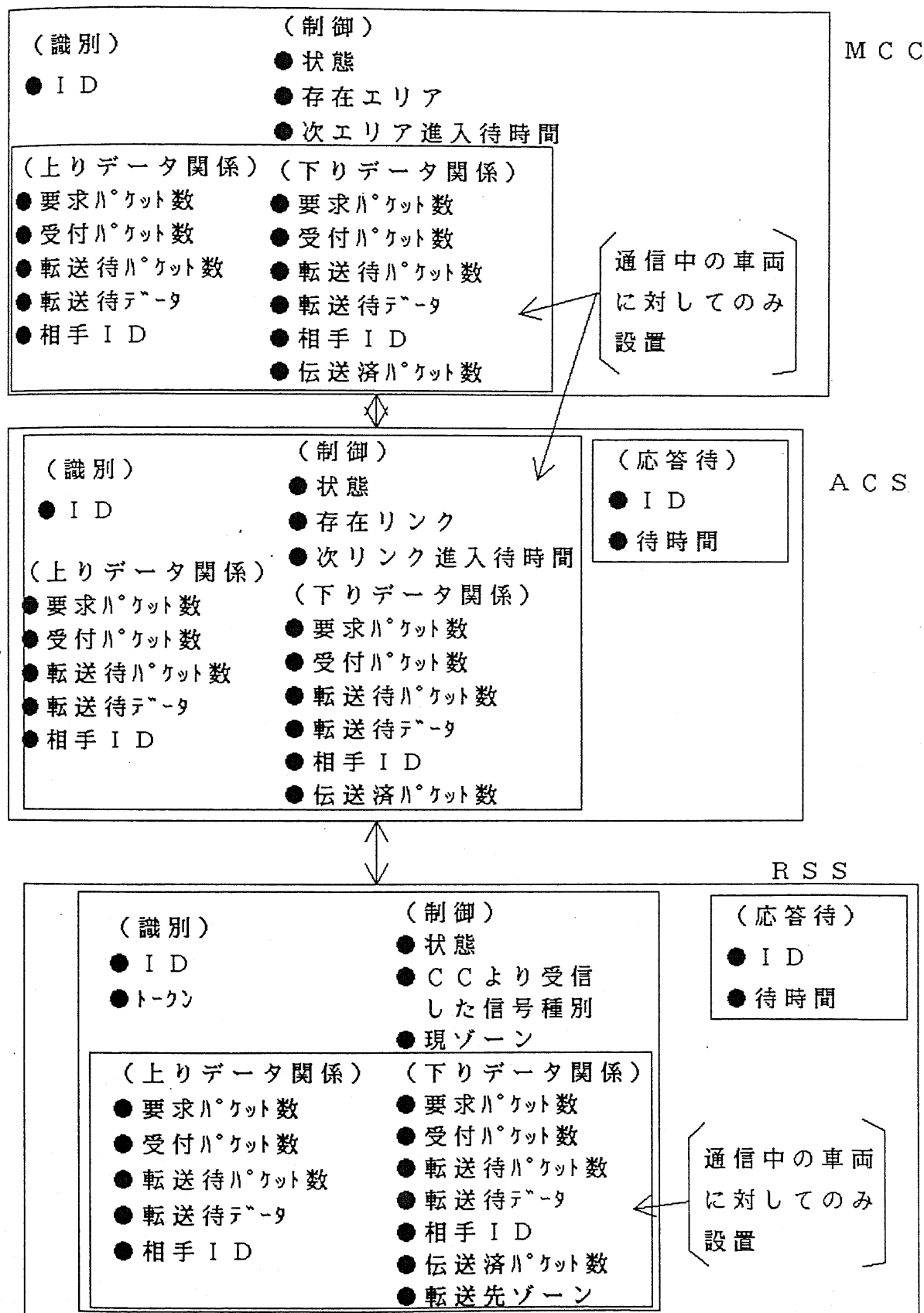


図4-5-2 汎用データ通信機能のために管理する情報内容

4. 6. 結 言

本章では、小ゾーン連続形自動車パケット通信システムにおいて、道路リンクより上位層の通信制御であるグローバルレベル制御について説明した。またその中において、車載機のインテリジェンスを積極的に利用した個別情報の自立分散型管理概念を導入し、さらに、リンク旅行時間予測を行うことと併せることにより効率的な制御が可能であることを示した。

なお、ここでは制御方式の評価手法としては簡単なマイクロモデルを用いたコンピュータシミュレーションを行うにとどめているが、今後は、後に述べるモデル実験システムに部分的にでも盛り込む等して、より現実的な条件を考慮した検討を行ってゆくことが必要であろうと考えられる。

5 . 走 行 誘 導 へ の 応 用

5. 1. 序 言

本章では、小ゾーン連続形自動車パケット通信システムの交通管理への応用について述べる。これまでも述べたように、本システムは、道路の全車線にまたがる長さ10m程度の通信ゾーンを相互の干渉が無視できる程度に接近させて連続的に配置するというこれまでにない形態を取っており、それにより道路上において個々の自動車を数十m程度の間隔できめ細かく捉えて、それらの走行を制御することが可能となる。

ここでは、この機能により実現される新たな交通管理の手法として走行誘導を提案し、まずその概念について説明する。そして、その具体例として高速道路分流点における動的車線規制と街路の信号交差点における右折車の待合わせ制御のアルゴリズムとを提案し、それらについてミクロモデルを用いた評価シミュレーションを行ってその有効性を示した。

5. 2. 走行誘導の概念

交通管理の高度化の手法としてこれまで考えられてきたものには、スポット状の通信ゾーンを設置して旅行時間等の情報を収集しそれをもとに経路誘導を行うこと[37][38][39][49]やさらに将来の構想として高速道路等における自動運転[51][52][53]といったものがある。経路誘導は道路ネットワークを交差点(ノード)とそれらを結ぶリンクとからなっていると考えると、各々のリンクへの動的な交通流配分を行うものである。一方、自動運転については前車への追従走行や道路の線形に合わせた操舵の自動化等が考えられているが、現状では追越しなどの高度な判断等も含めた完全な自動化を通常の走行速度下で行うことは車載機の処理能力等の面において困難が多い[59]。

これに対して本研究では、新たな交通管理の手法として走行誘導の概念を提案する。これは小ゾーン連続形自動車パケット通信システムにより、各道路リンク内における車両の位置や速度の情報を地上側に頻繁に収集し、地上側ではこの情報を基に車線選択や目標速度に関する指示を作成して各自動車に対して伝達する

ことにより、自動車側では運転者がこの指示を受けて運転操作を行ってゆくものである。これにより、道路リンク内における車両のミクロな誘導が可能になる。

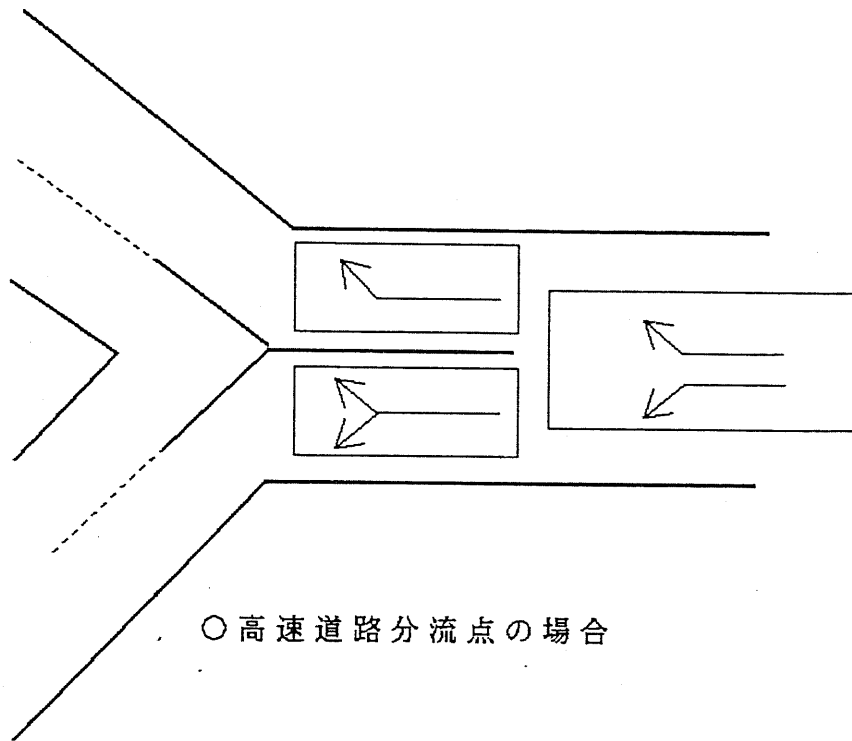
この応用として、交差点や分流点などのリンク出口においてのボトルネックを解消しその容量を増大することができる。以下にその基本的な考え方について説明する。

従来の固定的な車線利用方法では、図5-2-1に示したように道路上に描かれたペイント等により高速道路分流点や交差点等の手前で車両を目的方向別に横方向に整理し、層流化して捌いている。これでは次のような問題点がありボトルネックとなる場合がある。

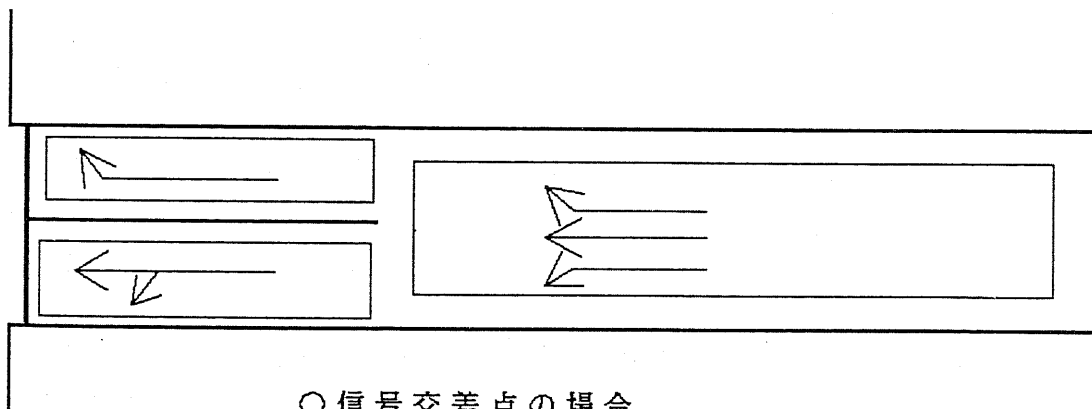
(A)車線の割当が固定的なため交通量の偏り等の状況の変動に対応できない。

(B)信号交差点出口では車線数が制限されるため信号の青時間を有効に活用できない。

これに対して小ゾーン連続形自動車パケット通信シ



○高速道路分流点の場合



○信号交差点の場合

図 5 - 2 - 1 従来の固定的な車線規制

システムを利用して走行誘導を行うことにより各道路のリンク内における車両の位置を計測し、それに基づいて車線選択や目標速度に関する指示を与え、車群を時間方向つまり縦方向に整理することを考える。これにより、道路リンク出口での車線規制を動的に変化させて、交通を捌くことにより交差点の容量の増大を図る。この具体例としてはここでは次の2つを提案する。

(A) 長期的動的車線利用

図5-2-2に示したように、目的方向別の交通量等の状況の変化に応じて比較的長期に亘り、高速道路分流点などの道路リンクでの車線規制を動的に切り換えることにより交通量の偏りに対応する。この切り換えの際に小ゾーン連続形自動車パケット通信システムにより交錯が生じないように車両の誘導を行う。

(B) 信号周期内動的車線利用

図5-2-3に示したように、1つの信号周期内で通過する車両を誘導し、車群を並べ変えることにより交差点出口での短周期の動的車線利用を行うことによ

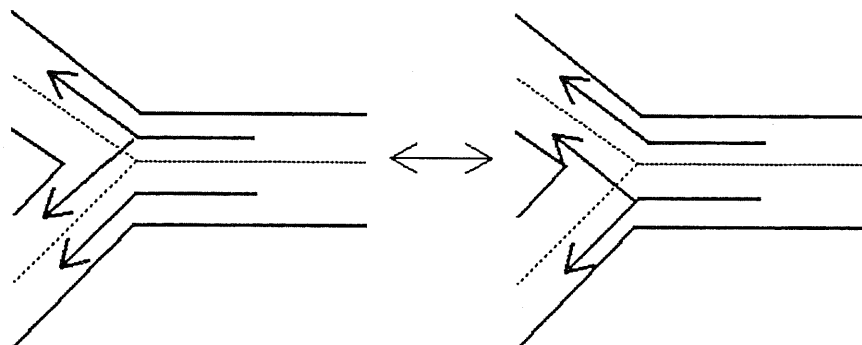


図 5 - 2 - 2 長期的動的車線利用

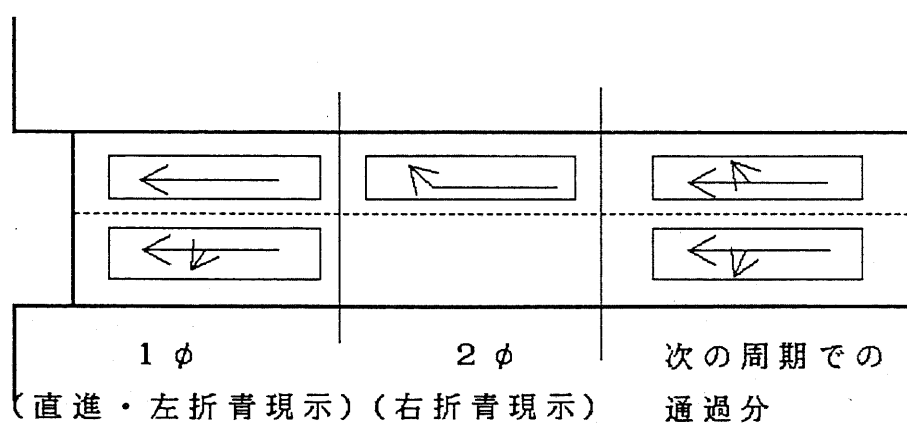


図 5 - 2 - 3 信号周期内動的車線利用

り、信号青時間内での車両の捌け量の増大を図ることが
できる。

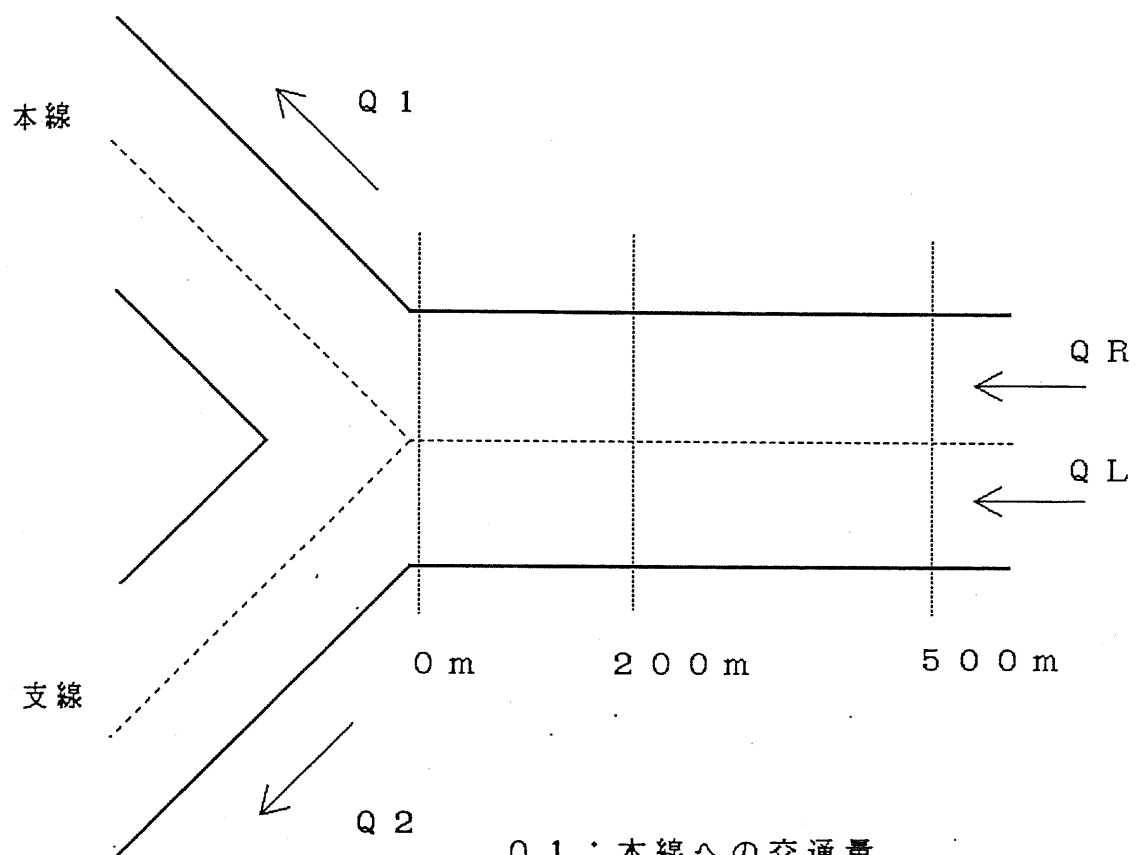
次節以降では、上記の概念をアルゴリズムとして具
体化し各々の効果について検討する。

5. 3. 高速道路分流出点での動的車線規制 [2][6][17] [18][19]

5. 3. 1. アルゴリズム概念

走行誘導アルゴリズムの最初の具体例として、高速道路の分流出点において目的方向別の交通需要の変動に応じて車線規制を動的に切り換えることにより交通量の偏りに対応することを考える。通常では、車線規制の切り換え時に分流出点における交錯が生じる危険があるが、ここでは走行誘導により個々の車両の走行をミクロに制御することにより安全に車線規制の切り換えを行う。

ここで扱った対象分流出点は図 5 - 3 - 1 に示すように、流入車線数が 2、流出車線数が本線、支線ともに 2 であるものを考える。この時、車線規制のパターンとしては図 5 - 3 - 2 に示す 2 通りのものが考えられる。ここでは、これらのモード 1 とモード 2 を切り換えながら安全な走行を行わせるための動的車線規制のアルゴリズムについて述べる。



Q_1 : 本線への交通量

Q_2 : 支線への交通量

Q_R : 右車線への流入交通量

Q_L : 左車線への流入交通量

Q : 総交通量

但し、

$$Q_R + Q_L = Q_1 + Q_2 = Q$$

が成り立つ。

図 5 - 3 - 1 対象分流点

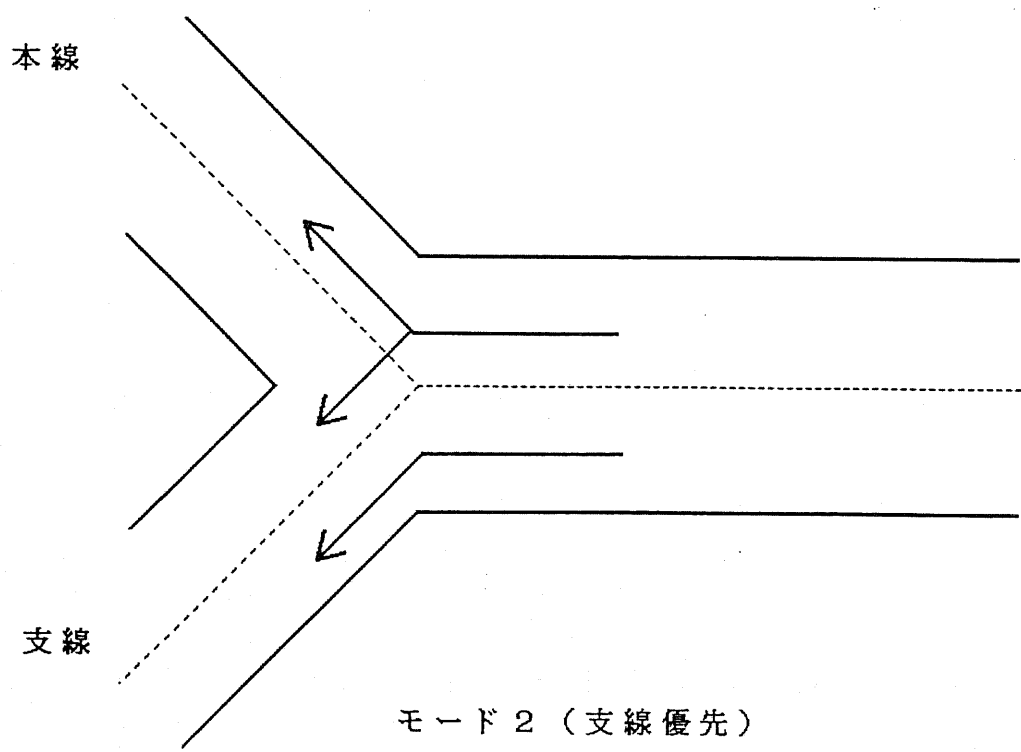
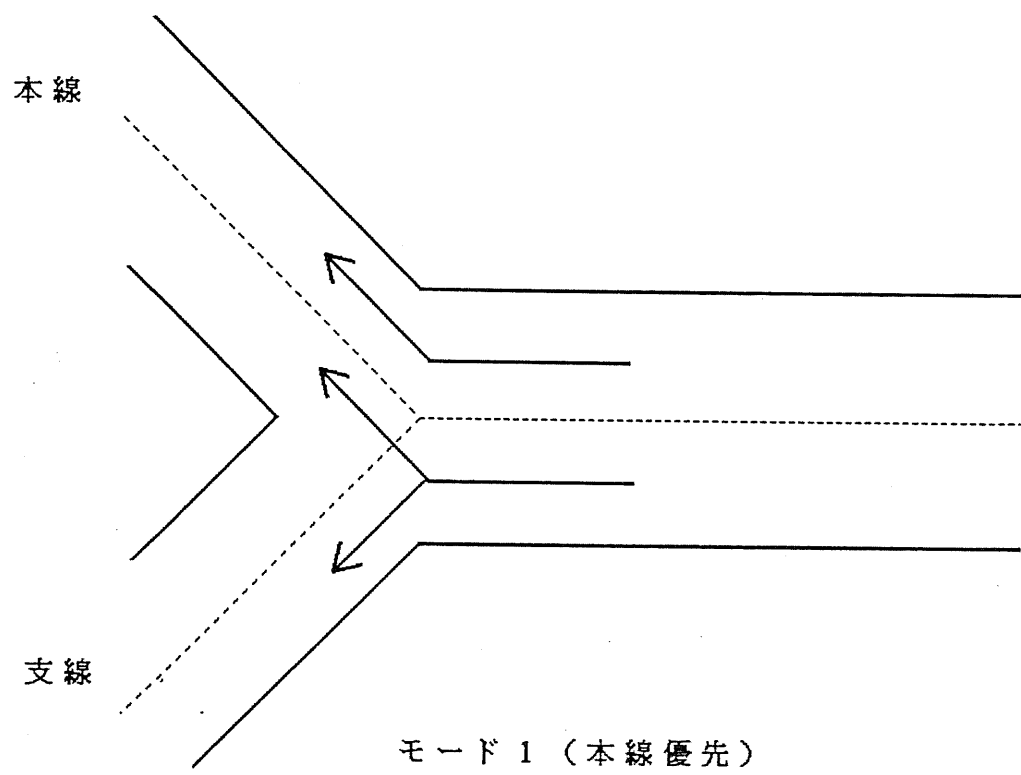


図 5 - 3 - 2 車線規制モード

5. 3. 2. 走行誘導指示作成方法

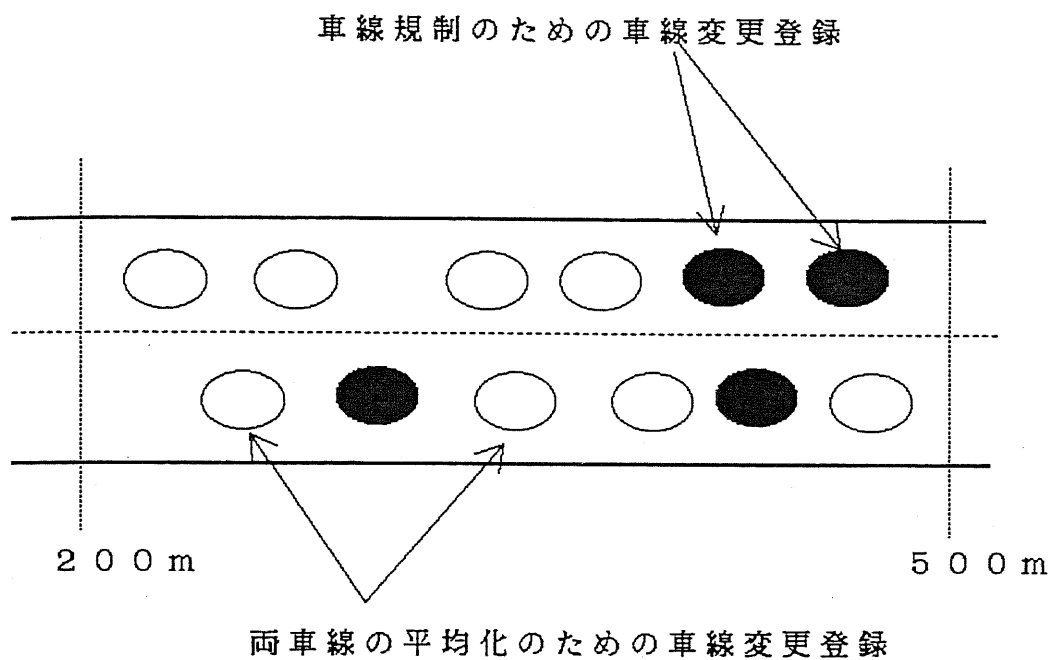
本アルゴリズムにおける具体的な誘導方法は以下のようになる。

(1) モードの決定

まず図5-3-1において、200m～500m区間に存在する車両の台数を目的方向別に計測する。 $Q_1 \geq Q_2$ ならばモード1、 $Q_1 < Q_2$ ならばモード2を選択する。ここで、以下の説明はモード1が選択された場合についてのものであるが、モード2が選択された場合には左と右、モード1とモード2、及び、本線と支線を各々逆に読み替える。

(2) 車線変更車の登録

車線規制がモード1となった時は、右車線から支線へ向かう車を左車線への車線変更指示を与える車として登録する。そして、それらの車が全て車線変更を行った後の左右車線の交通量を平均化するように、必要があれば左車線から本線へ向かう車の何台かを右車線への車線変更指示を与える車として登録する。この様子を図5-3-3に示す。



(この場合、 $Q1 = 8 > Q2 = 4$ であるために車線規制モードは1となる)

図5-3-3 モードの決定と車線変更車の登録

(3) 車 線 変 更 指 示 の 伝 達

これらの車が 2 0 0 m 地点に到達したら, 順次各車に車線変更指示を与えてゆき, 全ての登録車が 2 0 0 m 地点を通過したら (1) に戻る.

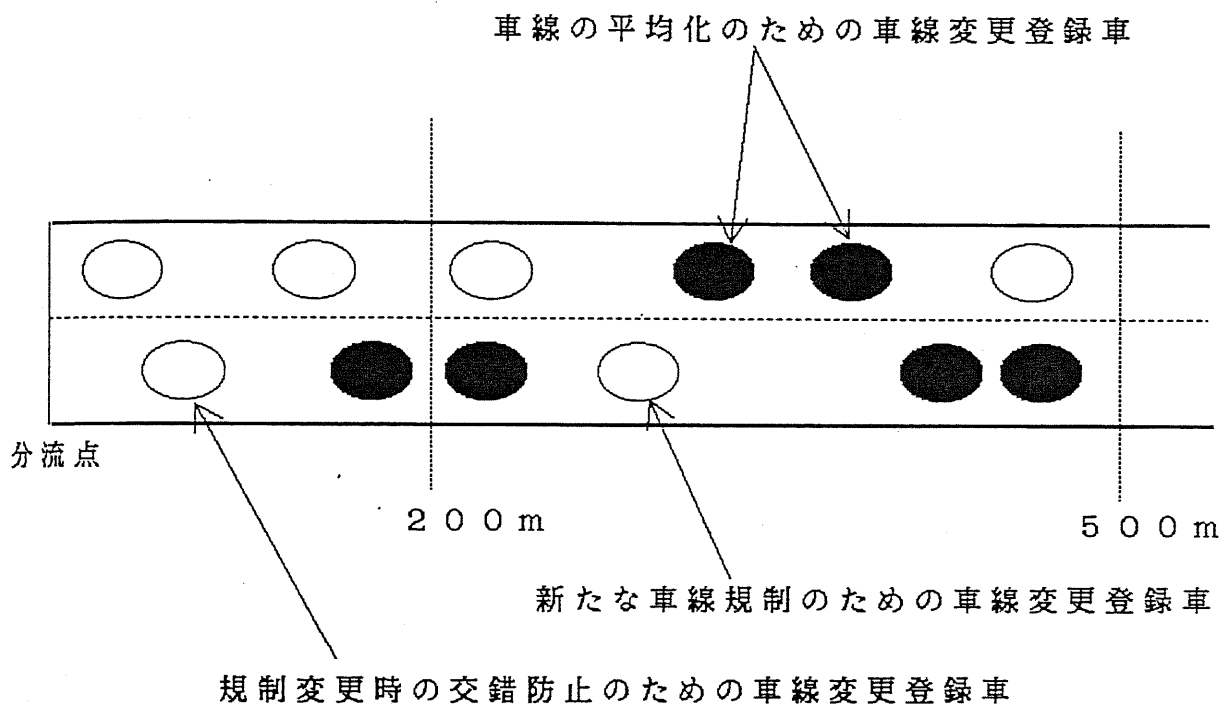
(4) モード切り換え時の指示

但し (2) において, モードが 1 であったのに対し, 新しくモード 2 に変化した場合には 0 ~ 2 0 0 m 区間に存在する左車線から本線に向かう車全てに右車線への車線変更指示を与えることにより, 分流点においての交錯を防止する. この様子を図 5 - 3 - 4 に示す.

これを行った後の車線変更車の登録で (2) と異なる点は, ここで車線変更の指示を与えられた車の台数分だけ補正を行い, 交通量の平均化のための車線変更車の登録台数を決定するところである.

5. 3. 3. シミュレーション評価

先と同様に, 文献 [26] のマイクロモデルを用いた評価シミュレーションを行って, このアルゴリズムの効果



○：本線へ向かう自動車

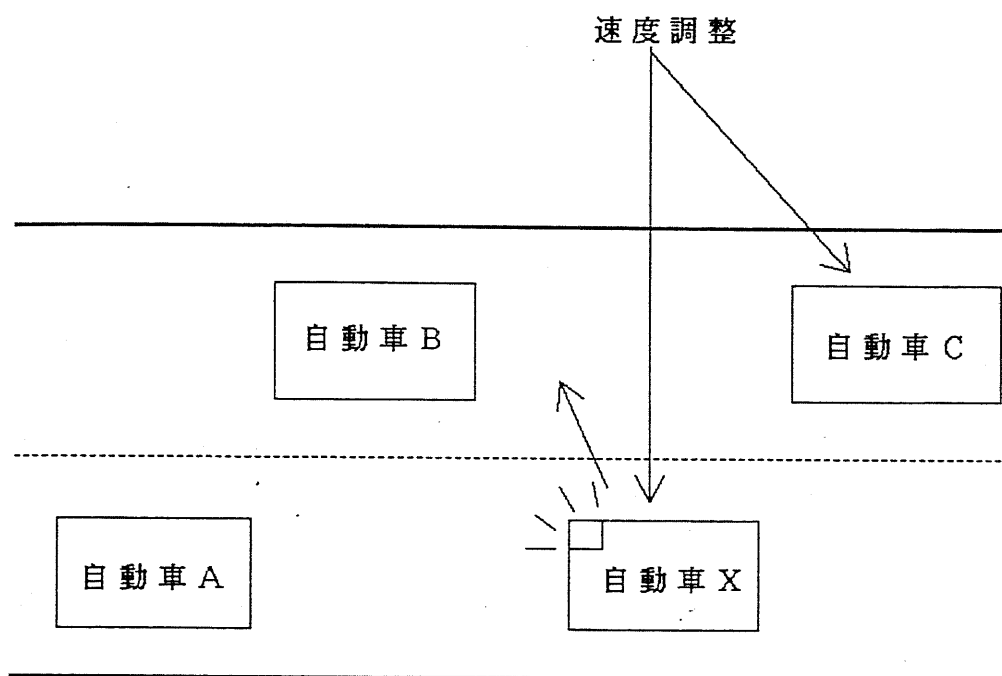
●：支線へ向かう自動車

$Q1 = 3 < Q2 = 5$ より、モード 1 からモード 2 に切り換わった場合

図 5 - 3 - 4 モード切り換え時の指示

を調べた。前述のように、このモデルでは道路を適当な区間毎に区切り、道路の状態は各区間に車が存在するか否かによって表す。車の走行のモデルとしては、先行車との車頭間隔から当該車の速度が求まるものとなっている。車頭間隔と速度の関係は、巨視的な特性がGreenbergのモデルに従うものとし、最大交通量が、 1800 台/h になるようにする。車両の移動は実時間の1秒毎に位置と速度を更新させる方法により行った。そして、位置の基本単位（前述の区間長）は5 m、速度単位は 4.5 km/h としている。

ここで、車線変更は次のようにモデル化した。図5-3-5で、X車が車線変更する場合を考える。これまでA車に追従走行してきたX車は、車線変更後B車にも追従走行できるように、最大加減速度の制限内でB車とX車の車間距離から求められる希望速度に最も近い速度まで減速する。X車の速度が希望値になったらC車がX車に追従走行できるように、最大加減速度の制限内でX車とC車の車間距離から求められる希望速度に最も近い速度まで減速する。C車の速度が希望値になったらX車は車線を変更する。



自動車 X が自動車 B と自動車 C の間に車線変更する様子

図 5 - 3 - 5 車線変更のモデル

また、流入交通流に関しては、車頭間隔時間の分布が指数分布をなすようにした。ただし、最小車頭時間を1秒としてある。発生させた車の初期速度は、前車との車頭間隔によって決まるものとした。ここで扱う区間は分流点の上流500mから分流点までとし、時刻0秒において車が全く存在しない状態から、600秒間シミュレーションを行った。流入交通量は、本線交通量率を含めて左右車線とも1200[台/h]としてある。

ここでは、時刻300秒から600秒の間に、流入した車のうち、時刻600秒までに分流点を通じた車の、500m地点から分流点通過までの平均旅行時間をOD-TIMEとして目的方向別に求めたものが図5-3-6と図5-3-7である。図中、本アルゴリズムにもとづいた誘導を行った場合をDynamic Lane Allocationとし、通常通り車線規制モードを1に固定した場合をFixed Lane Allocationとした。これらより、通常の場合は全体の交通量が一定でも支線に交通量が偏ると本線支線のどちらへ向かう車の平均旅行時間も大幅に増加するが、ここで提案したアルゴリズムによ

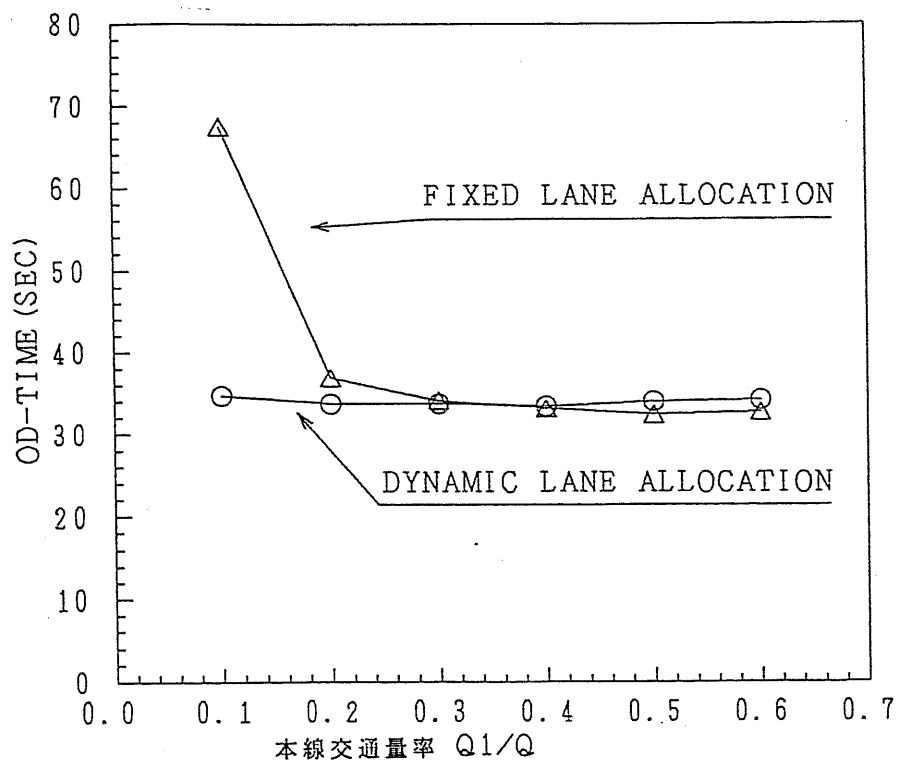


図 5 - 3 - 6 分流制御のシミュレーション結果（本線交通流）

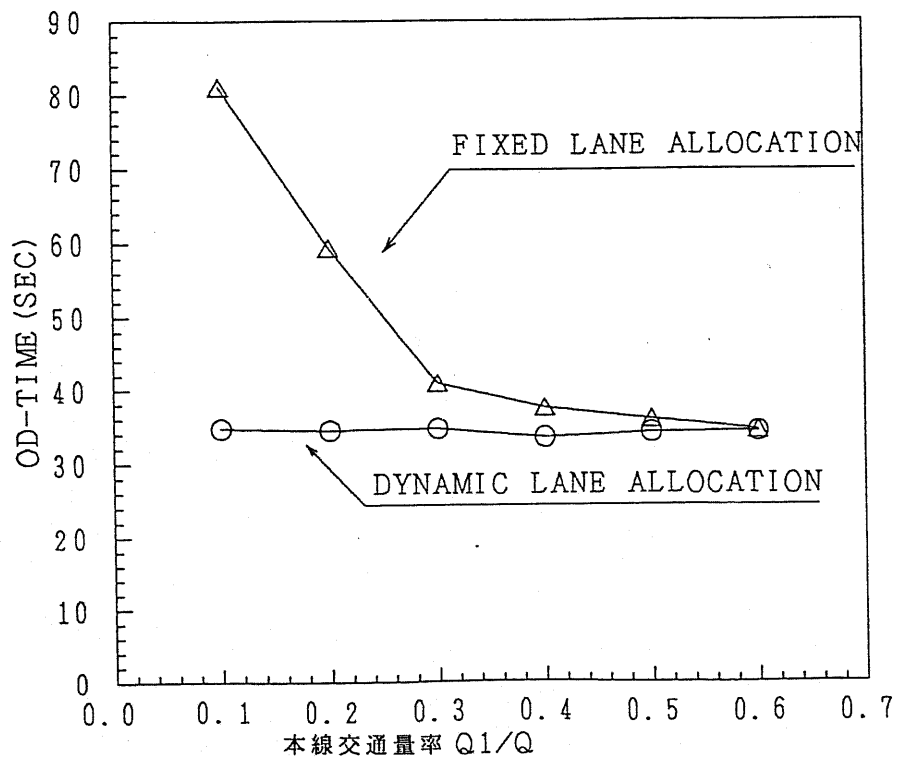


図 5 - 3 - 7 分流制御のシミュレーション結果（支線交通流）

る誘導を行った場合には全体交通量が一定ならば交通量の偏りがあってもほぼ一定の平均旅行時間を実現していることがわかる。

また、これらの計算実行時において、左車線から本線へ向かう車や右車線から支線へ向かう車が分流点に到達した時点で、これらと交錯する走行軌跡をとる車が分流点上流85mから分流点の間に存在するかどうかを評価シミュレーションを行った全ての場合についてチェックした。この結果、これに該当する車両は、どのケースでも1台も存在せず、ここで示した誘導アルゴリズムにより車線規制の安全な切り換えが行われることが示された。

5. 4. 街路の信号交差点での右折車の待合せ制御[2][6][13][17][19]

5. 4. 1. アルゴリズム概念

ここでは走行誘導のもう一つの具体例として右折車の待合せ制御のアルゴリズムを提案する。これは、街路の信号交差点において右折車を速度制御して交差点の後方に待たせておき右折車線を直進車にも利用させることにより交差点の容量を増大させるものである。

ここでは、通常図5-4-1のような車線規制が行われる2車線街路の信号交差点を対象とした。基本的な考え方としては図5-4-2に示したように、交差点手前において車両を誘導して直進・左折車と右折車の並べ替えを行うとともに、各右折車群が希望の時刻に交差点に到着するように各々の先頭車に対して速度制御を行う。そして図5-4-3に示したように信号現示が青の時は直進・左折車を2車線で捌き右折車が交差点に到着したら信号現示を右折青にして右折車を捌く。これにより交差点での右折車によるボトルネックの改善を図る。

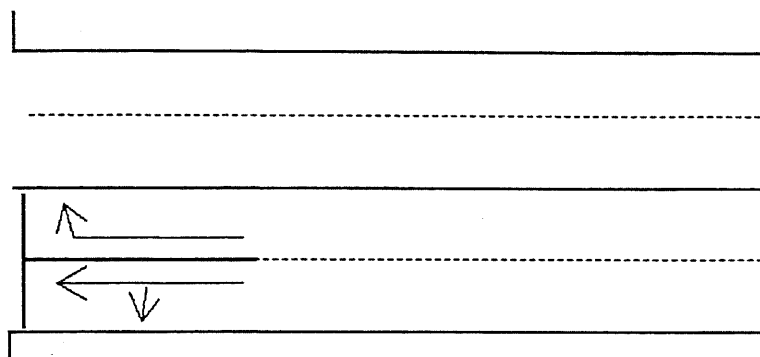
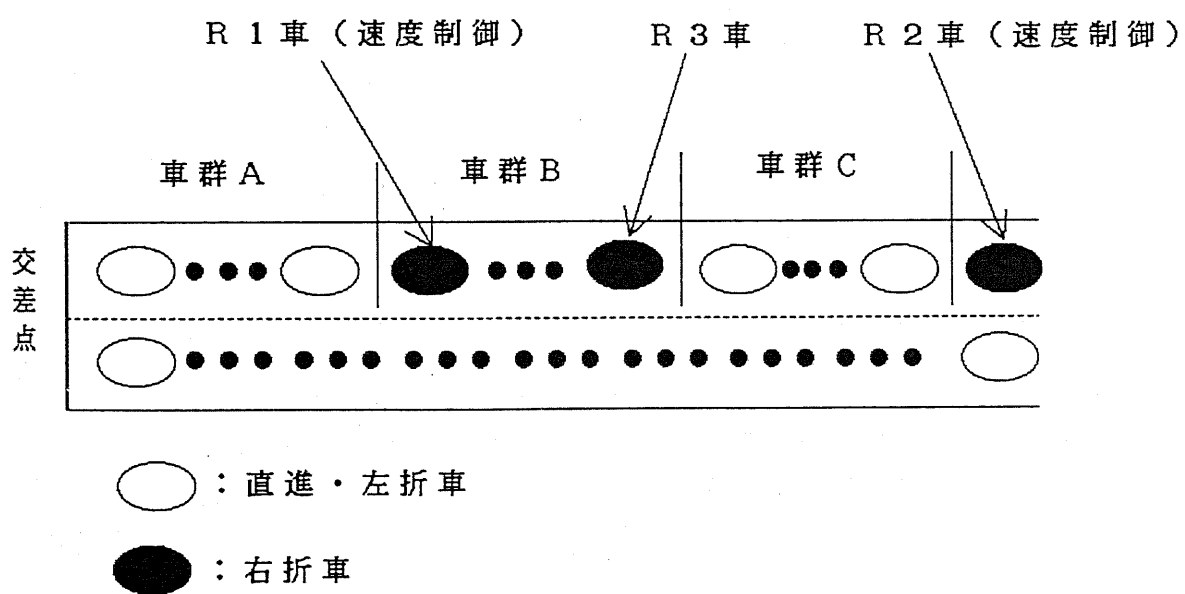


図 5 - 4 - 1 対象交差点



車群 A : 次の直進・左折青現示で通過する直進車

車群 B : 次の右折青現示で通過する右折車

車群 C : 次の次の直進・左折青現示で通過する直進車

図 5 - 4 - 2 車両の並べ変えと制御の概要

5. 4. 2. 誘導指示作成方法

ここでは、このための具体的な誘導方法について詳述する。

○車両の誘導方法

(1) 右・左折車への車線指示

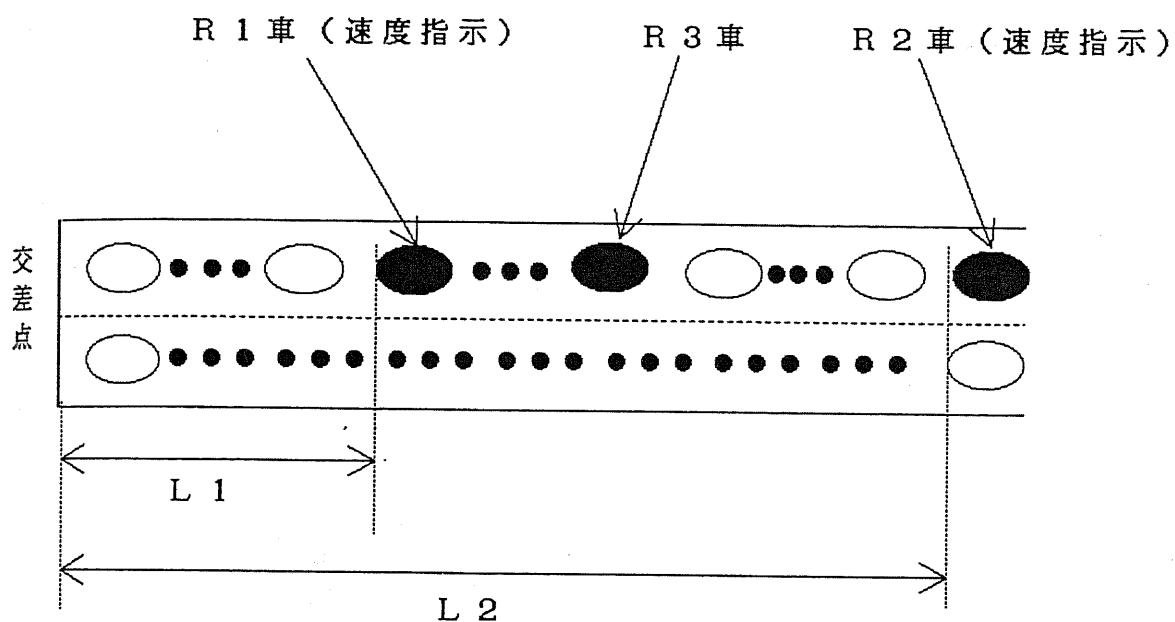
右折車に右車線を、左折車に左車線を各々指示する。

(2) 速度指示

信号現示が右折青以外のときに図5-4-2のように、次の右折青で交差点を通過させる右折先頭車をR1車、右折最後尾車をR3車、次の次の右折青で交差点を通過させる右折先頭車をR2車とする。なおこのR2車の決定方法は後で述べる。この時に、図5-4-4のようにR1車とR2車に対して等速で走行するとそれぞれの右折青の開始時に交差点に到着できるような速度を指示する。

(3) 直進車への車線指示

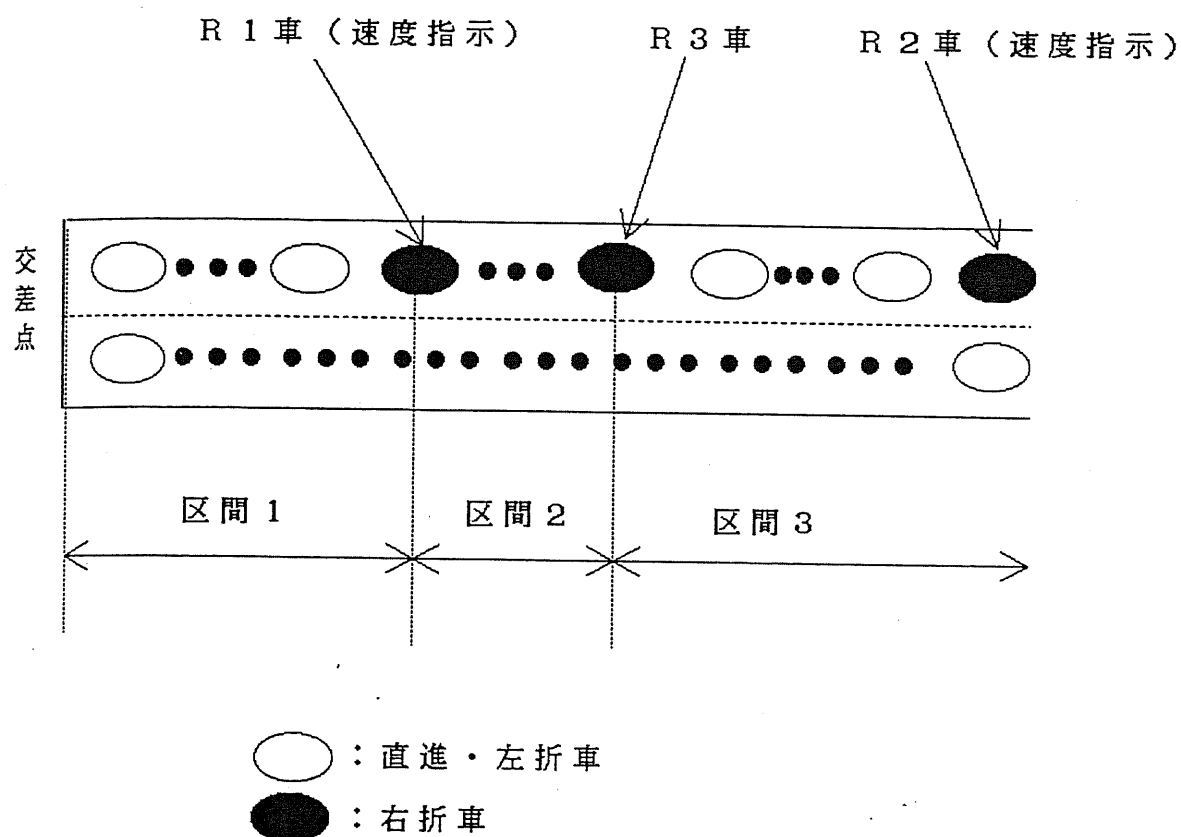
直進車に対しては、図5-4-5に示したように車



○ : 直進・左折車
 ● : 右折車

T 1 : 次の右折青現示開始までの時間
 T 2 : 次の次の右折青現示開始までの時間
 とすると
 R 1 車への指示速度 : $L 1 / T 1$
 R 2 車への指示速度 : $L 2 / T 2$
 となる。

図 5 - 4 - 4 速度指示の与え方



区間 1 : 両車線が同じ台数になるようにする

区間 2 : 右折車以外は左車線へ

区間 3 : 任意の車線選択を許可

図 5 - 4 - 5 直進車への車線指示

線指示を与える。まず、交差点直前からR1車の間の
区間1で両車線の直進・左折車の台数が等しくなるよ
うに、R1車を左車線から追い越した車に車線変更指
示を与える。次に、R1車からR3車の間の区間2の
車には、左車線を指示する。R3車より後ろの区間3
の車には、任意の車線選択を許可する。

○信号の制御

本アルゴリズムでは車両に対する誘導と連携して信
号の制御を行う必要がある。次にその具体的方法につ
いて述べる。まず、信号現示は直進・左折青→右折青
→赤→直進・左折青と変化させる。このとき各現示の
切り換えは以下のような手順に従うものとする。

(1) 直進・左折青現示

まず、一定時間(TG)直進・左折青現示を行う。

(2) 右折青現示

右車線の先頭車がR1車になれば信号現示を右折青
にする。

(3) 赤 現 示

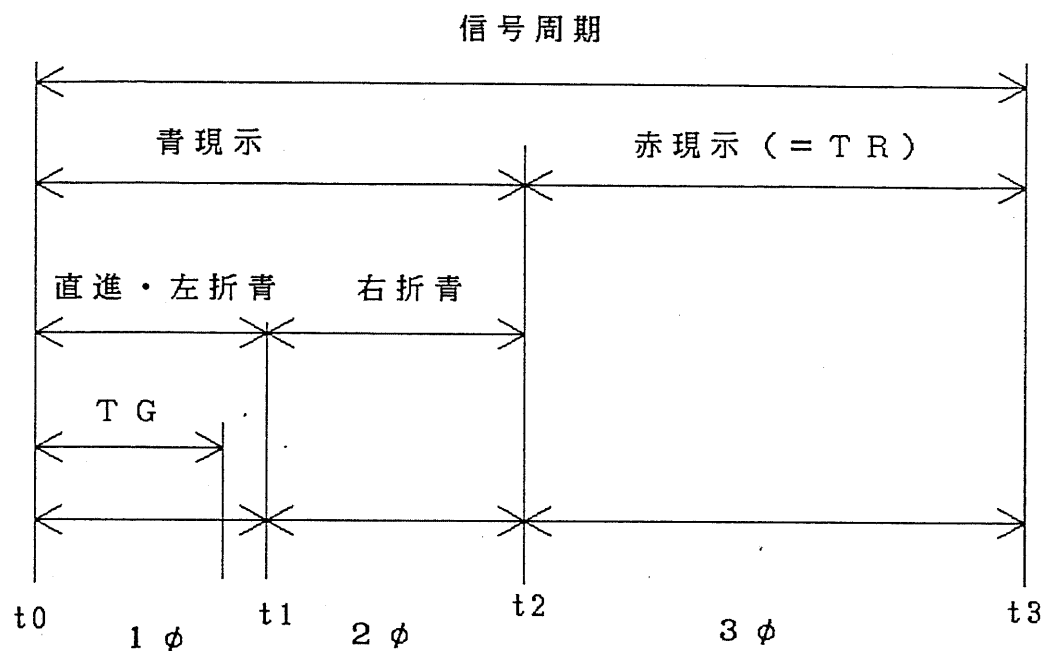
R 1 車 から R 3 車 ま で が 交 差 点 を 通 過 し た ら 信 号 現示を赤にする。そして、スプリットから決まる時間 ($= T R$) だけ経過したら信号現示を直進・左折青にする。

以上の様子を図 5 - 4 - 6 に説明する。

○ 新 R 2 車 の 決 定 方 法

信号現示を赤にした時点でこれまで R 2 車であった車両を R 1 車として新たに R 2 車と R 3 車を選定する。ここでは図 5 - 4 - 7 によりその方法を説明する。

まず、現在の信号周期内で交差点を通過できる車両の範囲として区間長 L の適当な初期値を仮定する。このとき R 1 車より後ろの直進車は左側の車線を通して R 1 車を追い越さなければならない。そこで、まず図の区間に存在する直進・左折車の台数 X を計測し、これらが一車線に平均に存在したときの車頭間隔 L / X を求める。そして 5. 3. で説明したマイクロモデルで



T G : 定められた直進・左折青時間

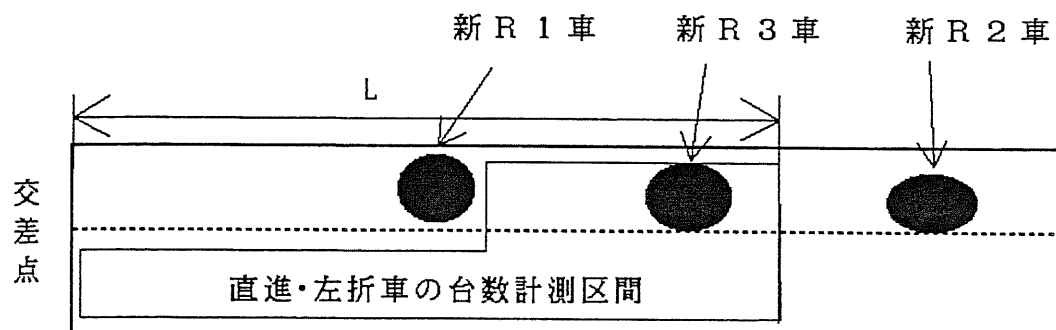
t_0 : 直進・左折青現示開始時刻

t_1 : 右折青現示開始時刻 ($t_0 + T G$ 以降 R 1 車が先頭車となった時刻)

t_2 : 赤現示開始時刻 (R 3 車が交差点を通過した時刻)

t_3 : 赤現示終了時刻 (スプリットにより決定)

図 5 - 4 - 6 信号の制御方法



T: 次の直進・左折青終了までの時間

X: 計測台数 L: 区間長

$L/T = F(L/X)$ となるような L を求める。

図 5 - 4 - 7 R 2 ・ R 3 車の決定方法

用いている車頭間隔と走行速度の関係を表したテーブル F からこれらの車両の平均の移動速度 $F (L / X)$ を求める。一方、直進・左折青終了時刻までの時間 T で、この区間長 L を割ってやることにより区間最後尾の車が直進・左折青現示の終了時刻にちょうど交差点を通過できるような平均速度 L / T を求める。そして L を適当に変化させて $F (L / X)$ と L / T が一致するような区間長 L を求める。こうして求められた長さ L の区間に関しては、その中の最後尾の車が交差点に到着する時刻が、次の直進・左折青終了時刻に対して、他の車と比べてこれを越えない範囲で最も近いものになると考えられる。従って、この区間外で最も交差点に近いものを $R2$ 車、 $R2$ 車の直前の右折車を $R3$ 車とすることにより、直進車と右折車の交差点通過に関しての平等を図ることができる。

5. 4. 3. システムへのインプリメント

このアルゴリズムを小ゾーン連続形自動車パケット通信システムに次のようにインプリメントする。

○ 速度指示の与え方

まず、R 1 車や R 2 車への指示速度はこれらが各通信ゾーンで通信する毎に、等速度で走行すると各々の右折青現示開始時刻に交差点に到着できるような速度を計算して求める。

○ 車線指示の与え方

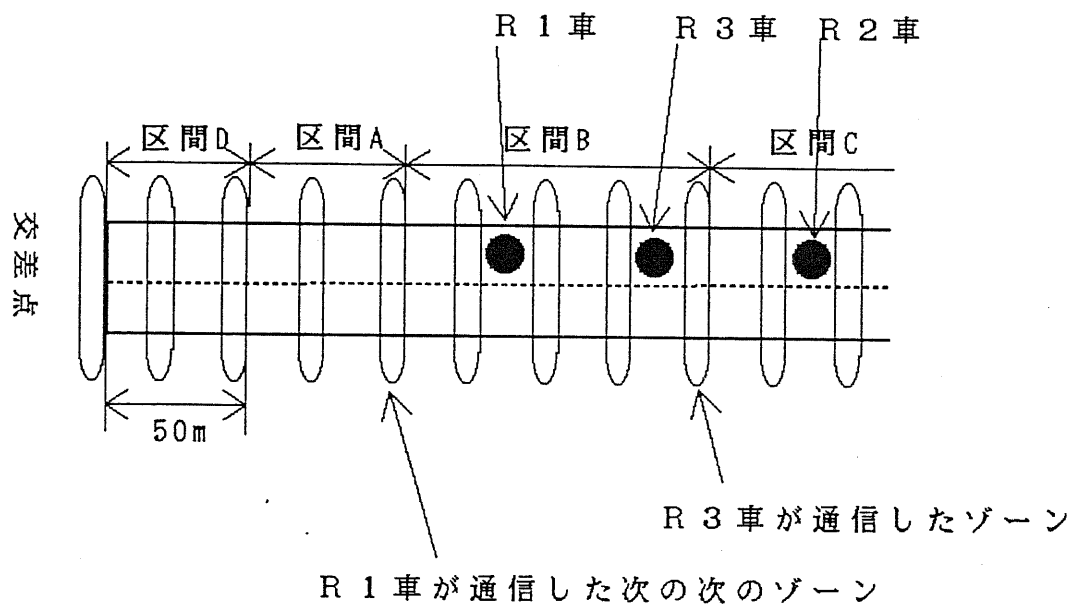
右・左折車に対しては、各々右・左車線が指示される。直進車に対する指示車線は通信ゾーンが図 5 - 4 - 8 のどの区間にあるかによって以下のように定める。

(区間 A)

ここでは、交差点直前からの両車線の直進・左折車の台数が等しくなるように R 1 車を左から追い越してきた直進車に対して車線指示を与える。

(区間 B)

この区間に存在する直進車には全て左車線を指示する。



(区間境界の設定規則)

図5-4-8 アルゴリズムのインプリメント

(区 間 C)

この区間では任意の車線選択を許可する.

(区 間 D)

交差点手前 5 0 m 以内であるこの区間内では車線変更禁止とした.

このようにして作成された指示は, 3. 5. で述べた情報伝送方式にしたがって次の通信ゾーンで伝送される. また, ゾーン間では車両が等速で走行するものとしてその位置を推定することとしている.

5. 4. 4. シミュレーション評価

このアルゴリズムの効果を小ゾーン連続形自動車パケット通信システムを想定して 2 5 m 毎にゾーンを配置して誘導を行った場合と右折車の待合せを行わない通常の場合に関して, 先と同様のミクロモデルを用いたシミュレーションを行って, 交差点通過に要する目的方向別の平均時間 (O D - T I M E) を求めて比較した. 誘導を行わない通常の場合では, 直進車は交差

点の 200 m 手前から左へ車線変更するものとした。

尚、このための評価プログラムを付録 A. 3. に示す。

ここで信号パラメタとしての T G は 30 s, スプリットは 0.5 とした。扱う区間は交差点の上流 2 km から交差点までとし、右折車の割合は 0.1, 0.2, 0.3 の 3 通りの場合について検討を行った。ここでは、1500 s 間シミュレーションを行い、その中で後半の 750 s 間に発生した直進・左折車と右折車の各々について O D - T I M E を求めた。その結果を図 5 - 4 - 9 ~ 図 5 - 4 - 12 に示す。図 5 - 4 - 9 は誘導を行わなかった場合の直進左折車、図 5 - 4 - 10 は誘導を行った場合の直進左折車、図 5 - 4 - 11 は誘導を行わなかった場合の右折車、図 5 - 4 - 12 は誘導を行った場合の右折車、各々について O D - T I M E を求めたものである。

このように、通常の場合では流入交通量が 900 台/h 以上となると交差点通過に要する時間が直進左折車及び右折車共に増大する。これは交差点の交通処理能力に対して直進左折車の流入交通量が大き過ぎるためにこれを捌ききれなくなり直進左折車が交差点手前

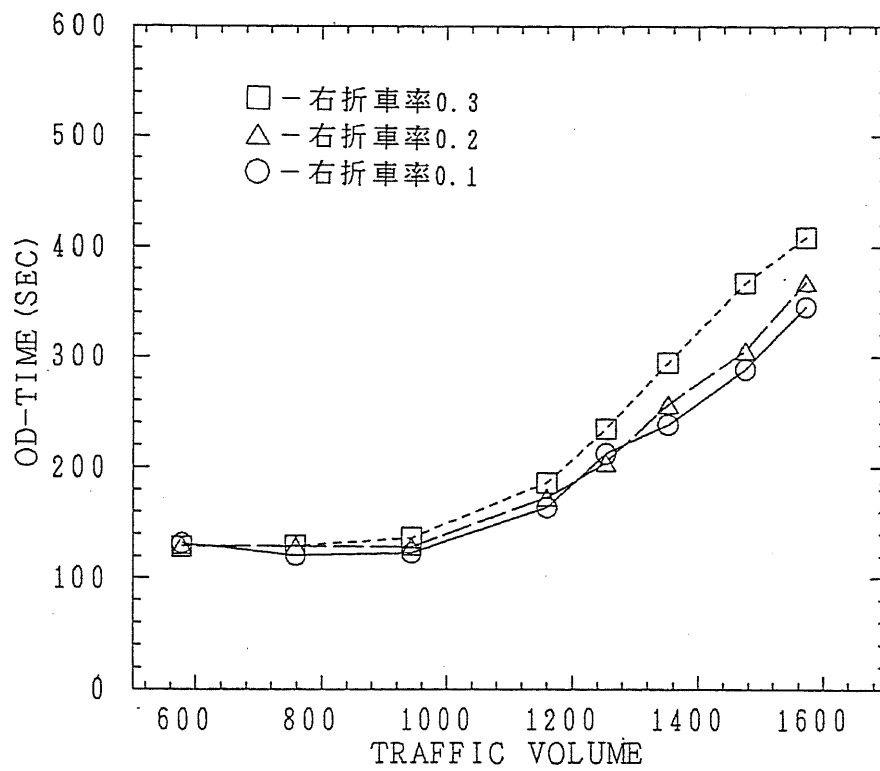


図 5 - 4 - 9 直進・左折車の OD-TIME (誘導無しの場合)

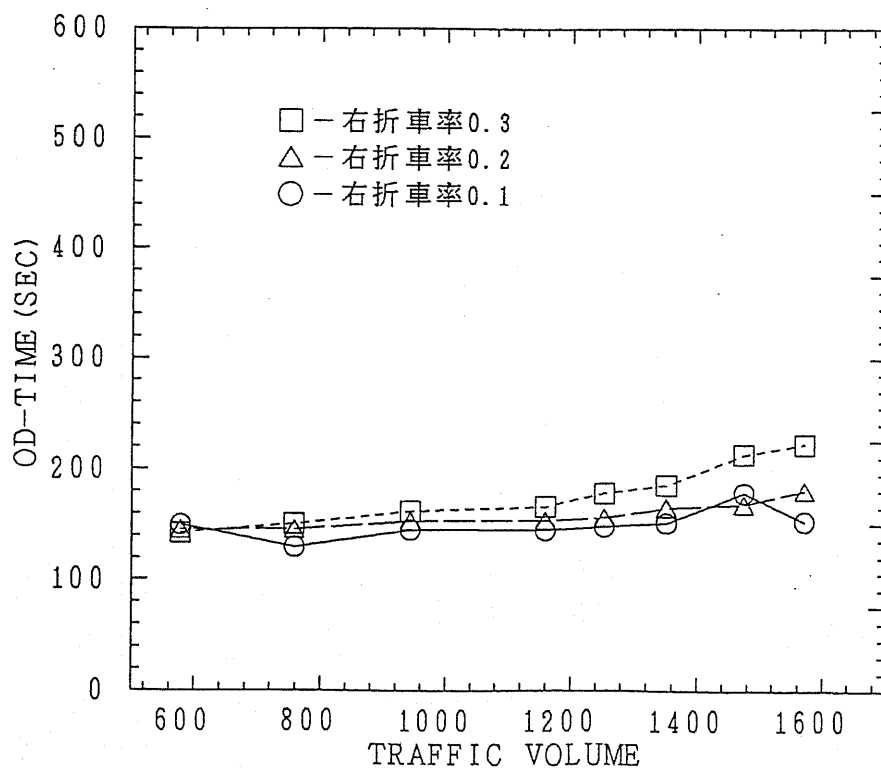


図 5 - 4 - 10 直進・左折車の OD-TIME (誘導した場合)

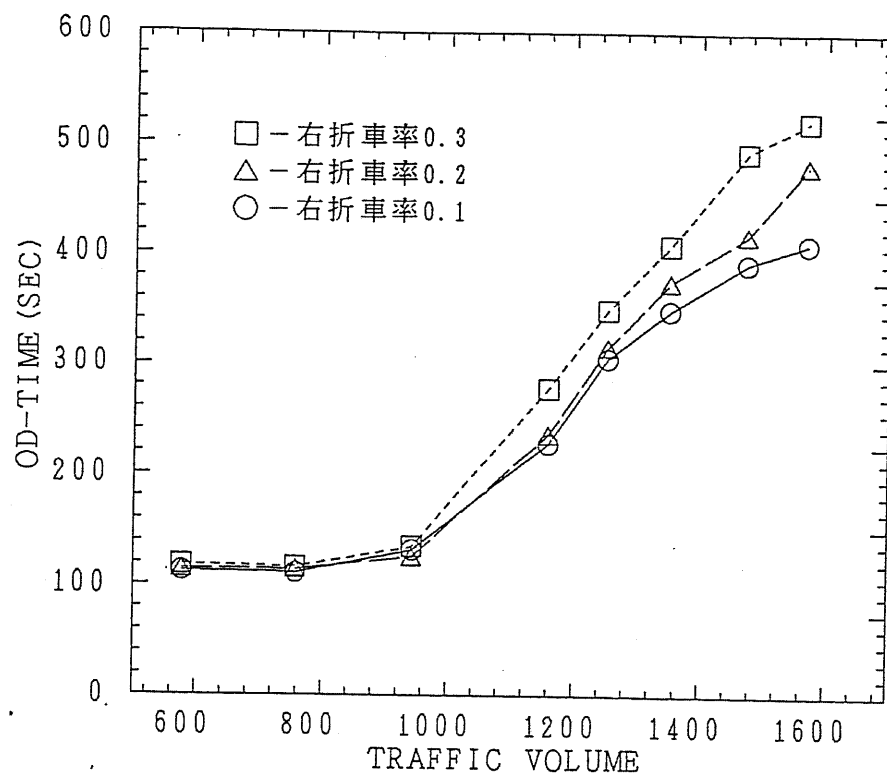


図 5 - 4 - 1 1 右折車の OD - TIME (誘導無しの場合)

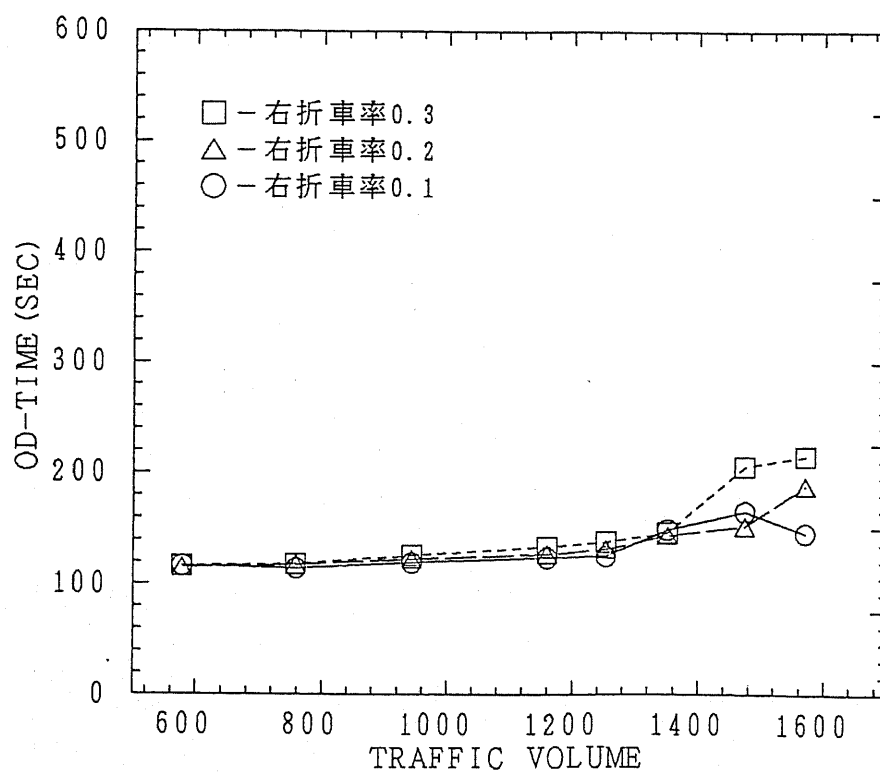


図 5 - 4 - 1 2 右折車の OD - TIME (誘導した場合)

において渋滞を生じ、右折車もこの影響を受けているものと考えられる。これに対して本システムにより右折車の持ち合わせ制御の誘導を行ってやることにより、数十%以上大きな流入交通量に対しても交差点通過時間の大幅な悪化は見らず、その容量を大きく増大できることが示された。またこの結果では、特に直進車に対する効果が顕著であるが、これは本アルゴリズムでは右折車線を直進車にも利用させるという発想に基づいているためと考えられる。

5. 4. 5. 通信ゾーンの配置間隔の検討

ここでは走行誘導への応用という面からみて必要な通信ゾーンの配置間隔に関する検討を行う。先に説明した街路の信号交差点における右折車の待ち合わせ制御のアルゴリズムに関して、同様のミクロモデルによるシミュレーションを行って、通信ゾーンの配置間隔と右折車の混入率をパラメータとして直進左折車及び右折車の交差点通過に要する時間の平均を5. 4. 4.と同様にO D - T I M Eとして求めた。ここで交通量は1 6 0 0 [台 / h], 1 4 0 0 [台 / h], 1 2 0

0 [台 / h] の 3 通りについてシミュレーションを行った。また、通信ゾーンは等間隔で配置することとしており、その他の条件は 5. 4. 4. におけるシミュレーションと同様としてある。

図 5 - 4 - 1 3 が 1 2 0 0 [台 / h] の場合の直進左折車、図 5 - 4 - 1 4 が 1 2 0 0 [台 / h] の場合の右折車、図 5 - 4 - 1 5 が 1 4 0 0 [台 / h] の場合の直進左折車、図 5 - 4 - 1 6 が 1 4 0 0 [台 / h] の場合の右折車、図 5 - 4 - 1 7 が 1 6 0 0 [台 / h] の場合の直進左折車、図 5 - 4 - 1 8 が 1 6 0 0 [台 / h] の場合の右折車、各々の結果である。これらより、交通量が 1 4 0 0 [台 / h] 以上の場合に、通信ゾーンの配置周期の差による影響が大きく表れてくる。特に交通量が多い 1 6 0 0 [台 / h] の場合に関して見れば、通信ゾーンを 2 0 m 以下の間隔で密に配置してもあまりアルゴリズムの効果は向上しないが、通信ゾーンが 3 0 m 以上となると、特に右折車の混入率が高い場合にはアルゴリズムの効果が低下し、交差点通過に要する時間が増大することがわかる。

ここで、通信ゾーンの配置間隔としては、 2. 4.

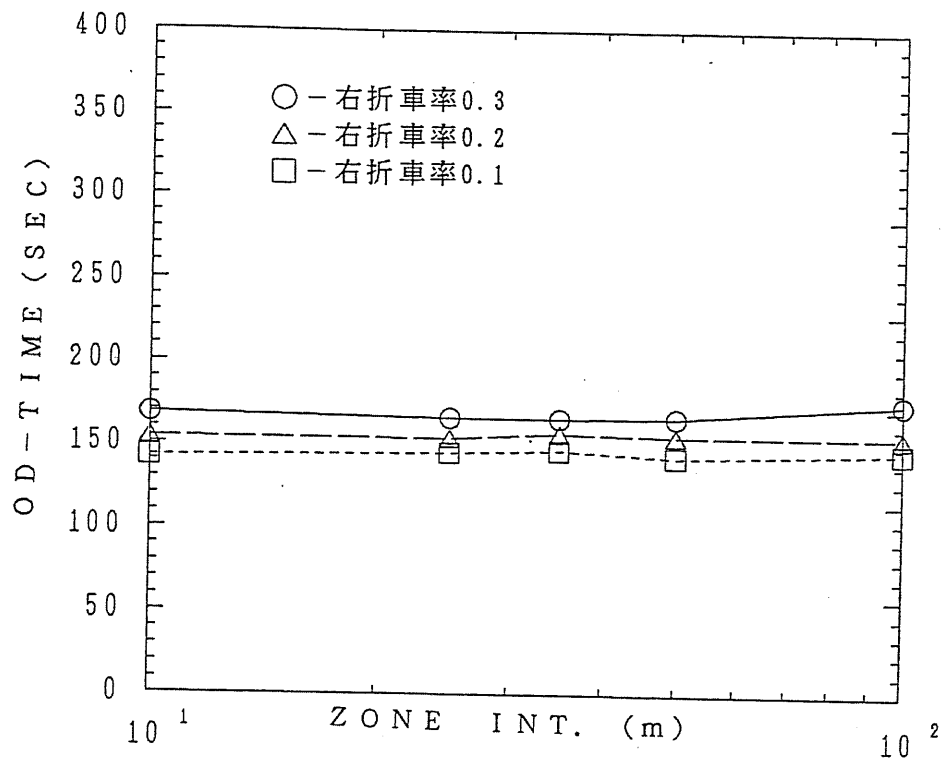


図5-4-13 通信ゾーンの配置間隔とOD-TIME
 (直進・左折車, 交通量: 1200 [台/時])

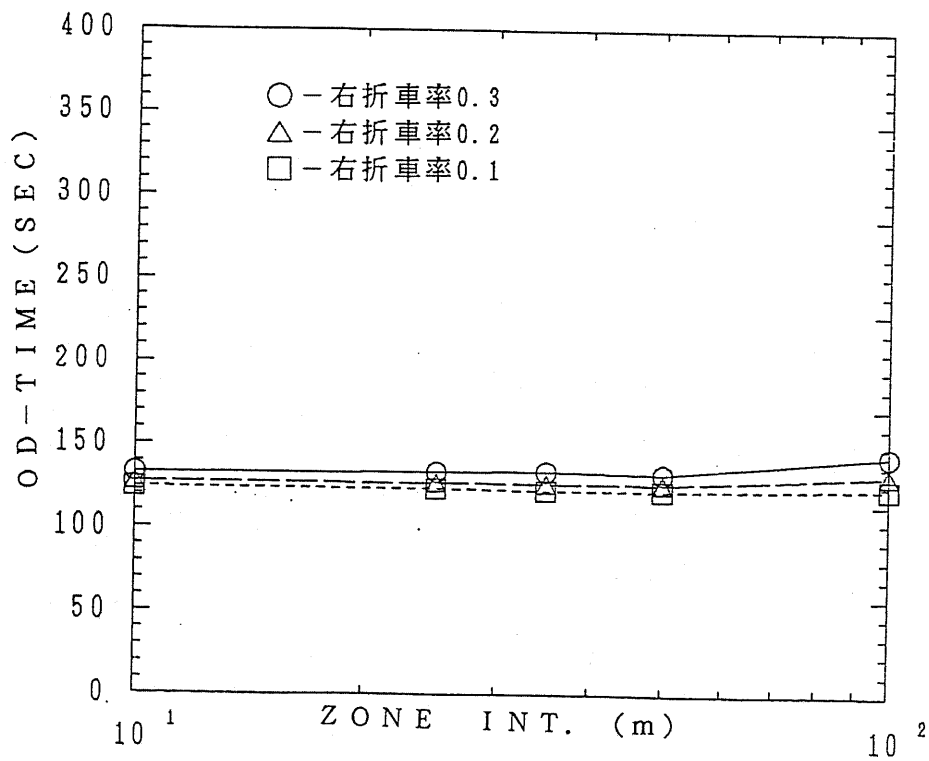


図5-4-14 通信ゾーンの配置間隔とOD-TIME
 (右折車, 交通量: 1200 [台/時])

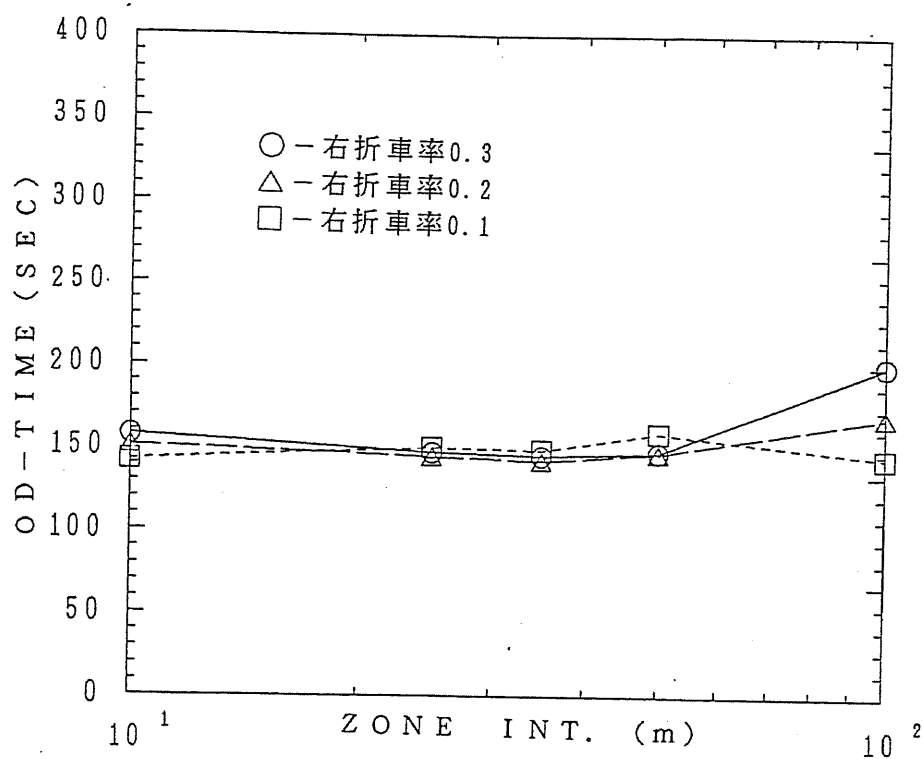


図5-4-15 通信ゾーンの配置間隔とOD-TIME
(直進・左折車, 交通量: 1400 [台/時])

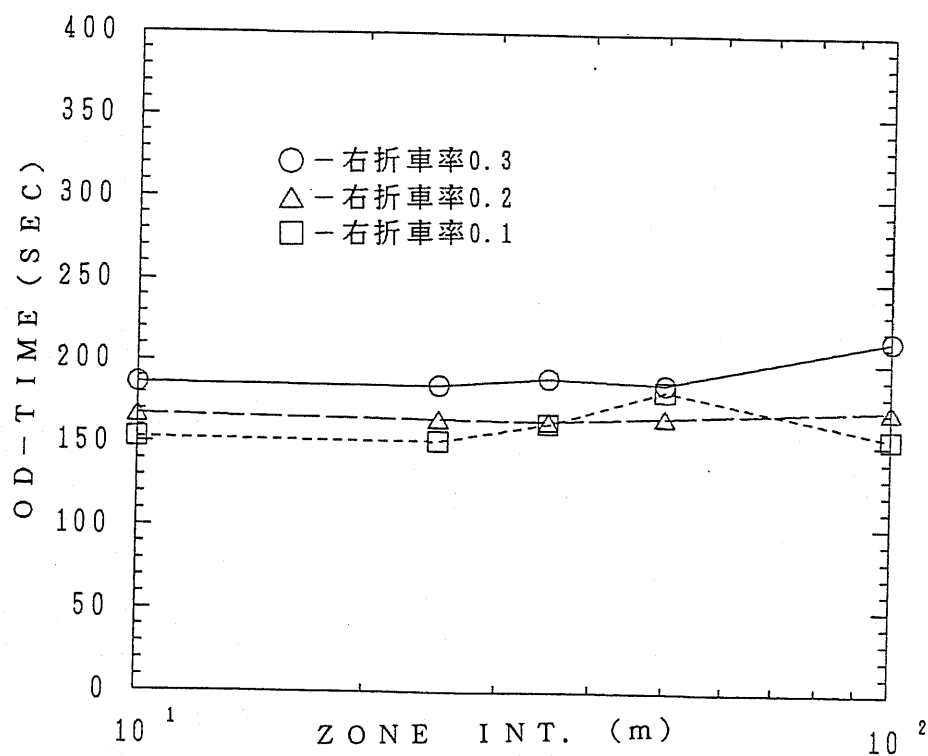


図5-4-16 通信ゾーンの配置間隔とOD-TIME
(右折車, 交通量: 1400 [台/時])

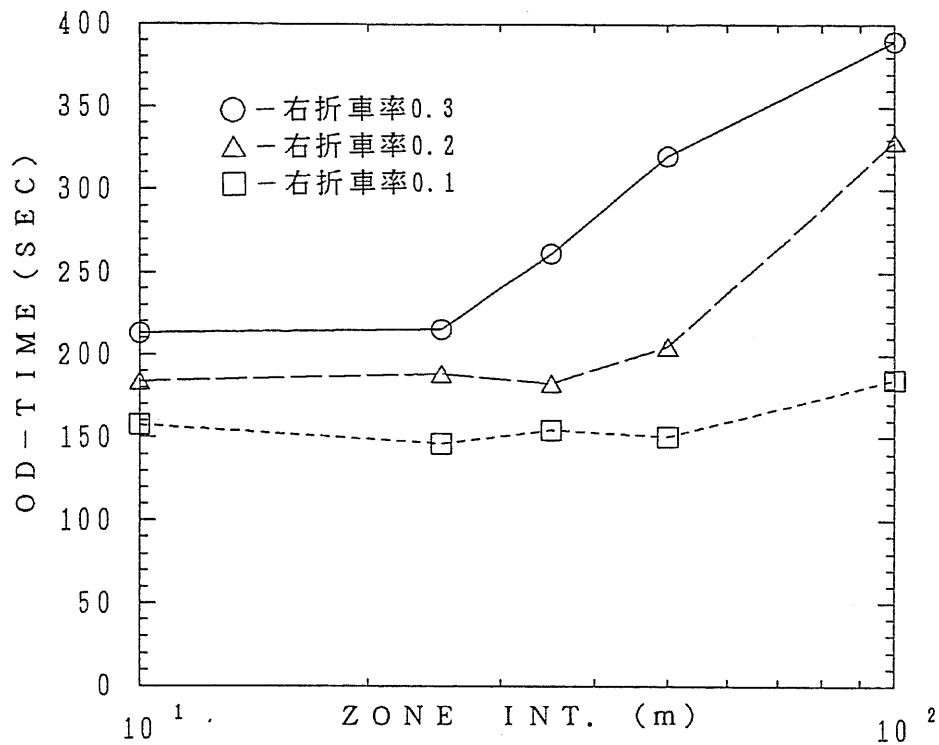


図5-4-17 通信ゾーンの配置間隔とOD-TIME
(直進・左折車, 交通量: 1600 [台/時])

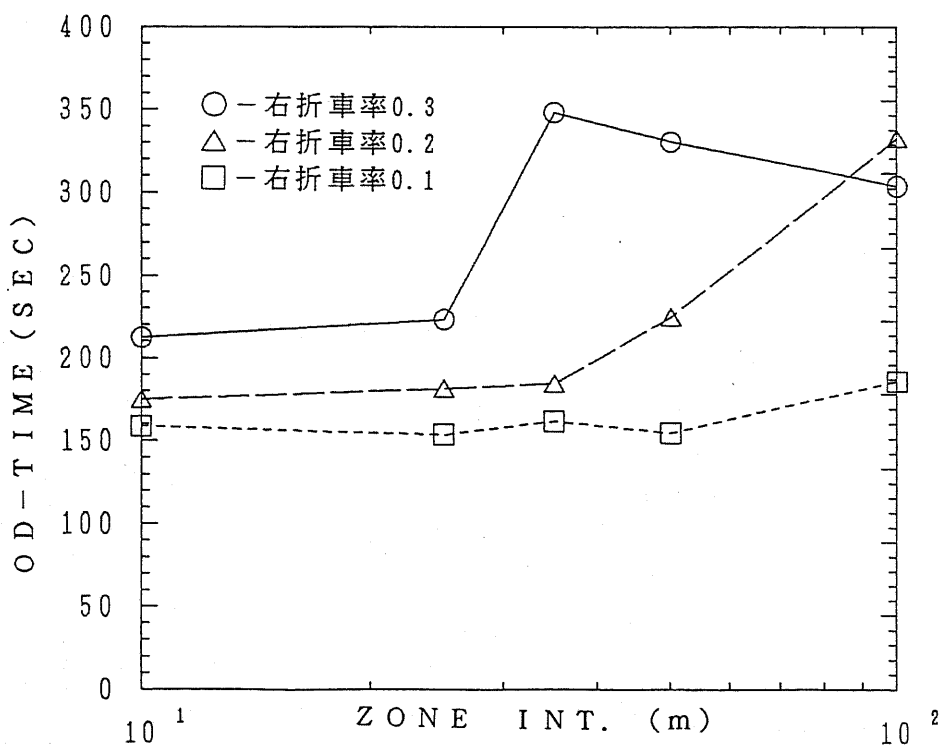


図5-4-18 通信ゾーンの配置間隔とOD-TIME
(右折車, 交通量: 1600 [台/時])

に述べたように交信干渉の影響を考慮したハードウェア面における検討より求められた制約条件として、26 m 以上離せばよいことがわかっている。ここでの検討より、これは交通量の多い場合であっても応用面における制約条件も満たしていることがわかった。

5. 5. 結 言

ここでは、小ゾーン連続形自動車パケット通信システムの交通管理への応用として、走行誘導を行うことについて述べた。ここでの検討では、具体的なアルゴリズムの例として、高速道路の分流点における動的車線規制と街路の信号交差点における右折車の待ち合わせ制御のアルゴリズムについて説明し、それらについてミクロモデルを用いた評価シミュレーションを行うことによりその有効性を示した。

但し、実際に走行誘導の手法の実現に当たっては、このような速度や車線に関する指示が運転中に頻繁に発せられるということ自体が社会的に容認される必要がある。これに関しては今後さらに人間工学的な見地からのより詳細な検討を行ってゆく必要があると考えられる。

6 . グローバルシステムへの応用

6. 1. 序言

第4章でも述べたように、本システムの通信機能においては、複数の道路リンクにまたがって移動する自動車を追跡してデータ転送を行うことも考えている。そこで、本章ではこのような通信機能を利用して、グローバルな情報提供を行うことに関して述べる。

まず、そのために必要な基本的な機能について、その実現形態を示す。次にそれらを利用して可能と考えられるシステムとして、地上側利用者との双方向データ通信、要求応答型の情報提供サービス、及び、放送型の情報提供サービスの3つの形態を示し、その各々について、どのようなシステムが応用例として考えられるかについてまとめた。

6. 2. 提供するグローバルな機能

ここでは、本システムの汎用データ通信の機能を利用して提供されるグローバルな機能について説明する。まず、本システムで提供する基本的なデータ通信機能のパターンとしては、接続要求、データ伝送、終了要求、放送型情報伝送がある。以下、各々の実現形態について述べる。但し、以下の説明において用いられている図において（A）は3. 4. で述べた上り方向へのパケット伝送を、（B）は下り方向への最初のパケット伝送を、（C）は下り方向への途中のパケット伝送を、各々表している。

（1）接続要求

自動車側からの接続要求のパターンを図6-2-1に、地上局側からの接続要求のパターンを図6-2-2に各々示す。このように発呼側からの接続要求信号に対して、着呼側より接続確認信号を応答する形態とする。

（2）データ伝送

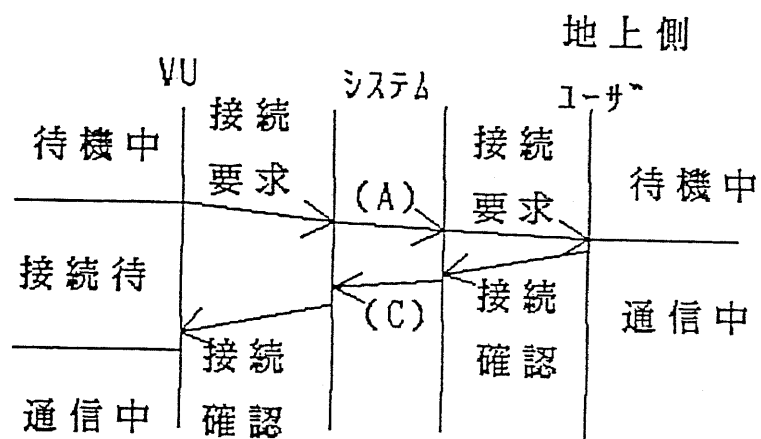


図 6 - 2 - 1 車載側からの接続要求

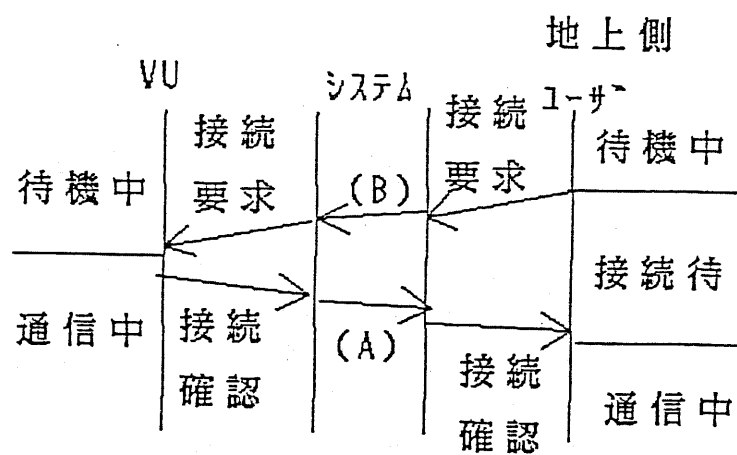


図 6 - 2 - 2 地上側からの接続要求

上りデータ伝送のパターンを図6-2-3に、下りデータ伝送のパターンを図6-2-4に各々示す。このように送信側からのデータに対して、受信側よりACKを返すことによりデータの到達確認を行う。

(3) 終了要求

自動車側からの終了要求のパターンを図6-2-5に、地上局側からの終了要求のパターンを図6-2-6に各々示す。このように、地上側または車載側より終了要求信号が発せられるとシステムより確認信号を返して通信を終了する。

(4) 放送型情報伝送

放送型の情報伝送のパターンを図6-2-7に示す。このように放送型で伝送するデータはシステム内のバッファに常駐させておき、自動車側からの受付可能信号に応じて地上側より伝送される。

以上が本システムで提供するグローバルな機能である。

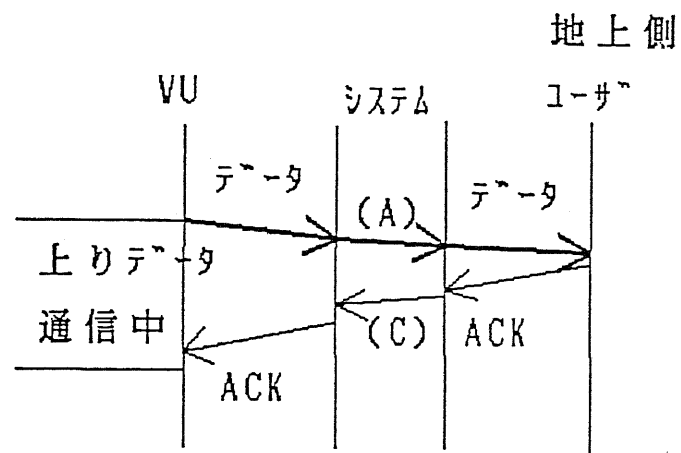


図 6 - 2 - 3 上り方向へのデータ伝送

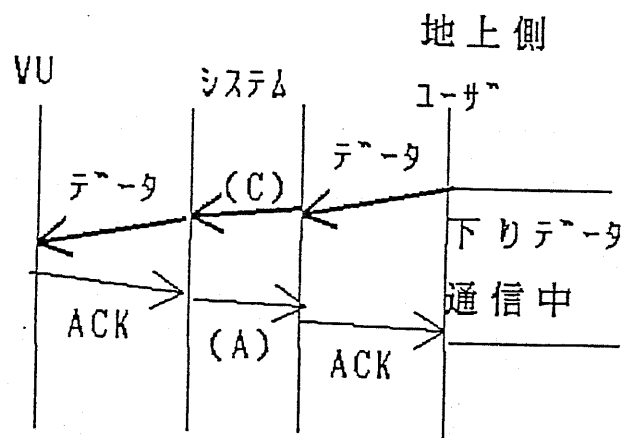


図 6 - 2 - 4 下り方向へのデータ伝送

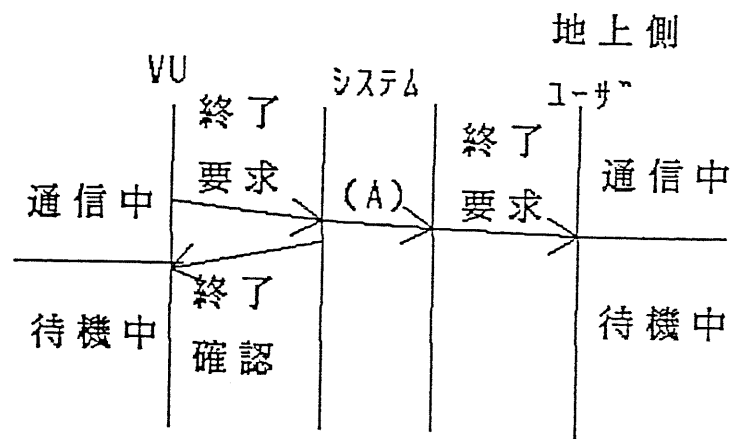


図 6 - 2 - 5 車載側からの終了要求

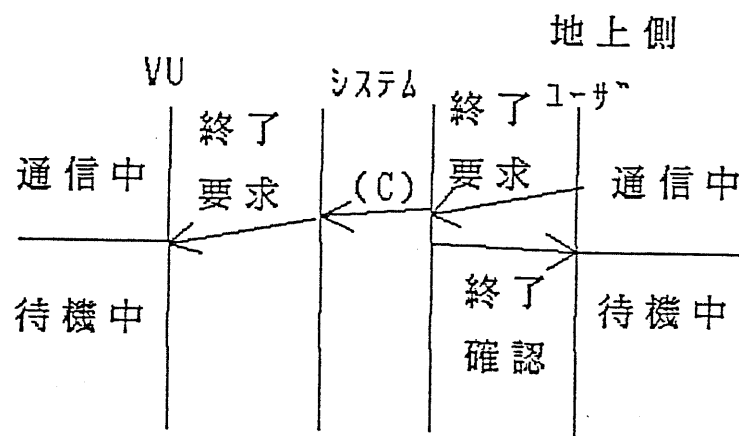


図 6 - 2 - 6 地上側からの終了要求

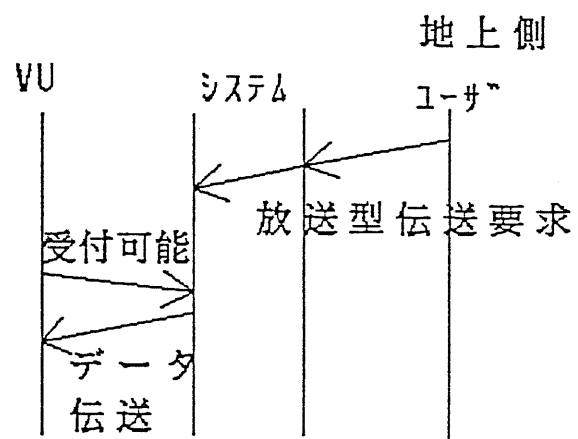


図 6 - 2 - 7 放送型データ伝送

6. 3. 応用システム例

6. 3. 1. 双方向データ通信

この利用形態では、システムは図 6 - 3 - 1 に示したようなものとなる。これにより地上側の利用者と車側の利用者との間でメッセージやデータを双方向で伝達する。このサービスを実現するための基本パターンの組合せは、

(地上側からの接続要求 or 車載側からの接続要求)

↓

(上りデータ伝送 or 下りデータ伝送 の繰り返し)

↓

(地上側からの終了要求 or 車載側からの終了要求)

となる。また、この形態の応用としては、トラック等の運行指令を含む業務連絡や一般の双方向データ通信システムなどが考えられる。

6. 3. 2. 要求応答型情報提供

この利用形態では、システムは図 6 - 3 - 2 に示し

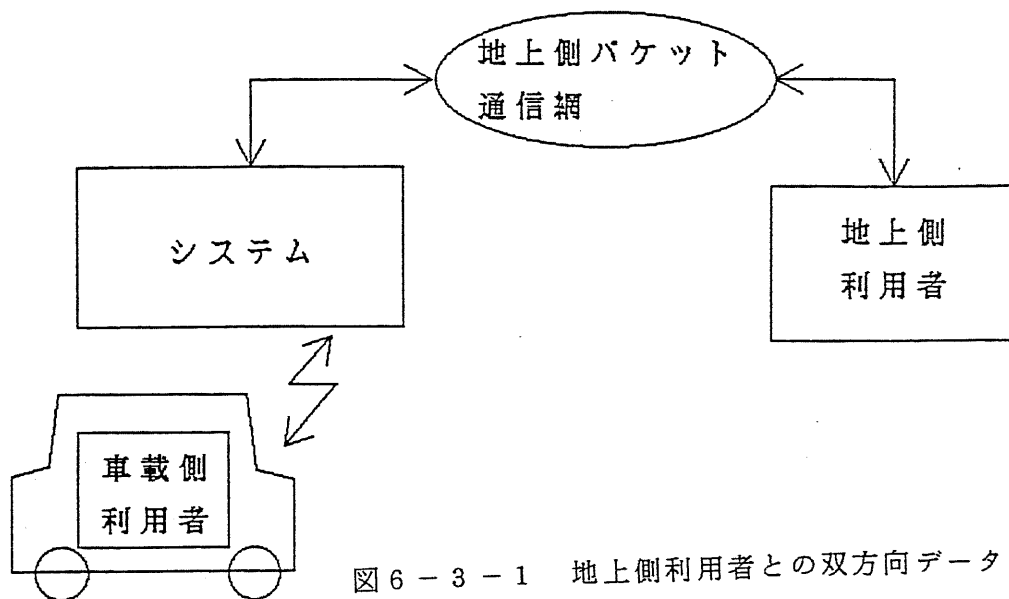


図 6 - 3 - 1 地上側利用者との双方向データ通信システム

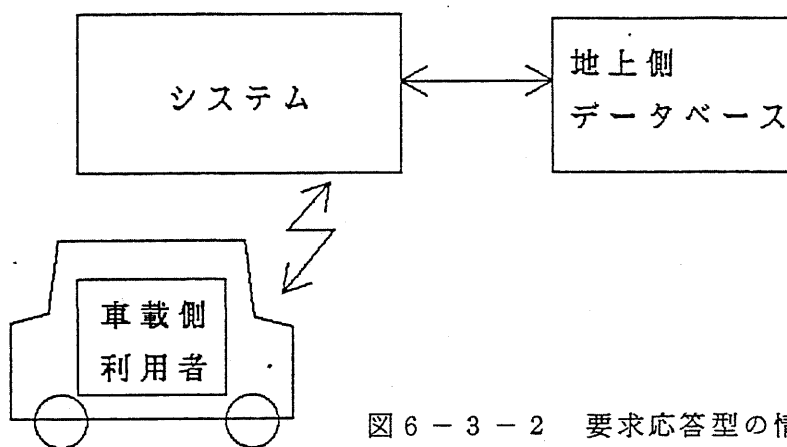


図 6 - 3 - 2 要求応答型の情報提供システム

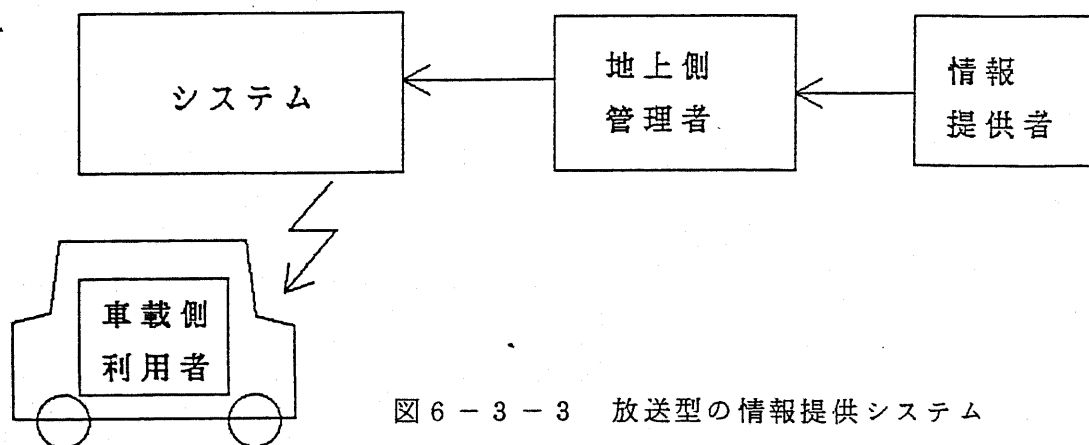


図 6 - 3 - 3 放送型の情報提供システム

たようなものとなる。そして車載側からの要求により、
地上側のデータベース業者などにアクセスして要求応
答型で種々の情報提供を行う。このサービスを実現す
るための基本パターンの組合せは、

(車載側からの接続要求)

↓

(上りデータ伝送)

↓

(下りデータ伝送)

↓

(車載側からの終了要求)

となる。また、この形態の応用としては、観光業者等
への予約・問い合わせや要求応答型の施設案内・ガイ
ド情報の提供等が考えられる。

6. 3. 3. 放送型情報提供

この利用形態では、システムは図6-3-3に示し
たようなものとなる。これにより道路上を走行してゆ
く上で、運転者が留意する必要がある、または、知っ
ていると便利な情報を、適当な場所で一方向的に車両

に伝達する。このサービスを実現するための基本パターンは

(放送型情報伝送)

である。

また、この形態の応用としては、事故、渋滞、道路異常、緊急情報、規制情報等の注意情報の伝達、及び、駐車場の空き情報やトイレ設置など、沿線施設関連の情報を提供することなどが考えられる。

6. 3. 4. 応用形態の定量的検討

本システムの汎用データの伝送容量に関する定量的な値としては、第3章における評価シミュレーションの結果及び第7章における実験結果が参考となる。まず、第3章の結果である図3-5-15及び3-5-16によれば、そのスループットは0.65程度である。一方、第7章の表7-5-2によれば、走行データの伝送所要時間が約40[ms]であるから、往復6車線の道路の全車線に交通量1800[veh/h]の飽和交通流が通るとすると、1秒間に走行データの伝送に使われる時間 T_m は、

$$T_m = (1800 [\text{veh/h}] / 3600 [\text{h/s}]) \cdot 6 [\text{車線}] \cdot 40 [\text{ms}] = 120 [\text{ms}] \cdots (\text{式 6-3-1})$$

となる。一方、同表から汎用データの情報伝送効率が約70%であるから、汎用データ伝送のスループットの実験値 S_{exp} は、

$$\begin{aligned} S_{\text{exp}} &= (70 [\%] / 100) \cdot (1000 [\text{ms}] - 120 [\text{ms}]) / 1000 [\text{ms}] \\ &= 0.62 \cdots (\text{式 6-3-2}) \end{aligned}$$

となり、第3章の結果と一致する。

ここで交通量 $1800 [\text{veh/h}]$ の場合の平均車頭間隔は 20 m であるから1つの通信ゾーン（長さ 10 m ）内の自動車の台数 N_v は、

$$N_v = 6 [\text{車線}] \cdot 10 [\text{m}] / 20 [\text{m}] = 3 \text{ 台} (\text{式 6-3-3})$$

となる。ここでゾーンの配置間隔が 20 m であるとし、また、スループットとして実験値 S_{exp} をとると一台当

りの可能なデータ伝送速度 B は,

$$B = 9600 \text{ [bps]} \cdot \text{Sexp} \cdot 10 \text{ [m]} / (10 \text{ [m]} + 20 \text{ [m]}) \cdot N_v = 661 \text{ [bps]} \cdot \dots \quad (\text{式 6-3-4})$$

となる。この伝送容量を考慮すると提供可能なサービス形態としては、少なくともテレメッセージや電子メール等のようなメッセージ伝送は十分行うことができ、さらにある程度圧縮した画像データの伝送も可能と考えられる。

また、先に述べた3つの応用形態の中では、上位層とのやり取りが比較的煩雑な双方向データ伝送においては、そのためのオーバーヘッドの影響を受けると考えられる。これに対して要求応答型では、駐車場案内などの各道路リンクと密接な関連を持つ情報を路側装置内で処理する等により、上記の伝送容量を充分活用できると考えられる。さらに放送型ではデータ伝送が一方的であり、オーバーヘッドの問題は生じないが、本システムのもつ各自動車に対する通信の個別性を充分活用できないうらみがある。

6. 4. 結 言

本章では、小ゾーン連続形自動車パケット通信システムの応用としてグローバルな情報提供を行うことを考え、いくつかの例を挙げてその実現形態について述べるとともに定量的検討を行った。

本章で述べたように、本システムでは種々の応用例が考えられる。今後、このシステムが情報通信システムとしてより魅力的なものとなってゆくためには、個々のニーズに応えられるような各種アプリケーションを充実させてゆくことも重要となってくるであろうと考えられる。

7. スケールモデルでの実現可能性検証実験

7. 1. 序言

本研究では、小ゾーン連続形自動車パケット通信システムが実現可能であることを具体的に示すためにモデル実験システムの構築を行った。ここで、実際のシステムにおいては、技術上の問題は主に各通信ゾーンでの個々の自動車の識別や各自動車との接続および通信ゾーン間の追跡などといった道路リンク内のきめ細かな制御の方式にあると考えられることから、本システムにおける1つの道路リンクを管理する部分を模擬したモデル実験システムを構築した。

本章ではこのモデル実験システムに関して、まずその構成を示す。次に、干渉波の影響の無視できる通信ゾーンの配置間隔について室内実験による検討を行いその妥当な値を調べた。また、構築したモデル実験システムに関して制御方式の動作確認を行うとともに処理時間や通信効率を測定した。さらに、本研究ではソフトウェアにより実現した交通流シミュレータと組み

合わせたハイブリッドシステムを構成し、実際に自動車
が移動する場合と等しいリアルタイムでの性能評価
試験を行って本システムの通信制御方式の実現可能性
を示した。

7. 2. 実験システムの装置構成[7][8][14][17][20]

7. 2. 1. 全体構成

本実験システムの構成を図7-2-1に示す。このように、本実験システムの規模は2つの通信ゾーンと車載機2台から成るものとした。これは後に述べる交信干渉や通信要求の衝突などの基本的な実験のために最低限必要となるものに限定したためである。

7. 2. 2. 路車間データ送受信装置

本システムでは、2. 4. でも述べたように、路車間の通信仕様としてはつくば自動車データ収集・提供システムなどの実システムにおいてすでに実績のある誘導通信方式を用いるものとする。ここではこれに基づいて、このモデル実験システムのために図7-2-2に示したような構成の路車間データ送受信装置（地上側：RSE，車載側：VU）を作成した。表7-2-1はその通信仕様である。実システムと同様に、誘導通信帯でMSKを用いて半二重で変復調を行う。車載アンテナは図7-2-3(a)に示すものとし、地上アンテナは室内での実験の便宜のために図7-2-3(b

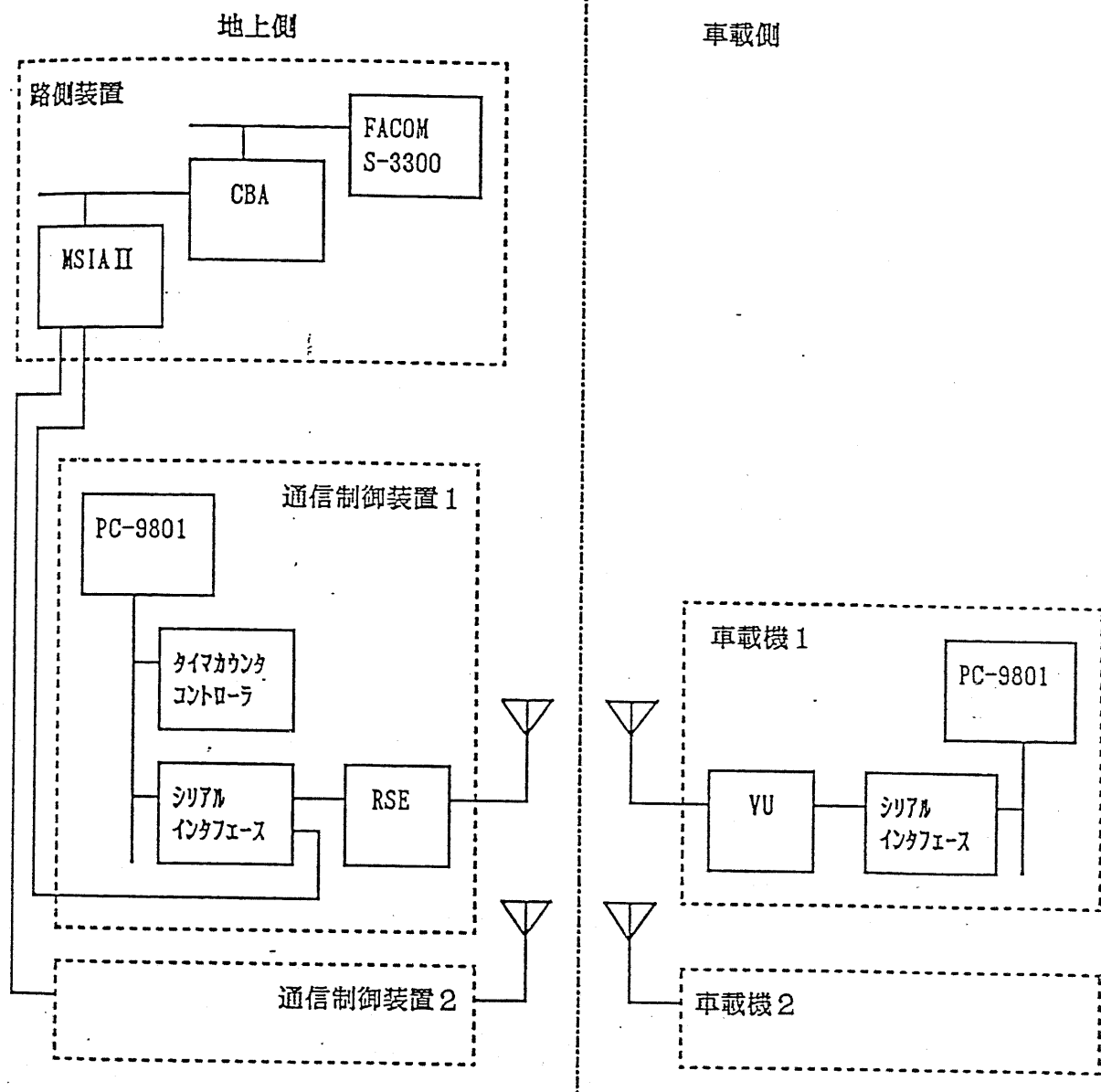


図 7 - 2 - 1 モデル実験システムのハードウェア構成

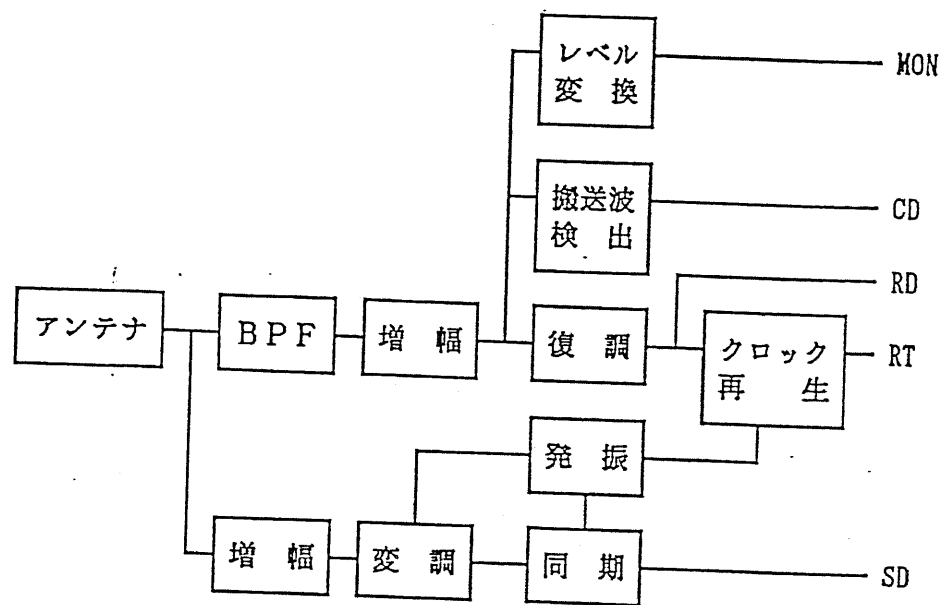
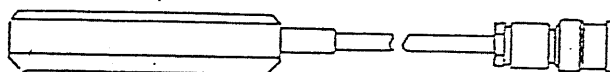
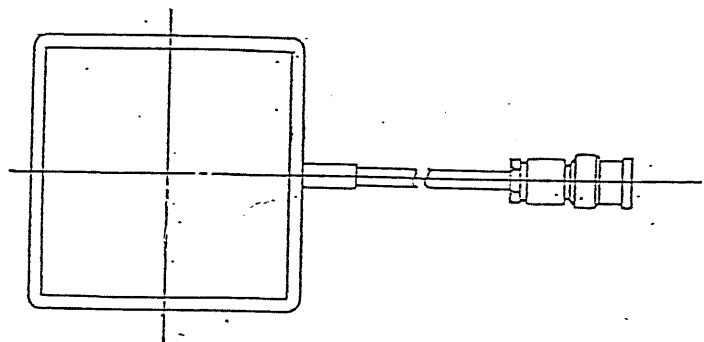


図 7 - 2 - 2 路車間データ送受信装置の構成

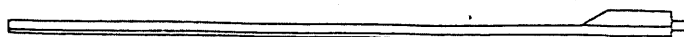
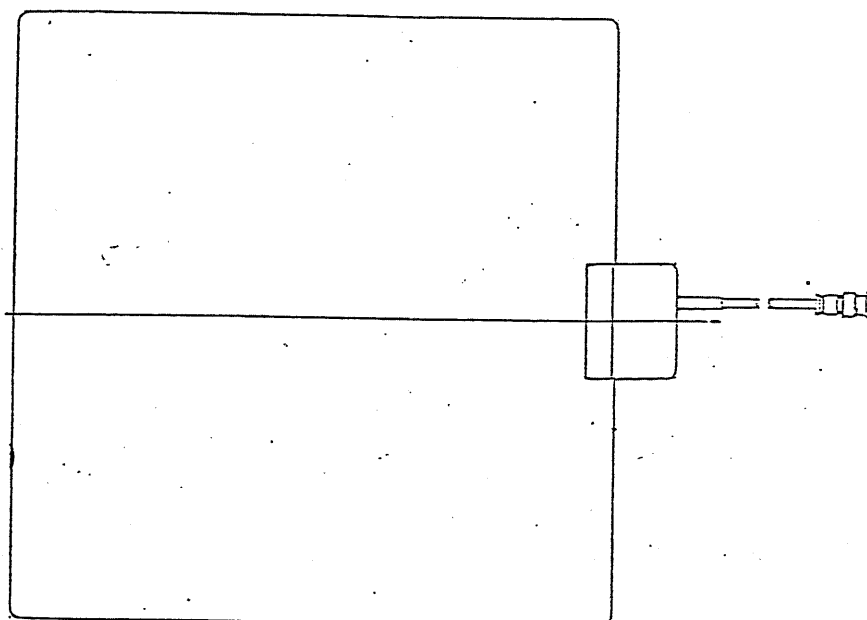
表 7 - 2 - 1 路車間データ送受信装置の仕様

項 目	路上機(RSE)	車載機(VU)
送信周波数	223.2KHz	304.8KHz
周波数偏移	±2.4KHz	←
周波数変動	100ppm以下	←
変調方式	MSK	←
復調方式	遅延検波	←
伝送速度	9600bps	←
送信出力	微弱電波の範囲	←
受信しきい値	MON端子で1V _{p-p}	←



寸法： 64(W)x15(H)x64(D)[mm]

(a)車載局用アンテナ



寸法： 320(W)x15(H)x355(D)[mm]

(b)地上局用アンテナ

図 7 - 2 - 3 アンテナの外観

)に示すループアンテナで実際の場合の1 / 50のスケールモデルとなるものを使用した。

7. 2. 3. 情報処理装置

本実験システムの処理装置としては、RSSの機能をミニコンピュータFACOM S-3300で、CCの機能および車載機の機能をパーソナルコンピュータPC-9801で模擬した。S-3300とPC-9801の間の接続は、実験システムの都合上汎用のシリアルインタフェースMSIA IIによりRS232C調歩同期9600bpsの全2重で実現した。PC-9801に付加したシリアルインタフェースボード(LSI RS422SIF)はRS232Cインタフェースを2回線分持っており、そのうちの1回線をHDL C 9600bpsに設定して路車間データ送受信装置と接続し、残りの1回線を調歩同期9600bpsに設定してS-3300と接続した。さらにCC側のPC-9801には、通信制御手順における時間管理のためにタイマカウンタコントローラボード(98T CNT(9))を装着した。

7. 3. 実験システムの制御プログラム [7][8][14][17][20]

7. 3. 1. 車載機用プログラム

本実験システムにおける制御プログラムとしては車載側と地上側のものがある。ここではその中の車載機用プログラムについて述べる。車載機側プログラムには、まず通信制御用として、すでにトークンが割り当てられている車両に対してその車両のみが応答するトリガ（専用トリガ）やリンクに新たに進入した自動車を捉えるコンテンションアクセスのためのトリガ（汎用トリガ）から始まる走行データ通信、及び、上り・下り方向の汎用データ通信の4つ機能に対応するソフトウェアを作成した。そして受信した地上側からの信号により通信種別を選択し、これらのソフトウェアが実行される形態となっている。

また、ここでは実験において必要な初期化プログラム、初期設定プログラム、通信内容・実行結果の表示プログラム、及び、通信内容・実行結果の保存プログラムも併せて作成した。

7. 3. 2. 地上局用プログラム

地上側のソフトウェアは通信制御装置と路側装置の2ヶ所に分散している。以下各々の構成について説明する。

○路側装置のプログラム

路側装置の制御ソフトウェアは、図7-3-1に示したように、受信タスク、個別管理情報送信タスク、下り汎用データ送信タスクおよびこれらを管理するタスクからなっている。

受信タスクは通信制御装置から送られた信号種別に応じて個別情報を更新し、上り汎用データを保存する働きをする。個別管理情報送信タスクは管理タスクにより起動されるもので、管理タスクから通知されるトークンおよび処理コードに応じて個別情報をもとに転送するデータを組み立てて通信制御装置に送信する働きをする。下り汎用データ送信タスクも管理タスクにより起動されるもので、管理タスクから通知されるトークンおよびパケット番号により必要な下り汎用データを検索し、通信制御装置に送信する働きをする。管

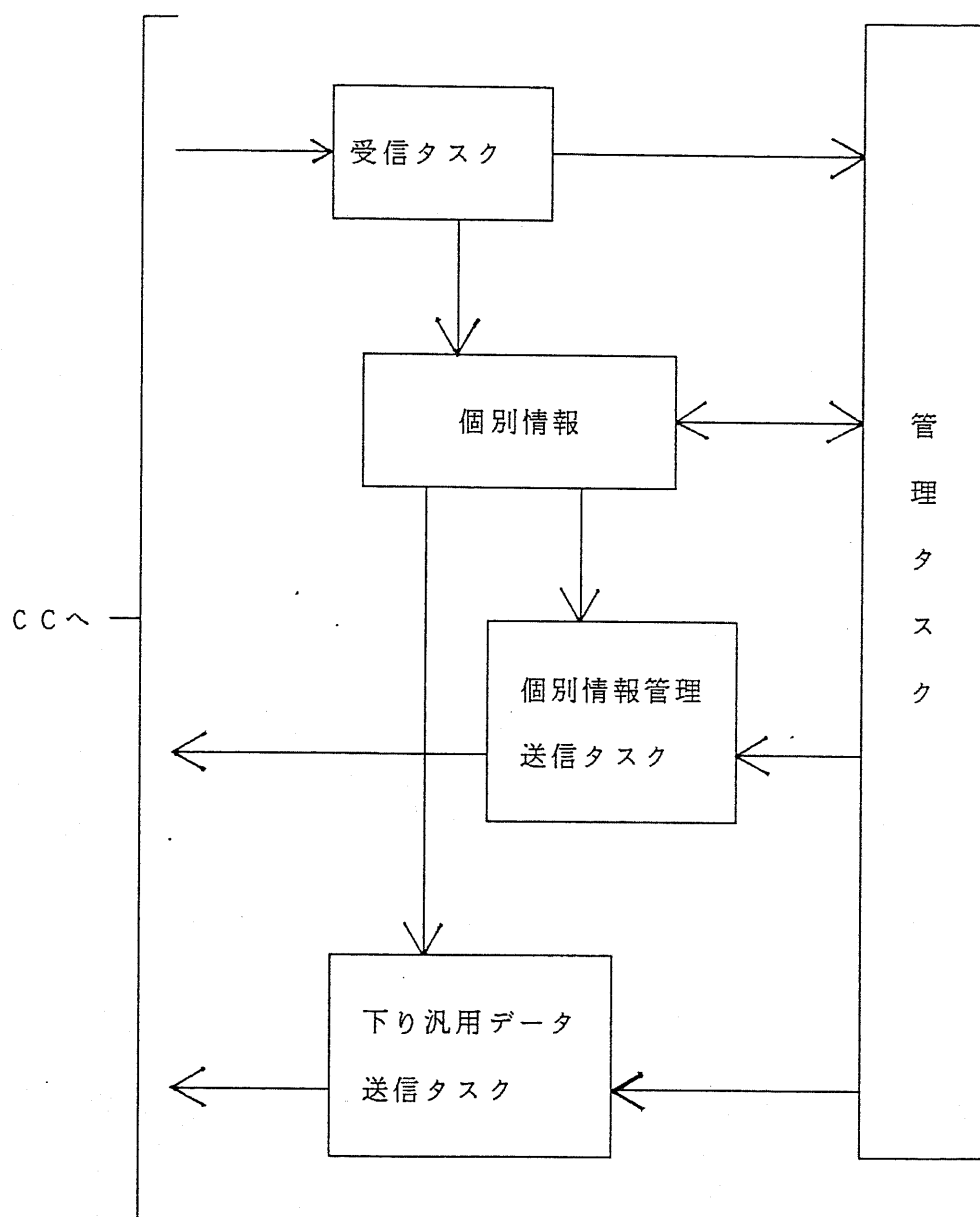


図 7 - 3 - 1 路側装置の制御プログラムの構成

理タスクは各自動車の次の通信ゾーンへの進入を予測し、そのトークンおよび処理コードを対応するゾーンへの個別管理情報送信タスクに送りそのタスクを起動し、また、下り汎用データ通信の要求がある時はその通信ゾーンへの下り汎用データ送信タスクにトークンおよびパケット番号を通知してそれを起動する働きをする。

○通信制御装置のプログラム

図 7 - 3 - 2 に通信制御装置の制御プログラムの構成概念を示す。ここでは、道路リンクレベルの制御を行う上で必要な機能として通信制御機能、チャネル割当機能および個別情報管理機能をインプリメントした。

通信制御としては、車載機の通信制御機能に対応して、専用トリガから始まる走行データ通信、汎用トリガから始まる走行データ通信、及び、上り・下り方向の汎用データ通信の4つの機能を持たせた。

チャネル割当としては、走行データ通信・汎用データ通信への割当と個々の自動車への割当の機能を持たせた。但し、汎用データ通信における個々の自動車へ

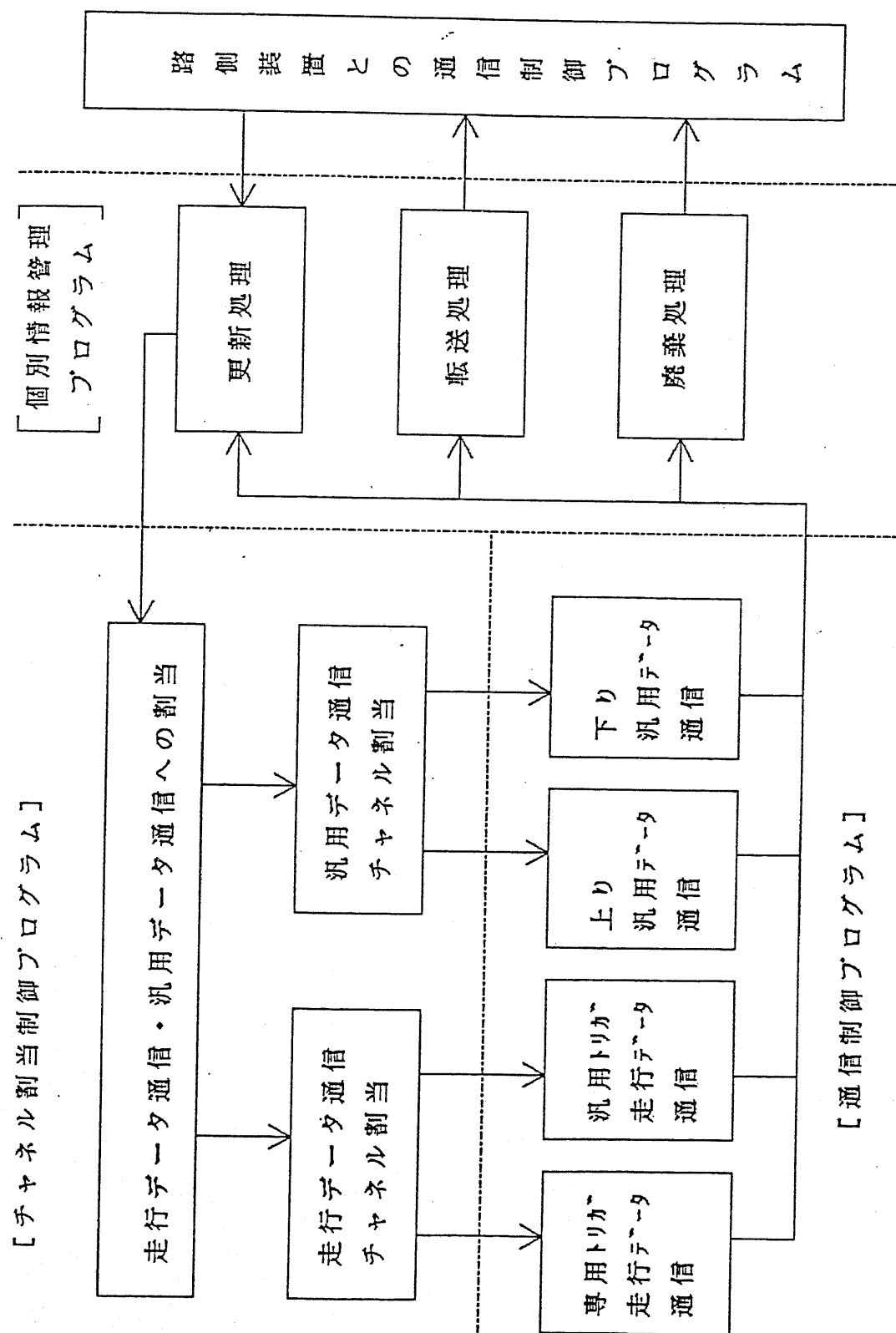


図 7-3-2 通信制御装置の制御プログラムの構成

の割当は現在のモデル実験システムでは、サイクリックに割り当てる方式を用いている。

個別情報管理としては、現在はトークン、走行データ通信の終了状況と終了時間、速度等の走行データ関連の情報と、上りと下りの汎用データ通信の伝送要求パケット数と伝送済みのパケット数等の汎用データ関連情報をシステムで管理しており、これらの情報の転送、更新、及び、廃棄の機能を持たせてある。

7. 4. 交信干渉の基礎実験 [12][17][20]

本研究では、完全なモデル実験システムの構築に先だって、図 7-4-1 に示したように路車間データ送受信装置と S-3300 の M S I A を調歩同期で接続して隣接ゾーン間の交信干渉の影響を調べるための基礎実験を行った。この時 R S E - 1 (V U - 1) を希望波送信用、R S E - 2 (V U - 2) を干渉波送信用、V U - 1 (R S E - 1) を受信用にしてビット誤り率の計測を行った。ここでは希望波の受信レベルを一定にし、干渉波の受信レベルをアンテナの相対位置により変化させることにより希望波対干渉波比 (D / U) を調整した。また、雑音は V U - 1 (R E S - 1) の B P F の前に混入させ、各信号の強度は V U - 1 (R S E - 1) 内において受信レベル測定のために設けられているモニタ端子にて尖頭値電圧で測定した。ここで、希望波は 80 ms 周期で 24 バイトずつ 10, 000 回送信し、干渉波はそのほぼ中間の 10 バイトに混入させた。ビット誤り率の計算に当たっては干渉波混入部の 10 バイトに関するものを有効なデータとして採用した。そして、1 回の伝送で発生する誤りは高

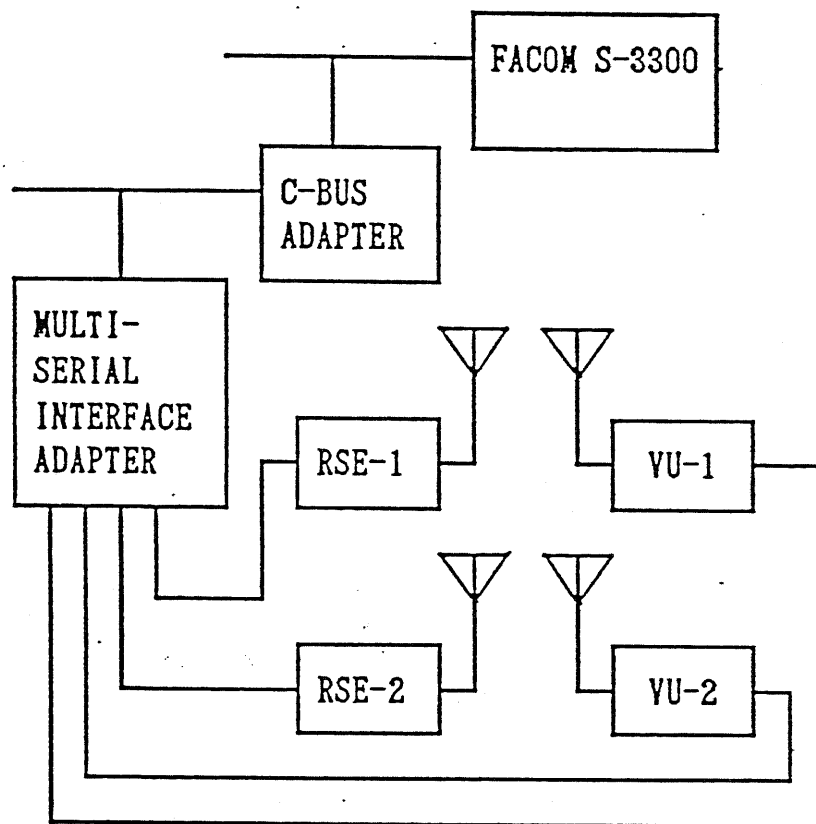


図 7 - 4 - 1 基礎実験のシステム構成

々 1 ビットであり、干渉波混入部分以外のビット誤り率は干渉波混入部分のビット誤り率に比べて十分小さく無視できるとし、ビット誤り率 P_e を

$$P_e = \frac{N_e}{N_r \times N_i \times N_b} \quad \dots (式 7 - 5 - 1)$$

但し、 N_e : 誤り発生回数

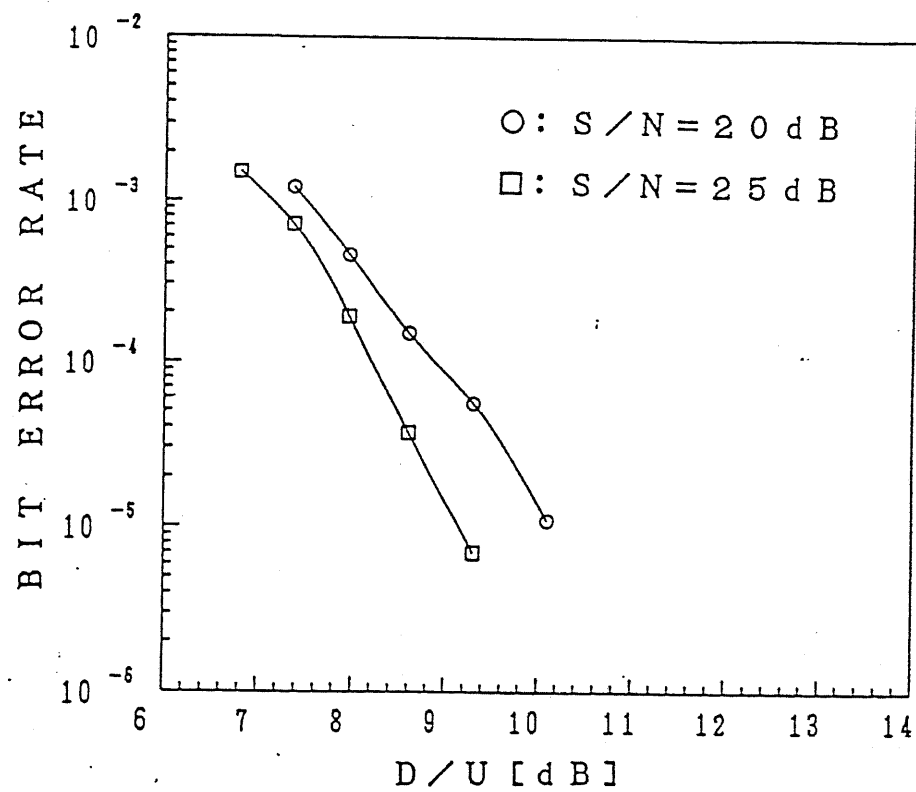
N_r : 伝送繰り返し回数

N_i : 干渉波混入部分のバイト数

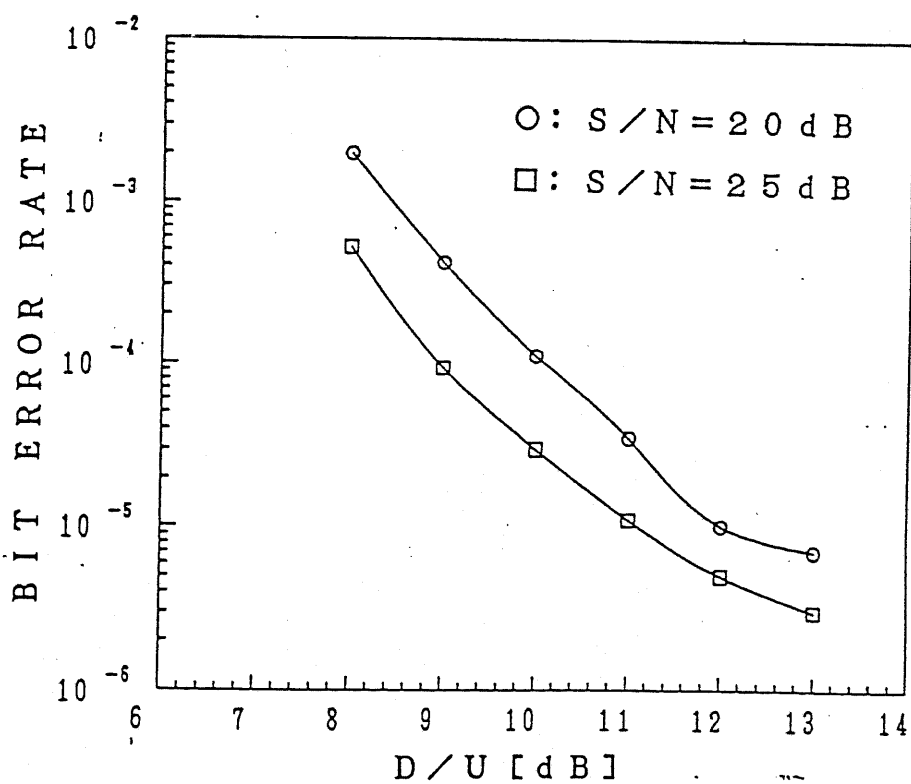
N_b : 調歩同期 1 バイトのビット数

によって求めた。

実験の結果を図 7 - 4 - 2 に示す。この結果よりビット誤り率 P_e を 10^{-5} 以下とするためには、 S/N が 20 dB の場合、 D/U をおよそ 12 dB 以上にする必要があることがわかる。ここで、実験システムを 1/50 のスケールモデルとみなすと模擬アンテナの受信電圧分布は図 7 - 4 - 3 に示すようになるので、こ



(a)車載局受信時



(b)地上局受信時

図 7 - 4 - 2 ビット誤り率の測定結果

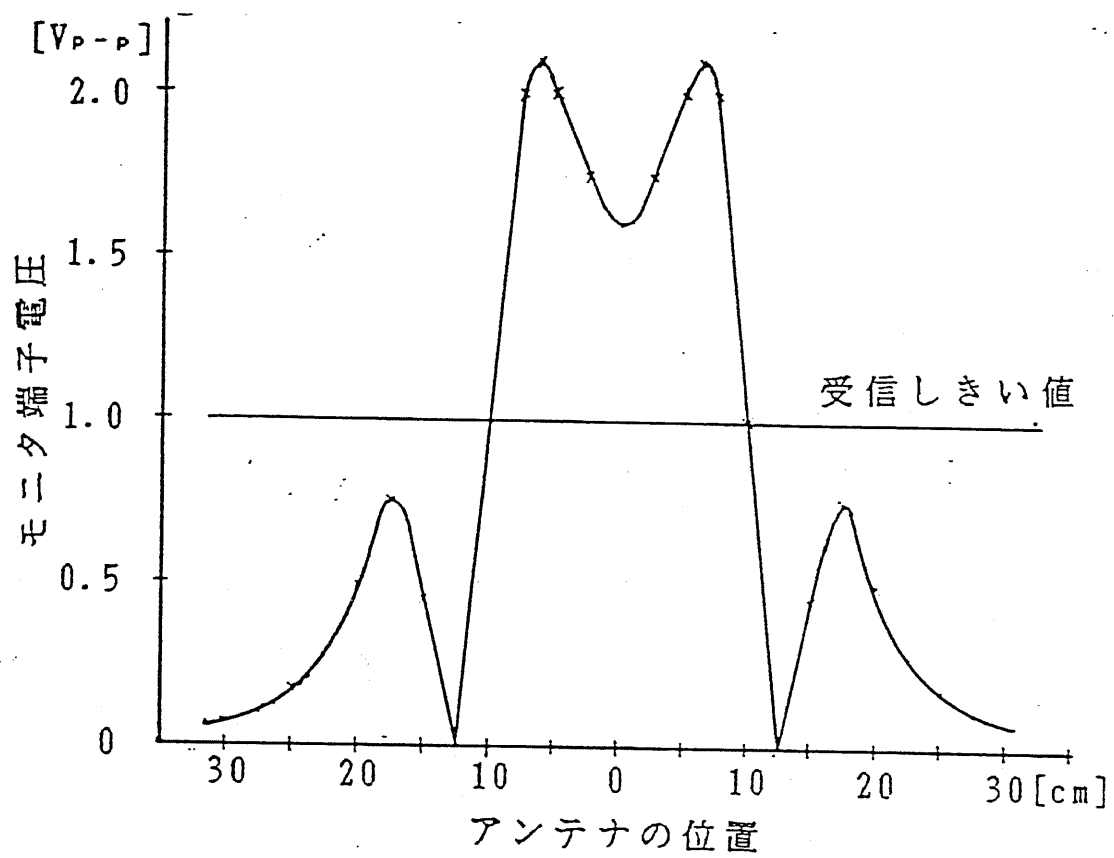


図 7 - 4 - 3 実験用地上アンテナの特性

れより通信ゾーン間隔はほぼ7 m以上必要になると考
えられる。

7. 5. 制御方式の動作確認と評価 [7][8][14][17][20]

7. 5. 1. 動作確認実験

ここでは、本実験システムにおいて制御プログラムを作成してインプレメントし、その動作の検証と性能評価実験を行った。その実験においては次のような方法を取った。まず、モデル実験システムの2つの通信ゾーンをリンク中間部の通信ゾーンと想定する。そして、自動車を2つの通信ゾーン間を往復運動させ、自動車が次に到着する通信ゾーンとして直前に存在した通信ゾーンと別の通信ゾーンを想定する。このようにすることにより、道路上に連続的に配置されている多数の通信ゾーンを自動車が走行し、これをシステムが追跡する形態を模擬した。ただし、ここでは希望する信号の消失などは必要に応じてソフトウェア的に設定している。

まず、走行データ通信の誤り回復動作の検証実験について述べる。ここでは作成した専用トリガによる走行データ通信の制御プログラムが所望の動作することを、各局の終了状態と、ロジックアナライザにより交

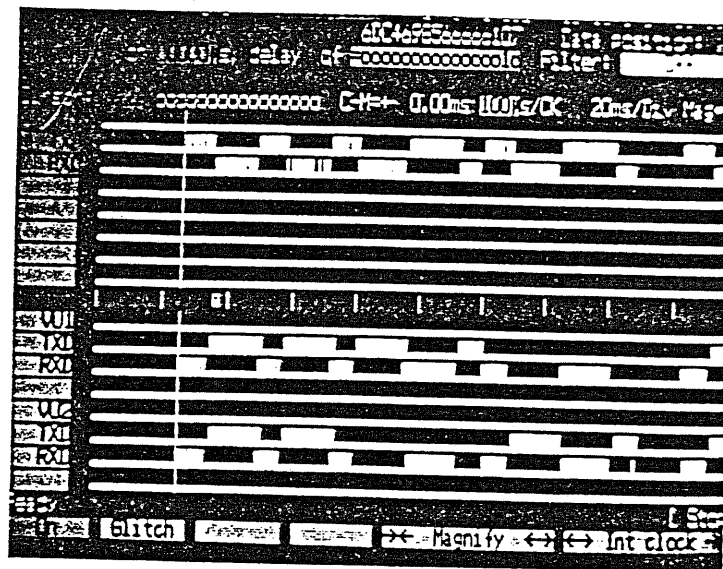
信シーケンスを観測することにより確認した。なお、この動作試験においては各信号が、正常受信、異常受信、消滅するようすを送信側でソフトウェア的に設定した。具体的には異常受信はデータを書き換えて送ることにより模擬し、消滅は信号を送信しないことにより模擬した。可能な全ての場合について、各局の終了状態を調べた結果を表7-5-1に示す。このように全て希望通りの結果となっている。

次に自動車の識別動作の確認を行う。ここではリンク入口の通信ゾーンに2台の自動車が同時に進入したときにコンテンションアクセスにより地上と交信が行われ、識別とトークンが付与されることを確かめた。この場合の交信シーケンスをロジックアナライザで観測した結果の例を図7-5-1(a)に示す。図7-5-1(b)は各信号の説明である。この例は通信要求の衝突を2度起こした後、車載局1、車載局2の順に交信が行われている様子を表している。

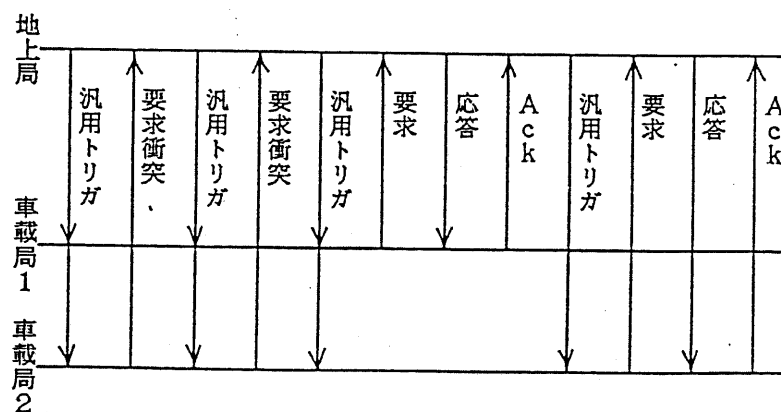
また、リンク中間部の通信ゾーンに2台の車載局が同時に進入する状況を想定して、2台の車載局と交信が行われることを確かめた。この場合の交信シーケン

表 7 - 5 - 1 誤り回復動作の試験結果

	設定						地上局終了状況	車載局終了状況
	地上局			車載局				
	トリガ	応答	再応答	要求	Ack	Nack		
1	正常	正常	—	正常	正常	—	正常終了	正常終了
2	正常	正常	—	正常	異常	—	確認異常受信	正常終了
3	正常	正常	—	正常	消滅	—	確認待ちタイムアウト	正常終了
4	正常	異常	正常	正常	正常	正常	正常終了	正常終了
5	正常	異常	正常	正常	異常	正常	確認異常受信	正常終了
6	正常	異常	正常	正常	消滅	正常	確認待ちタイムアウト	正常終了
7	正常	消滅	—	正常	—	—	確認待ちタイムアウト	応答待ちタイムアウト
8	正常	異常	異常	正常	—	正常	確認待ちタイムアウト	応答異常受信
9	正常	異常	消滅	正常	—	正常	確認待ちタイムアウト	応答待ちタイムアウト
10	正常	異常	—	正常	—	異常	確認異常受信	応答待ちタイムアウト
11	正常	異常	—	正常	—	消滅	確認待ちタイムアウト	応答待ちタイムアウト
12	正常	—	—	異常	—	—	要求異常受信	応答待ちタイムアウト
13	正常	—	—	消滅	—	—	要求待ちタイムアウト	応答待ちタイムアウト
14	異常	—	—	—	—	—	要求待ちタイムアウト	トリガ異常受信
15	消滅	—	—	—	—	—	要求待ちタイムアウト	終了せず



(a) 観測結果

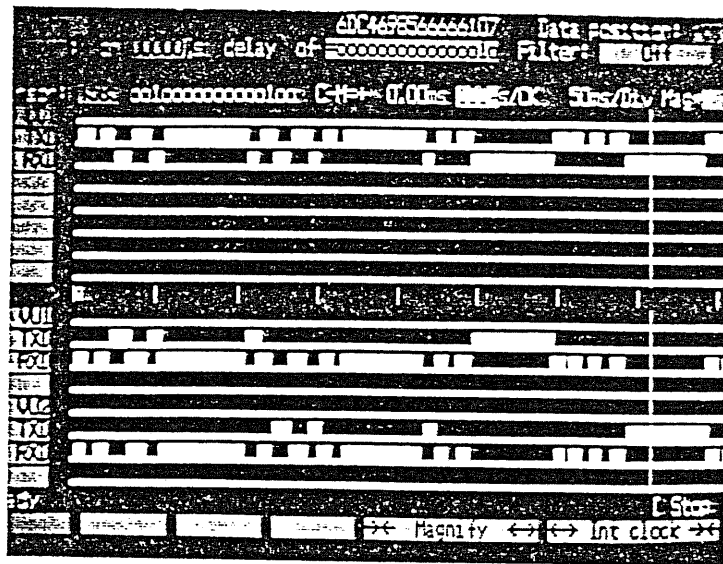


(b) 説明図

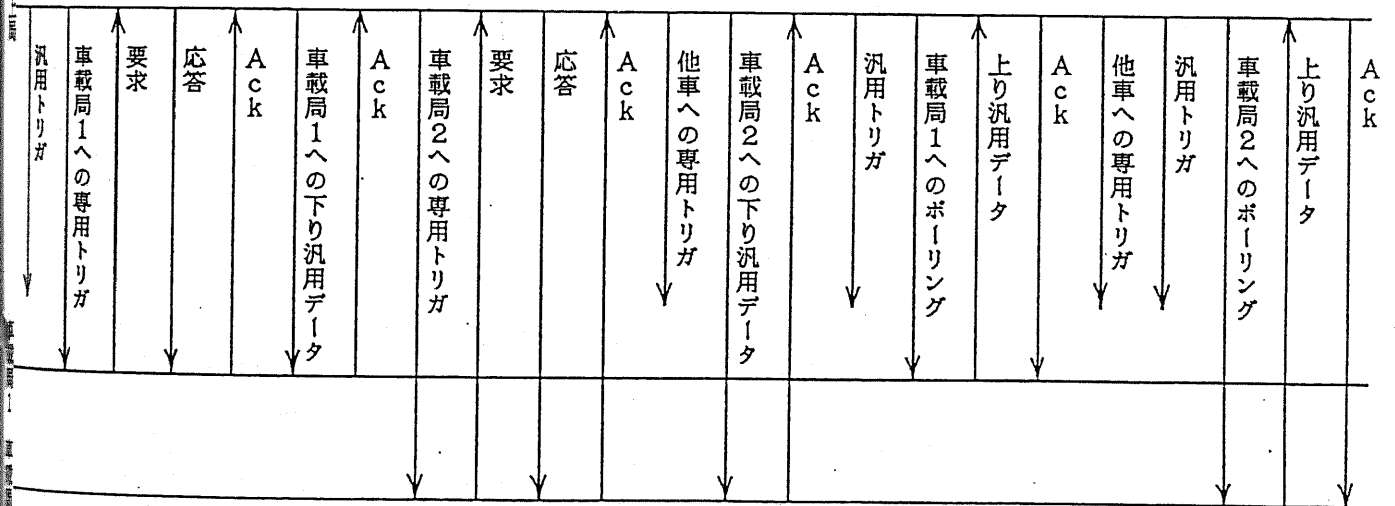
図 7-5-1 道路リンク入口部のゾーンでの交信シーケンス

スの観測結果の例を図7-5-2(a)に示す。図7-5-2(b)は各信号の説明である。これより、走行データ通信の専用トリガに対しては1台のみが要求信号を送出し、走行データ通信が1台ずつ行われていることがわかる。また、汎用データ通信についても同様に、各信号に対して2台の車載局が同時には信号を送出しないことがわかる。したがってトークンによる自動車の識別動作が正常に機能していると言える。

チャンネル割当の動作確認実験では、通信の対象となった自動車のトークンとその終了状態を示すコードおよびその終了時刻を実行の経過として地上局において記録して各装置の動作の確認を行った。走行データ通信・汎用データ通信への割当については、最低60msecに1回の割合で走行データ通信に割り当てられていることを確かめた。個々の自動車の走行データ通信への割当についてはサイクリックに行われていることなどを確かめた。汎用データ通信における各自動車への割当については、走行データ通信を終了した自動車が2台の場合においては交互に割り当てられていることなどを確かめた。



(a) 観測結果



(b) 説明図

図 7 - 5 - 2 道路リンク中間部のゾーンでの交信シーケンス

伝送パケット順序制御の動作確認実験では、車載局が通信ゾーンを移動したとき、下り汎用データ通信、上り汎用データ通信のデータパケットが重複なく順序通りに伝送されることの確認を行った。はじめに車載局または地上局からの確認信号が正常に送出されるように設定し、複数の下りまたは上り汎用データパケットの伝送中に車載局が通信ゾーンを数回移動したとき全ての下りまたは上り汎用データパケットが重複なく車載局に送られることを確認した。さらに車載局または地上局の確認信号が送出されないか、異常データを送出するように設定しても、次のゾーンにおける走行データ通信の中で再確認が行われることを確かめた。

7. 5. 2. 性能評価実験

○情報伝送効率

走行データ通信、下り汎用データ通信、上り汎用データ通信のそれぞれについて通信終了までの所要時間を測定し、情報伝送効率を計算した。正常に動作した場合の結果を表7-5-2に示す。この表の中で通信所要時間とは最初の信号の送信のキャリアが立ち上が

表 7 - 5 - 2 情報伝送効率の測定結果

			情報量 [byte]		75カ* CRC [byte]	通信 所要 時間 [ms]	情報 伝送 時間 [ms]	75カ* CRC 伝送 時間 [ms]	通信 処理 時間 [ms]	正味の 情報 伝送 効率 [%]	情報 伝送 効率 [%]
			走行 汎用	制御							
走行 データ 通信	要求	応答 6 byte	6	8	20	37	11.7	16.7	8.8	13.5	31.6
		5 byte 応答 7 byte	6	9	20	38	12.5	16.7	8.8	13.2	32.9
	要求	応答 6 byte	6	9	20	38	12.5	16.7	8.8	13.2	32.9
		6 byte 応答 7 byte	6	10	20	39	13.3	16.7	8.8	12.8	34.1
下り汎用データ通信			50	4	10	58	45.0	8.3	4.4	71.9	77.6
上り 汎用データ 通信	途中データ		50	6	10	59	46.7	8.3	4.4	70.6	79.1
	最終データ		50	8	15	67	48.3	12.5	6.6	62.2	72.1

ってから最後の送受信のキャリアが立ち下がるまでの時間をロジックアナライザにより測定したものである。ここで情報量には正味の情報である走行情報、汎用情報と、信号種別、トークン、汎用データ通信制御情報などを含んでいる。このように制御情報を含めた情報伝送効率の代表的な値が走行データ通信で32%、下り汎用データ通信で78%、上り汎用データ通信で79%である。走行データ通信は伝送情報量がわずかなため効率は低い、汎用データ通信は十分な効率が得られたと考えられる。

○ 処理時間の測定

通信制御装置においてある時点で送受信を行っている通信制御の各ルーチンから復帰して、次の信号の送信の準備が終わるまでの内部処理に要する時間を、復帰の状態と次の送出信号の組合せの中で代表的な場合について内蔵タイマにより測定した。この結果、内部処理に要する時間は路側装置への情報の転送が行われる場合は1～2 msecであり、転送データがないときは常に1 msec以下であることがわかった。一方、車載局

の処理時間は、いずれの場合も 1 msec以下であった。

以上の結果より、これらの処理時間は全体の通信の所要時間に比べて十分小さく、情報伝送効率にはほとんど影響しないと考えられる。

7. 6. ハイブリッドシミュレーション

7. 6. 1. ハイブリッドシミュレーションの概念

これまで述べたように本研究では小ゾーン連続形自動車パケット通信システムの実現可能性を確認するために1道路リンクを模擬した小規模のモデル実験システムを構築した。また、通信制御方式についてはコンピュータによるソフトウェアシミュレーションを行ってその有効性を示してきた。次の段階として、このハードウェアに関して実際の車両の移動に合わせてリアルタイムで性能評価を行うことが必要となる。しかしながら通常のフィールド実験による評価を行うためには実車に搭載してテストコースなどで走行実験を行うことになり、膨大なコストが必要となる。

ここではソフトウェアによる交通流シミュレータと通信システムのハードウェアを融合したハイブリッドシミュレーションを行うことを考える。具体的には、図7-6-1に示したように、ソフトウェアシミュレータにより各自動車の道路上における移動を模擬し、それを基に通信ゾーンへの進入退出を示すタイミング信号を発生する。通信システムのハードウェア側では

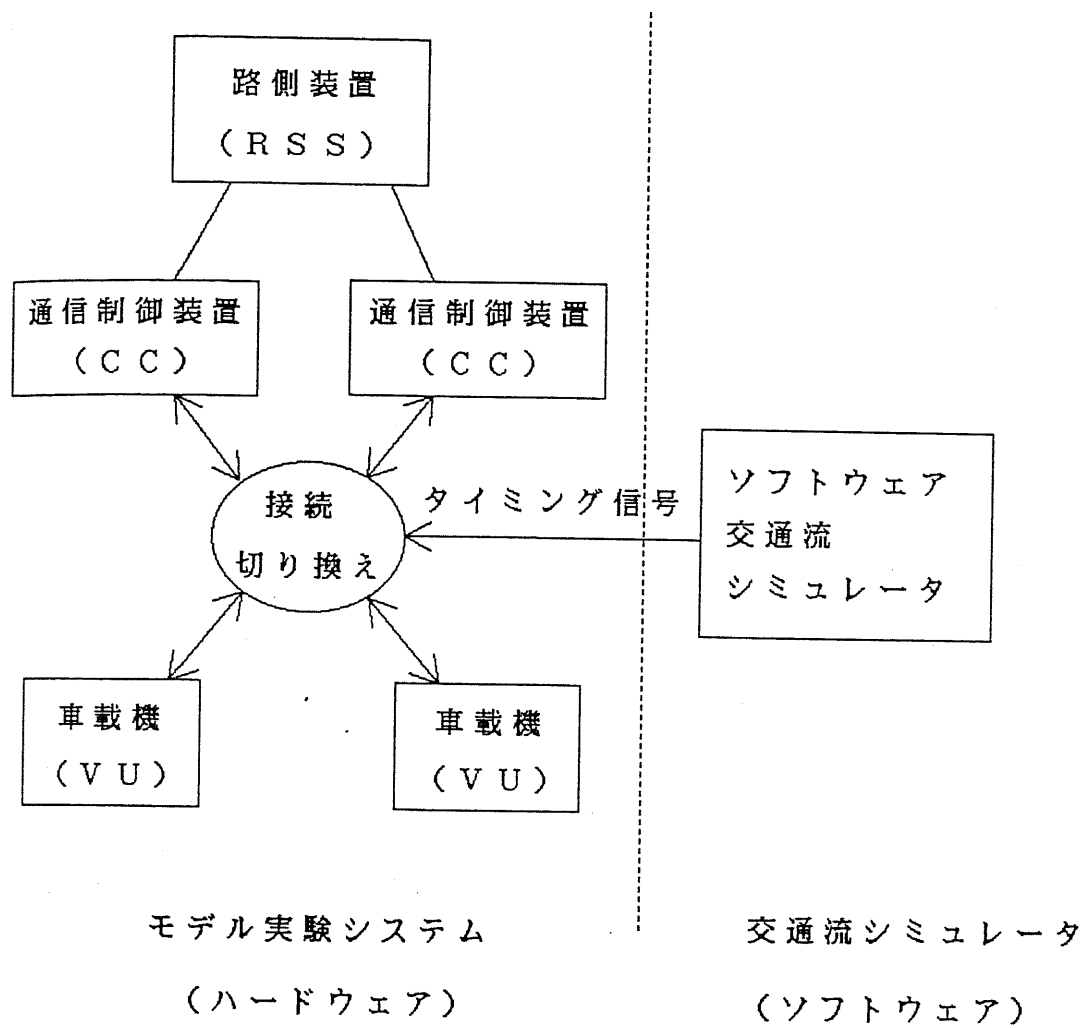


図 7 - 6 - 1 ハイブリッドシミュレーションの概念

この信号に基づいて通信制御装置（C C）と車載機（V U）の間の接続をハードウェア的に切り換えてやることにより自動車は現場と同様の速度で通信ゾーン上を通過してゆく様子が模擬できる。

7. 6. 2. シミュレータ装置構成

ここでは、以下の2通りの実験を行うことを想定してハイブリッドシミュレータを考える。

（実験1）

図7-6-2に示したように、1つの通信ゾーンと2つの車載機により2台の車両が任意の時間差・速度を持って通信ゾーンを通過する状況を模擬する。

（実験2）

図7-6-3に示したように、2つの通信ゾーンと1つの車載機により、車両が2つの通信ゾーンにまたがって通信を行ってゆく際に車両を追跡してデータ伝送を行う。

これにより自動車が任意の速度で通信ゾーンを通過する際にどの程度のデータが伝送可能かを調べる。

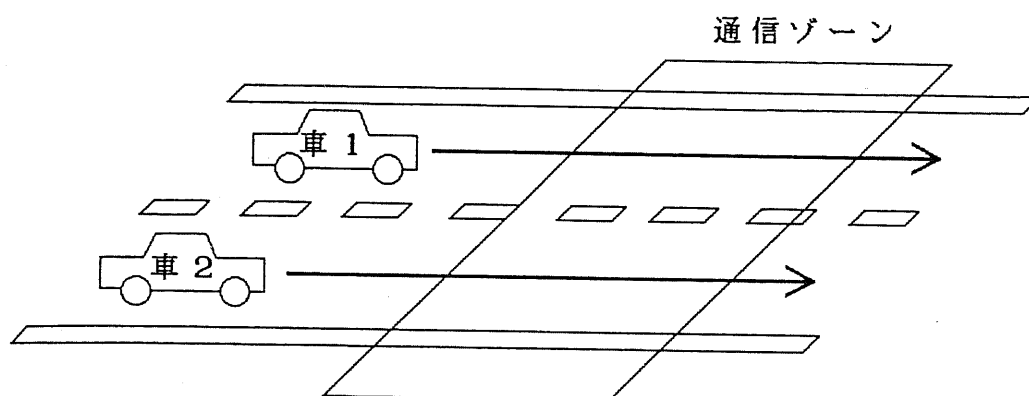


図 7 - 6 - 2 車両の進入実験

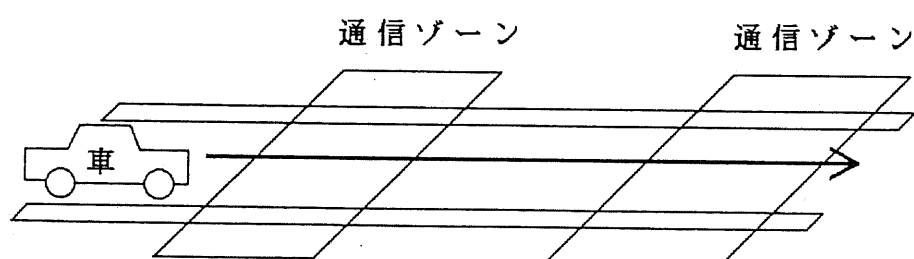


図 7 - 6 - 3 車両の追跡実験

このための、ハイブリッドシミュレータの装置構成を図7-6-4に示す。このように本装置では、交通流シミュレータとしてサンマイクロシステムズsun S3/50ワークステーションを用いる。この上でC言語により交通流シミュレータのソフトウェアを作成した。また、交通流シミュレータからの制御信号はRS232Cのシリアルインタフェースポート(2ポート実装)より取り出している。

交通流シミュレータにおいては、自動車は2つの実験のいずれにおいてもキーボードより設定した一定速度で移動するものとした。シミュレータの機能としては、デモンストレーション用の目的から自動車の動きを50ms毎にリアルタイムでグラフィック表示するモード1と表示を行わず内蔵タイマを用いてイベントスキャンにより精度1msでタイミング信号を発生できるモード2の2通りの動作が行えるようにした。

(実験1)

まず実験1のためのシステムを図7-6-5に示す。2台の車載機のアンテナを1つの路上アンテナと結合

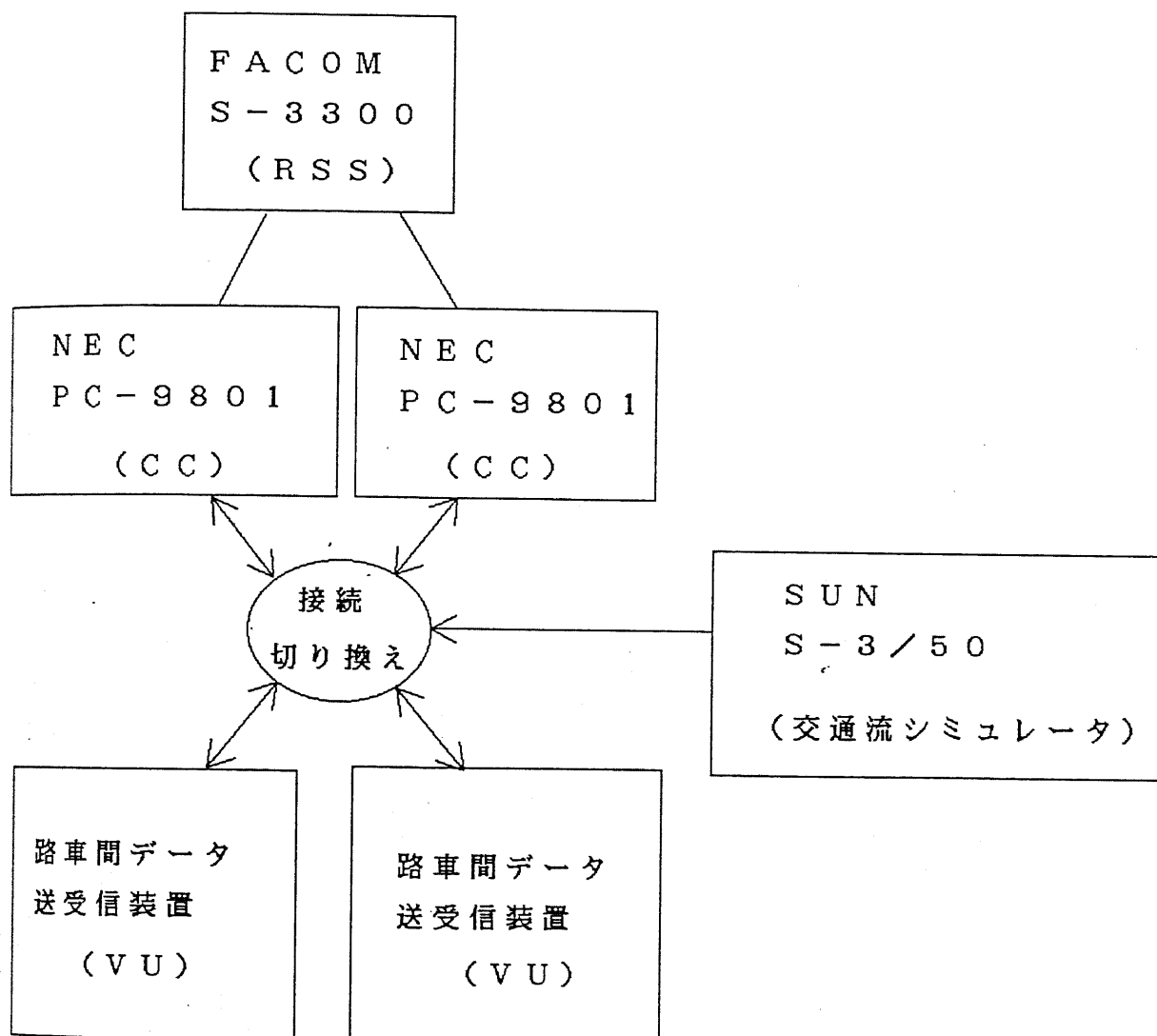


図 7-6-4 ハイブリッドシミュレータの装置構成

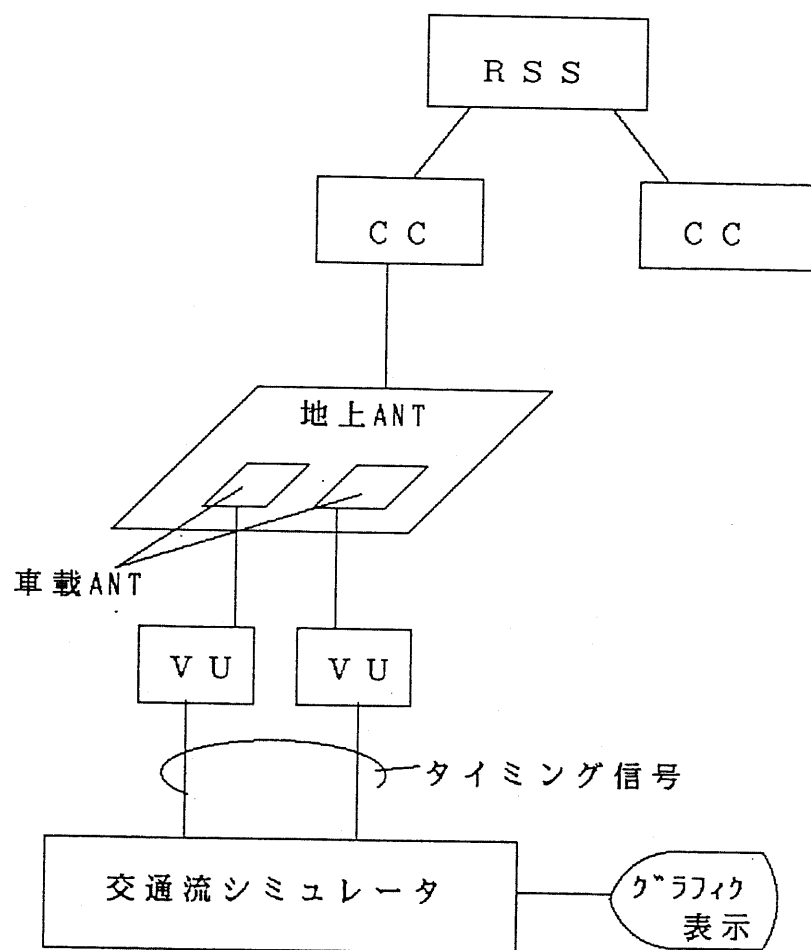


図 7 - 6 - 5 車両の進入実験のシステム

させておく。そして交通流シミュレータにおいて2台の車載機の通信ゾーンへの進入・退出のタイミングを示す信号を発生し、2台の車載機の通信制御用のLSIをON-OFFする。これにより、2台の自動車が同一の通信ゾーンを通過する状況を模擬する。

この時、交通流シミュレータよりVUへ転送される信号の内容と情報量を表7-6-1に示す。このように、進入信号においては、交通流シミュレータ側のキーボードより設定した、VUから地上側へ送出する上り汎用データの要求パケット数に関する情報を併せて伝送し、VUをON状態にすることとしている。そして、交通流シミュレータより退出信号が送出されるとVUはOFF状態となり、実行結果の保存と表示を行う。尚、VUと交通流シミュレータの間の信号は9600bpsの調歩同期により伝送されている。

尚、この実験のためのプログラムを付録A. 4. 1. に示す。

(実験2)

次に実験2のためのシステムを図7-6-6に示す。

表 7 - 6 - 1 実験 1 の信号

伝送方向	信号種別	内容	情報量
交通流 シミュレータ ↓ V U	進入信号	識別子	3 [byte]
		上り汎用データ 要求パケット数	
		終了キャラクタ	
	退出信号	識別子	2 [byte]
		終了キャラクタ	

表 7 - 6 - 2 実験 2 の信号

伝送方向	信号種別	内容	情報量
交通流 シミュレータ ↓ V U	進入信号	識別子	3 [byte]
		上り汎用データ 要求パケット数	
		終了キャラクタ	
	退出信号	識別子	2 [byte]
		終了キャラクタ	
前ゾーン 設置用 V U ↓ 後ゾーン 設置用 V U	引継信号	識別子	6 [byte]
		上り汎用データ 要求パケット数	
		上り汎用データ 残りパケット数	
		下り汎用データ 残りパケット数	
		上り汎用データ A C K 受信判断	
		終了キャラクタ	

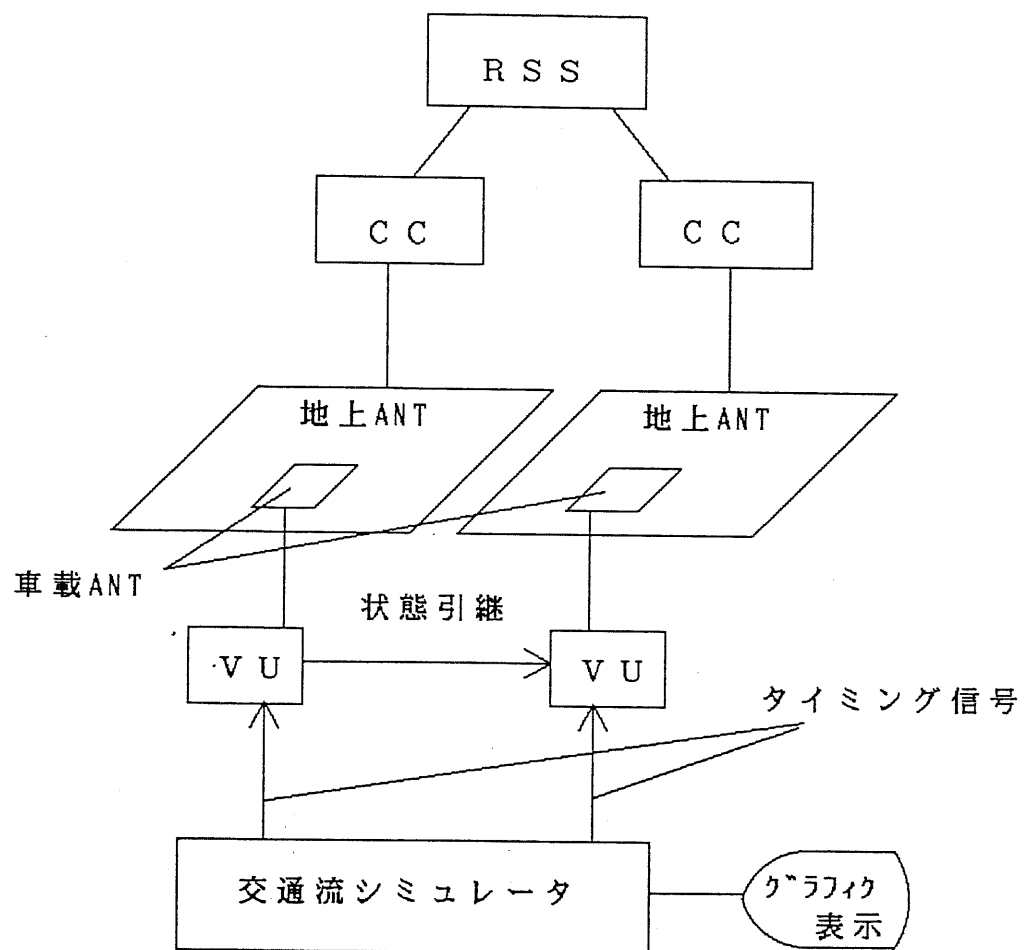


図 7 - 6 - 6 車両の追跡実験のシステム

2 台の車載機を同一のものと見なし、各々のアンテナを別々の地上アンテナと結合させておく。そして自動車は前の通信ゾーンより出たときにその終了状態を車載機間の結線によって引き継いでやり、また、交通流シミュレータからのタイミング信号に基づいて2台の車載機の通信制御用LSIのON-OFFを行う。これにより自動車が2つの通信ゾーンにまたがって移動する様子を模擬する。

この時各装置間で伝送される信号を表7-6-2にまとめた。交通流シミュレータとVUの間の信号に関しては実験1の場合と同様である。前側の通信ゾーンに設置されているVUより後側の通信ゾーンに設置されているVUへの引継信号としては、自動車に割り当てられたトークン、上り汎用データに関する情報、下り汎用データに関する情報を含ませる。交通流シミュレータからの退出信号を受けた前側の通信ゾーンに設置されているVUがこれらの引継情報を後側の通信ゾーンに設置されているVUに伝送すると、後側の通信ゾーンに設置されているVUは交通流シミュレータからの進入信号待の状態となる。そして、それを受信す

るとこの V U が O N 状態となり通信制御の動作を開始する。

尚、この実験のプログラムを付録の A. 4. 2. に示す。

7. 6. 3. 実験結果

ここでは、上りまたは下りデータのいずれかの伝送に関して、自動車側または地上側が 5 0 0 bit のパケット 2 0 個を伝送しようとする場合を想定して実験を行った。なお、各実験においてはシミュレータの動作モードは時間の正確さを保つためにモード 2 に設定して行った。

(実験 1)

実験 1 では、

(A) 1 台の自動車のみが通信ゾーンを通過したとき

(B) 2 台の自動車と同時に通信ゾーンを通過したとき

(C) 2 台の自動車 が車頭間隔 5 m で通信ゾーンを通過したとき

の3通りの場合について、各自動車の通信可能データ量を求めた。下りデータ及び上りデータの要求があるときに、自動車の速度をいろいろに変えて、通信完了したパケット数を測定したものが各々図7-6-7（上り・進入差0m）、図7-6-8（下り・進入差0m）、及び、図7-6-9（上り・進入差5m）、図7-6-10（下り・進入差5m）である。

当然のことであるが、自動車の速度が高くなると通信可能なデータ量が少なくなる傾向にある。データ量の具体的な値としては、自動車が1台の場合では自動車の速度が40km/hで7～9パケット、80km/hでは4～3パケットとほぼ速度に反比例する。また、自動車が2台の場合についてみれば、進入時間差が無い場合は1台の場合のほぼ半分になっていることがわかる。進入時間差がある場合では無い場合に比較して、伝送可能なデータ量が多くなっている。これは双方の自動車が通信ゾーンに存在する時間がずれるために各々が単独で通信できる時間があるためと考えられる。

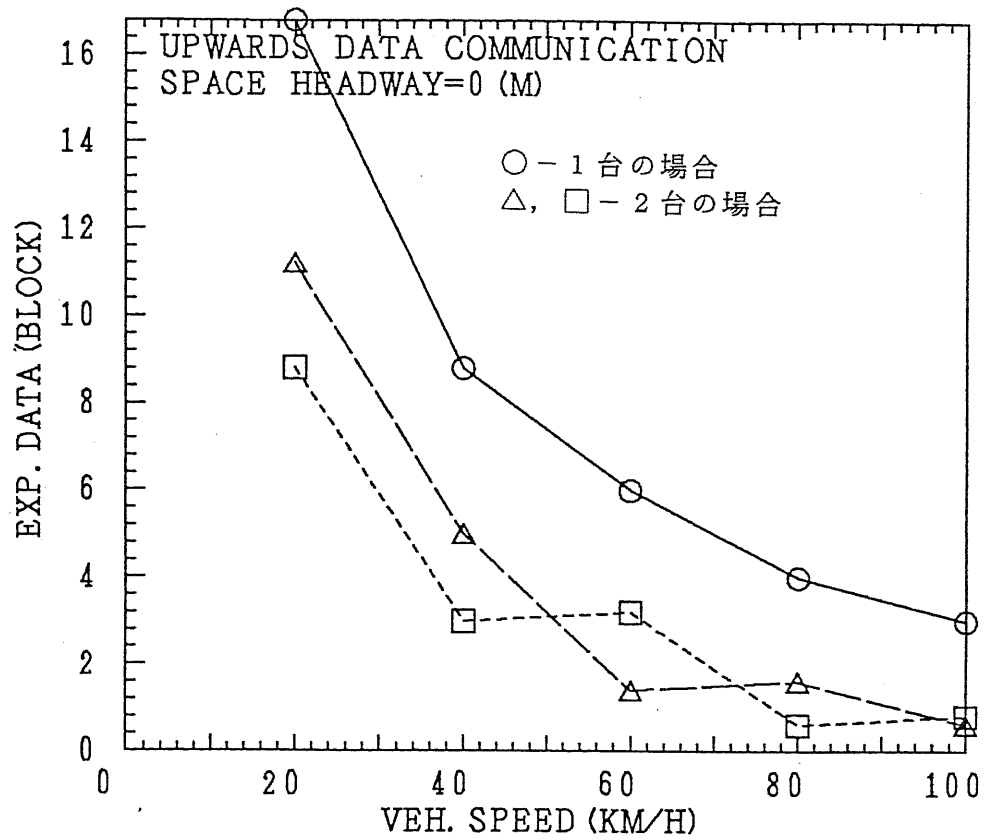


図 7 - 6 - 7 実験 1 の結果 (上り, 進入時間差無し)

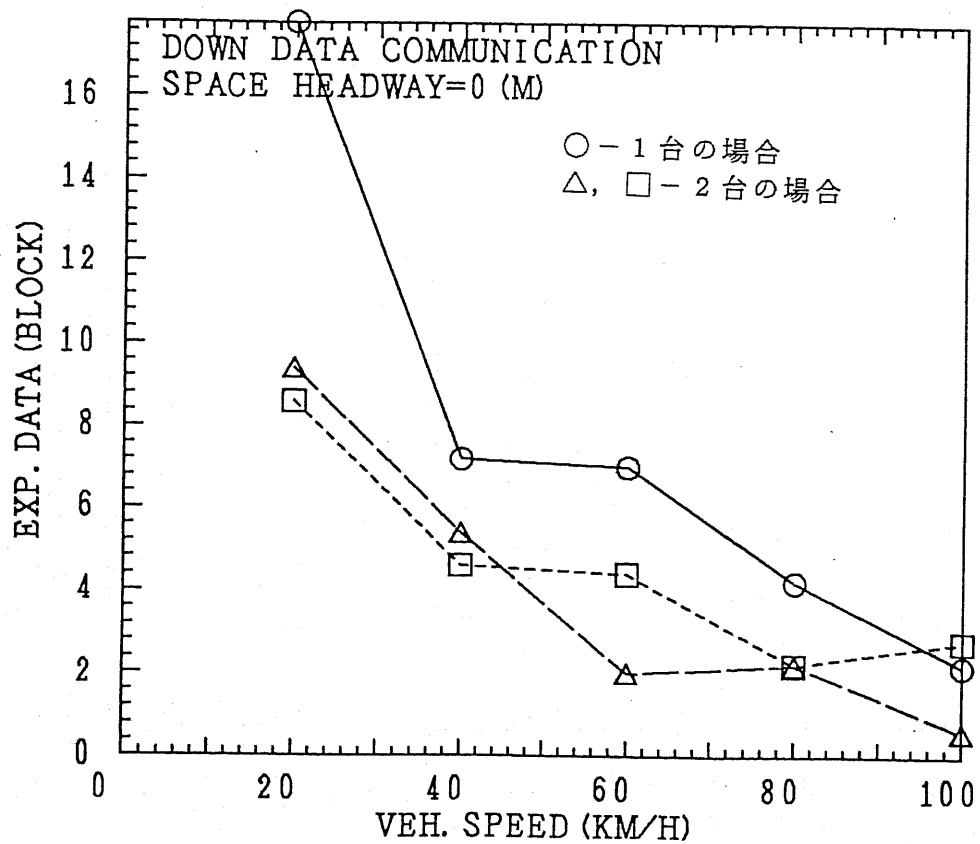


図 7 - 6 - 8 実験 1 の結果 (下り, 進入時間差無し)

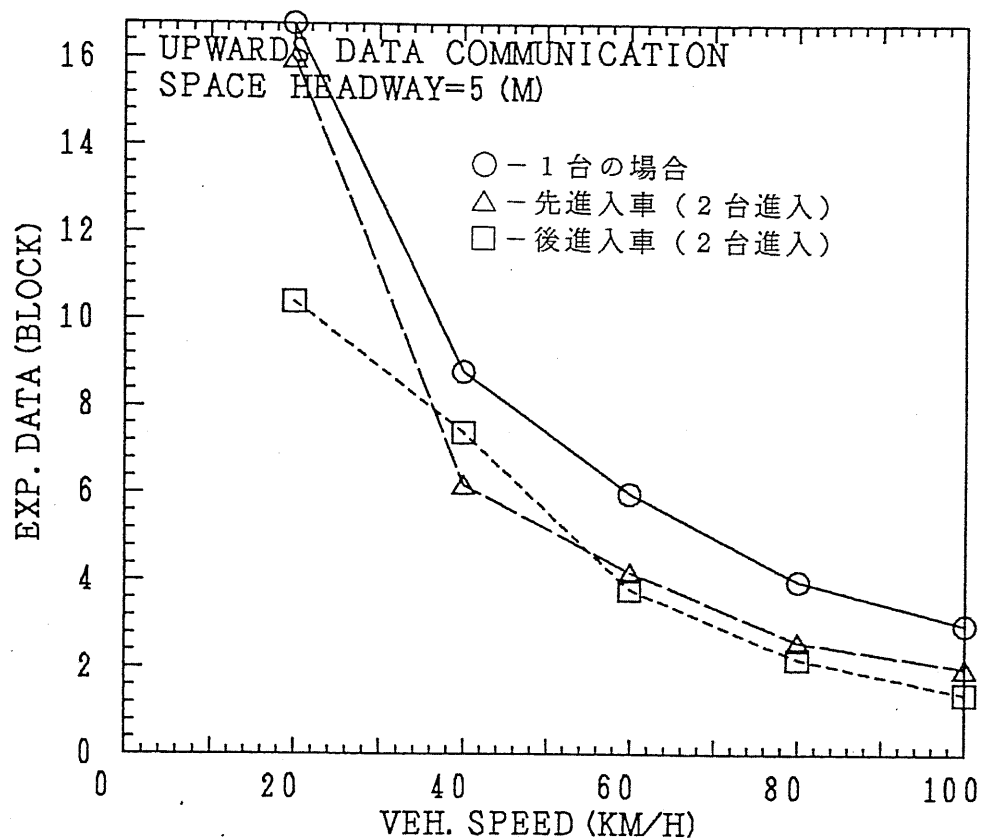


図 7 - 6 - 9 実験 1 の結果 (上り, 進入時間差 5 m)

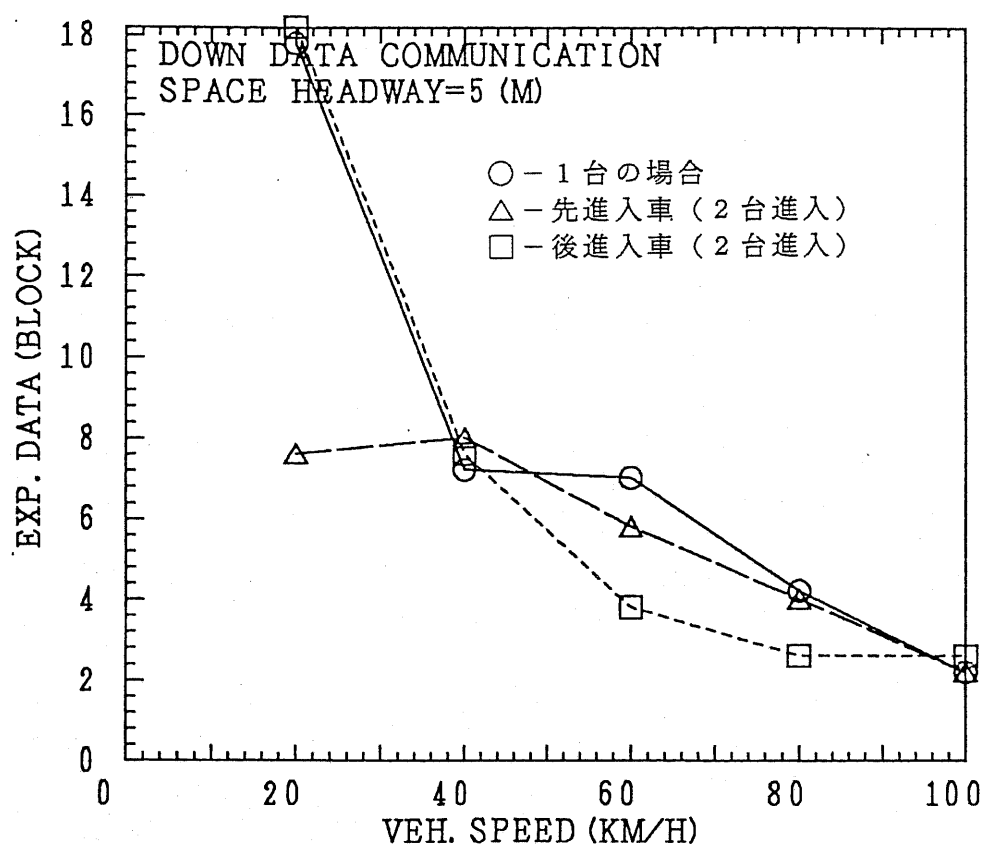


図 7 - 6 - 10 実験 1 の結果 (下り, 進入時間差 5 m)

(実験 2)

実験 2 では 20 個の packets からなる上りデータまたは下りデータの伝送要求を持つ 1 台の車載機が 2 つの通信ゾーンを種々の速度で通過するときに各通信ゾーンで通信できた packets 数を求めた。

図 7 - 6 - 1 1 は上りデータについて、図 7 - 6 - 1 2 は下りデータについて、各々の通信ゾーンで通信できた packets 数と 2 つの通信ゾーンでのトータルの伝送 packets 数を求めたものである。

この場合も、自動車の速度にほぼ反比例して伝送可能な packets 数は減少する。しかしながら自動車が 100 km / h 程度の高速で走行しても、これを追跡して通信が継続されている。なお、速度が 20 km / h の場合に後の通信ゾーンでの通信データ量が少なくなっているのは通信要求データ数を 20 個に設定しているためであり、実際にはより多くのデータが伝送できるものと考えられる。

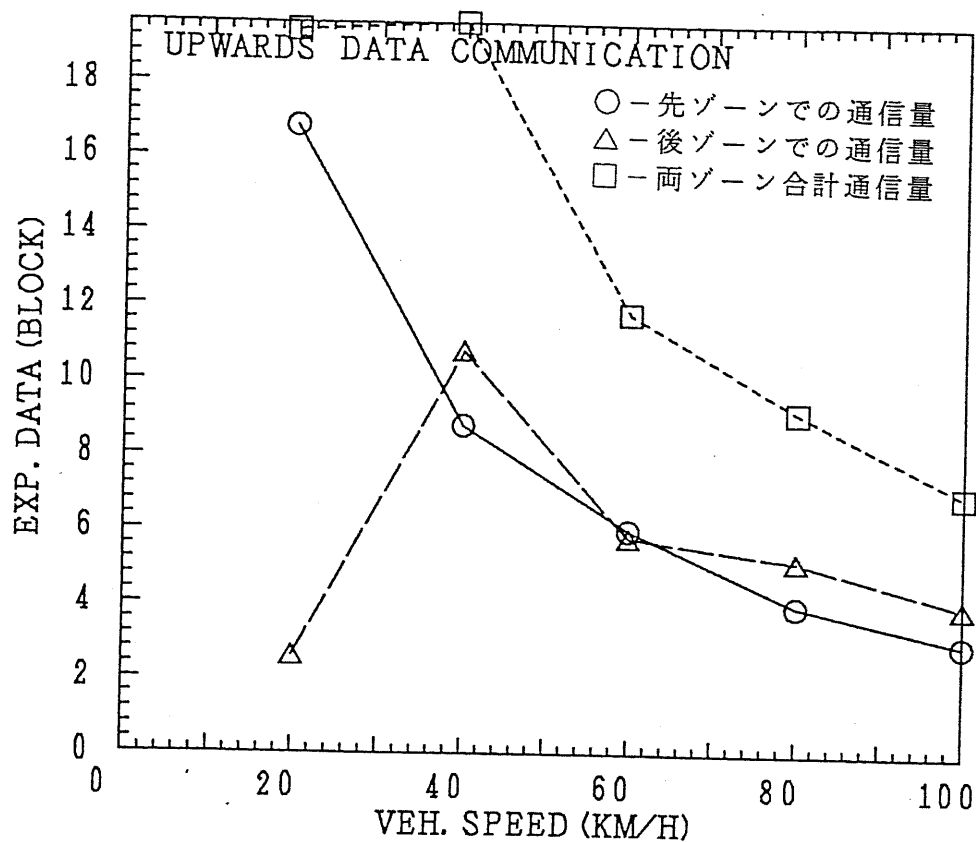


図 7 - 6 - 1 1 実験 2 の結果 (上り)

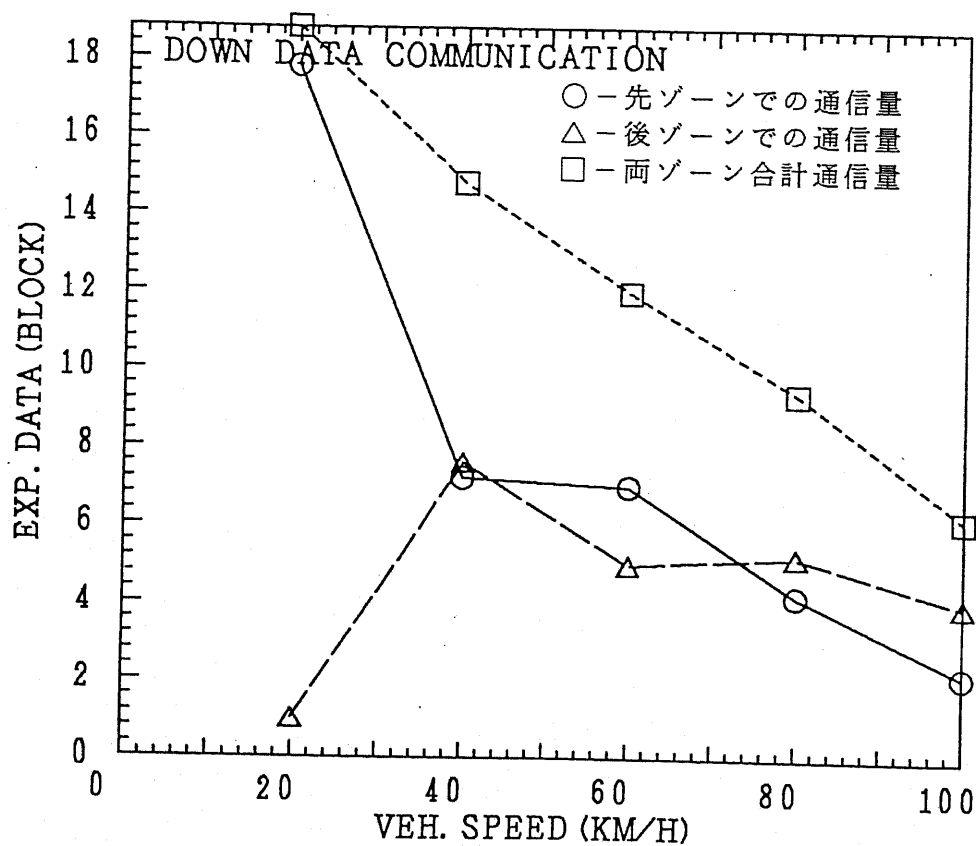


図 7 - 6 - 1 2 実験 2 の結果 (下り)

7. 7. 結 言

本章では、小ゾーン連続形パケット通信システムの通信制御方式においての特徴的な部分ともいえる道路リンクレベル通信制御方式の実現可能性を示すためにモデル実験システムの構築を行って通信制御のプログラムを実際にインプレメントし、動作確認や性能評価のための実験を行った。この中では、ソフトウェアにより構成した交通流シミュレータと組み合わせたハイブリッドシミュレーションによる実際の自動車の移動速度に合わせたリアルタイムでの性能評価試験も含まれている。これらの実験の結果より、当初の目標であった道路リンクレベルでのきめ細かな通信制御の実現可能性を示すことができたといえる。

8 . 結 び

8. 1. 研究成果

本論文では、将来における理想的な自動車情報通信システムの一つのあり方として、新たなコンセプトを持つ小ゾーン連続形自動車パケット通信システムを提案した。ここでは、まずその概念を示すとともに、通信ゾーンの構成に関して理論的検討を行い、実現可能と考えられる構成方法を示した。また、その通信制御方式に関しては、自動車の走行に関する情報を利用することにより通信制御方式の効率化を行うという考え方を導入し、その有効性をミクロモデルを用いた評価シミュレーションにより示した。

さらに本システムの交通管理への応用として、走行誘導の手法を提案し、これに関しても、ミクロモデルを用いた評価シミュレーションを行って、その効果を示し、加えて、本システムの情報通信システムとしての各種の利用形態の例を紹介した。

また、本研究ではスケールモデルを用いた小規模のモデル実験システムを構築して、室内実験を行うこと

により、本システムの道路リンクレベルでの通信制御方式の実現可能性をより具体的に示した。

以上、本論文では小ゾーン連続形自動車パケット通信システムの有効性を理論と実験の両面から証明することができたといえる。

8. 2. 今後の課題

第5章及び、第6章にも述べたように、本システムの導入による交通管理や自動車情報通信におけるメリットは大きいと考えられる。しかし、5. 5. にも述べたように本システムで考えるような走行誘導の手法が社会的に容認されるためには、人間工学的により詳しい検討を行ってゆくことが必要となろう。

また、本システムでは道路リンクレベルの通信制御方式に関してはスケールモデルによる実験により実現可能性を示すことができたが、グローバルレベルの通信制御方式では理論的検討にとどまっており、これに関しては更に現実的な検討が必要と考えられる。

最後に、このようなシステムでは利用者が多くなればなるほどシステムとしてのメリットが大きくなると考えられる。従って、本システムにおいてもそのアプリケーションの充実を図り、利用者側からみてより一層魅力溢れるものとしてゆく必要があるといえる。

◎ 謝 辞

5 年間にわたる本研究に当り、終始厳しく見守りつつ、親切かつ丁寧な御指導を頂いたのみでなく一個の人間としての心構えを様々な機会にお教え頂いた指導教官の高羽禎雄教授に心より深く感謝致します。

また、モデル実験システムの構築に関して多大なる貢献をして下さった関根富美、浜辺孝二郎、松浦慶典各氏、及び、走行誘導への応用面において多大なる貢献をして下さった西村健、森崎泰史両氏に御礼申し上げます。

その他、公私両面にわたってお世話頂いた研究室の諸氏、短期職員の方々にも謝意を表します。

◎参考文献

A) 著者と関係のある文献

A-1) 発表文献

- [1]酒井, 高羽: "小ゾーン連続形自動車パケット通信システムの接続制御方式", 信学論(B), Vol. J71-B, No. 3, pp. 383-390, 1988年3月.
- [2]S. SAKAI, K. NISHIMURA and S. TAKABA: "Mobile Packet Communication System Using Continuously Allocated Small Zones and Movement Control of Automobiles", Proceedings of The European Simulation Multiconference (Simulation in Traffic Control) June, 1988.
- [3]S. SAKAI and S. TAKABA: "Communication Protocol for Vehicle Packet Communication System Using Continuously Allocated Small Zones", Electronics and Communications in Japan. (掲載予定)
- [4]酒井, 高羽: "小ゾーン連続形自動車通信システムにおける評価シミュレーション", 日本シミュレーション学会第6回シミュレーションテクノロジー・コンファレンス, 9-5., 1987年6月.
- [5]酒井, 高羽: "小ゾーン連続形自動車パケット通信システムの情報伝送方式における評価シミュレーション", 日本シミュレーション学会第7回シミュレーションテクノロジー・コンファレンス, 12-4., 1988年6月.
- [6]酒井, 西村, 森崎, 高羽: "自動車の走行誘導とその小ゾーン連続形自動車パケット通信システムによる実現", 日本シミュレーション学会第8回シミュレーションテクノロジー・コンファレンス, 11-4., 1989年6月.
- [7]浜辺, 関根, 酒井, 高羽: "小ゾーン連続形自動車パケット通信システムのリンクレベル制御", 日本シミュレーション学会第8回シミュレーションテクノロジー・コンファレンス, 11-3., 1989年6月.
- [8]関根, 浜辺, 酒井, 高羽: "小ゾーン連続形自動車パケット通信システムのモデル実験", 電気学会道路交通研究会, RTA-89-15, 1989年12月.
- [9]酒井, 滝田, 高羽: "小ゾーン連続形自動車通信方式の一考察", 昭和6

1年度電子通信学会通信部門全国大会, 72. 1986年9月.

[10]酒井, 高羽: "小ゾーン連続形自動車通信システムにおける接続制御方式", 昭和62年電気学会全国大会, 462. 1987年4月.

[11]酒井, 高羽: "小ゾーン連続形自動車パケット通信システムにおける情報伝送方式", 昭和63年電子情報通信学会春季全国大会シンポジウム講演, SB-6-5. 1988年3月.

[12]関根, 酒井, 高羽: "小ゾーン連続形自動車パケット通信システムのモデル実験", 昭和63年電気学会全国大会, 1574. 1988年3月.

[13]酒井, 西村, 森崎, 高羽: "小ゾーン連続形自動車パケット通信システムによる右折車の待合せ制御", 平成元年電気学会全国大会, 1750. 1989年4月.

[14]関根, 浜辺, 酒井, 高羽: "小ゾーン連続形自動車パケット通信システムーモデル実験システムの構築ー", 平成元年電気学会全国大会, 1751. 1989年4月.

A-2) 学内文献

[15]酒井: "小ゾーン連続形自動車通信システム", 東京大学大学院修士論文, 昭和62年3月.

A-3) その他, 関係のある文献

[16]高羽: "小ゾーン連続形自動車パケット通信システム", OHM, Vol. 75, No. 9, pp. 84, 1988.

[17]高羽: "極小ゾーン縦続配置形自動車通信システムの研究", 昭和63年度科学研究費補助金(一般研究(B)) 研究成果報告書, 平成元年度3月.

[18]西村, 高羽: "走行誘導による高速道路の分流制御", 昭和63年電気学会全国大会, 1572.

[19]西村: "自動車の走行誘導による交通流制御", 東京大学大学院修士論文, 昭和63年3月.

[20]浜辺: "小ゾーン連続形自動車パケット通信システムのリンクレベル制

御”，東京大学大学院修士論文，平成元年3月。

[21]森崎：“小ゾーン連続形自動車パケット通信システムを用いた走行誘導”，千葉工業大学卒業論文，平成元年3月。

[22]松浦：“小ゾーン連続形自動車パケット通信システムの通信制御”，千葉工業大学卒業論文，平成元年3月。

B) 一般文献

B-1) 交通工学に関する文献

- [23]高羽: "交通管制機器の最近の進展", 信学誌, vol. 66, 6, 昭和58年6月.
- [24]高羽: "交通管制の新しい動向", 自動車技術, vol. 40, No. 5, 1986.
- [25]高羽: "交通管制 第2偏."
- [26]高羽, 小石: "高速道路交通流の実時間予測シミュレーションについて", 信学技報, SANE79-43, 昭和55年2月.
- [27]Bruce Douglas Greenshields, Frank Mark: "道路交通解析者のための統計学", 1978.
- [28]交通工学研究会: "交通工学ハンドブック", 1984.
- [29]交通工学研究会: "道路交通データブック", 昭和51年2月.
- [30]半場: "微視的交通流シミュレータ及びその応用に関する研究", 昭和52年6月.
- [31]田中, 渡辺, 重光: "自動車における車頭間隔の観測結果について", 運輸技研資料, No. 56, 一般-9, 昭和38年3月.

B-2) 自動車通信システムに関する文献

- [32]高羽: "自動車交通と通信技術", 自動車技術, vol. 38, No. 5, 1984.
- [33]高羽: "高度情報化と移動通信", 信学誌, vol. 68, 11, 昭和60年11月.
- [34]高羽: "ニューメディアと道路交通情報," 第35会・第36会交通工学講習会テキスト.
- [35]今村, 高羽: "情報の優先度に基づいてチャネル選択を行う自動車群と地上との間の通信方式", 昭和49年度信学全大, 2350.
- [36]佐々木, 高羽: "各種の自動車群-地上間通信方式の比較検討": 昭和51年度信学全大, 1455.
- [37]DAN A. ROSEN, FRANK J. MAMMANO, RINALDO FAVOUT: "An Electric Route-

Guidance System for Highway Vehicles", IEEE, Trans. on V.T., vol.VT-19, No.1, FEBURARY 1970.

[38]通産省工業技術院："自動車総合管制技術の研究開発（大型プロジェクト制度による7ヶ年間の研究成果）", 昭和54年9月.

[39]P.braegas:"Function, Equipment, and Field Testing of a Route Guidance and information System for Drives(ALI)," IEEE Trans.Veh.Tech nol., VT-29, No. 2, pp.216, 1980.

[40]自動車走行電子技術協会："路車間通信の共通仕様に基づく車載機・路上機の試作研究", 昭和58年3月.

[41]自動車走行電子技術協会："路車間通信の共通仕様に基づく車載機・路上機の試作研究", 昭和59年3月.

[42]自動車走行電子技術協会："自動車関連情報システムの高度化に関する研究 -路車間局地デジタル通信の利用-", 昭和61年3月.

[43]自動車走行電子技術協会："つくば自動車データ収集・提供システムとその実験 -新しい路車間通信仕様の応用-", 昭和61年3月.

[44]道路新産業開発機構："路車間情報システム研究会 ビーコン部会中間報告書", 昭和61年3月.

[45]自動車走行電子技術協会："-自動車の社会的効用の向上に関する調査研究報告書- 自動車における通信情報システムに関する調査研究", 昭和60年3月.

[46]自動車走行電子技術協会："自動車における情報ネットワークシステムに関する調査研究", 昭和61年3月.

[47]藤井："自動車のための総合情報システムを目指して", 機械振興, 22, 4, pp.17-22, 平成元年6月.

[48]M. Shibata:"Road Traffic Management Japan and The Development of The Road/Automobile Communication System(RACS)", Proc. of JSK International Symposium, pp.29-37, Nov. 1989.

[49]Dep. of Transport:"Autoguide Pilot Stage Proposals", A Consultation Document, Central Office of Information 1988.

[50]H. Okamoto:"Traffic Control in Japan and Development of The Advan

ced Mobile Traffic Information and Communication System (AMTICS)", Proc. of JSK International Symposium, pp.21-27, Nov. 1989.

[51]W. Weideman: "Future Automotive Technologies for Traffic Safety and Efficiency", Proc. of JSK International Symposium, pp.81-89, Nov. 1989.

[52]R. J. Betdold: "Intelligent Vehicle/Highway Systems for The United States -- An Emerging National Program", Proc. of JSK International Symposium, pp.53-59, Nov. 1989.

[53]新妻, 齊藤, 坂手: "21世紀の高速道路-自動運転システムの可能性-", 自動車技術, 41, 1, pp.76-85, 昭和62年.

[54]泉, 吉川, 浜田, 大井, 武内, 菊池: "自動車電話方式", 昭和56年1月, 信学誌, vol. 64, No. 1

[55]H. C. Furnage: "Development of Highway Advisory Radio", IEEE V.T. Conference, May 1982.

[56]SADAATSU OKASAKA: "Control Channel Traffic Design in a High-Capacity Land Mobile Telephone System", IEEE, Trans. on V.T., vol.VT-27, No.4, NOVEMBER 1978.

B-3) 運転自動化に関する文献

[57]安原: "制御と情報", 鉄道技術, vol. 42, 1, 1985, 1月.

[58]計測自動制御学会: "自動制御ハンドブック, 機器・応用編", 1983, 10月.

[59]谷口: "自動運転の自動化の見通し", 計測技術, '86増刊号, pp.105-114, 昭和61年.

B-4) 移動体デジタルデータ伝送に関する文献

[60]S. stein: "Unified Analysis of Certain Coherent and Noncoherent Binary Communication Systems", IEEE, Trans. on I.T., vol.IT-10, JANUARY 1964.

[61]R. R. ANDERSON, W. R. BENNETT, J. R. DAVEY, J. SALZ: "Differential Detec

tion of Binary FM", THE BELL SYSTEM TECH. JOUR., JANUARY 1965.

[62]川合, 秦, 木下: " デジタル型MSK遅延検波回路の検討", 信学技報, vol. CS80-55, 1980.

[63]鈴木: " GMSK遅延検波回路の誤り率特性", 信学技報, vol. CS79-129, 1979.

[64]T. MIKI, M. HATA: "Performance of 16Kbit/s GMSK Transmission with Postdetection Selection Diversity in Land Mobile Radio", IEEE, Trans. on V.T., vol. VT-33, AUGUST 1984.

[65]平出他: " 移動通信におけるデジタル伝送技術", 信学誌, Vol. 65, No. 2, 1982.

[66]堀越: " 高速デジタル伝送", 信学誌, Vol. 68, No. 11, 1985.

[67]Fumiyuki ADACHI: "Postdetection Selection Diversity Effects on Digital FM Land Mobile Radio," IEEE Trans. on Vehicular Tech. Vol. VT-31 No. 4, Nov. 1982.

[68]大久保他: " 誤り訂正符号の交錯法について", 信学論, Vol. J64-A, No. 5, 1981.

[69]神尾: " インターリーブの評価", 信学論, Vol. J67-A, No. 8, 1984

[70]神尾他: " デジタル陸上移動通信におけるインターリーブを組み合わせた誤り訂正能力の改善", 信学論, Vol. J68-B, No. 6, 1985

[71]浜田他: " 移動体通信におけるデータ伝送プロトコルの検討", 信学論, Vol. J70-B, No. 1, 1987.

[72]秦他: " 陸上移動無線における高速デジタル信号伝送特性", 信学論, Vol. J67-B, No. 2, 1984.

[73]Masaharu HATA: "Preferable Transmission Rate of MSK Land Mobile Radio with Differential Detection," The Trans. of The IECE of JAPAN, Vol. E65, No. 8 Aug. 1982.

[74]Toshio MIKI etc.: "Field Experiments on 128Kbit/s Digital Signal Transmission with Post-Detection Selection Diversity in Land Mobile Radio Channels," The Trans. of The IECE of JAPAN Vol. E 66, No. 9, Sept. 1983.

- [75]池上他：“移動通信の多重波フェージングにおけるアンテナ指向性の効果”，信学論，Vol.J68-B，No.3，1985.
- [76]Klaus-Dieter Eckert etc.：“The Fully Digital Cellular Radio Telephone System CD900,”NORDIC SEMMINAR ON DIGITAL LAND MOBILE RADIO COMMUNICATION, 5-7 Feb. 1985.
- [77]Michael Aldinger etc.：“Performance Analysis of Radiop Transmission in The Fully Digital Cellular Radio System CD900,”NORDIC SEMMINAR ON DIGITAL LAND MOBILE RADIO COMMUNICATION, 5-7 Feb. 1985.
- [78]Peter Kreutzer：“Experimental Investigations on a Digital Mobile Radio Telephone System Using TDMA and Spread Spectrum Techniques,”NORDIC SEMMINAR ON DIGITAL LAND MOBILE RADIO COMMUNICATION, 5-7 Feb. 1985.
- [79]三木他：“デジタル陸上移動無線におけるマルチキャリア伝送方式”，昭和59年度信学部門全大，647，1984.
- [80]三木他：“マルチキャリア高速デジタル移動無線伝送方式の検討”，信学技報，CS84-133，1984.
- [81]Susumu YOSHIDA etc.：“The Effect of Sample Timing on Bit Error Rate Performance in a Multipath Fading Channel,”IEEE Trans. on Vehicular Tech. Vol.VT-35, No.4 Nov.1986.
- [82]Sirikit Ariyavisitaku etc.：“Fractional-Bit Differential Detection of MSK:A Scheme to Avoid Outages Due to Frequency-Selective Fading,”IEEE Trans. on Vehicular Tech. Vol.VT-36 No.1, Feb. 1987.
- [83]MASAHARU HATA, TAKAYOSHI NAGATSU：“Mobile Location Using Signal Strength Measurements in a Cellular System”, IEEE, Trans. on V.T., vol.VT-29, No.2, MAY 1978.
- [84]CHRISTIAN NAMISLO：“Analysis of Mobile Radio Slotted ALOHA Networks”, IEEE, Trans. on V.T., vol.VT-33, No.3, AUGUST 1984.
- [85]RICHARD A. COMROE, DANIEL J. COSTELLO, J.R.：“ARQ Schemes for Data Transmission in Mobile Radio Systems”, IEEE, Trans. on V.T., vol.VT-33, No.3, AUGUST 1984.

B-5) 通信チャネルの制御方式に関する文献

[86]吉川, 岡坂, 駒形: "自動車電話における無線回線制御技術", 信学誌, vol. J61-B, No. 2, 1978年2月.

[87]野村, 吉田: "マルチパケット伝送方式に関するトラヒック解析", 信学誌, vol. J61-B, No. 2, 1978年2月.

[88]木下, 秦, 平出: "TD-FDMA移動通信方式の検討", 信学誌, vol. J64-B, No. 9, 1981年9月.

[89]椋本, 福田: "アイドル信号多元接続 (ISMA) 方式による無線パケット伝送システム", 信学誌, vol. J64-B, No. 10, 1981年10月.

[90]福田, 椋本: "双方向無線パケット通信システムにおける動的チャネル割当方式", 信学誌, vol. J65-B, No. 4, 1982年4月.

[91]渡辺, 石川: "移動通信用無線パケットアクセス方式の評価", 信学誌, vol. J67-B, No. 11, 1984年11月.

[92]SIMON S. LAM, LEONARD KLEINROCK: "Packet Switching in a Multiaccess Broadcast Channel: Dynamic Control Procedures", IEEE, Trans. on Com., vol. COM-23, No. 9, SEPTEMBER 1975.

[93]LEONARD KLEINROCK, FOUAD A. TOBAGI: "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics", IEEE, Trans. on Com., vol. COM-23, No. 12, DECEMBER 1975.

[94]FOUAD A. TOBAGI, LEONARD KLEINROCK: "Packet Switching in Radio Channels: Part II - The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution", IEEE, Trans. on Com., vol. COM-23, No. 12, DECEMBER 1975.

[95]FOUAD A. TOBAGI, LEONARD KLEINROCK: "Packet Switching in Radio Channels: Part III - Polling and (Dynamic) Split-Channel Reservation Multiple Access", IEEE, Trans. on Com., vol. COM-24, No. 8, AUGUST 1975

[96]NORMAN ABRAMSON: "The Throughput of Packet Broadcasting Channels"

, IEEE, Trans. on Com., vol.COM-25, No.1, JANUARY 1977.

B-8) 移動体用アンテナに関する文献

- [97]藤本: "小型アンテナに関する研究動向", 信学誌, Vol.70, No.8, p p.830-838, 1987年8月.
- [98]進士: "小型・薄型アンテナと無線通信システム", 信学論(B), V0 1.J-71-B, No.11, pp.1198-1205, 1988年11月.
- [99]奥村, 進士: "移動通信の基礎", 電子情報通信学会, コロナ社1986年10月.
- [100]吉田他: "アンテナと電波伝搬", 信学誌, vol.68, No.11, 1985.
- [101]山田, 恵比根, 中嶋, 奈良: "大容量移動通信方式基地局/移動局アンテナ構成技術", 研実法, Vol.35, No.10, 1986年10月.
- [102] KUNITOSHI NISHIKAWA and YOSHIKAZU ASAO: "Vertical Patterns of Mobile Antenna in UHF Band", IEEE trans., VOL. VT-33, No.2, MAY 1986.
- [103]池上, 竹内, 吉田: "移動通信の多重波フェージングにおけるアンテナ指向性の効果", 信学論(B), VOL.J68-B, No.3, 1985年3月.
- [104]BENJAMIN M. HALPERN and PAUL E. MAYES: "The Monopole Slot as a Two-Port Diversity Antenna for UHF Land-Mobile Radio Systems", IEEE trans. VOL.VT-33, No.2, MAY 1984.
- [105]恵比根, 山田: "車載用ダイバーシチアンテナ", 信学技報, CS85-120, 1985年11月.
- [106]TOKIO TAGA and KOUICHI TSUNEKAWA: "Performance Analysis of a Built-In Planar Inverted F Antenna for 800MHz Band Portable Radio Units", IEEE trans. VOL.SAC-5, No.5, JUNE 1987.
- [107]向, 常川, 山田: "携帯機ダイバーシチアンテナの相関係数", 1989年信学春季全大, B-817, 1989年3月.
- [108]INDIRA CHATTERJEE, YONG-GON GU and OM P. GANDHI: "Quantification of Electromagnetic Absorption in Humans from Body-Mounted Communication Transceivers", IEEE trans. VOL.VT-34, No.2, MAY 1985.

B-6) デジタル情報サービスに関する文献

[109]電子通信学会: "電子通信ハンドブック", pp 1162-1172, 昭和54年10月.

[110]吹抜: "静止画像の符号化", 信学誌, vol. 67, 1, 昭和51年1月.

[111]田中, 清水, 松原, 星野: "ハードウェア/ハイエンド", 日経バイト vol.10, No.64, pp.197-281, 1989.

[112]NTT発行: "DDX-P料金表."

B-7) 通信ケーブルの特性に関する文献

[113]小林: "同軸ケーブルにおけるパルス波の電幅", 藤倉電線技報, vol. 65, 昭和58年3月.

◎付録

○付録目次

項目	頁
A. 1. 情報伝送方式の評価プログラム	
A. 1. 1. 従来方式の評価プログラム	〈 2 〉
A. 1. 2. 新方式の評価プログラム	〈 1 6 〉
A. 2. グローバルレベル制御方式の評価プログラム（両方式の比較）	〈 3 2 〉
A. 3. 右折車の待ち合わせ制御アルゴリズムの評価プログラム	
A. 3. 1. 誘導を行わない場合の評価プログラム	〈 4 6 〉
A. 3. 2. 誘導を行った場合の評価プログラム	〈 5 8 〉
A. 4. ハイブリッドシミュレーション用プログラム	
A. 4. 1. 進入実験用プログラム	
A. 4. 1. 1. 交通流シミュレータ用プログラム	〈 8 2 〉
A. 4. 1. 2. 車載機用プログラム	〈 8 6 〉
A. 4. 2. 追跡実験用プログラム	
A. 4. 2. 1. 交通流シミュレータ用プログラム	〈 9 1 〉
A. 4. 2. 2. 車載機用プログラム	〈 9 5 〉

A. 1. 情報伝送方式の評価プログラム

A. 1. 1. 従来方式の評価プログラム

```

C** COMMUNICATION SIMULATION FOR CONVENTIONAL METHOD *****
C
COMMON /GDAT/IENTRY, IDMY, NDEV, NATN
*, NGDRR1, NGDRR2, NGDRD, NGDRZ, NGDRC, NGDRM
*, IOPTR1, IOPTR2, IOPTD, IOPTZ, IOPTC, IOPTM
*, GXL, GXM, GYL, GYM
*/EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)
*, NRQ(1100), ITPGND(1100)
*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)
*, TPSGND(1000), INXTGN(1000)
*/QDAT/IQCMGN(20), NQCMGN, IQRSGN(20), NQRSGN
*, IQCMMV(50), NQCMMV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN
*/COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND
*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGARMX
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN
C
1112 CONTINUE
READ(5, *) IEXEC
IF (IEXEC.EQ.0) GOTO 1111
C
C DATA PCALL/1.0, 999*0.5/
PCALL(1)=1.0
DO 111 IX=2, 1000
111 PCALL(IX)=0.5
C
C DATA TTRIG, TRQ, TACK, TMVD, TGND, TWRQ
C */0.00333, 0.00604, 0.00604, 0.0170, 0.0614, 0.0003/
TTRIG=0.00333
TRQ=0.00604
TACK=0.00604
TMVD=0.0170
TGND=0.0614
TWRQ=0.0003
C
C DATA ICARST/1100*0/
DO 112 IX=1, 1100
112 ICARST(IX)=0
C
C DATA NQCMGN, NQRSGN, NQCMMV, NQRSMV, NQCNGN, NCMTRG, NSGNL/7*0/
NQCMGN=0
NQRSGN=0
NQCMMV=0
NQRSMV=0
NQCNGN=0
NCMTRG=0
NSGNL=0
C

```

C	DATA ISTGND/1000*0/	00005100
	DO 113 IX=1,1000	00005200
113	ISTGND(IX)=0	00005300
C		00005400
C	DATA IGND, ICAR/2*0/	00005500
	IGND=0	00005600
	ICAR=0	00005700
C		00005800
C		00005900
C	DATA TGNRMX, TGNRMN, FG NRMX/100., 0.7, 1.5/	00006000
	TGNRMX=100.	00006100
	TGNRMN=0.7	00006200
	FG NRMX=1.5	00006300
C		00006400
C	DATA IRAN/0/	00006500
	IRAN=0	00006600
C		00006700
C	DATA DD, DL, NL, NZ/16.0, 10.0, 6, 3/	00006800
	DD=16.0	00006900
	DL=10.0	00007000
	NL=6	00007100
	NZ=3	00007200
C		00007300
C	DATA XLC/4.0/	00007400
	XLC=4.0	00007500
C		00007600
C	DATA GXL, GXM, GYL, GYM/50.0, 4050.0, 500.0, 2500.0/	00007700
	GXL=50.0	00007800
	GXM=4050.0	00007900
	GYL=500.0	00008000
	GYM=2500.0	00008100
C		00008200
C	DATA RWC/0.6/	00008300
	RWC=0.6	00008400
C		00008500
C	DATA QLGNR/6*1300./	00008600
	DO 114 IX=1,6	00008700
114	QLGNR(IX)=1300.	00008800
C		00008900
C	DATA VLGNR/6*10./	00009000
	DO 115 IX=1,6	00009100
115	VLGNR(IX)=10.0	00009200
C		00009300
	TCNTL=0.0114	00009400
C		00009500
	XLR=DL+FLOAT(NZ-1)*(DD+DL)	00009600
	DO 40 IZ=1,NZ	00009700
	XZN(IZ,1)=FLOAT(IZ-1)*(DD+DL)	00009800
	XZN(IZ,2)=XZN(IZ,1)+DL	00009900
40	CONTINUE	00010000
C		00010100
	GYM=GYM*50./XLR	00010200
C		00010300
	READ(5,*) IPRTGN	00010400
	READ(5,*) PPCKT	00010500

READ(5,*) LPCKT	00010600
READ(5,*) RSPMV	00010700
READ(5,*) RSPGN	00010800
READ(5,*) TITRG	00010900
READ(5,*) TSIMMX	00011000
WRITE(6,*) '*****'	00011100
WRITE(6,*) 'PRINTOUT STATUS?(YES=1) -> ', IPRTGN	00011200
WRITE(6,*) 'PROBABIRITY OF GENERATING DATA -> ', PPCKT	00011300
WRITE(6,*) 'LENGTH OF GENERATING DATA -> ', LPCKT	00011400
WRITE(6,*) 'RESPONSE TIME OF MOVEMENT DATA -> ', RSPMV	00011500
WRITE(6,*) 'RESPONSE TIME OF GENERAL DATA -> ', RSPGN	00011600
WRITE(6,*) 'INTERVAL TIME OF TRIGGER -> ', TITRG	00011700
WRITE(6,*) 'TOTAL SIMULATION TIME -> ', TSIMMX	00011800
WRITE(6,*)	00011900
TSIM=0.0	00012000
C	00012100
C *** COMMUNICATION PARAMETER INITIALIZATION ***	00012200
TIWTRG=TITRG	00012300
PSTIM=0.	00012400
C	00012500
C *** GENERATOR INITIALIZATION ***	00012600
CALL GNRINI	00012700
C	00012800
C *** SIMULARION START ***	00012900
100 TSIM=TSIM+PSTIM	00013000
IF (TSIM.GE.TSIMMX) GOTO 200	00013100
NSGNL=0	00013200
CALL CRGNR	00013300
CALL CARMV	00013400
CALL CMZPRM	00013500
CALL CMTPRM	00013600
CALL CMCNTL	00013700
ISG=1	00013800
60 CONTINUE	00013900
IF (ISGNL(ISG,1).NE.1) GOTO 70	00014000
IF (ISG.GE.NSGNL) GOTO 50	00014100
ISG=ISG+1	00014200
GOTO 60	00014300
70 CONTINUE	00014400
C	00014500
C *** DISPLAYING STATUS OF COMMUNICATION ***	00014600
IF (IPRTGN.EQ.1) THEN	00014700
CALL PRTGND	00014800
END IF	00014900
C	00015000
50 CONTINUE	00015100
C	00015200
GOTO 100	00015300
200 CONTINUE	00015400
C	00015500
C *** CALCULATING RESULTS ***	00015600
C ** CALCULATING FAILURE RATE **	00015700
IF (ICAR.GT.0) THEN	00015800
IFCNCT=0	00015900
IFRSP=0	00016000

IFNSH=0	00016100
RFRSP=0.	00016200
RFCNCT=0.	00016300
DO 300 IC=1, ICAR	00016400
IF (ICARST(IC).GE.20) THEN	00016500
IFNSH=IFNSH+1	00016600
IF ((ICARST(IC).EQ.23).OR.(ICARST(IC).EQ.20))	00016700
* IFCNCT=IFCNCT+1	00016800
IF ((ICARST(IC).EQ.24).OR.(ICARST(IC).EQ.21))	00016900
* IFRSP=IFRSP+1	00017000
END IF	00017100
300 CONTINUE	00017200
IF (IFNSH.GT.0) THEN	00017300
RFCNCT=FLOAT(IFCNCT)/FLOAT(IFNSH)	00017400
RFRSP=FLOAT(IFRSP)/FLOAT(IFNSH)	00017500
ELSE	00017600
WRITE(6,*) 'NO CARS FINISH.'	00017700
RFCNCT=-1.	00017800
RFRSP=-1.	00017900
END IF	00018000
END IF	00018100
C ** CALCULATING DELAY TIME OF DATA AND THROUGHPUT **	00018200
TDRY=0.	00018300
ITBLK=0	00018400
STH=0.	00018500
ADRYT=0.	00018600
IF (IGND.GT.0) THEN	00018700
DO 400 IG=1, IGND	00018800
IF (ISTGND(IG).GT.0) THEN	00018900
ITBLK=ITBLK+(LPCKT-IBGND(IG))	00019000
TDRY=TDRY+TPSGND(IG)	00019100
END IF	00019200
400 CONTINUE	00019300
TCOM=FLOAT(ITBLK)*(TGND-TCNTL)	00019400
* +FLOAT(IFNSH-IFRSP-IFCNCT)*(TMVD-TCNTL)	00019500
STH=TCOM/TSIM	00019600
IF (ITBLK.GT.0) THEN	00019700
ADRYT=(TDRY/FLOAT(ITBLK))*LPCKT	00019800
ELSE	00019900
WRITE(6,*) 'NO DATA CUMMUNICATED SUCCESFULLY.'	00020000
ADRYT=1000.	00020100
END IF	00020200
END IF	00020300
C	00020400
C *** PRINTOUT RESULTS ***	00020500
WRITE(6,*) 'RESULTS OF CONTENTION'	00020600
WRITE(6,*) 'GENERATED CARS=', ICAR	00020700
WRITE(6,*) 'GENERATED DATA=', IGND	00020800
WRITE(6,*) 'FAILURE RATE OF MISSCONNECT=', RFCNCT	00020900
WRITE(6,*) 'FAILURE RATE OF MISSRESPONSE=', RFRSP	00021000
WRITE(6,*) 'AVERAGE DELAY TIME OF DATA=', ADRYT	00021100
WRITE(6,*) 'THROUGHPUT OF DATA=', STH	00021200
C	00021300
WRITE(2) PPCKT, LPCKT, RSPMV, RSPGN, TITRG, RFCNCT, RFRSP, ADRYT, STH	00021400
C	00021500

GOTO 1112	00021600
1111 CONTINUE	00021700
STOP	00021800
END	00021900
C	00022000
C	00022100
C *** GENERATE CARS ***	00022200
SUBROUTINE CRGMR	00022300
COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00022400
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00022500
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00022600
*, NRQ(1100), ITPGND(1100)	00022700
*/QDAT/IQCMGN(20), NQCMGN, IQRSGN(20), NQRSGN	00022800
*, IQCMV(50), NQCMV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN	00022900
*/COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00023000
*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00023100
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00023200
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00023300
C	00023400
C	00023500
DO 10 IL=1, NL	00023600
TWGNR(IL)=TWGNR(IL)-PSTIM	00023700
IF (TWGNR(IL).LE.0.0) THEN	00023800
C	00023900
C ** SET INITIAL CAR PARM **	00024000
ICAR=ICAR+1	00024100
ICARST(ICAR)=1	00024200
CARVEL(ICAR)=VLGNR(IL)	00024300
ICARLN(ICAR)=IL	00024400
CARPOS(ICAR)=0.	00024500
NRQ(ICAR)=0	00024600
CALL RANU2(IRAN, RAND, 1, ICON)	00024700
PBOUT(ICAR)=XZN(1, 1)+(XZN(1, 2)-XZN(1, 1))*RAND	00024800
C	00024900
C ** RESET TWGNR **	00025000
100 CONTINUE	00025100
CALL RANU2(IRAN, RAND, 1, ICON)	00025200
TG=TGNRMN+(TGNRMX-TGNRMN)*RAND	00025300
CALL RANU2(IRAN, RAND, 1, ICON)	00025400
PG=RAND*FGNRMX	00025500
IF (GNRFNC(QLGNR(IL), TGNRMN, TG).LE.PG) GOTO 100	00025600
TWGNR(IL)=TG	00025700
C	00025800
END IF	00025900
10 CONTINUE	00026000
RETURN	00026100
END	00026200
C	00026300
C	00026400
C *** GENERATOR INITIALIZATION ***	00026500
SUBROUTINE GNRINI	00026600
COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00026700
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00026800
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00026900
C	00027000

DO 10 IL=1, NL	00027100
100 CONTINUE	00027200
CALL RANU2(IRAN, RAND, 1, ICON)	00027300
TG=TGNRMN+(TGNRMX-TGNRMN)*RAND	00027400
CALL RANU2(IRAN, RAND, 1, ICON)	00027500
PG=RAND*FGNRMX	00027600
IF (GNRFNC(QLG NR(IL), TGNRMN, TG). LE. PG) GOTO 100	00027700
TWGNR(IL)=TG	00027800
C	00027900
10 CONTINUE	00028000
RETURN	00028100
END	00028200
C	00028300
C	00028400
C *** GERARATOR FUNCTION OF CARS ***	00028500
REAL FUNCTION GNRFNC(Q, TO, T)	00028600
RAMDA=Q/3600.	00028700
IF (T. GT. TO) THEN	00028800
TRAMDA=1./RAMDA	00028900
FGNR=EXP(-(T-TO)/(TRAMDA-TO))	00029000
ELSE	00029100
FGNR=0.	00029200
END IF	00029300
GNRFNC=FGNR	00029400
RETURN	00029500
END	00029600
C	00029700
C	00029800
C *** MOVING CARS ***	00029900
SUBROUTINE CARMV	00030000
COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00030100
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00030200
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00030300
*, NRQ(1100), ITPGND(1100)	00030400
*/COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00030500
*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00030600
*/SIMDAT/TWGNR(6), QLG NR(6), VLG NR(6), TGNRMX, TGNRMN, FGNRMX	00030700
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00030800
C	00030900
IF (ICAR. GT. 0) THEN	00031000
DO 10 IIC=1, ICAR	00031100
IF (ICARST(IIC). GT. 0) THEN	00031200
IF (CARPOS(IIC). LE. XLR) THEN	00031300
CARPOS(IIC)=CARPOS(IIC)+PSTIM*CARVEL(IIC)*1000./3600.	00031400
END IF	00031500
ENDIF	00031600
10 CONTINUE	00031700
END IF	00031800
RETURN	00031900
END	00032000
C	00032100
C	00032200
C *** PRINT OUT STATUS OF DATA, CARS AND COMMUNICATED SIGNALS ***	00032300
SUBROUTINE PRGND	00032400
COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00032500

* , ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00032600
* , NRQ(1100), ITPGND(1100)	00032700
* /GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00032800
* , TPSGND(1000), INXTGN(1000)	00032900
* /QDAT/IQCMGN(20), NQCMGN, IQRSGN(20), NQRSGN	00033000
* , IQCMMV(50), NQCMMV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN	00033100
* /COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00033200
* , NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00033300
* /SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00033400
* , TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00033500
C	00033600
WRITE(6, 3000)	00033700
WRITE(6, *) 'SIMULATION TIME=', TSIM	00033800
WRITE(6, *)	00033900
WRITE(6, *) 'COMMUNICATED SIGNALS'	00034000
IF (NSGNL.EQ.0) THEN	00034100
WRITE(6, *) 'NON SIGNALS COMMUNICATED'	00034200
ELSE	00034300
WRITE(6, *) 'SIGNAL CAR'	00034400
DO 500 IS=1, NSGNL	00034500
IF (ISGNL(IS, 1).EQ.1) THEN	00034600
WRITE(6, *) 'TRG'	00034700
ELSE IF (ISGNL(IS, 1).EQ.2) THEN	00034800
IF (ISGNL(IS, 2).GT.0) THEN	00034900
WRITE(6, *) 'REQ', ISGNL(IS, 2)	00035000
ELSE IF (ISGNL(IS, 2).EQ.0) THEN	00035100
WRITE(6, *) 'REQ CONTENTION'	00035200
WRITE(6, *) 'CALLING CARS-', ((ICLL(ICON), '-'), ICON=1, NCLL)	00035300
END IF	00035400
ELSE IF (ISGNL(IS, 1).EQ.3) THEN	00035500
WRITE(6, *) 'ACK', ISGNL(IS, 2)	00035600
ELSE IF (ISGNL(IS, 1).EQ.4) THEN	00035700
WRITE(6, *) 'MOV CON', ISGNL(IS, 2)	00035800
ELSE IF (ISGNL(IS, 1).EQ.5) THEN	00035900
WRITE(6, *) 'GEN DAT', ISGNL(IS, 2)	00036000
END IF	00036100
500 CONTINUE	00036200
END IF	00036300
WRITE(6, *) 'END OF SIGNLS'	00036400
WRITE(6, *)	00036500
WRITE(6, *) 'STATUS OF CARS'	00036600
IF (ICAR.EQ.0) THEN	00036700
WRITE(6, *) 'NO CARS EXIST.'	00036800
ELSE	00036900
WRITE(6, *)	00037000
* NO. STA POS LANE TOP-GND R-MV W-MV'	00037100
DO 600 IC=1, ICAR	00037200
IF (((ICARST(IC).GE.10).AND.(ICARST(IC).LT.20)).OR.	00037300
* ((ICARST(IC).GE.23).AND.(ICARST(IC).LE.27))) THEN	00037400
WRITE(6, 4000) IC, ICARST(IC), CARPOS(IC), ICARLN(IC), ITPGND(IC)	00037500
C * , IRMVST(IC), IWMVST(IC)	00037600
ELSE	00037700
* IF (((ICARST(IC).GE.1).AND.(ICARST(IC).LT.10)).OR.	00037800
* ((ICARST(IC).GE.20).AND.(ICARST(IC).LT.23))) THEN	00037900
WRITE(6, 4500) IC, ICARST(IC), CARPOS(IC), ICARLN(IC)	00038000

C	*	, IRMVST(IC), IWMVST(IC)	00038100
		END IF	00038200
600		CONTINUE	00038300
		END IF	00038400
		WRITE(6,*) 'END OF CARS'	00038500
		WRITE(6,*)	00038600
		WRITE(6,*) 'STATUS OF GENERAL DATA'	00038700
		IF (IGND.EQ.0) THEN	00038800
		WRITE(6,*) 'NO DATA EXIST.'	00038900
		ELSE	00039000
		WRITE(6,*) ' NO. STA CAR NEXT REMAINT'	00039100
		DO 300 IG=1, IGND	00039200
300		WRITE(6,1000) IG, ISTGND(IG), ICRGND(IG), INXTGN(IG), IBGND(IG)	00039300
		END IF	00039400
		WRITE(6,*) 'END OF GENERAL DATA'	00039500
		WRITE(6,*)	00039600
		WRITE(6,*) 'QUE FOR CONT.'	00039700
		IF (NQCNGN.EQ.0) THEN	00039800
		WRITE(6,*) 'NO QUE EXIST.'	00039900
		ELSE	00040000
		WRITE(6,*) ' NO. DATA'	00040100
		DO 400 IQ=1, NQCNGN	00040200
400		WRITE(6,2000) IQ, IQCNGN(IQ)	00040300
		END IF	00040400
		WRITE(6,*) 'END OF QUE'	00040500
		WRITE(6,*)	00040600
C			00040700
1000		FORMAT(1H , 6I10)	00040800
2000		FORMAT(1H , 2I10)	00040900
3000		FORMAT(1H1)	00041000
4000		FORMAT(1H , 2I10, F10.3, 2I10, 2I5)	00041100
4500		FORMAT(1H , 2I10, F10.3, I10, ' NO DATA', 2I5)	00041200
		RETURN	00041300
		END	00041400
C			00041500
C			00041600
C		*** CHANGE STATUS OF CARS AND DATA BY OUT OR INTO ZONE ***	00041700
		SUBROUTINE CMZPRM	00041800
		COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00041900
		*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00042000
		*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00042100
		*, NRQ(1100), ITPGND(1100)	00042200
		*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00042300
		*, TPSGND(1000), INXTGN(1000)	00042400
		*/QDAT/IQCMGN(20), NQCNGN, IQRSGN(20), NQRSGN	00042500
		*, IQCMNV(50), NQCMMV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN	00042600
		*/COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00042700
		*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00042800
		*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00042900
		*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00043000
C			00043100
		ZI=XZN(2, 1)	00043200
		ZO=XZN(2, 2)	00043300
		ZPO=XZN(1, 2)	00043400
C			00043500

IF (ICAR.GT.0) THEN	00043600
DO 10 IIC=1, ICAR	00043700
IST=ICARST(IIC)	00043800
IF ((IST.GT.0).AND. (IST.LT.20)) THEN	00043900
XC=CARPOS(IIC)	00044000
IZOGND=0	00044100
C	00044200
IF (IST.EQ.1) THEN	00044300
IF (XC.GT.PBOUT(IIC)) THEN	00044400
IST=2	00044500
END IF	00044600
ELSE IF (IST.EQ.2) THEN	00044700
IF (XC.GT.ZPO) THEN	00044800
IST=3	00044900
C	00045000
C ** CONT.GENERAL DATA **	00045100
IF (NQCNGN.GT.0) THEN	00045200
C	00045300
C * CHANGE STATUS OF DATA *	00045400
IG=IQCNGN(1)	00045500
15 ISTG=ISTGND(IG)	00045600
IF (ISTG.EQ.21) THEN	00045700
ISTG=11	00045800
ELSE	00045900
WRITE(6,*) 'ERROR IN CHECKPOINT-1 OF SUBROUTINE CMZPRM!'	00046000
END IF	00046100
ISTGND(IG)=ISTG	00046200
IG=INXTGN(IG)	00046300
IF (IG.EQ.0) GOTO 25	00046400
GOTO 15	00046500
25 CONTINUE	00046600
C	00046700
C * MOV DATA TO CAR *	00046800
IG=IQCNGN(1)	00046900
16 ICRGND(IG)=IIC	00047000
IG=INXTGN(IG)	00047100
IF (IG.EQ.0) GOTO 26	00047200
GOTO 16	00047300
26 CONTINUE	00047400
ITPGND(IIC)=IQCNGN(1)	00047500
C	00047600
C * REMOVE DATA FROM CONT.QUE *	00047700
NQCNGN=NQCNGN-1	00047800
IF (NQCNGN.GT.0) THEN	00047900
DO 20 IQ=1,NQCNGN	00048000
20 IQCNGN(IQ)=IQCNGN(IQ+1)	00048100
END IF	00048200
C	00048300
C * CHANGE STATUS OF CAR *	00048400
IST=11	00048500
END IF	00048600
C	00048700
C ** GENERATE NEW DATA **	00048800
CALL RANU2(IRAN,RAND,1,ICON)	00048900
IF (RAND.LT.PPCKT) THEN	00049000

	IGND=IGND+1	00049100
	ISTGND(IGND)=11	00049200
	ICRGND(IGND)=IIC	00049300
	IBGND(IGND)=LPCKT	00049400
	TPSGND(IGND)=0.0	00049500
	INXTGN(IGND)=0	00049600
	IF(IST.EQ.3) THEN	00049700
	ITPGND(IIC)=IGND	00049800
	IST=11	00049900
	ELSE IF (IST.EQ.11) THEN	00050000
	IG=ITPGND(IIC)	00050100
17	IF (INXTGN(IG).EQ.0) GOTO 27	00050200
	IG=INXTGN(IG)	00050300
	GOTO 17	00050400
27	CONTINUE	00050500
	INXTGN(IG)=IGND	00050600
	ELSE	00050700
	WRITE(6,*) 'ERROR IN CHECKPOINT-2 OF SUBROUTINE CMZPRM!'	00050800
	END IF	00050900
	END IF	00051000
C		00051100
	END IF	00051200
	ELSE IF (IST.EQ.3) THEN	00051300
	IF (XC.GT.ZI) IST=4	00051400
	ELSE IF (IST.EQ.4) THEN	00051500
	IF (XC.GT.ZO) IST=20	00051600
	ELSE IF (IST.EQ.5) THEN	00051700
	IF (XC.GT.ZO) IST=21	00051800
	ELSE IF (IST.EQ.11) THEN	00051900
	IF (XC.GT.ZI) IST=12	00052000
	ELSE IF (IST.EQ.12) THEN	00052100
	IF (XC.GT.ZO) THEN	00052200
	IST=23	00052300
	IZOGND=ITPGND(IIC)	00052400
	ENDIF	00052500
	ELSE IF (IST.EQ.13) THEN	00052600
	IF (XC.GT.ZO) THEN	00052700
	IST=24	00052800
	IZOGND=ITPGND(IIC)	00052900
	ENDIF	00053000
	ELSE IF (IST.EQ.14) THEN	00053100
	IF (XC.GT.ZO) THEN	00053200
	IST=25	00053300
	IZOGND=ITPGND(IIC)	00053400
	ENDIF	00053500
	ELSE IF (IST.EQ.15) THEN	00053600
	IF (XC.GT.ZO) THEN	00053700
	IST=26	00053800
	IZOGND=ITPGND(IIC)	00053900
	ENDIF	00054000
	END IF	00054100
	ICARST(IIC)=IST	00054200
C		00054300
C ** ADD DATA TO CONT. QUE **		00054400
	IF (IZOGND.GT.0) THEN	00054500

	NQCNGN=NQCNGN+1	00054600
	IQCMGN(NQCNGN)=IZOGND	00054700
C *	CHANGE STATUS OF DATA *	00054800
	IG=IZOGND	00054900
11	IST=ISTGND(IG)	00055000
	IF (IST.EQ.11) THEN	00055100
	IST=21	00055200
	ELSE IF (IST.EQ.12) THEN	00055300
	IST=21	00055400
	ELSE IF (IST.EQ.13) THEN	00055500
	IST=21	00055600
	ELSE IF (IST.EQ.14) THEN	00055700
	IST=21	00055800
C *	REMOVE DATA FROM COMMUNICATION QUE *	00055900
	IF (NQCMGN.GT.0) THEN	00056000
	IQCM=1	00056100
12	CONTINUE	00056200
	IF (IQCMGN(IQCM).EQ.IG) GOTO 22	00056300
	IF (IQCM.GE.NQCMGN) GOTO 32	00056400
	IQCM=IQCM+1	00056500
	GOTO 12	00056600
22	NQCMGN=NQCMGN-1	00056700
	IF (NQCMGN.GE.IQCM) THEN	00056800
	DO 42 II=IQCM,NQCMGN	00056900
42	IQCMGN(II)=IQCMGN(II+1)	00057000
	END IF	00057100
	GOTO 100	00057200
32	WRITE(6,*) 'ERROR IN CHECKPOINT 3 IN CMZPRM!'	00057300
100	CONTINUE	00057400
	ELSE	00057500
	WRITE(6,*) 'ERROR IN CHECKPOINT 4 IN CMZPRM!'	00057600
	END IF	00057700
	ELSE	00057800
	WRITE(6,*) 'ERROR IN CHECKPOINT 5 IN CMZPRM!'	00057900
	END IF	00058000
	ISTGND(IG)=IST	00058100
	IG=INXTGN(IG)	00058200
	IF (IG.EQ.0) GOTO 21	00058300
	GOTO 11	00058400
21	CONTINUE	00058500
C		00058600
	END IF	00058700
	END IF	00058800
10	CONTINUE	00058900
	END IF	00059000
	RETURN	00059100
	END	00059200
C		00059300
C		00059400
C ***	COMMUNICATION CONTROL ***	00059500
	SUBROUTINE CMCNTL	00059600
	COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00059700
	*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00059800
	*, NRQ(1100), ITPGND(1100)	00059900
	*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00060000

* ,TPSGND(1000), INXTGN(1000)	00060100
*/QDAT/IQCMGN(20), NQCMGN, IQRSGN(20), NQRSGN	00060200
*, IQCMMV(50), NQCMMV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN	00060300
*/COMDAT/RSPGN, RSPMV, ISGNL(100, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00060400
*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00060500
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGARMX	00060600
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00060700
C	00060800
PSTIM=0.	00060900
IF (NQCMMV.EQ.0) GOTO 10	00061000
C ** TRANSMISSION OF MOVEMENT CONTROL DATA	00061100
DO 20 IQ=1, NQCMMV	00061200
ICR=IQCMMV(IQ)	00061300
IST=ICARST(ICR)	00061400
IF (IST.EQ.5) THEN	00061500
PSTIM=PSTIM+TMVD	00061600
IST=22	00061700
ELSE IF (IST.EQ.13) THEN	00061800
PSTIM=PSTIM+TMVD	00061900
IST=14	00062000
ELSE	00062100
WRITE(6,*) 'ERROR IN CHECKPOINT 1 IN CMCNTL!'	00062200
END IF	00062300
ICARST(ICR)=IST	00062400
NSGNL=NSGNL+1	00062500
ISGNL(NSGNL, 1)=4	00062600
ISGNL(NSGNL, 2)=ICR	00062700
20 CONTINUE	00062800
NQCMMV=0	00062900
10 CONTINUE	00063000
C	00063100
C ** TRANSMISSION OF GENERAL DATA **	00063200
IF ((NQCMGN.GT.0).AND.(NCMTRG.EQ.0)) THEN	00063300
IG=IQCMGN(1)	00063400
NQCMGN=NQCMGN-1	00063500
IF (NQCMGN.GT.0) THEN	00063600
DO 40 IQ=1, NQCMGN	00063700
40 IQCMGN(IQ)=IQCMGN(IQ+1)	00063800
END IF	00063900
PSTIM=PSTIM+TGND	00064000
IBGND(IG)=IBGND(IG)-1	00064100
NSGNL=NSGNL+1	00064200
ISGNL(NSGNL, 1)=5	00064300
ISGNL(NSGNL, 2)=ICRGND(IG)	00064400
IF (IBGND(IG).EQ.0) THEN	00064500
ISTGND(IG)=25	00064600
IF (INXTGN(IG).EQ.0) THEN	00064700
ICARST(ICRGND(IG))=27	00064800
ELSE	00064900
ICARST(ICRGND(IG))=14	00065000
ITPGND(ICRGND(IG))=INXTGN(IG)	00065100
END IF	00065200
ELSE	00065300
NQCMGN=NQCMGN+1	00065400
IQCMGN(NQCMGN)=IG	00065500

END IF	00065600
C	00065700
C ** TRIGGER GENERATION **	00065800
ELSE	00065900
IF (NCMTRG.EQ.0) THEN	00066000
TIWTRG=TITRG	00066100
NCMTRG=1	00066200
END IF	00066300
DO 50 IT=1, NCMTRG	00066400
PSTIM=PSTIM+TTRIG	00066500
NCL=0	00066600
NSGNL=NSGNL+1	00066700
ISGNL(NSGNL, 1)=1	00066800
ISGNL(NSGNL, 2)=0	00066900
NCLL=0	00067000
IF (ICAR.GT.0) THEN	00067100
DO 60 IIC=1, ICAR	00067200
IF ((ICARST(IIC).EQ.4).OR.(ICARST(IIC).EQ.12)) THEN	00067300
NRQ(IIC)=NRQ(IIC)+1	00067400
CALL RANU2(IRAN, RAND, 1, ICON)	00067500
IF (RAND.LT.PCALL(NRQ(IIC))) THEN	00067600
NCL=NCL+1.	00067700
ICL=IIC	00067800
NCLL=NCLL+1	00067900
ICLL(NCLL)=IIC	00068000
END IF	00068100
END IF	00068200
60 CONTINUE	00068300
END IF	00068400
IF (NCL.EQ.1) THEN	00068500
PSTIM=PSTIM+TRQ+TACK	00068600
TRMVD(ICL)=RSPMV	00068700
NSGNL=NSGNL+1	00068800
ISGNL(NSGNL, 1)=2	00068900
ISGNL(NSGNL, 2)=ICL	00069000
NSGNL=NSGNL+1	00069100
ISGNL(NSGNL, 1)=3	00069200
ISGNL(NSGNL, 2)=ICL	00069300
IF (ICARST(ICL).EQ.4) THEN	00069400
ICARST(ICL)=5	00069500
ELSE IF (ICARST(ICL).EQ.12) THEN	00069600
ICARST(ICL)=13	00069700
IG=ITPGND(ICL)	00069800
12 TRGND(IG)=RSPGN	00069900
ISTGND(IG)=12	00070000
IG=INXTGN(IG)	00070100
IF (IG.EQ.0) GOTO 11	00070200
GOTO 12	00070300
11 CONTINUE	00070400
ELSE	00070500
WRITE(6,*) 'ERROR IN CHECKPOINT 2 IN CMCNTL!'	00070600
END IF	00070700
ELSE IF (NCL.EQ.0) THEN	00070800
PSTIM=PSTIM+TWRQ	00070900
ELSE	00071000

PSTIM=PSTIM+TRQ	00071100
NSGNL=NSGNL+1	00071200
ISGNL(NSGNL,1)=2	00071300
ISGNL(NSGNL,2)=0	00071400
END IF	00071500
50 CONTINUE	00071600
NCMTRG=0	00071700
END IF	00071800
C	00071900
C ** CHANGE STATUS OF DATA AND CARS BY REPLY OF DATA	00072000
IF (ICAR.GT.0) THEN	00072100
DO 30 IC=1, ICAR	00072200
IF (ICARST(IC).EQ.14) THEN	00072300
IG=ITPGND(IC)	00072400
IF (ISTGND(IG).EQ.13) THEN	00072500
NQCMGN=NQCMGN+1	00072600
IQCMGN(NQCMGN)=IG	00072700
ISTGND(IG)=14	00072800
ICARST(IC)=15	00072900
END IF	00073000
END IF	00073100
30 CONTINUE	00073200
END IF	00073300
C	00073400
100 RETURN	00073500
END	00073600
C	00073700
C	00073800
C *** CHANGE COMMUNICATION TIME PARAMETER ***	00073900
SUBROUTINE CMTPRM	00074000
COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00074100
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00074200
*, NRQ(1100), ITPGND(1100)	00074300
*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00074400
*, TPSGND(1000), INXTGN(1000)	00074500
*/QDAT/IQCMGN(20), NQCMGN, IQRSGN(20), NQRSGN	00074600
*, IQCMNV(50), NQCMNV, IQRSMV(50), NQRSMV, IQCNGN(30), NQCNGN	00074700
*/COMDAT/RSPGN, RSPMV, ISGNL(100,2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00074800
*, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00074900
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00075000
*, TSIMMX, TSIM, ICAR, ICLL(1100), NCLL, IRAN	00075100
C	00075200
C ** RESPONSETIME OF MOVEMENT CONTROL DATA **	00075300
IF (ICAR.GT.0) THEN	00075400
DO 10 IC=1, ICAR	00075500
IF ((ICARST(IC).EQ.5).OR.(ICARST(IC).EQ.13)) THEN	00075600
TRMVD(IC)=TRMVD(IC)-PSTIM	00075700
IF (TRMVD(IC).LE.0) THEN	00075800
NQCMNV=NQCMNV+1	00075900
IQCMNV(NQCMNV)=IC	00076000
END IF	00076100
END IF	00076200
10 CONTINUE	00076300
END IF	00076400
C	00076500

C ** RESPONSE TIME OF GENERAL DATA **	00076600
IF (IGND.GT.0) THEN	00076700
DO 20 IG=1,IGND	00076800
IF ((ISTGND(IG).GT.0).AND.(ISTGND(IG).LT.20)) THEN	00076900
TPSGND(IG)=TPSGND(IG)+PSTIM	00077000
IF (ISTGND(IG).EQ.12) THEN	00077100
TRGND(IG)=TRGND(IG)-PSTIM	00077200
IF (TRGND(IG).LE.0.) THEN	00077300
ISTGND(IG)=13	00077400
END IF	00077500
END IF	00077600
END IF	00077700
20 CONTINUE	00077800
END IF	00077900
C	00078000
C ** TRIGGER GENERATION **	00078100
TIWTRG=TIWTRG-PSTIM	00078200
IF (TIWTRG.LE.0.) THEN	00078300
30 NCMTRG=NCMTRG+1	00078400
TIWTRG=TIWTRG+TITRG	00078500
IF (TIWTRG.GT.0.) GOTO 40	00078600
GOTO 30	00078700
40 CONTINUE	00078800
END IF	00078900
C	00079000
RETURN	00079100
END	00079200

A. 1. 2. 新方式の評価プログラム

C** COMMUNICATION SIMULATION FOR NEW METHOD *****	00000100
C	00000200
COMMON /GDAT/IENTRY, IDMY, NDEV, NATN	00000300
*, NGDRR1, NGDRR2, NGDRD, NGDRZ, NGDRC, NGDRM	00000400
*, IOPTR1, IOPTR2, IOPTD, IOPTZ, IOPTC, IOPTM	00000500
*, GXL, GXM, GYL, GYM	00000600
*/EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00000700
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00000800
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00000900
*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00001000
*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00001100
*, TPSGND(1000), INXTGN(1000)	00001200
*/QDAT/IQCMGN(20), NQCMGN	00001300
*, IQCMNV(50), NQCMNV, IQCNGN(100), NQCNGN	00001400
*/COMDAT/RSPGN, RSPMV, ISGNL(1000, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00001500
*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00001600
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00001700
*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00001800
C	00001900
1112 CONTINUE	00002000
READ(5, *) IEXEC	00002100
IF (IEXEC.EQ.0) GOTO 1111	00002200
C	00002300
C DATA PCALL/1.0, 999*0.5/	00002400

	PCALL(1)=1.0	00002500
	DO 111 IX=2,1000	00002600
111	PCALL(IX)=0.5	00002700
C		00002800
C	DATA TTRIG, TRQ, TACK, TMVD, TGND, TWRQ	00002900
C	* /0.00604, 0.0, 0.0, 0.0170, 0.0614, 0.0003/	00003000
	TTRIG=0.00604	00003100
	TRQ=0.0	00003200
	TACK=0.0	00003300
	TMVD=0.0170	00003400
	TGND=0.0614	00003500
	TWRQ=0.0003	00003600
C		00003700
C	DATA ICARST/1100*0/	00003800
	DO 112 IX=1,1100	00003900
112	ICARST(IX)=0	00004000
C		00004100
C	DATA NQCMGN, NQCMMV, NQCNGN, NCMTRG, NSGNL/5*0/	00004200
	NQCMGN=0	00004300
	NQCMMV=0	00004400
	NQCNGN=0	00004500
	NCMTRG=0	00004600
	NSGNL=0	00004700
C		00004800
C	DATA ISTGND/1000*0/	00004900
	DO 113 IX=1,1000	00005000
113	ISTGND(IX)=0	00005100
C		00005200
C	DATA IGND, ICAR/2*0/	00005300
	IGND=0	00005400
	ICAR=0	00005500
C		00005600
C	DATA TGNRMX, TGNRMN, FGNRMX/100., 0.7, 1.5/	00005700
	TGNRMX=100.	00005800
	TGNRMN=0.7	00005900
	FGNRMX=1.5	00006000
C		00006100
C	DATA IRAN/0/	00006200
	IRAN=0	00006300
C		00006400
C	DATA DD, DL, NL, NZ/16.0, 10.0, 6.3/	00006500
	DD=16.0	00006600
	DL=10.0	00006700
	NL=6	00006800
	NZ=3	00006900
C		00007000
C	DATA XLC/4.0/	00007100
	XLC=4.0	00007200
C		00007300
C	DATA GXL, GXM, GYL, GYM/50.0, 4050.0, 500.0, 2500.0/	00007400
	GXL=50.0	00007500
	GXM=4050.0	00007600
	GYL=500.0	00007700
	GYM=2500.0	00007800
C		00007900

C	DATA RWC/0.6/ RWC=0.6	00008000 00008100
C		00008200
C	DATA QLGNR/6*1300./ DO 114 IX=1,6	00008300 00008400
114	QLGNR(IX)=1300.	00008500
C		00008600
C	DATA VLGNR/6*10./ DO 115 IX=1,6	00008700 00008800
115	VLGNR(IX)=10.0	00008900
C		00009000
C	DATA WMVD/0.6/ WMVD=0.6	00009100 00009200
C		00009300
C	DATA IRMVST/1100*0/ DO 116 IX=1,1100	00009400 00009500
116	IRMVST(IX)=0	00009600
C		00009700
C	DATA IWMVST/1100*0/ DO 117 IX=1,1100	00009800 00009900
117	IWMVST(IX)=0	00010000
C		00010100
C	TCNTL=0.0114	00010200
C		00010300
	XLR=DL+FLOAT(NZ-1)*(DD+DL)	00010400
	DO 40 IZ=1,NZ	00010500
	XZN(IZ,1)=FLOAT(IZ-1)*(DD+DL)	00010600
	XZN(IZ,2)=XZN(IZ,1)+DL	00010700
40	CONTINUE	00010800
C		00010900
C	GYM=GYM*50./XLR	00011000
C		00011100
	READ(5,*) IPRTGN	00011200
	READ(5,*) PPCKT	00011300
	READ(5,*) LPCKT	00011400
	READ(5,*) RSPMV	00011500
	READ(5,*) RSPGN	00011600
	READ(5,*) TITRG	00011700
	READ(5,*) TSIMMX	00011800
	WRITE(6,*) '*****'	00011900
	WRITE(6,*) 'PRINTOUT STATUS?(YES=1) -> ',IPRTGN	00012000
	WRITE(6,*) 'PROBABILITY OF GENERATING DATA -> ',PPCKT	00012100
	WRITE(6,*) 'LENGTH OF GENERATING DATA -> ',LPCKT	00012200
	WRITE(6,*) 'RESPONSE TIME OF MOVEMENT DATA -> ',RSPMV	00012300
	WRITE(6,*) 'RESPONSE TIME OF GENERAL DATA -> ',RSPGN	00012400
	WRITE(6,*) 'INTERVAL TIME OF TRIGGER -> ',TITRG	00012500
	WRITE(6,*) 'TOTAL SIMULATION TIME -> ',TSIMMX	00012600
	WRITE(6,*)	00012700
	TSIM=0.0	00012800
C		00012900
C	*** COMMUNICATION PARAMETER INITIALIZATION ***	00013000
	TIWTRG=TITRG	00013100
	PSTIM=0.	00013200
C		00013300
C	*** GENERATOR INITIALIZATION ***	00013400

CALL GNRINI	00013500
C	00013600
C *** SIMULARION START ***	00013700
100 TSIM=TSIM+PSTIM	00013800
IF (TSIM.GE.TSIMMX) GOTO 200	00013900
NSGNL=0	00014000
CALL CRGMR	00014100
CALL CARMV	00014200
CALL CMZPRM	00014300
CALL CMTPRM	00014400
CALL CMCNTL	00014500
ISG=1	00014600
60 CONTINUE	00014700
IF (ISGNL(ISG,1).NE.1) GOTO 70	00014800
IF (ISG.GE.NSGNL) GOTO 50	00014900
ISG=ISG+1	00015000
GOTO 60	00015100
70 CONTINUE	00015200
C	00015300
C *** DISPLAYING STATUS OF COMMUNICATION ***	00015400
IF (IPRTGN.EQ.1) THEN	00015500
CALL PRTGND	00015600
END IF	00015700
C	00015800
50 CONTINUE	00015900
C	00016000
GOTO 100	00016100
200 CONTINUE	00016200
C	00016300
C *** CALCULATING RESULTS ***	00016400
C ** CALCULATING FAILURE RATE **	00016500
IF (ICAR.GT.0) THEN	00016600
IFCNCT=0	00016700
IFRSP=0	00016800
IFNSH=0	00016900
RFRSP=0.	00017000
RFCNCT=0.	00017100
DO 300 IC=1,ICAR	00017200
IF (ICARST(IC).GE.20) THEN	00017300
IFNSH=IFNSH+1	00017400
IF ((ICARST(IC).EQ.23).OR.(ICARST(IC).EQ.20))	00017500
* IFCNCT=IFCNCT+1	00017600
IF ((ICARST(IC).EQ.24).OR.(ICARST(IC).EQ.21))	00017700
* IFRSP=IFRSP+1	00017800
END IF	00017900
300 CONTINUE	00018000
IF (IFNSH.GT.0) THEN	00018100
RFCNCT=FLOAT(IFCNCT)/FLOAT(IFNSH)	00018200
RFRSP=FLOAT(IFRSP)/FLOAT(IFNSH)	00018300
ELSE	00018400
WRITE(6,*) 'NO CARS FINISH.'	00018500
RFCNCT=-1.	00018600
RFRSP=-1.	00018700
END IF	00018800
END IF	00018900

C ** CALCULATING DELAY TIME OF DATA AND THROUGHPUT **	00019000
TDRY=0.	00019100
ITBLK=0	00019200
ADRYT=0.	00019300
STH=0.	00019400
IF (IGND.GT.0) THEN	00019500
DO 400 IG=1,IGND	00019600
IF (1STGND(IG).GT.0) THEN	00019700
ITBLK=ITBLK+(LPCKT-IBGND(IG))	00019800
TDRY=TDRY+TPSGND(IG)	00019900
END IF	00020000
400 CONTINUE	00020100
TCOM=FLOAT(ITBLK)*(TGND-TCNTL)	00020200
* +FLOAT(IFNSH-IFRSP-IFCNCT)*(TMVD-TCNTL)	00020300
STH=TCOM/TSIM	00020400
IF (ITBLK.GT.0) THEN	00020500
ADRYT=(TDRY/FLOAT(ITBLK))*LPCKT	00020600
ELSE	00020700
WRITE(6,*) 'NO DATA CUMMUNICATED SUCCESFULLY.'	00020800
ADRYT=1000.	00020900
END IF	00021000
END IF	00021100
C	00021200
C *** PRINTOUT RESULTS ***	00021300
WRITE(6,*) 'RESULTS OF NEW METHOD'	00021400
WRITE(6,*) 'GENERATED CARS=',ICAR	00021500
WRITE(6,*) 'GENERATED DATA=',IGND	00021600
WRITE(6,*) 'FAILURE RATE OF MISSCONNECT=',RFCNCT	00021700
WRITE(6,*) 'FAILURE RATE OF MISSRESPONSE=',RFRSP	00021800
WRITE(6,*) 'AVERAGE DELAY TIME OF DATA=',ADRYT	00021900
WRITE(6,*) 'THROUGHPUT OF DATA=',STH	00022000
C	00022100
WRITE(2) PPCKT,LPCKT,RSPMV,RSPGN,TITRG,RFCNCT,RFRSP,ADRYT,STH	00022200
C	00022300
GOTO 1112	00022400
1111 CONTINUE	00022500
C	00022600
STOP	00022700
END	00022800
C	00022900
C	00023000
C *** GENERATE CARS ***	00023100
SUBROUTINE CRGMR	00023200
COMMON /EDAT/XLR,XLC,DD,DL,NL,NZ,XZN(10,2),RWC	00023300
*/CARDAT/CARPOS(1100),CARVEL(1100),ICARLN(1100)	00023400
*,ICARST(1100),PBOUT(1100),TRMVD(1100),TWMVD(1100)	00023500
*,NRQ(1100),ITPGND(1100),IRMVST(1100),IWMVST(1100)	00023600
*/QDAT/IQCMGN(20),NQCNGN	00023700
*,IQCMNV(50),NQCMMV,IQCNGN(100),NQCNGN	00023800
*/COMDAT/RSPGN,RSPMV,ISGNL(1000,2),NCMTRG,TITRG,TIWTRG,PSTIM,IGND	00023900
*,WMVD,NSGNL,TTRIG,TRQ,TACK,TMVD,TGND,TWRQ,PCALL(1000),LPCKT,PPCKT	00024000
*/SIMDAT/TWGNR(6),QLGMR(6),VLGMR(6),TGNRMX,TGNRMN,FGNRMX	00024100
*,TSIMMX,TSIM,ICAR,ICLL(20),NCLL,IRAN	00024200
C	00024300
C	00024400

DO 10 IL=1,NL	00024500
TWGNR(IL)=TWGNR(IL)-PSTIM	00024600
IF (TWGNR(IL).LE.0.0) THEN	00024700
C	00024800
C ** SET INITIAL CAR PARM **	00024900
ICAR=ICAR+1	00025000
ICARST(ICAR)=1	00025100
CARVEL(ICAR)=VLGNR(IL)	00025200
ICARLN(ICAR)=IL	00025300
CARPOS(ICAR)=0.	00025400
NRQ(ICAR)=0	00025500
CALL RANU2(IRAN,RAND,1,ICON)	00025600
PBOUT(ICAR)=XZN(1,1)+(XZN(1,2)-XZN(1,1))*RAND	00025700
C	00025800
C ** RESET TWGNR **	00025900
100 CONTINUE	00026000
CALL RANU2(IRAN,RAND,1,ICON)	00026100
TG=TGNRMN+(TGNRMX-TGNRMN)*RAND	00026200
CALL RANU2(IRAN,RAND,1,ICON)	00026300
PG=RAND*FGNRMX	00026400
IF (GNRFNC(QLGNR(IL),TGNRMN,TG).LE.PG) GOTO 100	00026500
TWGNR(IL)=TG	00026600
C	00026700
END IF	00026800
10 CONTINUE	00026900
RETURN	00027000
END	00027100
C	00027200
C	00027300
C *** GENERATOR INITIALIZATION ***	00027400
SUBROUTINE GNRINI	00027500
COMMON /EDAT/XLR,XLC,DD,DL,NL,NZ,XZN(10,2),RWC	00027600
*/SIMDAT/TWGNR(6),QLGNR(6),VLGNR(6),TGNRMX,TGNRMN,FGNRMX	00027700
*,TSIMMX,TSIM,ICAR,ICLL(20),NCLL,IRAN	00027800
C	00027900
DO 10 IL=1,NL	00028000
100 CONTINUE	00028100
CALL RANU2(IRAN,RAND,1,ICON)	00028200
TG=TGNRMN+(TGNRMX-TGNRMN)*RAND	00028300
CALL RANU2(IRAN,RAND,1,ICON)	00028400
PG=RAND*FGNRMX	00028500
IF (GNRFNC(QLGNR(IL),TGNRMN,TG).LE.PG) GOTO 100	00028600
TWGNR(IL)=TG	00028700
C	00028800
10 CONTINUE	00028900
RETURN	00029000
END	00029100
C	00029200
C	00029300
C *** GERARATOR FUNCTION OF CARS ***	00029400
REAL FUNCTION GNRFNC(Q,T0,T)	00029500
RAMDA=Q/3600.	00029600
IF (T.GT.T0) THEN	00029700
TRAMDA=1./RAMDA	00029800
FGNR=EXP(-(T-T0)/(TRAMDA-T0))	00029900

ELSE	00030000
FGNR=0.	00030100
END IF	00030200
GNRFNC=FGNR	00030300
RETURN	00030400
END	00030500
C	00030600
C	00030700
C *** MOVING CARS ***	00030800
SUBROUTINE CARMV	00030900
COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00031000
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00031100
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00031200
*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00031300
*/COMDAT/RSPGN, RSPMV, ISGNL(1000, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00031400
*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00031500
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00031600
*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00031700
C	00031800
IF (ICAR.GT.0) THEN	00031900
DO 10 IIC=1, ICAR	00032000
IF (ICARST(IIC).GT.0) THEN	00032100
IF (CARPOS(IIC).LE.XLR) THEN	00032200
CARPOS(IIC)=CARPOS(IIC)+PSTIM*CARVEL(IIC)*1000./3600.	00032300
END IF	00032400
ENDIF	00032500
10 CONTINUE	00032600
END IF	00032700
RETURN	00032800
END	00032900
C	00033000
C	00033100
C *** PRINT OUT STATUS OF DATA, CARS AND COMMUNICATED SIGNALS ***	00033200
SUBROUTINE PRTGND	00033300
COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00033400
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00033500
*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00033600
*/GNDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00033700
*, TPSGND(1000), INXTGN(1000)	00033800
*/QDAT/IQCMGN(20), NQCMGN	00033900
*, IQCMMV(50), NQCMMV, IQCNGN(100), NQCNGN	00034000
*/COMDAT/RSPGN, RSPMV, ISGNL(1000, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00034100
*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00034200
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00034300
*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00034400
C	00034500
WRITE(6, 3000)	00034600
WRITE(6, *) 'SIMULATION TIME=', TSIM	00034700
WRITE(6, *)	00034800
WRITE(6, *) 'COMMUNICATED SIGNALS'	00034900
IF (NSGNL.EQ.0) THEN	00035000
WRITE(6, *) 'NON SIGNALS COMMUNICATED'	00035100
ELSE	00035200
WRITE(6, *) ' SIGNAL CAR'	00035300
DO 500 IS=1, NSGNL	00035400

IF (ISGNL(IS,1).EQ.1) THEN	00035500
IF (ISGNL(IS,2).EQ.0) THEN	00035600
WRITE(6,*) ' TRG DUMMY'	00035700
ELSE	00035800
WRITE(6,*) ' TRG ' , ISGNL(IS,2)	00035900
END IF	00036000
ELSE IF (ISGNL(IS,1).EQ.4) THEN	00036100
WRITE(6,*) 'MOV CON ' , ISGNL(IS,2)	00036200
ELSE IF (ISGNL(IS,1).EQ.5) THEN	00036300
WRITE(6,*) 'GEN DAT ' , ISGNL(IS,2)	00036400
END IF	00036500
500 CONTINUE	00036600
END IF	00036700
WRITE(6,*) 'END OF SIGNLS'	00036800
WRITE(6,*)	00036900
WRITE(6,*) 'STATUS OF CARS'	00037000
IF (ICAR.EQ.0) THEN	00037100
WRITE(6,*) 'NO CARS EXIST.'	00037200
ELSE	00037300
WRITE(6,*)	00037400
* NO. STA POS LANE TOP-GND R-MV W-MV'	00037500
DO 600 IC=1, ICAR	00037600
IF (((ICARST(IC).GE.10).AND.(ICARST(IC).LT.20)).OR.	00037700
* ((ICARST(IC).GE.23).AND.(ICARST(IC).LE.27))) THEN	00037800
WRITE(6,4000) IC, ICARST(IC), CARPOS(IC), ICARLN(IC), ITPGND(IC)	00037900
* , IRMVST(IC), IWMVST(IC)	00038000
ELSE	00038100
* IF (((ICARST(IC).GE.1).AND.(ICARST(IC).LT.10)).OR.	00038200
* ((ICARST(IC).GE.20).AND.(ICARST(IC).LT.23))) THEN	00038300
WRITE(6,4500) IC, ICARST(IC), CARPOS(IC), ICARLN(IC)	00038400
* , IRMVST(IC), IWMVST(IC)	00038500
END IF	00038600
600 CONTINUE	00038700
END IF	00038800
WRITE(6,*) 'END OF CARS'	00038900
WRITE(6,*)	00039000
WRITE(6,*) 'STATUS OF GENERAL DATA'	00039100
IF (IGND.EQ.0) THEN	00039200
WRITE(6,*) 'NO DATA EXIST.'	00039300
ELSE	00039400
WRITE(6,*) ' NO. STA CAR NEXT REMAINT'	00039500
DO 300 IG=1, IGND	00039600
300 WRITE(6,1000) IG, ISTGND(IG), ICRGND(IG), INXTGN(IG), IBGND(IG)	00039700
END IF	00039800
WRITE(6,*) 'END OF GENERAL DATA'	00039900
WRITE(6,*)	00040000
WRITE(6,*) 'QUE FOR CONT.'	00040100
IF (NQCNGN.EQ.0) THEN	00040200
WRITE(6,*) 'NO QUE EXIST.'	00040300
ELSE	00040400
WRITE(6,*) ' NO. DATA'	00040500
DO 400 IQ=1, NQCNGN	00040600
400 WRITE(6,2000) IQ, IQCNGN(IQ)	00040700
END IF	00040800
WRITE(6,*) 'END OF QUE'	00040900

WRITE(6,*)	00041000
C	00041100
1000 FORMAT(1H , 6I10)	00041200
2000 FORMAT(1H , 2I10)	00041300
3000 FORMAT(1H1)	00041400
4000 FORMAT(1H , 2I10, F10. 3, 2I10, 2I5)	00041500
4500 FORMAT(1H , 2I10, F10. 3, I10, ' NO DATA', 2I5)	00041600
RETURN	00041700
END	00041800
C	00041900
C	00042000
C *** CHANGE STATUS OF CARS AND DATA BY OUT OR INTO ZONE ***	00042100
SUBROUTINE CMZPRM	00042200
COMMON /EDAT/XLR, XLC, DD, DL, NL, NZ, XZN(10, 2), RWC	00042300
*/CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00042400
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00042500
*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00042600
*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00042700
*, TPSGND(1000), INXTGN(1000)	00042800
*/QDAT/IQCMGN(20), NQCMGN	00042900
*, IQCMMV(50), NQCMMV, IQCNGN(100), NQCNGN	00043000
*/COMDAT/RSPGN, RSPMV, ISGNL(1000, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00043100
*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00043200
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00043300
*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00043400
C	00043500
ZI=XZN(2, 1)	00043600
ZO=XZN(2, 2)	00043700
ZPO=XZN(1, 2)	00043800
C	00043900
IF (ICAR.GT.0) THEN	00044000
DO 10 IIC=1, ICAR	00044100
IST=ICARST(IIC)	00044200
IF ((IST.GT.0).AND. (IST.LT.20)) THEN	00044300
XC=CARPOS(IIC)	00044400
IZOGND=0	00044500
C	00044600
IF (IST.EQ.1) THEN	00044700
IF (XC.GT.PBOUT(IIC)) THEN	00044800
IST=2	00044900
TRMVD(IIC)=RSPMV	00045000
TWMVD(IIC)=WMVD	00045100
END IF	00045200
ELSE IF (IST.EQ.2) THEN	00045300
IF (XC.GT.ZPO) THEN	00045400
IST=3	00045500
C	00045600
C ** CONT. GENERAL DATA **	00045700
IF (NQCNGN.GT.0) THEN	00045800
C	00045900
C * CHANGE STATUS OF DATA *	00046000
IG=IQCNGN(1)	00046100
15 ISTG=ISTGND(IG)	00046200
IF (ISTG.EQ.21) THEN	00046300
ISTG=11	00046400

	ELSE IF (ISTG.EQ.22) THEN	00046500
	ISTG=12	00046600
	ELSE IF (ISTG.EQ.23) THEN	00046700
	ISTG=13	00046800
	ELSE IF (ISTG.EQ.24) THEN	00046900
	ISTG=13	00047000
	ELSE	00047100
	WRITE(6,*) 'ERROR IN CHECKPOINT-1 OF SUBROUTINE CMZPRM!'	00047200
	END IF	00047300
	ISTGND(IG)=ISTG	00047400
	IG=INXTGN(IG)	00047500
	IF (IG.EQ.0) GOTO 25	00047600
	GOTO 15	00047700
25	CONTINUE	00047800
C		00047900
C	* MOV DATA TO CAR *	00048000
	IG=IQCNGN(1)	00048100
16	ICRGND(IG)=IIC	00048200
	IG=INXTGN(IG)	00048300
	IF (IG.EQ.0) GOTO 26	00048400
	GOTO 16	00048500
26	CONTINUE .	00048600
	ITPGND(IIC)=IQCNGN(1)	00048700
C		00048800
C	* REMOVE DATA FROM CONT. QUE *	00048900
	NQCNGN=NQCNGN-1	00049000
	IF (NQCNGN.GT.0) THEN	00049100
	DO 20 IQ=1,NQCNGN	00049200
20	IQCNGN(IQ)=IQCNGN(IQ+1)	00049300
	END IF	00049400
C		00049500
C	* CHANGE STATUS OF CAR *	00049600
	IST=11	00049700
	END IF	00049800
C		00049900
C	** GENERATE NEW DATA **	00050000
	CALL RANU2(IRAN,RAND,1,ICON)	00050100
	IF (RAND.LT.PPCKT) THEN	00050200
	IGND=IGND+1	00050300
	ISTGND(IGND)=11	00050400
	ICRGND(IGND)=IIC	00050500
	IBGND(IGND)=LPCKT	00050600
	TPSGND(IGND)=0.0	00050700
	INXTGN(IGND)=0	00050800
	IF(IST.EQ.3) THEN	00050900
	ITPGND(IIC)=IGND	00051000
	IST=11	00051100
	ELSE IF (IST.EQ.11) THEN	00051200
	IG=ITPGND(IIC)	00051300
17	IF (INXTGN(IG).EQ.0) GOTO 27	00051400
	IG=INXTGN(IG)	00051500
	GOTO 17	00051600
27	CONTINUE	00051700
	INXTGN(IG)=IGND	00051800
	ELSE	00051900

	WRITE(6,*) 'ERROR IN CHECKPOINT-2 OF SUBROUTINE CMZPRM!'	00052000
	END IF	00052100
	END IF	00052200
C		00052300
	END IF	00052400
	ELSE IF (IST.EQ.3) THEN	00052500
	IF (XC.GT.ZI) IST=4	00052600
	ELSE IF (IST.EQ.4) THEN	00052700
	IF (XC.GT.ZO) THEN	00052800
	IST=20	00052900
	IF ((IRMVST(IIC).EQ.1).AND.(IWMVST(IIC).EQ.1)) THEN	00053000
C *	REMOVE CARS FROM TRIGGER QUE (HOLDING NO DATA)*	00053100
	IF (NQCMMV.GT.0) THEN	00053200
	IQCM=1	00053300
13	CONTINUE	00053400
	IF (IQCMMV(IQCM).EQ.IIC) GOTO 23	00053500
	IF (IQCM.GE.NQCMMV) GOTO 33	00053600
	IQCM=IQCM+1	00053700
	GOTO 13	00053800
23	NQCMMV=NQCMMV-1	00053900
	IF (NQCMMV.GE.IQCM) THEN	00054000
	DO 43 II=IQCM,NQCMMV	00054100
43	IQCMMV(II)=IQCMMV(II+1)	00054200
	END IF	00054300
	GOTO 130	00054400
33	WRITE(6,*) 'ERROR IN CHECKPOINT 3 IN CMZPRM!'	00054500
130	CONTINUE	00054600
	ELSE	00054700
	WRITE(6,*) 'ERROR IN CHECKPOINT 4 IN CMZPRM!'	00054800
	END IF	00054900
	END IF	00055000
C		00055100
	END IF	00055200
	ELSE IF (IST.EQ.11) THEN	00055300
	IF (XC.GT.ZI) IST=12	00055400
	ELSE IF (IST.EQ.12) THEN	00055500
	IF (XC.GT.ZO) THEN	00055600
	IST=23	00055700
	IF ((IRMVST(IIC).EQ.1).AND.(IWMVST(IIC).EQ.1)) THEN	00055800
C *	REMOVE CARS FROM TRIGGER QUE (HOLDING DATA)*	00055900
	IF (NQCMMV.GT.0) THEN	00056000
	IQCM=1	00056100
14	CONTINUE	00056200
	IF (IQCMMV(IQCM).EQ.IIC) GOTO 24	00056300
	IF (IQCM.GE.NQCMMV) GOTO 34	00056400
	IQCM=IQCM+1	00056500
	GOTO 14	00056600
24	NQCMMV=NQCMMV-1	00056700
	IF (NQCMMV.GE.IQCM) THEN	00056800
	DO 44 II=IQCM,NQCMMV	00056900
44	IQCMMV(II)=IQCMMV(II+1)	00057000
	END IF	00057100
	GOTO 140	00057200
34	WRITE(6,*) 'ERROR IN CHECKPOINT 6 IN CMZPRM!'	00057300
140	CONTINUE	00057400

ELSE	00057500
WRITE(6,*) 'ERROR IN CHECKPOINT 7 IN CMZPRM!'	00057600
END IF	00057700
END IF	00057800
C	00057900
IZOGND=ITPGND(IIC)	00058000
ENDIF	00058100
ELSE IF (IST.EQ.14) THEN	00058200
IF (XC.GT.ZO) THEN	00058300
IST=25	00058400
IZOGND=ITPGND(IIC)	00058500
ENDIF	00058600
ELSE IF (IST.EQ.15) THEN	00058700
IF (XC.GT.ZO) THEN	00058800
IST=26	00058900
IZOGND=ITPGND(IIC)	00059000
ENDIF	00059100
END IF	00059200
ICARST(IIC)=IST	00059300
C	00059400
C ** ADD DATA TO CONT. QUE **	00059500
IF (IZOGND.GT.0) THEN	00059600
NQCNGN=NQCNGN+1	00059700
IQCNGN(NQCNGN)=IZOGND	00059800
C * CHANGE STATUS OF DATA *	00059900
IG=IZOGND	00060000
11 IST=ISTGND(IG)	00060100
IF (IST.EQ.11) THEN	00060200
IST=21	00060300
ELSE IF (IST.EQ.12) THEN	00060400
IST=22	00060500
ELSE IF (IST.EQ.13) THEN	00060600
IST=23	00060700
ELSE IF (IST.EQ.14) THEN	00060800
IST=24	00060900
C * REMOVE DATA FROM COMMUNICATION QUE *	00061000
IF (NQCMGN.GT.0) THEN	00061100
IQCM=1	00061200
12 CONTINUE	00061300
IF (IQCMGN(IQCM).EQ.IG) GOTO 22	00061400
IF (IQCM.GE.NQCMGN) GOTO 32	00061500
IQCM=IQCM+1	00061600
GOTO 12	00061700
22 NQCMGN=NQCMGN-1	00061800
IF (NQCMGN.GE.IQCM) THEN	00061900
DO 42 II=IQCM, NQCMGN	00062000
42 IQCMGN(II)=IQCMGN(II+1)	00062100
END IF	00062200
GOTO 100	00062300
32 WRITE(6,*) 'ERROR IN CHECKPOINT 9 IN CMZPRM!'	00062400
100 CONTINUE	00062500
ELSE	00062600
WRITE(6,*) 'ERROR IN CHECKPOINT 10 IN CMZPRM!'	00062700
END IF	00062800
ELSE	00062900

	WRITE(6,*) 'ERROR IN CHECKPOINT 11 IN CMZPRM!'	00063000
	END IF	00063100
	ISTGND(IG)=IST	00063200
	IG=INXTGN(IG)	00063300
	IF (IG.EQ.0) GOTO 21	00063400
	GOTO 11	00063500
21	CONTINUE	00063600
C		00063700
	END IF	00063800
	END IF	00063900
10	CONTINUE	00064000
	END IF	00064100
	RETURN	00064200
	END	00064300
C		00064400
C		00064500
C	*** COMMUNICATION CONTROL ***	00064600
	SUBROUTINE CMCNTL	00064700
	COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00064800
	*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00064900
	*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00065000
	*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00065100
	*, TPSGND(1000), INXTGN(1000)	00065200
	*/QDAT/IQCMGN(20), NQCMGN	00065300
	*, IQCMMV(50), NQCMMV, IQCNGN(100), NQCNGN	00065400
	*/COMDAT/RSPGN, RSPMV, ISGNL(1000, 2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00065500
	*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00065600
	*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00065700
	*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00065800
C		00065900
	PSTIM=0.	00066000
C		00066100
C	** TRANSMISSION OF GENERAL DATA **	00066200
	IF (NQCMMV.EQ.0) THEN	00066300
	NCMTRG=0	00066400
	END IF	00066500
	IF ((NQCMGN.GT.0).AND.(NCMTRG.EQ.0)) THEN	00066600
	IG=IQCMGN(1)	00066700
	NQCMGN=NQCMGN-1	00066800
	IF (NQCMGN.GT.0) THEN	00066900
	DO 40 IQ=1, NQCMGN	00067000
40	IQCMGN(IQ)=IQCMGN(IQ+1)	00067100
	END IF	00067200
	PSTIM=PSTIM+TGND	00067300
	IBGND(IG)=IBGND(IG)-1	00067400
	NSGNL=NSGNL+1	00067500
	ISGNL(NSGNL, 1)=5	00067600
	ISGNL(NSGNL, 2)=ICRGND(IG)	00067700
	IF (IBGND(IG).EQ.0) THEN	00067800
	ISTGND(IG)=25	00067900
	IF (INXTGN(IG).EQ.0) THEN	00068000
	ICARST(ICRGND(IG))=27	00068100
	ELSE	00068200
	ICARST(ICRGND(IG))=14	00068300
	ITPGND(ICRGND(IG))=INXTGN(IG)	00068400

	END IF	00068500
	ELSE	00068600
	NQCMMV=NQCMMV+1	00068700
	IQCMGN(NQCMMV)=IG	00068800
	END IF	00068900
C		00069000
C **	TRIGGER GENERATION AND TRANSMISSION OF MOVEMENT CONTROL DATA **	00069100
	ELSE IF (NQCMMV.GT.0) THEN	00069200
	IF (NCMTRG.EQ.0) THEN	00069300
	TIWTRG=TITRG	00069400
	NCMTRG=1	00069500
	END IF	00069600
	DO 50 IT=1,NCMTRG	00069700
	IF (NQCMMV.EQ.0) GOTO 50	00069800
	ICT=IQCMV(1)	00069900
	NQCMMV=NQCMMV-1	00070000
	IF (NQCMMV.GT.0) THEN	00070100
	DO 31 IQ=1,NQCMMV	00070200
31	IQCMV(IQ)=IQCMV(IQ+1)	00070300
	END IF	00070400
	PSTIM=PSTIM+TTRIG	00070500
	NSGNL=NSGNL+1	00070600
	ISGNL(NSGNL,1)=1	00070700
	ISGNL(NSGNL,2)=ICT	00070800
	IF (ICARST(ICT).EQ.4) THEN	00070900
	ICARST(ICT)=22	00071000
	PSTIM=PSTIM+TMVD	00071100
	NSGNL=NSGNL+1	00071200
	ISGNL(NSGNL,1)=4	00071300
	ISGNL(NSGNL,2)=ICT	00071400
	ELSE IF (ICARST(ICT).EQ.12) THEN	00071500
	ICARST(ICT)=14	00071600
	PSTIM=PSTIM+TMVD	00071700
	NSGNL=NSGNL+1	00071800
	ISGNL(NSGNL,1)=4	00071900
	ISGNL(NSGNL,2)=ICT	00072000
	IG=ITPGND(ICT)	00072100
12	IF (ISTGND(IG).EQ.11) THEN	00072200
	TRGND(IG)=RSPGN	00072300
	ISTGND(IG)=12	00072400
	END IF	00072500
	IG=INXTGN(IG)	00072600
	IF (IG.EQ.0) GOTO 11	00072700
	GOTO 12	00072800
11	CONTINUE	00072900
	ELSE IF ((ICARST(ICT).EQ.2).OR.(ICARST(ICT).EQ.3).	00073000
*	OR.(ICARST(ICT).EQ.11)) THEN	00073100
	PSTIM=PSTIM+TWRQ	00073200
	NQCMMV=NQCMMV+1	00073300
	IQCMV(NQCMMV)=ICT	00073400
	ELSE	00073500
	WRITE(6,*) 'ERROR IN CHECKPOINT 2 IN CMCNTL!'	00073600
	END IF	00073700
50	CONTINUE	00073800
	NCMTRG=0	00073900

ELSE	00074000
PSTIM=PSTIM+TTRIG+TWRQ	00074100
NSGNL=NSGNL+1	00074200
ISGNL(NSGNL,1)=1	00074300
ISGNL(NSGNL,2)=0	00074400
END IF	00074500
C	00074600
C ** CHANGE STATUS OF DATA AND CARS BY REPLY OF DATA	00074700
IF (ICAR.GT.0) THEN	00074800
DO 30 IC=1, ICAR	00074900
IF (ICARST(IC).EQ.14) THEN	00075000
IG=ITPGND(IC)	00075100
IF (ISTGND(IG).EQ.13) THEN	00075200
NQCMGN=NQCMGN+1	00075300
IQCMGN(NQCMGN)=IG	00075400
ISTGND(IG)=14	00075500
ICARST(IC)=15	00075600
END IF	00075700
END IF	00075800
30 CONTINUE	00075900
END IF	00076000
C	00076100
100 RETURN	00076200
END	00076300
C	00076400
C	00076500
C *** CHANGE COMMUNICATION TIME PARAMETER ***	00076600
SUBROUTINE CMTPRM	00076700
COMMON /CARDAT/CARPOS(1100), CARVEL(1100), ICARLN(1100)	00076800
*, ICARST(1100), PBOUT(1100), TRMVD(1100), TWMVD(1100)	00076900
*, NRQ(1100), ITPGND(1100), IRMVST(1100), IWMVST(1100)	00077000
*/GNDDAT/ISTGND(1000), ICRGND(1000), TRGND(1000), IBGND(1000)	00077100
*, TPSGND(1000), INXTGN(1000)	00077200
*/QDAT/IQCMGN(20), NQCMGN	00077300
*, IQCMMV(50), NQCMMV, IQCNGN(100), NQCNGN	00077400
*/COMDAT/RSPGN, RSPMV, ISGNL(1000,2), NCMTRG, TITRG, TIWTRG, PSTIM, IGND	00077500
*, WMVD, NSGNL, TTRIG, TRQ, TACK, TMVD, TGND, TWRQ, PCALL(1000), LPCKT, PPCKT	00077600
*/SIMDAT/TWGNR(6), QLGNR(6), VLGNR(6), TGNRMX, TGNRMN, FGNRMX	00077700
*, TSIMMX, TSIM, ICAR, ICLL(20), NCLL, IRAN	00077800
C	00077900
C ** RESPONSETIME OF MOVEMENT CONTROL DATA **	00078000
IF (ICAR.GT.0) THEN	00078100
DO 10 IC=1, ICAR	00078200
IF (((ICARST(IC).EQ.2).OR.(ICARST(IC).EQ.3).OR.	00078300
* (ICARST(IC).EQ.4).OR.(ICARST(IC).EQ.12).OR.	00078400
* (ICARST(IC).EQ.11)).AND.	00078500
* ((IRMVST(IC).EQ.0).OR.(IWMVST(IC).EQ.0))) THEN	00078600
C	00078700
IF (IWMVST(IC).EQ.0) THEN	00078800
TWMVD(IC)=TWMVD(IC)-PSTIM	00078900
IF (TWMVD(IC).LE.0) THEN	00079000
IWMVST(IC)=1	00079100
END IF	00079200
END IF	00079300
IF (IRMVST(IC).EQ.0) THEN	00079400

TRMVD(IC)=TRMVD(IC)-PSTIM	00079500
IF (TRMVD(IC).LE.0) THEN	00079600
IRMVST(IC)=1	00079700
END IF	00079800
END IF	00079900
IF ((IRMVST(IC).EQ.1).AND.(IWMVST(IC).EQ.1)) THEN	00080000
NQCMMV=NQCMMV+1	00080100
IQCMMV(NQCMMV)=IC	00080200
END IF	00080300
END IF	00080400
10 CONTINUE	00080500
END IF	00080600
C	00080700
C ** RESPONSE TIME OF GENERAL DATA **	00080800
IF (IGND.GT.0) THEN	00080900
DO 20 IG=1, IGND	00081000
IF ((ISTGND(IG).GT.0).AND.(ISTGND(IG).LT.20)) THEN	00081100
TPSGND(IG)=TPSGND(IG)+PSTIM	00081200
IF (ISTGND(IG).EQ.12) THEN	00081300
TRGND(IG)=TRGND(IG)-PSTIM	00081400
IF (TRGND(IG).LE.0.) THEN	00081500
ISTGND(IG)=13	00081600
END IF	00081700
END IF	00081800
END IF	00081900
20 CONTINUE	00082000
END IF	00082100
C	00082200
C ** TRIGGER GENERATION **	00082300
TIWTRG=TIWTRG-PSTIM	00082400
IF (TIWTRG.LE.0.) THEN	00082500
30 NCMTRG=NCMTRG+1	00082600
TIWTRG=TIWTRG+TITRG	00082700
IF (TIWTRG.GT.0.) GOTO 40	00082800
GOTO 30	00082900
40 CONTINUE	00083000
END IF	00083100
C	00083200
RETURN	00083300
END	00083400

A. 2. グローバルレベル制御方式の評価プログラム (両方式の比較)

```

C***** SIMULATION FOR GROBAL LEVEL CONTROL *****
      INTEGER IROAD(6,500,6), NIROAD(6,500,6), TIME, F(0:17),
+AC(0:16), IX1, IX2, ICON,
+TE, GECHAR, ENDSGT, NSIG,
+NL, LMAX, NDMAX, LRED, LBLUE
      REAL GETIME(6), Q(6), QH, A(1)
      CHARACTER DSPR(6,500), NDSPR(6,500), CSQ(20), NMQ(10)
      REAL PDAT, BLPS
      INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,
+      NCOMDT(2), NCOMCT(2)
      INTEGER CTBF(100), NCTBSZ, ICTWR, ICTRD, IWERR
      REAL RQ(500), RP(500), RDC(500), RCC(500), RDN(500), RCN(500),
+      RMMC(500), RMMN(500), RAMC(500), RAMN(500)
      INTEGER NRSLT, NRLD(500), NRWS(500)
      INTEGER MEMMAX(2)
      REAL AVRMEM(2)

C
      COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,
+      /FT/F, AC, /NMQ/NMQ,
+      /NRD/NIROAD, /CNRD/NDSPR, /RND/IX1, IX2,
+      /SIG/NSIG, ENDSGT, LRED, LBLUE, /NMAX/NL, LMAX, NDMAX,
+      /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,
+      NCOMDT, NCOMCT, PDAT, BLPS,
+      /CNT/CTBF, NCTBSZ, ICTWR, ICTRD, /ERR/IWERR

C
C      IWM=11
C      IWC=12
C      IWR=10
      IWR=6
      IWERR=6
      IWF=1

C
C      OPEN(IWM, FILE='AA30850. CARMOV. DATA', STATUS='OLD')
C      OPEN(IWC, FILE='AA30850. COMEXE. DATA', STATUS='OLD')
C      OPEN(IWR, FILE='AA30850. RESULT. DATA', STATUS='OLD')
C
      NL=6
      NDMAX=6
      NCTBSZ=100

C
      LZONE=2
      LDEAD=3
      BPS=9600.
      LBDAT=500
      DWNRT=0.5
      VF=67.5
      RKJ=107.

C
      NRSLT=0

1010 READ(5,100) ICNTL
100  FORMAT(I1)
      IF(ICNTL.EQ.0) GOTO 1000

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900
00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200

```

C	READ(5, 110) LRED, LBLUE, PDAT, LDAT, NWSIZ, QH, LMAX, TE, IX1, IX2	00005300
110	FORMAT(2I5, F5. 2, 2I5, F5. 0, 2I5, 2I2)	00005400
C		00005500
	WRITE(IWR, 200)	00005600
	WRITE(IWR, 210) PDAT	00005700
	WRITE(IWR, 220) LDAT	00005800
	WRITE(IWR, 230) NWSIZ	00005900
	WRITE(IWR, 240) QH	00006000
	WRITE(IWR, 250) LMAX	00006100
	WRITE(IWR, 260) TE	00006200
	WRITE(IWR, 270) IX1, IX2	00006300
	WRITE(IWR, 280)	00006400
		00006500
C		00006600
C	WRITE(IWC, 200)	00006700
C	WRITE(IWC, 210) PDAT	00006800
C	WRITE(IWC, 220) LDAT	00006900
C	WRITE(IWC, 230) NWSIZ	00007000
C	WRITE(IWC, 240) QH	00007100
C	WRITE(IWC, 250) LMAX	00007200
C	WRITE(IWC, 260) TE	00007300
C	WRITE(IWC, 270) IX1, IX2	00007400
C	WRITE(IWC, 280)	00007500
C		00007600
200	FORMAT(1H1, '***** CONDITION SETUP *****')	00007700
210	FORMAT(1H, 'DATA GENERATION PROB. (/ZONE)=', F5. 2)	00007800
220	FORMAT(1H, 'DATA LENGTH(BLOCKS)=', I5)	00007900
230	FORMAT(1H, 'WINDOW SIZE(BLOCKS)=', I2)	00008000
240	FORMAT(1H, 'TRAFFIC(VEH/H*LANE)=', F5. 0)	00008100
250	FORMAT(1H, 'LENGTH OF SIMULATED AREA(BLOCK)=', I5)	00008200
260	FORMAT(1H, 'TOTAL SIMULATION TIME(SEC)=', I5)	00008300
270	FORMAT(1H, 'RANDOMIZE PARAM. RND1 RND2 =', I2, 1X, I2)	00008400
280	FORMAT(1H, '*****')	00008500
C		00008600
	LDIST=LZONE+LDEAD	00008700
	NZONE=INT(LMAX/LDIST)	00008800
	STHRU=0. 55 -(0. 0132/(3600. /(QH*FLOAT(NL))))	00008900
	BLPS=(BPS*STHRU/FLOAT(LBDAT))*DWNRT	00009000
	AVRSPD=(VF/2. 0)*(1+(1. 0-(4. 0*QH/(RKJ*VF)))*0. 5)	00009100
	AVRTRP=FLOAT(LMAX)*5. 0/(AVRSPD*1000./3600.)	00009200
	AVRDEN=(QH/AVRSPD)/1000.	00009300
	NADAT=BLPS*(FLOAT(LZONE)/FLOAT(LDIST))*AVRTRP/(1+(NL-1)*AVRDEN*10)	00009400
C		00009500
	WRITE(IWR, 300)	00009600
	WRITE(IWR, 310) STHRU	00009700
	WRITE(IWR, 320) BLPS	00009800
	WRITE(IWR, 330) NADAT	00009900
	WRITE(IWR, 340)	00010000
C		00010100
C	WRITE(IWC, 300)	00010200
C	WRITE(IWC, 310) STHRU	00010300
C	WRITE(IWC, 320) BLPS	00010400
C	WRITE(IWC, 330) NADAT	00010500
C	WRITE(IWC, 340)	00010600
C		00010700

300	FORMAT(1H , '***** CALCULATED DATA *****')	00010800
310	FORMAT(1H , 'THROUGHPUT=', F5.2)	00010900
320	FORMAT(1H , 'COMMUNICABLE BLOCKS /SEC=', F5.2)	00011000
330	FORMAT(1H , 'EXPECTED COMUNICABLE BLOCKS /LINK*VEH=', I5)	00011100
340	FORMAT(1H , '*****')	00011200
C		00011300
	DO 12 I=1, NL	00011400
12	Q(I)=3600./QH	00011500
C		00011600
	DO 13 I=1, 2	00011700
	NCOMDT(I)=0	00011800
13	NCOMCT(I)=0	00011900
	DO 14 I=1, 2	00012000
	AVRMEM(I)=0.	00012100
14	MEMMAX(I)=0	00012200
	ICTWR=1	00012300
	ICTRD=1	00012400
C		00012500
	DO 10 I=1, NL	00012600
	DO 10 J=1, LMAX	00012700
	IROAD(I, J, 1)=0	00012800
	DSPR(I, J)='.'	00012900
10	CONTINUE	00013000
	DO 11 I=1, NL	00013100
	CALL RANE2(Q(I)-1, IX1, A, 1, ICON)	00013200
	GETIME(I)=1+A(1)	00013300
11	CONTINUE	00013400
	GECHAR=1	00013500
	NSIG=1	00013600
	ENDSGT=LRED	00013700
C		00013800
	DO 20 TIME=1, TE	00013900
	CALL STCAR(Q)	00014000
	CALL SIGCN	00014100
	CALL COMZN	00014200
C		00014300
	MMCT=0	00014400
	IF(ICTWR.GT. ICTRD) THEN	00014500
	DO 21 I=ICTRD, (ICTWR-1)	00014600
21	MMCT=MMCT+CTBF(I)	00014700
	ELSE IF(ICTRD.GT. ICTWR) THEN	00014800
	DO 22 I=ICTRD, NCTBSZ	00014900
22	MMCT=MMCT+CTBF(I)	00015000
	DO 23 I=1, (ICTWR-1)	00015100
23	MMCT=MMCT+CTBF(I)	00015200
	ENDIF	00015300
	MMSIZ1=MMCT	00015400
	MMSIZ2=MMCT	00015500
	DO 15 I=1, NL	00015600
	DO 15 J=1, LMAX	00015700
	IF(IROAD(I, J, 1).EQ. 1) THEN	00015800
	MMSIZ1=MMSIZ1+(IROAD(I, J, 3)-IROAD(I, J, 4))+	00015900
+	INT(J/LDIST)*IROAD(I, J, 4)	00016000
	MMSIZ2=MMSIZ2+IROAD(I, J, 5)+	00016100
+	INT(J/LDIST)*IROAD(I, J, 6)	00016200

	ENDIF	00016300
15	CONTINUE	00016400
	AVRMEM(1)=AVRMEM(1)+FLOAT(MMSIZ1)	00016500
	IF(MMSIZ1.GT.MEMMAX(1)) MEMMAX(1)=MMSIZ1	00016600
	AVRMEM(2)=AVRMEM(2)+FLOAT(MMSIZ2)	00016700
	IF(MMSIZ2.GT.MEMMAX(2)) MEMMAX(2)=MMSIZ2	00016800
C		00016900
C	CALL PRINT(IWM)	00017000
C	CALL PRTCOM(IWC)	00017100
	CALL CARMOV	00017200
20	CONTINUE	00017300
C		00017400
	NRSLT=NRSLT+1	00017500
	RP(NRSLT)=PDAT	00017600
	NRLD(NRSLT)=LDAT	00017700
	NRWS(NRSLT)=NWSIZ	00017800
	RQ(NRSLT)=QH	00017900
	RDC(NRSLT)=FLOAT(NCOMDT(1)*LBDAT)/FLOAT(TE*1000)	00018000
	RCC(NRSLT)=FLOAT(NCOMCT(1))/FLOAT(TE)	00018100
	RDN(NRSLT)=FLOAT(NCOMDT(2)*LBDAT)/FLOAT(TE*1000)	00018200
	RCN(NRSLT)=FLOAT(NCOMCT(2))/FLOAT(TE)	00018300
	DO 16 I=1, 2	00018400
16	AVRMEM(I)=AVRMEM(I)/FLOAT(TE)	00018500
	RMMC(NRSLT)=FLOAT(MEMMAX(1)*LBDAT)/8000.	00018600
	RAMC(NRSLT)=AVRMEM(1)*FLOAT(LBDAT)/8000.	00018700
	RMMN(NRSLT)=FLOAT(MEMMAX(2)*LBDAT)/8000.	00018800
	RAMN(NRSLT)=AVRMEM(2)*FLOAT(LBDAT)/8000.	00018900
C		00019000
	WRITE(IWR, 400)	00019100
	WRITE(IWR, 410) RDC(NRSLT), RDN(NRSLT)	00019200
	WRITE(IWR, 420) RCC(NRSLT), RCN(NRSLT)	00019300
	WRITE(IWR, 430)	00019400
C		00019500
C	WRITE(IWC, 400)	00019600
C	WRITE(IWC, 410) RDC(NRSLT), RDN(NRSLT)	00019700
C	WRITE(IWC, 420) RCC(NRSLT), RCN(NRSLT)	00019800
C	WRITE(IWC, 430)	00019900
C		00020000
C		00020100
	WRITE(IWR, 500)	00020200
	WRITE(IWR, 510) RMMC(NRSLT), RMMN(NRSLT)	00020300
	WRITE(IWR, 520) RAMC(NRSLT), RAMN(NRSLT)	00020400
	WRITE(IWR, 530)	00020500
C		00020600
400	FORMAT(1H, '*****SIMULATION RESULT*****')	00020700
410	FORMAT(1H, 'LOAD OF DATA(KBPS) CONV=', F7.3, ' NEW=', F7.3)	00020800
420	FORMAT(1H, 'LOAD OF CONT(TMS/S) CONV=', F7.2, ' NEW=', F7.2)	00020900
430	FORMAT(1H, '*****')	00021000
500	FORMAT(1H, '*****MEMORY UTILIZATION*****')	00021100
510	FORMAT(1H, 'MAX MEM(KBYTE) CONV=', F7.1, ' NEW=', F7.1)	00021200
520	FORMAT(1H, 'AVR MEM(KBYTE) CONV=', F7.1, ' NEW=', F7.1)	00021300
530	FORMAT(1H, '*****')	00021400
C		00021500
C	CLOSE(IWM)	00021600
C	CLOSE(IWC)	00021700

C	CLOSE(IWR)	00021800
C		00021900
	GOTO 1010	00022000
C		00022100
	1000 WRITE(IWR, 600)	00022200
	600 FORMAT(1H1, 'SIMULATION ENDED')	00022300
C		00022400
	WRITE(IWF) NRSLT	00022500
	DO 1030 I=1, NRSLT	00022600
	1030 WRITE(IWF) RP(I), NRLD(I), NRWS(I), RQ(I), RDC(I), RCC(I), RDN(I), RCN(I),	00022700
	+ RMMC(I), RAMC(I), RMMN(I), RAMN(I)	00022800
	WRITE(IWR, 610)	00022900
	610 FORMAT(1H , 'RESULTS ARE SAVED')	00023000
C		00023100
	STOP	00023200
	END	00023300
C		00023400
C		00023500
C		00023600
	BLOCK DATA	00023700
	INTEGER F(0:17), AC(0:16)	00023800
	CHARACTER CSQ(20), NMQ(10)	00023900
	COMMON /FT/F, AC, /CST/CSQ, /NMQ/NMQ	00024000
	DATA F/0, 0, 4, 6, 8, 9, 10, 11, 11, 12, 13, 13, 14, 14, 14, 15, 15, 16/	00024100
	DATA AC/4, 4, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 16, 16/	00024200
	DATA CSQ/'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',	00024300
	+ 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T' /	00024400
	DATA NMQ/'0', '1', '2', '3', '4', '5', '6', '7', '8', '9' /	00024500
	END	00024600
C		00024700
C		00024800
C		00024900
	SUBROUTINE STCAR(Q)	00025000
C		00025100
	REAL Q(6)	00025200
C		00025300
	INTEGER IROAD(6, 500, 6), TIME, IX1, F(0:17), GECHAR	00025400
	REAL GETIME(6)	00025500
	CHARACTER DSPR(6, 500), CSQ(20)	00025600
	COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,	00025700
	+ /RND/IX1, /FT/F, /NMAX/NL, LMAX	00025800
C		00025900
	REAL A(1)	00026000
C		00026100
	DO 10 I=1, NL	00026200
	IF(TIME. GE. GETIME(I)) THEN	00026300
	ISP=F(IFD(I, LMAX))	00026400
	IP=INT(LMAX-ISP*(TIME-GETIME(I))/4)	00026500
	IROAD(I, IP, 1)=1	00026600
	IROAD(I, IP, 2)=ISP	00026700
	CALL STDAT(I, IP)	00026800
	DSPR(I, IP)=CSQ(GECHAR)	00026900
	IF(GECHAR. EQ. 20) THEN	00027000
	GECHAR=1	00027100
	ELSE	00027200

GECHAR=GECHAR+1	00027300
ENDIF	00027400
CALL RANE2(Q(I)-1, IX1, A, 1, ICON)	00027500
GETIME(I)=GETIME(I)+1+A(I)	00027600
ENDIF	00027700
10 CONTINUE	00027800
RETURN	00027900
END	00028000
C	00028100
C	00028200
C	00028300
SUBROUTINE STDAT(I, J)	00028400
C	00028500
INTEGER I, J	00028600
REAL A(1),	00028700
+ PDAT, BLPS	00028800
INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00028900
+ NCOMDT(2), NCOMCT(2),	00029000
+ IROAD(6, 500, 6), IX2, ICON	00029100
C	00029200
COMMON /RD/IROAD,	00029300
+ /RND/IX1, IX2,	00029400
+ /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00029500
+ NCOMDT, NCOMCT, PDAT, BLPS	00029600
C	00029700
CALL RANU2(IX2, A, 1, ICON)	00029800
IF(A(1).LE.PDAT) THEN	00029900
IROAD(I, J, 3)=LDAT	00030000
IROAD(I, J, 5)=LDAT	00030100
ELSE	00030200
IROAD(I, J, 3)=0	00030300
IROAD(I, J, 5)=0	00030400
ENDIF	00030500
LDCNT=LCTRD()	00030600
IROAD(I, J, 3)=IROAD(I, J, 3)+LDCNT	00030700
IROAD(I, J, 5)=IROAD(I, J, 5)+LDCNT	00030800
C	00030900
IF(IROAD(I, J, 3).GT.0) THEN	00031000
NCOMCT(1)=NCOMCT(1)+1	00031100
IF(IROAD(I, J, 3).GT.NWSIZ) THEN	00031200
IROAD(I, J, 4)=NWSIZ	00031300
NCOMDT(1)=NCOMDT(1)+NWSIZ	00031400
ELSE	00031500
IROAD(I, J, 4)=IROAD(I, J, 3)	00031600
NCOMDT(1)=NCOMDT(1)+IROAD(I, J, 3)	00031700
ENDIF	00031800
ELSE	00031900
IROAD(I, J, 4)=0	00032000
ENDIF	00032100
C	00032200
IF(IROAD(I, J, 5).GT.0) THEN	00032300
NCOMCT(2)=NCOMCT(2)+1	00032400
IF(IROAD(I, J, 5).GT.NADAT) THEN	00032500
IROAD(I, J, 6)=NADAT	00032600
NCOMDT(2)=NCOMDT(2)+NADAT	00032700

ELSE	00032800
IROAD(I, J, 6)=IROAD(I, J, 5)	00032900
NCOMDT(2)=NCOMDT(2)+IROAD(I, J, 5)	00033000
ENDIF	00033100
ELSE	00033200
IROAD(I, J, 6)=0	00033300
ENDIF	00033400
C	00033500
RETURN	00033600
END	00033700
C	00033800
C	00033900
C	00034000
SUBROUTINE SIGCN	00034100
C	00034200
INTEGER ENDSGT, NSIG, LRED, LBLUE, TIME	00034300
COMMON /SIG/NSIG, ENDSGT, LRED, LBLUE,	00034400
+ /T/ TIME	00034500
C	00034600
IF(NSIG.EQ.1) THEN	00034700
IF(TIME.GE.ENDSGT) THEN	00034800
NSIG=2	00034900
ENDSGT=TIME+LBLUE	00035000
ENDIF	00035100
ENDIF	00035200
IF(NSIG.EQ.2) THEN	00035300
IF(TIME.GE.ENDSGT) THEN	00035400
NSIG=1	00035500
ENDSGT=TIME+LRED	00035600
ENDIF	00035700
ENDIF	00035800
RETURN	00035900
END	00036000
C	00036100
C	00036200
C	00036300
SUBROUTINE COMZN	00036400
C	00036500
INTEGER IROAD(6, 500, 6),	00036600
+NL, LMAX, NDMAX	00036700
REAL PDAT, BLPS	00036800
INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00036900
+ NCOMDT(2), NCOMCT(2)	00037000
C	00037100
COMMON /RD/IROAD,	00037200
+ /NMAX/NL, LMAX, NDMAX,	00037300
+ /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00037400
+ NCOMDT, NCOMCT, PDAT, BLPS	00037500
C	00037600
DO 10 IZONE=1, NZONE	00037700
IZMAX=LMAX-(IZONE-1)*LDIST-LDEAD	00037800
IZMIN=LMAX-IZONE*LDIST+1	00037900
C	00038000
NDATZN=INT(BLPS)	00038100
ICOMCR=1	00038200

DO 10 WHILE((NDATZN.GT.0).AND.(ICOMCR.EQ.1))	00038300
ICOMCR=0	00038400
DO 10 I=1,NL	00038500
DO 10 J=IZMIN,IZMAX	00038600
IF(NDATZN.GT.0) THEN	00038700
IF(IROAD(I,J,1).EQ.1) THEN	00038800
IF(IROAD(I,J,3).GT.0) THEN	00038900
ICOMCR=1	00039000
NDATZN=NDATZN-1	00039100
IROAD(I,J,3)=IROAD(I,J,3)-1	00039200
IF(MOD(IROAD(I,J,3),LDAT).EQ.0) THEN	00039300
NCOMCT(1)=NCOMCT(1)+1	00039400
NCOMCT(2)=NCOMCT(2)+1	00039500
IROAD(I,J,5)=IROAD(I,J,3)	00039600
ENDIF	00039700
IROAD(I,J,4)=IROAD(I,J,4)-1	00039800
IROAD(I,J,6)=IROAD(I,J,6)-1	00039900
IF(IROAD(I,J,4).LT.NWSIZ) THEN	00040000
IF(IROAD(I,J,3).GE.NWSIZ) THEN	00040100
IROAD(I,J,4)=NWSIZ	00040200
NCOMDT(1)=NCOMDT(1)+1	00040300
NCOMCT(1)=NCOMCT(1)+1	00040400
ENDIF	00040500
ENDIF	00040600
IF(IROAD(I,J,6).LT.NWSIZ) THEN	00040700
IF(IROAD(I,J,3).GE.NWSIZ) THEN	00040800
IROAD(I,J,5)=IROAD(I,J,3)	00040900
IROAD(I,J,6)=NWSIZ	00041000
NCOMDT(2)=NCOMDT(2)+1	00041100
NCOMCT(2)=NCOMCT(2)+1	00041200
ENDIF	00041300
ENDIF	00041400
ENDIF	00041500
ENDIF	00041600
10 CONTINUE	00041700
C	00041800
RETURN	00041900
END	00042000
C	00042100
C	00042200
C	00042300
C	00042400
SUBROUTINE CARMOV	00042500
C	00042600
INTEGER IROAD(6,500,6),NIROAD(6,500,6),F(0:17),NSIG,	00042700
+ NL,LMAX,NDMAX	00042800
CHARACTER DSPR(6,500),NDSPR(6,500)	00042900
COMMON /RD/IROAD,/CRD/DSPR,/NRD/NIROAD,/CNRD/NDSPR,/SIG/NSIG,	00043000
+ /FT/F,	00043100
+ /NMAX/NL,LMAX,NDMAX	00043200
C	00043300
INTEGER TOP	00043400
C	00043500
DO 10 I=1,NL	00043600
DO 10 J=1,LMAX	00043700

NIROAD(I, J, 1)=0	00043800
NDSPR(I, J)=' '	00043900
10 CONTINUE	00044000
DO 20 I=1, NL	00044100
TOP=0	00044200
DO 20 J=1, LMAX	00044300
IF(IROAD(I, J, 1).EQ.1)THEN	00044400
ISP=IVEL(I, J)	00044500
IF(TOP.EQ.0)THEN	00044600
IF(NSIG.EQ.1) THEN	00044700
K=MIN(J, 17)	00044800
ISP=MIN(ISP, F(K))	00044900
TOP=1	00045000
ENDIF	00045100
ENDIF	00045200
IROAD(I, J, 2)=ISP	00045300
CALL MOVE(I, J, ISP)	00045400
ENDIF	00045500
20 CONTINUE	00045600
DO 30 I=1, NL	00045700
DO 30 J=1, LMAX	00045800
DSPR(I, J)=NDSPR(I, J)	00045900
DO 30 K=1, NDMAX	00046000
IROAD(I, J, K)=NIROAD(I, J, K)	00046100
30 CONTINUE	00046200
RETURN	00046300
END	00046400
C	00046500
C	00046600
C	00046700
SUBROUTINE MOVE(I, J, ISP)	00046800
C	00046900
INTEGER I, J, ISP,	00047000
+ IROAD(6, 500, 6), NIROAD(6, 500, 6), TIME,	00047100
+ NL, LMAX, NDMAX	00047200
CHARACTER DSPR(6, 500), NDSPR(6, 500)	00047300
COMMON /RD/IROAD, /CRD/DSPR, /NRD/NIROAD, /CNRD/NDSPR, /T/TIME,	00047400
+ /NMAX/NL, LMAX, NDMAX	00047500
C	00047600
K=MOD(ISP, 4)	00047700
L=(ISP-K)/4	00047800
M=MOD(TIME, 4)	00047900
IF(K*M. GE. 3)L=L+1	00048000
IF(J. GT. L)THEN	00048100
NDSPR(I, J-L)=DSPR(I, J)	00048200
DO 10 N=1, NDMAX	00048300
NIROAD(I, J-L, N)=IROAD(I, J, N)	00048400
10 CONTINUE	00048500
ELSE	00048600
IF(IROAD(I, J, 3).GT.0) THEN	00048700
CALL FNDAT(IROAD(I, J, 3), IROAD(I, J, 4))	00048800
ENDIF	00048900
ENDIF	00049000
RETURN	00049100
END	00049200

C		00049300
C		00049400
C		00049500
	SUBROUTINE FNDAT(IR, IM)	00049600
C		00049700
	INTEGER IR, IM	00049800
	INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00049900
	+ NCOMDT(2), NCOMCT(2)	00050000
C		00050100
	COMMON /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00050200
	+ NCOMDT, NCOMCT, PDAT, BLPS	00050300
C		00050400
	CALL CTWR(IR)	00050500
C		00050600
	NCOMCT(1)=NCOMCT(1)+1	00050700
	NCOMDT(1)=NCOMDT(1)+IM	00050800
C		00050900
	NCOMCT(2)=NCOMCT(2)+1	00051000
C		00051100
	RETURN	00051200
	END	00051300
C		00051400
C		00051500
C		00051600
	SUBROUTINE PRINT(IW)	00051700
C		00051800
	INTEGER IW	00051900
	INTEGER IROAD(6, 500, 6), TIME, NL, LMAX	00052000
	INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00052100
	+ NCOMDT(2), NCOMCT(2)	00052200
	CHARACTER DSPR(6, 500)	00052300
	COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /SIG/NSIG,	00052400
	+ /NMAX/NL, LMAX,	00052500
	+ /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00052600
	+ NCOMDT, NCOMCT, PDAT, BLPS	00052700
C		00052800
	CHARACTER DD(6, 500), C(500), CHSIG(2)	00052900
C		00053000
	CHSIG(1)=' R'	00053100
	CHSIG(2)=' G'	00053200
C		00053300
	DO 10 I=1, NL	00053400
	DO 10 J=1, LMAX	00053500
	DD(I, J)=DSPR(I, J)	00053600
10	CONTINUE	00053700
C		00053800
	DO 20 I=1, LMAX	00053900
	C(I)=' '	00054000
20	CONTINUE	00054100
	DO 21 IZONE=1, NZONE	00054200
	IZMAX=LMAX-(IZONE-1)*LDIST-LDEAD	00054300
	IZMIN=LMAX-IZONE*LDIST+1	00054400
	DO 21 I=IZMIN, IZMAX	00054500
21	C(I)=' Z'	00054600
C		00054700

	LDSP=1+INT((LMAX-1)/50)	00054800
100	FORMAT(1H , '@' , 50A1)	00054900
110	FORMAT(1H , '***** TIME=' , I5 , ' **** NOW SIGNAL=' , A1 , '*****')	00055000
120	FORMAT(1H ,)	00055100
	WRITE(IW, 110) TIME, CHSIG(NSIG)	00055200
	DO 30 I=1, LDSP	00055300
	LDMIN=1+(I-1)*50	00055400
	LDMAX=MIN((I*50), LMAX)	00055500
	WRITE(IW, 100) (C(J), J=LDMIN, LDMAX)	00055600
	DO 40 J=1, NL	00055700
40	WRITE(IW, 100) (DD(J, K), K=LDMIN, LDMAX)	00055800
	WRITE(IW, 100) (C(J), J=LDMIN, LDMAX)	00055900
30	CONTINUE	00056000
	WRITE(IW, 120)	00056100
	RETURN	00056200
	END	00056300
C		00056400
C		00056500
C		00056600
	SUBROUTINE PRTCOM(IW)	00056700
C		00056800
	INTEGER IW	00056900
	INTEGER IROAD(6, 500, 6),	00057000
	+NL, LMAX, NDMAX, TIME, NSIG	00057100
	REAL PDAT, BLPS	00057200
	INTEGER LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00057300
	+ NCOMDT(2), NCOMCT(2)	00057400
	INTEGER CTBF(100), NCTBSZ, ICTWR, ICTRD	00057500
	CHARACTER DD(6, 1500, 4), NMQ(10), C(1500), LN(1500), CHSIG(2)	00057600
C		00057700
	COMMON /RD/IROAD, /NMQ/NMQ, /T/TIME, /SIG/NSIG,	00057800
	+ /NMAX/NL, LMAX, NDMAX,	00057900
	+ /COM/LZONE, LDEAD, LDIST, NZONE, LDAT, NWSIZ, NADAT,	00058000
	+ NCOMDT, NCOMCT, PDAT, BLPS,	00058100
	+ /CNT/CTBF, NCTBSZ, ICTWR, ICTRD	00058200
C		00058300
	CHSIG(1)=' R'	00058400
	CHSIG(2)=' G'	00058500
C		00058600
	DO 10 I=1, NL	00058700
	DO 10 J=1, LMAX	00058800
	JB=3*(J-1)	00058900
	IF(IROAD(I, J, 1).EQ.1) THEN	00059000
	DD(I, JB+1, 1)=NMQ(1+INT(IROAD(I, J, 3)/10))	00059100
	DD(I, JB+2, 1)=NMQ(1+MOD(IROAD(I, J, 3), 10))	00059200
	DD(I, JB+3, 1)=' '	00059300
	DD(I, JB+1, 2)=NMQ(1+INT(IROAD(I, J, 4)/10))	00059400
	DD(I, JB+2, 2)=NMQ(1+MOD(IROAD(I, J, 4), 10))	00059500
	DD(I, JB+3, 2)=' '	00059600
	DD(I, JB+1, 3)=NMQ(1+INT(IROAD(I, J, 5)/10))	00059700
	DD(I, JB+2, 3)=NMQ(1+MOD(IROAD(I, J, 5), 10))	00059800
	DD(I, JB+3, 3)=' '	00059900
	DD(I, JB+1, 4)=NMQ(1+INT(IROAD(I, J, 6)/10))	00060000
	DD(I, JB+2, 4)=NMQ(1+MOD(IROAD(I, J, 6), 10))	00060100
	DD(I, JB+3, 4)=' '	00060200

```

ELSE
DO 11 K=1,3
DO 11 L=1,4
11 DD(I,JB+K,L)=' '
ENDIF
10 CONTINUE
C
DO 20 I=1,(LMAX*3)
C(I)='.'
LN(I)='- '
20 CONTINUE
DO 21 IZONE=1,NZONE
IZMAX=LMAX-(IZONE-1)*LDIST-LDEAD
IZMIN=LMAX-IZONE*LDIST+1
DO 21 I=IZMIN,IZMAX
IB=3*(I-1)
DO 21 J=1,3
C(IB+J)='Z'
21 LN(IB+J)='Z'
C
100 FORMAT(1H,'***** TIME=',I5,' ** NOW SIGNAL=',A1,'*****')
110 FORMAT(1H,'LOAD OF DATA(BLOCK) CONV=',I5,' NEW=',I5)
120 FORMAT(1H,'LOAD OF CONT.(TIME) CONV=',I5,' NEW=',I5)
130 FORMAT(1H)
140 FORMAT(1H,'*****CONTINUED DATA*****')
150 FORMAT(1H,'-----NON DATA-----')
160 FORMAT(1H,30I5)
170 FORMAT(1H)
200 FORMAT(1H,'@',100A1)
210 FORMAT(1H)
C
WRITE(IW,100)TIME,CHSIG(NSIG)
WRITE(IW,110) (NCOMDT(I),I=1,2)
WRITE(IW,120) (NCOMCT(I),I=1,2)
WRITE(IW,130)
WRITE(IW,140)
IF(ICTWR.EQ.ICTRD) THEN
WRITE(IW,150)
ELSEIF(ICTWR.GT.ICTRD) THEN
WRITE(IW,160) (CTBF(I),I=ICTRD,ICTWR-1)
ELSE
WRITE(IW,160) (CTBF(I),I=ICTRD,NCTBSZ), (CTBF(I),I=1,ICTWR-1)
ENDIF
WRITE(IW,170)
C
LDSP=1+INT((3*LMAX-1)/100)
DO 30 I=1,LDSP
LDMIN=1+(I-1)*100
LDMAX=MIN((I*100),(3*LMAX))
WRITE(IW,200) (C(J),J=LDMIN,LDMAX)
DO 40 J=1,NL
WRITE(IW,200) (DD(J,K,1),K=LDMIN,LDMAX)
WRITE(IW,200) (DD(J,K,2),K=LDMIN,LDMAX)
WRITE(IW,200) (DD(J,K,3),K=LDMIN,LDMAX)
WRITE(IW,200) (DD(J,K,4),K=LDMIN,LDMAX)

```

IF(J.LT.NL) THEN	00065800
WRITE(IW,200)(LN(K),K=LDMIN,LDMAX)	00065900
ENDIF	00066000
40 CONTINUE	00066100
WRITE(IW,200)(C(J),J=LDMIN,LDMAX)	00066200
30 CONTINUE	00066300
WRITE(IW,210)	00066400
WRITE(IW,210)	00066500
RETURN	00066600
END	00066700
C	00066800
C	00066900
C	00067000
INTEGER FUNCTION IVEL(I,J)	00067100
C	00067200
INTEGER I,J	00067300
C	00067400
INTEGER IROAD(6,500,6),F(0:17),AC(0:16)	00067500
COMMON /RD/IROAD,/FT/F,AC	00067600
C	00067700
IVF=F(IFD(I,J))	00067800
ISP=IROAD(I,J,2)	00067900
IVG=AC(ISP)	00068000
IVEL=MIN(IVF,IVG)	00068100
RETURN	00068200
END	00068300
C	00068400
C	00068500
C	00068600
INTEGER FUNCTION IFD(I,J)	00068700
C	00068800
INTEGER I,J	00068900
C	00069000
INTEGER IROAD(6,500,6)	00069100
COMMON /RD/IROAD	00069200
C	00069300
L=1	00069400
DO 10 WHILE(IROAD(I,J-L,1).EQ.0.AND.J.NE.L.AND.L.NE.17)	00069500
L=L+1	00069600
10 CONTINUE	00069700
IF(J.EQ.L)L=17	00069800
IFD=L	00069900
RETURN	00070000
END	00070100
C	00070200
C	00070300
C	00070400
INTEGER FUNCTION LCTRD()	00070500
INTEGER CTBF(100),NCTBSZ,ICTWR,ICTRD	00070600
COMMON /CNT/CTBF,NCTBSZ,ICTWR,ICTRD	00070700
IF(ICTRD.NE.ICTWR) THEN	00070800
LCTRD=CTBF(ICTRD)	00070900
ICTRD=ICTRD+1	00071000
IF(ICTRD.GT.NCTBSZ) ICTRD=1	00071100
ELSE	00071200

	LCTRD=0	00071300
	ENDIF	00071400
C		00071500
	RETURN	00071600
	END	00071700
C		00071800
C		00071900
C		00072000
	SUBROUTINE CTWR(I)	00072100
	INTEGER I, IWERR	00072200
	INTEGER CTBF(100), NCTBSZ, ICTWR, ICTRD	00072300
	COMMON /CNT/CTBF, NCTBSZ, ICTWR, ICTRD, /ERR/IWERR	00072400
	CTBF(ICTWR)=I	00072500
	ICTWR=ICTWR+1	00072600
	IF(ICTWR.GT.NCTBSZ) ICTWR=1	00072700
	IF(ICTWR.EQ.ICTRD) THEN	00072800
	WRITE(IWERR, 100)	00072900
100	FORMAT(1H, 'BUFFER FULL!')	00073000
	ICTWR=ICTWR-1	00073100
	IF(ICTWR.EQ.0) ICTWR=NCTBSZ	00073200
	ENDIF	00073300
	RETURN	00073400
	END	00073500

A. 3. 右折車の待ち合わせ制御アルゴリズムの評価プログラム

A. 3. 1. 誘導を行わない場合の評価プログラム

```

C***** SIMULATION FOR NOT YUDO AT AN INTERSECTION *****
C
    INTEGER IROAD(2,400,6), NIROAD(2,400,6), TIME, F(0:17), BR(0:16),
+AC(0:16), IX1, IX2, DT(2,2,1000), DP(2),
+TE, GECHAR(2), SIGCT(400), SIGCP, ENDSGT, NSIG, TDS, TDE, TDI,
+DSW, DSL
    REAL GETIME(2), Q(2)
    CHARACTER DSPR(2,400), NDSPR(2,400), CSQ(20), CSIGC(400)
C
    COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,
+      /FT/F, BR, AC,
+      /NRD/NIROAD, /CNRD/NDSPR, /RND/IX1, IX2,
+      /ODT/DT, DP, /SIG/NSIG, ENDSGT, SIGCT, SIGCP, /CSIG/CSIGC
C
54 CONTINUE
    READ(5,51) ICONT
51 FORMAT(I2)
    IF(ICONT.EQ.0) GOTO 52
C
C
    DO 10 I=1,2
        DO 10 J=1,400
            IROAD(I,J,1)=0
            DSPR(I,J)=' '
            SIGCT(J)=0
            CSIGC(J)=' '
10 CONTINUE
        DO 15 I=1,2
            DO 15 J=1,1000
                DT(I,1,J)=0
15 CONTINUE
            DO 17 I=1,2
                DP(I)=1
                GETIME(I)=0
17 CONTINUE
            GECHAR(1)=1
            GECHAR(2)=11
            NSIG=1
            ENDSGT=30
C*
            SIGCP=1
C
C
            WRITE(6,110)
110 FORMAT(1H0,'RIGHT-TURNING VEHICLE NOT CONTROL')
            READ(5,210) TE
210 FORMAT(I5)
            WRITE(6,211) TE
211 FORMAT(1H0,'SIMULATION TIME LENGTH=',I5)
            READ(5,220) TDS, TDE, TDI

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900
00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000

```

220	FORMAT(3I5)	00005100
	WRITE(6,221) TDS,TDE,TDI	00005200
221	FORMAT(1H0,'DISPLAYED TIME =',I5,' ->',I5,' INTERVAL=',I5)	00005300
	READ(5,230) DSW,DSL	00005400
230	FORMAT(2I5)	00005500
	WRITE(6,231) DSW,DSL	00005600
231	FORMAT(1H0,'DISPLAYED WIDTH =',I5,' DISPLAYED LINES=',I5,)	00005700
	READ(5,310) IX1,IX2	00005800
310	FORMAT(2I2)	00005900
	WRITE(6,311) IX1,IX2	00006000
311	FORMAT(1H0,'RANDOMIZING PARAMETER=',2I2)	00006100
	READ(5,410) Q(1),Q(2)	00006200
410	FORMAT(2F5.3)	00006300
	WRITE(6,411) Q(1),Q(2)	00006400
411	FORMAT(1H0,'AVR. TIME HEADWAY, 1ST HALF,2ND HALF=',2F7.4)	00006500
	READ(5,510) QR1,QR2	00006600
510	FORMAT(2F5.3)	00006700
	WRITE(6,511) QR1,QR2	00006800
511	FORMAT(1H0,'RIGHT-TURNING CAR RATIO IN 1ST HALF,2ND HALF=',2F7.4)	00006900
C		00007000
	DO 20 TIME=1,TE	00007100
	IF(TIME.LT.TE/2)THEN	00007200
	QGEN=Q(1)	00007300
	CALL STCAR2(QGEN,QR1)	00007400
	ELSE	00007500
	QGEN=Q(2)	00007600
	CALL STCAR2(QGEN,QR2)	00007700
	ENDIF	00007800
	CALL SIGCN2	00007900
	CALL CHCAR3	00008000
	CALL CHEXE1	00008100
	IF(TIME.GE.TDS.AND.TIME.LE.TDE.AND.	00008200
	+ MOD((TIME-TDS),TDI).EQ.0) CALL PRINT2(DSW,DSL)	00008300
	CALL MOV2	00008400
	20 CONTINUE	00008500
C		00008600
	WRITE(6,610)	00008700
610	FORMAT(1H0,'SIGNAL CHANGE ')	00008800
	I=1	00008900
	DO 60 WHILE(SIGCT(I).NE.0)	00009000
	WRITE(6,611) SIGCT(I),CSIGC(I)	00009100
	I=I+1	00009200
	60 CONTINUE	00009300
611	FORMAT(1H0,' T=',I5,' SIGNAL=',A1)	00009400
C		00009500
	DO 30 I=1,2	00009600
	WRITE(6,710)	00009700
	WRITE(6,711) I	00009800
	J=1	00009900
	AV1=0	00010000
	AV2=0	00010100
	N1=0	00010200
	N2=0	00010300
	M1=0	00010400
	M2=0	00010500

DO 40 WHILE(DT(I, 1, J).NE. 0)	00010600
IF(DT(I, 1, J).LE. TE/2)THEN	00010700
AV1=AV1+DT(I, 2, J)	00010800
N1=N1+1	00010900
ELSE	00011000
AV2=AV2+DT(I, 2, J)	00011100
N2=N2+1	00011200
ENDIF	00011300
J=J+1	00011400
40 CONTINUE	00011500
IF(I.EQ. 1)THEN	00011600
C OPEN(11, FILE='AA30851.@DES1.DATA', STATUS='OLD',	00011700
C + FORM='UNFORMATTED')	00011800
ELSE	00011900
C OPEN(11, FILE='AA30851.@DES2.DATA', STATUS='OLD',	00012000
C + FORM='UNFORMATTED')	00012100
ENDIF	00012200
C WRITE(11) J-1, (FLOAT(DT(I, 1, K)), FLOAT(DT(I, 2, K)), K=1, J-1)	00012300
C CLOSE(11)	00012400
C	00012500
IF(I.EQ. 1)THEN	00012600
DO 41 L1=1, 2	00012700
DO 41 L2=1, 400	00012800
IF(IROAD(L1, L2, 1).EQ. 1.AND. IROAD(L1, L2, 2).NE. 2)THEN	00012900
IF(IROAD(L1, L2, 6).LE. (TE/2))THEN	00013000
M1=M1+1	00013100
ELSE	00013200
M2=M2+1	00013300
ENDIF	00013400
ENDIF	00013500
41 CONTINUE	00013600
ELSE	00013700
DO 42 L1=1, 2	00013800
DO 42 L2=1, 400	00013900
IF(IROAD(L1, L2, 1).EQ. 1.AND. IROAD(L1, L2, 2).EQ. 2)THEN	00014000
IF(IROAD(L1, L2, 6).LE. (TE/2))THEN	00014100
M1=M1+1	00014200
ELSE	00014300
M2=M2+1	00014400
ENDIF	00014500
ENDIF	00014600
42 CONTINUE	00014700
ENDIF	00014800
C	00014900
WRITE(6, 717) M1+N1	00015000
IF(N1.NE. 0)THEN	00015100
FN1=AV1/N1	00015200
WRITE(6, 712) N1, FN1	00015300
ELSE	00015400
WRITE(6, 713) N1	00015500
ENDIF	00015600
WRITE(6, 718) M2+N2	00015700
IF(N2.NE. 0)THEN	00015800
FN2=AV2/N2	00015900
WRITE(6, 714) N2, FN2	00016000

ELSE	00016100
WRITE(6,715) N2	00016200
ENDIF	00016300
WRITE(6,716) (N1+N2), ((AV1+AV2)/(N1+N2))	00016400
30 CONTINUE	00016500
710 FORMAT(1H0, '.....')	00016600
711 FORMAT(1H0, 'DESTINATION=' I1)	00016700
712 FORMAT(1H0, 'NUMBER OF CAR(1) =' , I4, ' AVARAGE TIME(1)=' , F7.2)	00016800
713 FORMAT(1H0, 'NUMBER OF CAR(1) =' , I4)	00016900
714 FORMAT(1H0, 'NUMBER OF CAR(2) =' , I4, ' AVARAGE TIME(1)=' , F7.2)	00017000
715 FORMAT(1H0, 'NUMBER OF CAR(2) =' , I4)	00017100
716 FORMAT(1H0, 'TOTAL NUMBER OF CAR=' , I4, ' AVARAGE TIME=' , F7.2)	00017200
717 FORMAT(1H0, 'GENERATED NUM. OF CAR(1) =' , I4)	00017300
718 FORMAT(1H0, 'GENERATED NUM. OF CAR(2) =' , I4)	00017400
C CLOSE(1)	00017500
C CLOSE(2)	00017600
C	00017700
GOTO 54	00017800
C	00017900
52 WRITE(6,53)	00018000
53 FORMAT(1H0, 'SIMULATION ENDED')	00018100
STOP	00018200
END	00018300
C	00018400
C	00018500
C	00018600
BLOCK DATA	00018700
INTEGER F(0:17), BR(0:16), AC(0:16)	00018800
CHARACTER CSQ(20)	00018900
COMMON /FT/F, BR, AC, /CST/CSQ	00019000
DATA F/0, 0, 4, 6, 8, 9, 10, 11, 11, 12, 13, 13, 14, 14, 14, 15, 15, 16/	00019100
DATA BR/0, 0, 0, 0, 0, 0, 0, 0, 4, 4, 6, 8, 10, 11, 12, 13, 14/	00019200
DATA AC/4, 4, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 16, 16/	00019300
DATA CSQ/'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',	00019400
+ '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' /	00019500
END	00019600
C	00019700
C	00019800
C	00019900
SUBROUTINE STCAR2(QGEN, QR)	00020000
C	00020100
REAL QGEN, QR	00020200
C	00020300
INTEGER IROAD(2, 400, 6), TIME, IX1, IX2, F(0:17), GECHAR(2)	00020400
REAL GETIME(2)	00020500
CHARACTER DSPR(2, 400), CSQ(20)	00020600
COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,	00020700
+ /RND/IX1, IX2, /FT/F	00020800
C	00020900
REAL A(1)	00021000
C	00021100
DO 10 I=1, 2	00021200
IF(TIME. GE. GETIME(I)) THEN	00021300
ISP=F(IFD(I, 400, I))	00021400
IP=INT(400-ISP*(TIME-GETIME(I))/4)	00021500

IROAD(1, IP, 1)=1	00021600
CALL RANU2(IX1, A, 1, ICON)	00021700
IF(I. EQ. 1. AND. A(1). LT. 2*QR) THEN	00021800
IROAD(1, IP, 2)=1	00021900
DSPR(1, IP)=CSQ(GECHAR(1))	00022000
IF(GECHAR(1). EQ. 10) THEN	00022100
GECHAR(1)=1	00022200
ELSE	00022300
GECHAR(1)=GECHAR(1)+1	00022400
ENDIF	00022500
ELSE	00022600
IROAD(1, IP, 2)=2	00022700
DSPR(1, IP)=CSQ(GECHAR(2))	00022800
IF(GECHAR(2). EQ. 20) THEN	00022900
GECHAR(2)=11	00023000
ELSE	00023100
GECHAR(2)=GECHAR(2)+1	00023200
ENDIF	00023300
ENDIF	00023400
IROAD(1, IP, 3)=0	00023500
IROAD(1, IP, 4)=16	00023600
IROAD(1, IP, 5)=ISP	00023700
IROAD(1, IP, 6)=TIME	00023800
CALL RANE2(QGEN-1, IX2, A, 1, ICON)	00023900
GETIME(I)=GETIME(I)+1+A(1)	00024000
ENDIF	00024100
10 CONTINUE	00024200
RETURN	00024300
END	00024400
C	00024500
C	00024600
C	00024700
SUBROUTINE SIGCN2	00024800
C	00024900
INTEGER SIGCP, ENDSGT, NSIG, SIGCT(400), TIME, IROAD(2, 400, 6)	00025000
CHARACTER CSIGC(400), CHSIG(3)	00025100
COMMON /SIG/NSIG, ENDSGT, SIGCT, SIGCP, /CSIG/CSIGC, /RD/IROAD,	00025200
+ /T/TIME	00025300
C	00025400
CHSIG(1)=' G'	00025500
CHSIG(2)=' Y'	00025600
CHSIG(3)=' R'	00025700
IF(NSIG. EQ. 1) THEN	00025800
IF(TIME. GE. ENDSGT) THEN	00025900
NSIG=2	00026000
SIGCT(SIGCP)=TIME	00026100
CSIGC(SIGCP)=CHSIG(NSIG)	00026200
SIGCP=SIGCP+1	00026300
ENDIF	00026400
ENDIF	00026500
IF(NSIG. EQ. 2) THEN	00026600
IF(IROAD(1, IBD(1, 0, 1), 2). NE. 1. OR.	00026700
+ IBD(1, 0, 1). EQ. 17) THEN	00026800
NSIG=3	00026900
SIGCT(SIGCP)=TIME	00027000

CSIGC(SIGCP)=CHSIG(NSIG)	00027100
SIGCP=SIGCP+1	00027200
ENDSGT=TIME+30+TIME-ENDSGT	00027300
ENDIF	00027400
ENDIF	00027500
IF(NSIG.EQ.3)THEN	00027600
IF(TIME.GE.ENDSGT)THEN	00027700
NSIG=1	00027800
SIGCT(SIGCP)=TIME	00027900
CSIGC(SIGCP)=CHSIG(NSIG)	00028000
SIGCP=SIGCP+1	00028100
ENDSGT=TIME+30	00028200
ENDIF	00028300
ENDIF	00028400
RETURN	00028500
END	00028600
C	00028700
C	00028800
C	00028900
SUBROUTINE CHCAR3	00029000
C	00029100
CALL CHLLFT(J,40)	00029200
CALL CHLVEL(41,400)	00029300
RETURN	00029400
END	00029500
C	00029600
C	00029700
C	00029800
SUBROUTINE CHLVEL(L1,L2)	00029900
C	00030000
INTEGER L1,L2	00030100
C	00030200
INTEGER IROAD(2,400,6),F(0:17)	00030300
COMMON /RD/IROAD,/FT/F	00030400
C	00030500
INTEGER IV1,IV2	00030600
C	00030700
DO 10 I=1,2	00030800
K=MOD(I,2)+1	00030900
DO 10 J=L1,L2	00031000
IF(IROAD(I,J,1).EQ.1.AND.IROAD(I,J,2).EQ.2)THEN	00031100
IV1=F(IFD(I,J,I))	00031200
IV2=F(IFD(I,J,K))	00031300
IF(IV1.LE.IV2-4)THEN	00031400
IROAD(I,J,3)=1	00031500
ENDIF	00031600
IF(IROAD(I,J,3).EQ.1.AND.IV1.GE.IV2)THEN	00031700
IROAD(I,J,3)=0	00031800
ENDIF	00031900
ENDIF	00032000
10 CONTINUE	00032100
RETURN	00032200
END	00032300
C	00032400
C	00032500

C	SUBROUTINE CHLLFT(L1,L2)	00032600
C		00032700
C	INTEGER L1,L2	00032800
C		00032900
C	INTEGER IROAD(2,400,6)	00033000
C	COMMON /RD/IROAD	00033100
C		00033200
	J=L1	00033300
	DO 10 WHILE(J.LT.L2)	00033400
	IF(IROAD(1,J,1).EQ.1.AND.IROAD(1,J,2).EQ.2)THEN	00033500
	IROAD(1,J,3)=1	00033600
	ENDIF	00033700
	J=J+1	00033800
10	CONTINUE	00033900
	RETURN	00034000
	END	00034100
C		00034200
C		00034300
C		00034400
	SUBROUTINE CHEXE1	00034500
C		00034600
	INTEGER IROAD(2,400,6),F(0:17),BR(0:16),TIME	00034700
	CHARACTER DSPR(2,400)	00034800
	COMMON /RD/IROAD,/CRD/DSPR,/FT/F,BR,/T/TIME	00034900
C		00035000
	DO 10 I=1,2	00035100
	K=MOD(I,2)+1	00035200
	DO 10 J=1,400	00035300
	IF(IROAD(1,J,1).EQ.0.OR.IROAD(1,J,3).NE.1)GOTO 10	00035400
	ISP1=IROAD(1,J,5)	00035500
	IFG=IFD(1,J,K)	00035600
	IBG=IBD(1,J,K)	00035700
	IF(IBG.EQ.17)THEN	00035800
	ISP2=0	00035900
	ELSE	00036000
	ISP2=IROAD(K,J+IBG,5)	00036100
	ENDIF	00036200
	IF(F(IFG).GT.BR(ISP1))THEN	00036300
	IF(F(IBG).GT.BR(ISP2))THEN	00036400
	IF(IROAD(K,J,1).EQ.0)THEN	00036500
	L=J	00036600
	ELSEIF(IROAD(K,J-1,1).EQ.0)THEN	00036700
	L=J-1	00036800
	ELSE	00036900
	L=-1	00037000
	ENDIF	00037100
	IF(L.GE.1)THEN	00037200
	DO 20 M=1,6	00037300
	IROAD(K,L,M)=IROAD(1,J,M)	00037400
20	CONTINUE	00037500
	IROAD(K,L,3)=0	00037600
	DSPR(K,L)=DSPR(1,J)	00037700
	IROAD(1,J,1)=0	00037800
	DSPR(1,J)=	00037900
		00038000

ELSEIF(L.EQ.0)THEN	00038100
CALL TRC2(I,J)	00038200
IROAD(I,J,1)=0	00038300
DSPR(I,J)=' '	00038400
ENDIF	00038500
ELSEIF(IROAD(K,J+IBG,3).EQ.0)THEN	00038600
IF(IVEL0(I,J).GT.0.OR.IBG.GT.1)THEN	00038700
IROAD(K,J+IBG,3)=2	00038800
ENDIF	00038900
ENDIF	00039000
10 CONTINUE	00039100
RETURN	00039200
END	00039300
C	00039400
C	00039500
C	00039600
SUBROUTINE MOV2	00039700
C	00039800
INTEGER IROAD(2,400,6),NIROAD(2,400,6),F(0:17),BR(0:16),NSIG	00039900
C	00040000
INTEGER IROAD(2,400,6),NIROAD(2,400,6),F(0:17),NSIG	00040100
CHARACTER DSPR(2,400),NDSPR(2,400)	00040200
COMMON /RD/IROAD,/CRD/DSPR,/NRD/NIROAD,/CNRD/NDSPR,/SIG/NSIG,	00040300
+ /FT/F,BR	00040400
C	00040500
INTEGER TOP	00040600
C	00040700
DO 10 I=1,2	00040800
DO 10 J=1,400	00040900
NIROAD(I,J,1)=0	00041000
NDSPR(I,J)=' '	00041100
10 CONTINUE	00041200
DO 20 I=1,2	00041300
TOP=0	00041400
DO 20 J=1,400	00041500
IF(IROAD(I,J,1).EQ.1)THEN	00041600
ISP=IVEL(I,J)	00041700
IF(TOP.EQ.0)THEN	00041800
IF(NSIG.EQ.3.OR.(NSIG.EQ.2.AND.IROAD(I,J,2).EQ.2).OR.	00041900
+ (NSIG.EQ.1.AND.IROAD(I,J,2).NE.2))THEN	00042000
K=MIN(J,17)	00042100
ISP=MIN(ISP,F(K))	00042200
TOP=1	00042300
ENDIF	00042400
ENDIF	00042500
IROAD(I,J,5)=ISP	00042600
CALL MOVE(I,J,ISP)	00042700
ENDIF	00042800
20 CONTINUE	00042900
DO 30 I=1,2	00043000
DO 30 J=1,400	00043100
DSPR(I,J)=NDSPR(I,J)	00043200
DO 30 K=1,6	00043300
IROAD(I,J,K)=NIROAD(I,J,K)	00043400
30 CONTINUE	00043500

	RETURN	00043600
	END	00043700
C		00043800
C		00043900
C		00044000
	SUBROUTINE MOVE(I, J, ISP)	00044100
C		00044200
	INTEGER I, J, ISP	00044300
C		00044400
	INTEGER IROAD(2, 400, 6), NIROAD(2, 400, 6), TIME	00044500
	CHARACTER DSPR(2, 400), NDSPR(2, 400)	00044600
	COMMON /RD/IROAD, /CRD/DSPR, /NRD/NIROAD, /CNRD/NDSPR, /T/TIME	00044700
C		00044800
	K=MOD(ISP, 4)	00044900
	L=(ISP-K)/4	00045000
	M=MOD(TIME, 4)	00045100
	IF(K*M. GE. 3)L=L+1	00045200
	IF(J. GT. L) THEN	00045300
	NDSPR(I, J-L)=DSPR(I, J)	00045400
	DO 10 N=1, 6	00045500
	NIROAD(I, J-L, N)=IROAD(I, J, N)	00045600
10	CONTINUE	00045700
	IF(NIROAD(I, J-L, 3). EQ. 2) NIROAD(I, J-L, 3)=0	00045800
	ELSE	00045900
	CALL TRC2(I, J)	00046000
	ENDIF	00046100
	RETURN	00046200
	END	00046300
C		00046400
C		00046500
C		00046600
	SUBROUTINE TRC2(I, J)	00046700
C		00046800
	INTEGER I, J	00046900
C		00047000
	INTEGER IROAD(2, 400, 6), TIME, DT(2, 2, 1000), DP(2)	00047100
	COMMON /RD/IROAD, /T/TIME, /ODT/DT, DP	00047200
C		00047300
	IF(IROAD(I, J, 2). EQ. 2) THEN	00047400
	K=2	00047500
	ELSE	00047600
	K=1	00047700
	ENDIF	00047800
	DT(K, 1, DP(K))=IROAD(I, J, 6)	00047900
	DT(K, 2, DP(K))=TIME-IROAD(I, J, 6)	00048000
	DP(K)=DP(K)+1	00048100
	RETURN	00048200
	END	00048300
C		00048400
C		00048500
C		00048600
	SUBROUTINE PRINT2(JW, J1)	00048700
C		00048800
	INTEGER JW, J1	00048900
	INTEGER IROAD(2, 400, 6), TIME	00049000

CHARACTER DSPR(2,400)	00049100
COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /SIG/NSIG	00049200
C CHARACTER DD(2,400), C(400), CHSIG(3)	00049300
C	00049400
CHSIG(1)=' G'	00049500
CHSIG(2)=' Y'	00049600
CHSIG(3)=' R'	00049700
DO 10 I=1, 2	00049800
DO 10 J=1, 400	00049900
DD(I, J)=DSPR(I, J)	00050000
IF(IROAD(I, J, 1).EQ.1.AND.IROAD(I, J, 3).EQ.1)DD(I, J)=' +'	00050100
IF(IROAD(I, J, 1).EQ.1.AND.IROAD(I, J, 3).EQ.2)DD(I, J)=' -'	00050200
10 CONTINUE	00050300
DO 20 I=1, 100	00050400
IF(INT(I/10)*10.EQ.I)THEN	00050500
C(I)=' @'	00050600
ELSE	00050700
C(I)=' .'	00050800
ENDIF	00050900
20 CONTINUE	00051000
C	00051100
WRITE(6, 101)	00051200
WRITE(6, 102) TIME, CHSIG(NSIG)	00051300
DO 21 K=1, J1	00051400
K1=JW*(K-1)+1	00051500
K2=JW*K	00051600
WRITE(6, 100) (DD(1, J), J=K1, K2)	00051700
WRITE(6, 100) (DD(2, J), J=K1, K2)	00051800
WRITE(6, 100) (C(I), I=K1, K2)	00051900
21 CONTINUE	00052000
WRITE(6, 101)	00052100
100 FORMAT(' @', 100A1)	00052200
101 FORMAT(1H0, '*****')	00052300
102 FORMAT(1H0, ' TIME=', I5, ' NOW SIGNAL=', A1)	00052400
RETURN	00052500
END	00052600
C	00052700
C	00052800
C	00052900
C	00053000
INTEGER FUNCTION IVEL(I, J)	00053100
C	00053200
INTEGER I, J	00053300
C	00053400
INTEGER IROAD(2, 400, 6)	00053500
COMMON /RD/IROAD	00053600
C	00053700
IF(IROAD(I, J, 3).EQ.0)THEN	00053800
IVEL=IVEL0(I, J)	00053900
ELSEIF(IROAD(I, J, 3).EQ.1)THEN	00054000
IVEL=IVEL1(I, J)	00054100
ELSE	00054200
IVEL=IVEL2(I, J)	00054300
ENDIF	00054400
RETURN	00054500

END	00054600
C	00054700
C	00054800
C	00054900
INTEGER FUNCTION IVEL0(I, J)	00055000
C	00055100
INTEGER I, J	00055200
C	00055300
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16), AC(0:16)	00055400
COMMON /RD/IROAD, /FT/F, BR, AC	00055500
C	00055600
IVF=F(IFD(I, J, 1))	00055700
ISP=IROAD(I, J, 5)	00055800
IVG=AC(ISP)	00055900
ISP=BR(ISP)	00056000
IVEL0=MIN(IVF, IVG)	00056100
RETURN	00056200
END	00056300
C	00056400
C	00056500
C	00056600
INTEGER FUNCTION IVEL1(I, J)	00056700
C	00056800
INTEGER I, J	00056900
C	00057000
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16)	00057100
COMMON /RD/IROAD, /FT/F, BR	00057200
C	00057300
IV=IVEL0(I, J)	00057400
K=MOD(I, 2)+1	00057500
IFG=IFD(I, J, K)	00057600
IF (F(IFG).LT. IV. AND. (F(IFG).GT. 0. OR. J.LT. 40)) THEN	00057700
IVEL1=MAX(F(IFG), BR(IROAD(I, J, 5)))	00057800
ELSE	00057900
IVEL1=IV	00058000
ENDIF	00058100
RETURN	00058200
END	00058300
C	00058400
C	00058500
C	00058600
INTEGER FUNCTION IVEL2(I, J)	00058700
C	00058800
INTEGER I, J	00058900
C	00059000
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16)	00059100
COMMON /RD/IROAD, /FT/F, BR	00059200
C	00059300
IV=IVEL0(I, J)	00059400
K=MOD(I, 2)+1	00059500
IFG=IFDC(I, J, K, 1)	00059600
IF (F(IFG).LT. IV) THEN	00059700
IVEL2=MAX(F(IFG), BR(IROAD(I, J, 5)))	00059800
ELSE	00059900
IVEL2=IV	00060000

ENDIF	00060100
RETURN	00060200
END	00060300
C	00060400
C	00060500
C	00060600
INTEGER FUNCTION IFD(I, J, K)	00060700
C	00060800
INTEGER I, J, K	00060900
C	00061000
INTEGER IROAD(2, 400, 6)	00061100
COMMON /RD/IROAD	00061200
C	00061300
IF (I. NE. K. AND. IROAD(K, J, 1). NE. 0) THEN	00061400
IF ((IROAD(I, J, 3). NE. 1. AND. IROAD(K, J, 3). EQ. 1). OR.	00061500
+ (IROAD(I, J, 3). EQ. 1. AND. IROAD(K, J, 3). EQ. 1. AND. I. EQ. 2)) THEN	00061600
IFD=0	00061700
RETURN	00061800
ENDIF	00061900
ENDIF	00062000
L=1	00062100
DO 10 WHILE (IROAD(K, J-L, 1). EQ. 0. AND. J. NE. L. AND. L. NE. 17)	00062200
L=L+1	00062300
10 CONTINUE	00062400
IF (J. EQ. L) L=17	00062500
IFD=L	00062600
RETURN	00062700
END	00062800
C	00062900
C	00063000
C	00063100
INTEGER FUNCTION IBD(I, J, K)	00063200
C	00063300
INTEGER I, J, K	00063400
C	00063500
INTEGER IROAD(2, 400, 6)	00063600
COMMON /RD/IROAD	00063700
C	00063800
IF (I. NE. K. AND. IROAD(K, J, 1). NE. 0) THEN	00063900
IF ((IROAD(I, J, 3). EQ. 1. AND. IROAD(K, J, 3). NE. 1). OR.	00064000
+ (IROAD(I, J, 3). EQ. 1. AND. IROAD(K, J, 3). EQ. 1. AND. I. EQ. 1)) THEN	00064100
IBD=0	00064200
RETURN	00064300
ENDIF	00064400
ENDIF	00064500
L=1	00064600
DO 10 WHILE (IROAD(K, J+L, 1). EQ. 0. AND. J+L. NE. 400. AND. L. NE. 17)	00064700
L=L+1	00064800
10 CONTINUE	00064900
IF (J+L. EQ. 400) L=17	00065000
IBD=L	00065100
RETURN	00065200
END	00065300
C	00065400
C	00065500

C	INTEGER FUNCTION IFDC(I, J, K, L)	00065600
		00065700
C	INTEGER I, J, K, L	00065800
		00065900
C	INTEGER IROAD(2, 400, 6)	00066000
	COMMON /RD/IROAD	00066100
		00066200
C	M=IFD(I, J, K)	00066300
	DO 10 WHILE(IROAD(K, J-M, 3). NE. L. AND. M. LT. 17)	00066400
	M=IFD(K, J-M, K)+M	00066500
		00066600
10	CONTINUE	00066700
	IF(M. GT. 17)M=17	00066800
	IFDC=M	00066900
	RETURN	00067000
	END	00067100
C		00067200
C		00067300
C		00067400
	INTEGER FUNCTION IBDC(I, J, K, L)	00067500
C		00067600
	INTEGER I, J, K, L.	00067700
C		00067800
	INTEGER IROAD(2, 400, 6)	00067900
	COMMON /RD/IROAD	00068000
C		00068100
	M=IBD(I, J, K)	00068200
	DO 10 WHILE(IROAD(K, J+M, 2). NE. L. AND. M. LT. 17)	00068300
	M=IBD(K, J+M, K)+M	00068400
10	CONTINUE	00068500
	IF(M. GT. 17)M=17	00068600
	IBDC=M	00068700
	RETURN	00068800
	END	00068900

A. 3. 2. 誘導を行った場合の評価プログラム

C*****	SIMULATION FOR YUDO AT INTERSECTION *****	00000100
C		00000200
	INTEGER IROAD(2, 400, 6), NIROAD(2, 400, 6), TIME, F(0:17), BR(0:16),	00000300
	+AC(0:16), IX1, IX2, DT(2, 2, 1000), DP(2),	00000400
	+TE, GECHAR(2), SIGCT(400), SIGCP, ENDSGT, NSIG, LYTL, TDS, TDE, TDI,	00000500
	+IROADC(2, 400, 7), ICOMZN(400, 2), IZN(400), NIRDC(2, 400, 7), DSW, DSL	00000600
	REAL GETIME(2), Q(2)	00000700
	CHARACTER DSPR(2, 400), NDSPR(2, 400), CSQ(20), CSIGC(400)	00000800
C		00000900
	COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,	00001000
	+ /FT/F, BR, AC,	00001100
	+ /NRD/NIROAD, /CNRD/NDSPR, /RND/IX1, IX2,	00001200
	+ /ODT/DT, DP, /SIG/NSIG, ENDSGT, LYTL, SIGCT, SIGCP, /CSIG/CSIGC,	00001300
	+ /COM/IROADC, ICOMZN, /NRDC/NIRDC	00001400
C		00001500
54	CONTINUE	00001600
	READ(5, 51) ICONT	00001700

51	FORMAT(I2)	00001800
	IF(ICONT.EQ.0) GOTO 52	00001900
C		00002000
	DO 10 I=1, 2	00002100
	DO 10 J=1, 400	00002200
	IROAD(I, J, 1)=0	00002300
	DSPR(I, J)=' '	00002400
	SIGCT(J)=0	00002500
	CSIGC(J)=' '	00002600
10	CONTINUE	00002700
	DO 15 I=1, 2	00002800
	DO 15 J=1, 1000	00002900
	DT(I, 1, J)=0	00003000
15	CONTINUE	00003100
	DO 17 I=1, 2	00003200
	DP(I)=1	00003300
	GETIME(I)=0	00003400
17	CONTINUE	00003500
	GECHAR(1)=1	00003600
	GECHAR(2)=11	00003700
	NSIG=1	00003800
	ENDSGT=30	00003900
	SIGCP=1	00004000
C		00004100
	WRITE(6, 110)	00004200
110	FORMAT(1H0, 'RIGHT-TURNING VEHICLE CONTROL BY SMALL. ZONE')	00004300
C		00004400
	READ(5, 111) JZ	00004500
111	FORMAT(I3)	00004600
	WRITE(6, 112) JZ	00004700
112	FORMAT(1H0, 'INTERVAL OF COMMUNICATION ZONE=', I2)	00004800
	DO 3 I=1, 400	00004900
	IZN(I)=0	00005000
	IF(MOD(I, JZ).EQ.0) IZN(I)=1	00005100
3	CONTINUE	00005200
	I=1	00005300
	DO 5 J=1, 400	00005400
	ICOMZN(J, 1)=IZN(J)	00005500
	ICOMZN(J, 2)=I	00005600
	IF(IZN(J).EQ.1) I=I+1	00005700
5	CONTINUE	00005800
C		00005900
	READ(5, 210) TE	00006000
210	FORMAT(I5)	00006100
	WRITE(6, 211) TE	00006200
211	FORMAT(1H0, 'SIMULATION TIME LENGTH=', I5)	00006300
	READ(5, 220) TDS, TDE, TDI	00006400
220	FORMAT(3I5)	00006500
	WRITE(6, 221) TDS, TDE, TDI	00006600
221	FORMAT(1H0, 'DISPLAYED TIME =', I5, ' ->', I5, ' INTERVAL=', I5)	00006700
	READ(5, 230) DSW, DSL	00006800
230	FORMAT(2I5)	00006900
	WRITE(6, 231) DSW, DSL	00007000
231	FORMAT(1H0, 'DISPLAYED WIDTH =', I5, ' DISPLAYED LINES=', I5,)	00007100
	READ(5, 310) IX1, IX2	00007200

310	FORMAT(2I2)		00007300
	WRITE(6,311) IX1,IX2		00007400
311	FORMAT(1H0,'RANDOMIZING PARAMETER=',2I2)		00007500
	READ(5,410) Q(1),Q(2)		00007600
410	FORMAT(2F5.3)		00007700
	WRITE(6,411) Q(1),Q(2)		00007800
411	FORMAT(1H0,'AVR. TIME HEADWAY, 1ST HALF, 2ND HALF=',2F7.4)		00007900
	READ(5,510) QR1,QR2		00008000
510	FORMAT(2F5.3)		00008100
	WRITE(6,511) QR1,QR2		00008200
511	FORMAT(1H0,'RIGHT-TURNING CAR RATIO IN 1ST HALF, 2ND HALF=',2F7.4)		00008300
C			00008400
C	OPEN(1,FILE='AA30851. @SIG1. DATA',STATUS='OLD',		00008500
C	+ FORM='FORMATTED')		00008600
C	OPEN(2,FILE='AA30851. @SIG2. DATA',STATUS='OLD',		00008700
C	+ FORM='FORMATTED')		00008800
C			00008900
	DO 20 TIME=1,TE		00009000
	IF(TIME.LT.TE/2)THEN		00009100
	QGEN=Q(1)		00009200
	CALL STCAR2(QGEN,QR1)		00009300
	ELSE		00009400
	QGEN=Q(2)		00009500
	CALL STCAR2(QGEN,QR2)		00009600
	ENDIF		00009700
	CALL SIGCHN		00009800
	CALL CHCAR2		00009900
	CALL CHEXE1		00010000
	IF(TIME.GE.TDS.AND.TIME.LE.TDE.AND.		00010100
	+ MOD((TIME-TDS),TDI).EQ.0) CALL PRINT2(DSW,DSL)		00010200
	CALL MOV2		00010300
	CALL COMMN		00010400
	20 CONTINUE		00010500
C			00010600
	WRITE(6,610)		00010700
610	FORMAT(1H0,'SIGNAL CHANGE	')	00010800
	I=1		00010900
	DO 60 WHILE(SIGCT(I).NE.0)		00011000
	WRITE(6,611) SIGCT(I),CSIGC(I)		00011100
	I=I+1		00011200
	60 CONTINUE		00011300
611	FORMAT(1H0,'	T=',I5,' SIGNAL=',A1)	00011400
C			00011500
	DO 30 I=1,2		00011600
	WRITE(6,710)		00011700
	WRITE(6,711) I		00011800
	J=1		00011900
	AV1=0		00012000
	AV2=0		00012100
	N1=0		00012200
	N2=0		00012300
	M1=0		00012400
	M2=0		00012500
	DO 40 WHILE(DT(I,1,J).NE.0)		00012600
	IF(DT(I,1,J).LE.TE/2)THEN		00012700

	AV1=AV1+DT(I, 2, J)	00012800
	N1=N1+1	00012900
	ELSE	00013000
	AV2=AV2+DT(I, 2, J)	00013100
	N2=N2+1	00013200
	ENDIF	00013300
	J=J+1	00013400
40	CONTINUE	00013500
	IF(I.EQ.1)THEN	00013600
C	OPEN(11, FILE='AA30851.@DES1.DATA', STATUS='OLD',	00013700
C	+ FORM='UNFORMATTED')	00013800
	ELSE	00013900
C	OPEN(11, FILE='AA30851.@DES2.DATA', STATUS='OLD',	00014000
C	+ FORM='UNFORMATTED')	00014100
	ENDIF	00014200
C	WRITE(11)J-1, (FLOAT(DT(I, 1, K)), FLOAT(DT(I, 2, K)), K=1, J-1)	00014300
C	CLOSE(11)	00014400
C		00014500
	IF(I.EQ.1)THEN	00014600
	DO 41 L1=1, 2	00014700
	DO 41 L2=1, 400	00014800
	IF(IROAD(L1, L2, 1).EQ.1.AND. IROAD(L1, L2, 2).NE.2)THEN	00014900
	IF(IROAD(L1, L2, 6).LE. (TE/2))THEN	00015000
	M1=M1+1	00015100
	ELSE	00015200
	M2=M2+1	00015300
	ENDIF	00015400
	ENDIF	00015500
41	CONTINUE	00015600
	ELSE	00015700
	DO 42 L1=1, 2	00015800
	DO 42 L2=1, 400	00015900
	IF(IROAD(L1, L2, 1).EQ.1.AND. IROAD(L1, L2, 2).EQ.2)THEN	00016000
	IF(IROAD(L1, L2, 6).LE. (TE/2))THEN	00016100
	M1=M1+1	00016200
	ELSE	00016300
	M2=M2+1	00016400
	ENDIF	00016500
	ENDIF	00016600
42	CONTINUE	00016700
	ENDIF	00016800
C		00016900
	WRITE(6, 717) M1+N1	00017000
	IF(N1.NE.0)THEN	00017100
	FN1=AV1/N1	00017200
	WRITE(6, 712) N1, FN1	00017300
	ELSE	00017400
	WRITE(6, 713) N1	00017500
	ENDIF	00017600
	WRITE(6, 718) M2+N2	00017700
	IF(N2.NE.0)THEN	00017800
	FN2=AV2/N2	00017900
	WRITE(6, 714) N2, FN2	00018000
	ELSE	00018100
	WRITE(6, 715) N2	00018200

ENDIF	00018300
WRITE(6, 716) (N1+N2), ((AV1+AV2)/(N1+N2))	00018400
30 CONTINUE	00018500
710 FORMAT(1H0, '.....')	00018600
711 FORMAT(1H0, 'DESTINATION=' I1)	00018700
712 FORMAT(1H0, 'NUMBER OF CAR(1) =', I4, ' AVARAGE TIME(1)=' , F7.2)	00018800
713 FORMAT(1H0, 'NUMBER OF CAR(1) =', I4)	00018900
714 FORMAT(1H0, 'NUMBER OF CAR(2) =', I4, ' AVARAGE TIME(1)=' , F7.2)	00019000
715 FORMAT(1H0, 'NUMBER OF CAR(2) =', I4)	00019100
716 FORMAT(1H0, 'TOTAL NUMBER OF CAR=' , I4, ' AVARAGE TIME=' , F7.2)	00019200
717 FORMAT(1H0, 'GENERATED NUM. OF CAR(1) =', I4)	00019300
718 FORMAT(1H0, 'GENERATED NUM. OF CAR(2) =', I4)	00019400
C CLOSE(1)	00019500
C CLOSE(2)	00019600
C	00019700
GOTO 54	00019800
C	00019900
52 WRITE(6, 53)	00020000
53 FORMAT(1H0, 'SIMULATION ENDED')	00020100
STOP	00020200
END	00020300
C	00020400
C	00020500
C	00020600
BLOCK DATA	00020700
INTEGER F(0:17), BR(0:16), AC(0:16)	00020800
CHARACTER CSQ(20)	00020900
COMMON /FT/F, BR, AC, /CST/CSQ	00021000
DATA F/0, 0, 4, 6, 8, 9, 10, 11, 11, 12, 13, 13, 14, 14, 14, 15, 15, 16/	00021100
DATA BR/0, 0, 0, 0, 0, 0, 0, 0, 4, 4, 6, 8, 10, 11, 12, 13, 14/	00021200
DATA AC/4, 4, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 16, 16/	00021300
DATA CSQ/'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',	00021400
+ '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' /	00021500
END	00021600
C	00021700
C	00021800
C	00021900
SUBROUTINE STCAR2(QGEN, QR)	00022000
C	00022100
REAL QGEN, QR	00022200
C	00022300
INTEGER IROAD(2, 400, 6), TIME, IX1, IX2, F(0:17), GECHAR(2)	00022400
REAL GETIME(2)	00022500
CHARACTER DSPR(2, 400), CSQ(20)	00022600
COMMON /RD/IROAD, /CRD/DSPR, /T/TIME, /ST/GETIME, GECHAR, /CST/CSQ,	00022700
+ /RND/IX1, IX2, /FT/F	00022800
C	00022900
REAL A(1)	00023000
C	00023100
DO 10 I=1, 2	00023200
IF(TIME. GE. GETIME(I)) THEN	00023300
ISP=F(IPD(I, 400, I))	00023400
IP=INT(400-ISP*(TIME-GETIME(I))/4)	00023500
IROAD(I, IP, 1)=1	00023600
CALL RANU2(IX1, A, 1, ICON)	00023700

IF(I.EQ.1.AND.A(1).LT.2*QR)THEN	00023800
IROAD(I,IP,2)=1	00023900
DSPR(I,IP)=CSQ(GECHAR(1))	00024000
IF(GECHAR(1).EQ.10)THEN	00024100
GECHAR(1)=1	00024200
ELSE	00024300
GECHAR(1)=GECHAR(1)+1	00024400
ENDIF	00024500
ELSE	00024600
IROAD(I,IP,2)=2	00024700
DSPR(I,IP)=CSQ(GECHAR(2))	00024800
IF(GECHAR(2).EQ.20)THEN	00024900
GECHAR(2)=11	00025000
ELSE	00025100
GECHAR(2)=GECHAR(2)+1	00025200
ENDIF	00025300
ENDIF	00025400
IROAD(I,IP,3)=0	00025500
IROAD(I,IP,4)=16	00025600
IROAD(I,IP,5)=ISP	00025700
IROAD(I,IP,6)=TIME	00025800
CALL COMINI(I,IP)	00025900
CALL RANE2(QGEN-1,IX2,A,1,ICON)	00026000
GETIME(I)=GETIME(I)+1+A(1)	00026100
ENDIF	00026200
10 CONTINUE	00026300
RETURN	00026400
END	00026500
C	00026600
C	00026700
C	00026800
SUBROUTINE SIGCHN	00026900
C	00027000
INTEGER SIGCP,ENDSGT,LYTL,NSIG,SIGCT(400),TIME,IROAD(2,400,6)	00027100
CHARACTER CSIGC(400),CHSIG(3)	00027200
COMMON /SIG/NSIG,ENDSGT,LYTL,SIGCT,SIGCP,/CSIG/CSIGC,/RD/IROAD,	00027300
+ /T/TIME	00027400
C	00027500
INTEGER ICK1,ICK2,ICK3	00027600
C	00027700
CHSIG(1)='G'	00027800
CHSIG(2)='Y'	00027900
CHSIG(3)='R'	00028000
IBD101=IBD(1,0,1)	00028100
IF(NSIG.EQ.1)THEN	00028200
IF(TIME.GE.ENDSGT.AND.(IROAD(1,IBD101,2).NE.2.OR.	00028300
+ IBD101.EQ.17))THEN	00028400
NSIG=2	00028500
SIGCT(SIGCP)=TIME	00028600
CSIGC(SIGCP)=CHSIG(NSIG)	00028700
SIGCP=SIGCP+1	00028800
ENDIF	00028900
ENDIF	00029000
IF(NSIG.EQ.2)THEN	00029100
CALL RCARCK(ICK1,ICK2,ICK3)	00029200

	IF((ICK3.EQ.0.AND.ICK1.EQ.400.	00029300
+	AND.(IROAD(1,IBD101,2).NE.1.OR.IBD101.EQ.17)).	00029400
+	OR.(IBD101.NE.17.	00029500
+	AND.(IROAD(1,IBD101,2).NE.1.	00029600
+	AND.IROAD(1,IBD101,2).NE.3.	00029700
+	AND.IROAD(1,IBD101,2).NE.4))) THEN	00029800
	NSIG=3	00029900
	SIGCT(SIGCP)=TIME	00030000
	CSIGC(SIGCP)=CHSIG(NSIG)	00030100
	SIGCP=SIGCP+1	00030200
	LYTL=TIME-ENDSGT	00030300
	ENDSGT=TIME+30+LYTL	00030400
	CALL DEFTPR	00030500
	ENDIF	00030600
	ENDIF	00030700
	IF(NSIG.EQ.3) THEN	00030800
	IF(TIME.GE.ENDSGT) THEN	00030900
	NSIG=1	00031000
	SIGCT(SIGCP)=TIME	00031100
	CSIGC(SIGCP)=CHSIG(NSIG)	00031200
	SIGCP=SIGCP+1	00031300
	ENDSGT=TIME+30	00031400
	ENDIF	00031500
	ENDIF	00031600
	RETURN	00031700
	END	00031800
C		00031900
C		00032000
C		00032100
	SUBROUTINE DEFTPR	00032200
C		00032300
	INTEGER IROAD(2,400,6)	00032400
	COMMON /RD/IROAD	00032500
C		00032600
	CALL RCARCK(ICK1,ICK2,ICK3)	00032700
C		00032800
	IF(ICK1.NE.400) THEN	00032900
	IF(ICK2.EQ.400) THEN	00033000
	CALL DFTPR2(I2)	00033100
	IF(I2.NE.400.AND.I2.GT.ICK1) THEN	00033200
	IROAD(1,I2,2)=4	00033300
	ENDIF	00033400
	ENDIF	00033500
	ELSE	00033600
	IF(ICK2.EQ.400) THEN	00033700
	K=1	00033800
	DO 10 WHILE((IROAD(1,K,2).NE.1.OR.IROAD(1,K,1).NE.1).AND.	00033900
+	K.NE.400)	00034000
	K=K+1	00034100
10	CONTINUE	00034200
	IF(K.NE.400) THEN	00034300
	IROAD(1,K,2)=3	00034400
	CALL DFTPR2(I2)	00034500
	IF(I2.NE.400.AND.I2.GT.K) THEN	00034600
	IROAD(1,I2,2)=4	00034700

ENDIF	00034800
ENDIF	00034900
ELSEIF(ICK3.EQ.0)THEN	00035000
K=ICK2	00035100
IROAD(1,K,2)=3	00035200
CALL DFTPR2(I2)	00035300
IF(I2.NE.400.AND.I2.GT.K)THEN	00035400
IROAD(1,I2,2)=4	00035500
ENDIF	00035600
ELSE	00035700
IROAD(1,ICK2,2)=1	00035800
K=1	00035900
DO 20 WHILE((IROAD(1,K,2).NE.1.OR.IROAD(1,K,1).NE.1).AND.	00036000
+ K.NE.400)	00036100
K=K+1	00036200
20 CONTINUE	00036300
IF(K.NE.400)THEN	00036400
IROAD(1,K,2)=3	00036500
CALL DFTPR2(I2)	00036600
IF(I2.NE.400.AND.I2.GT.K)THEN	00036700
IROAD(1,I2,2)=4	00036800
ENDIF	00036900
ENDIF	00037000
ENDIF	00037100
ENDIF	00037200
C RETURN	00037300
END	00037400
C	00037500
C	00037600
C	00037700
C	00037800
SUBROUTINE DFTPR2(I2)	00037900
C	00038000
INTEGER IROAD(2,400,6),IROADC(2,400,7),ICOMZN(400,2),TIME	00038100
COMMON /RD/IROAD,/COM/IROADC,ICOMZN,/T/TIME	00038200
C	00038300
K=1	00038400
DO 10 WHILE((IROAD(1,K,2).NE.3.OR.IROAD(1,K,1).NE.1).AND.	00038500
+ K.NE.400)	00038600
K=K+1	00038700
10 CONTINUE	00038800
L=200	00038900
I=NCTT(L,K)	00039000
IF(I.LT.60)THEN	00039100
L=L+100	00039200
ELSE	00039300
L=L-100	00039400
ENDIF	00039500
I=NCTT(L,K)	00039600
IF(I.LT.60)THEN	00039700
L=L+50	00039800
ELSE	00039900
L=L-50	00040000
ENDIF	00040100
I=NCTT(L,K)	00040200

IF(I. LT. 60) THEN	00040300
L=L+25	00040400
ELSE	00040500
L=L-25	00040600
ENDIF	00040700
I=NCTT(L, K)	00040800
IF(I. LT. 60) THEN	00040900
L=L+13	00041000
ELSE	00041100
L=L-13	00041200
ENDIF	00041300
I=NCTT(L, K)	00041400
IF(I. LT. 60) THEN	00041500
L=L+7	00041600
ELSE	00041700
L=L-7	00041800
ENDIF	00041900
I=NCTT(L, K)	00042000
IF(I. LT. 60) THEN	00042100
L=L+4	00042200
ELSE	00042300
L=L-4	00042400
ENDIF	00042500
I=NCTT(L, K)	00042600
IF(I. LT. 60) THEN	00042700
L=L+2	00042800
ELSE	00042900
L=L-2	00043000
ENDIF	00043100
I=NCTT(L, K)	00043200
IF(I. LT. 60) THEN	00043300
L=L+1	00043400
ELSE	00043500
L=L-1	00043600
ENDIF	00043700
IF(L. LT. 400) THEN	00043800
LM=L	00043900
DO 30 WHILE(ICOMZN(LM, 1). NE. 1. AND. LM. NE. 400)	00044000
LM=LM+1	00044100
30 CONTINUE	00044200
LRAS=400	00044300
LRS=0	00044400
LR=LM	00044500
DO 40 WHILE(ICOMZN(LR, 2). EQ. ICOMZN(LM, 2). AND. LR. NE. 0)	00044600
IF(IROAD(1, LR, 1). EQ. 1. AND. IROAD(1, LR, 2). NE. 2) THEN	00044700
LRA=LM-(IROADC(1, LR, 3)*(TIME-IROADC(1, LR, 5))/4)	00044800
IF(LRA. GT. L. AND. LRA. LT. LRAS) THEN	00044900
LRAS=LRA	00045000
LRS=LR	00045100
ENDIF	00045200
ENDIF	00045300
LR=LR-1	00045400
40 CONTINUE	00045500
IF(LRS. EQ. 0) THEN	00045600
IF(LM. LT. 400) THEN	00045700

	L=LM+1	00045800
	DO 50 WHILE((IROAD(1, L, 2).EQ. 2. OR. IROAD(1, L, 1).NE. 1).	00045900
+	AND. L. NE. 400)	00046000
	L=L+1	00046100
50	CONTINUE	00046200
	ELSE	00046300
	L=400	00046400
	ENDIF	00046500
	ELSE	00046600
	L=LRS	00046700
	ENDIF	00046800
	ELSE	00046900
	L=400	00047000
	ENDIF	00047100
	I2=L	00047200
	RETURN	00047300
	END	00047400
C		00047500
C		00047600
C		00047700
	SUBROUTINE CHLCNC(L1, L2)	00047800
C		00047900
	INTEGER L1, L2	00048000
C		00048100
	INTEGER IROAD(2, 400, 6)	00048200
	COMMON /RD/IROAD	00048300
C		00048400
	DO 10 I=1, 2	00048500
	DO 10 J=L1, L2	00048600
	IF(IROAD(I, J, 1).EQ. 1) IROAD(I, J, 3)=0	00048700
10	CONTINUE	00048800
	RETURN	00048900
	END	00049000
C		00049100
C		00049200
C		00049300
	SUBROUTINE CHLVEL(I, J)	00049400
C		00049500
	INTEGER I, J	00049600
C		00049700
	INTEGER IROAD(2, 400, 6), F(0:17)	00049800
	COMMON /RD/IROAD, /FT/F	00049900
C		00050000
	INTEGER IV1, IV2	00050100
C		00050200
	K=MOD(I, 2)+1	00050300
	IF(IROAD(I, J, 1).EQ. 1. AND. IROAD(I, J, 2).EQ. 2) THEN	00050400
	IV1=F(IFD(I, J, I))	00050500
	IV2=F(IFD(I, J, K))	00050600
	IF(IV1. LE. IV2-4) THEN	00050700
	IROAD(I, J, 3)=1	00050800
	ENDIF	00050900
	IF(IROAD(I, J, 3).EQ. 1. AND. IV1. GE. IV2) THEN	00051000
	IROAD(I, J, 3)=0	00051100
	ENDIF	00051200

ENDIF	00051300
RETURN	00051400
END	00051500
C	00051600
C	00051700
C	00051800
SUBROUTINE RCARCK(ICK1, ICK2, ICK3)	00051900
C	00052000
INTEGER ICK1, ICK2, ICK3	00052100
C	00052200
INTEGER IROAD(2, 400, 6)	00052300
COMMON /RD/IROAD	00052400
C	00052500
ICK1=1	00052600
DO 10 WHILE((IROAD(1, ICK1, 2).NE. 3. OR. IROAD(1, ICK1, 1).NE. 1). AND.	00052700
+ ICK1.LT. 400)	00052800
ICK1=ICK1+1	00052900
10 CONTINUE	00053000
ICK2=1	00053100
DO 20 WHILE((IROAD(1, ICK2, 2).NE. 4. OR. IROAD(1, ICK2, 1).NE. 1). AND.	00053200
+ ICK2.LT. 400)	00053300
ICK2=ICK2+1	00053400
20 CONTINUE	00053500
C	00053600
IF(ICK2.NE. 400)THEN	00053700
ICK3=ICK2	00053800
IF(ICK1.LT. ICK2)THEN	00053900
DO 31 WHILE(ICK3.NE. ICK1. AND.	00054000
+ (IROAD(1, ICK3, 2).NE. 1. OR. IROAD(1, ICK3, 1).NE. 1))	00054100
ICK3=ICK3-1	00054200
31 CONTINUE	00054300
IF(ICK3.EQ. ICK1) ICK3=0	00054400
ELSEIF(ICK1.EQ. 400)THEN	00054500
DO 32 WHILE(ICK3.NE. 0. AND.	00054600
+ (IROAD(1, ICK3, 2).NE. 1. OR. IROAD(1, ICK3, 1).NE. 1))	00054700
ICK3=ICK3-1	00054800
32 CONTINUE	00054900
ELSE	00055000
ICK3=0	00055100
ENDIF	00055200
ELSE	00055300
ICK3=0	00055400
ENDIF	00055500
RETURN	00055600
END	00055700
C	00055800
C	00055900
C	00056000
SUBROUTINE CHEXE1	00056100
C	00056200
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16), TIME,	00056300
+ IROADC(2, 400, 7), ICOMZN(400, 2)	00056400
CHARACTER DSPR(2, 400)	00056500
COMMON /RD/IROAD, /CRD/DSPR, /FT/F, BR, /T/TIME,	00056600
+ /COM/IROADC, ICOMZN	00056700

C

DO 10 I=1, 2	00056800
K=MOD(I, 2)+1	00056900
DO 10 J=1, 400	00057000
IF(IROAD(I, J, 1).EQ. 0. OR. IROAD(I, J, 3).NE. 1)GOTO 10	00057100
ISP1=IROAD(I, J, 5)	00057200
IFG=IFD(I, J, K)	00057300
IBG=IBD(I, J, K)	00057400
IF (IBG. EQ. 17) THEN	00057500
ISP2=0	00057600
ELSE	00057700
ISP2=IROAD(K, J+IBG, 5)	00057800
ENDIF	00057900
IF (F(IFG). GT. BR (ISP1)) THEN	00058000
IF (F (IBG). GT. BR (ISP2)) THEN	00058100
IF (IROAD(K, J, 1). EQ. 0) THEN	00058200
L=J	00058300
ELSEIF (IROAD(K, J-1, 1). EQ. 0) THEN	00058400
L=J-1	00058500
ELSE	00058600
L=-1	00058700
ENDIF	00058800
IF (L. GE. 1) THEN	00058900
DO 20 M=1, 6	00059000
IROAD(K, L, M)=IROAD(I, J, M)	00059100
20 CONTINUE	00059200
DO 21 M=1, 7	00059300
IROADC(K, L, M)=IROADC(I, J, M)	00059400
21 CONTINUE	00059500
IROAD(K, L, 3)=0	00059600
DSPR(K, L)=DSPR(I, J)	00059700
IROAD(I, J, 1)=0	00059800
DSPR(I, J)=	00059900
ELSEIF (L. EQ. 0) THEN	00060000
CALL TRC2(I, J)	00060100
IROAD(I, J, 1)=0	00060200
DSPR(I, J)=	00060300
ENDIF	00060400
ELSEIF (IROAD(K, J+IBG, 3). EQ. 0) THEN	00060500
IF (IVELO(I, J). GT. 0. OR. IBG. GT. 1) THEN	00060600
IROAD(K, J+IBG, 3)=2	00060700
ENDIF	00060800
ENDIF	00060900
ENDIF	00061000
10 CONTINUE	00061100
RETURN	00061200
END	00061300
C	00061400
C	00061500
C	00061600
SUBROUTINE MOV2	00061700
C	00061800
INTEGER IROAD(2, 400, 6), NIROAD(2, 400, 6), F(0:17), NSIG,	00061900
+ IROADC(2, 400, 7), ICOMZN(400, 2), NIRD(2, 400, 7)	00062000
CHARACTER DSPR(2, 400), NDSPR(2, 400)	00062100
	00062200

COMMON /RD/IROAD, /CRD/DSPR, /NRD/NROAD, /CNRD/NDSPR, /SIG/NSIG,	00062300
+ /FT/F,	00062400
+ /COM/IROADC, ICOMZN, /NRDC/NIRDC	00062500
C INTEGER TOP	00062600
C	00062700
DO 10 I=1, 2	00062800
DO 10 J=1, 400	00062900
NROAD(I, J, 1)=0	00063000
NDSPR(I, J)=	00063100
10 CONTINUE	00063200
DO 20 I=1, 2	00063300
TOP=0	00063400
DO 20 J=1, 400	00063500
IF(IROAD(I, J, 1).EQ.1)THEN	00063600
ISP=IVEL3(I, J)	00063700
IF(TOP.EQ.0)THEN	00063800
IF(NSIG.EQ.3.OR.(NSIG.EQ.2.AND.IROAD(I, J, 2).EQ.2).OR.	00063900
+ (NSIG.EQ.1.AND.IROAD(I, J, 2).NE.2))THEN	00064000
K=MIN(J, 17)	00064100
ISP=MIN(ISP, F(K))	00064200
TOP=1	00064300
ENDIF	00064400
ENDIF	00064500
IROAD(I, J, 5)=ISP	00064600
CALL MOVE(I, J, ISP)	00064700
ENDIF	00064800
20 CONTINUE	00064900
DO 30 I=1, 2	00065000
DO 30 J=1, 400	00065100
DSPR(I, J)=NDSPR(I, J)	00065200
DO 31 K=1, 7	00065300
IROADC(I, J, K)=NIRDC(I, J, K)	00065400
31 CONTINUE	00065500
DO 30 K=1, 6	00065600
IROAD(I, J, K)=NROAD(I, J, K)	00065700
30 CONTINUE	00065800
RETURN	00065900
END	00066000
C	00066100
C	00066200
C	00066300
C SUBROUTINE MOVE(I, J, ISP)	00066400
C	00066500
C INTEGER I, J, ISP	00066600
C	00066700
C	00066800
INTEGER IROAD(2, 400, 6), NROAD(2, 400, 6), TIME,	00066900
+ IROADC(2, 400, 7), ICOMZN(400, 2), NIRDC(2, 400, 7)	00067000
CHARACTER DSPR(2, 400), NDSPR(2, 400)	00067100
COMMON /RD/IROAD, /CRD/DSPR, /NRD/NROAD, /CNRD/NDSPR, /T/TIME	00067200
+ , /COM/IROADC, ICOMZN, /NRDC/NIRDC	00067300
C	00067400
K=MOD(ISP, 4)	00067500
L=(ISP-K)/4	00067600
M=MOD(TIME, 4)	00067700

IF(K*M.GE.3)L=L+1	00067800
IF(J.GT.L)THEN	00067900
NDSPR(I,J-L)=DSPR(I,J)	00068000
DO 11 N=1,7	00068100
NIRDC(I,J-L,N)=IROADC(I,J,N)	00068200
11 CONTINUE	00068300
DO 10 N=1,6	00068400
NIROAD(I,J-L,N)=IROAD(I,J,N)	00068500
10 CONTINUE	00068600
IF(NIROAD(I,J-L,3).EQ.2)NIROAD(I,J-L,3)=0	00068700
ELSE	00068800
CALL TRC2(I,J)	00068900
ENDIF	00069000
RETURN	00069100
END	00069200
C	00069300
C	00069400
C	00069500
SUBROUTINE TRC2(I,J)	00069600
C	00069700
INTEGER I,J	00069800
C	00069900
INTEGER IROAD(2,400,6),TIME,DT(2,2,1000),DP(2)	00070000
COMMON /RD/IROAD,/T/TIME,/ODT/DT,DP	00070100
C	00070200
IF(IROAD(I,J,2).EQ.2)THEN	00070300
K=2	00070400
ELSE	00070500
K=1	00070600
ENDIF	00070700
DT(K,1,DP(K))=IROAD(I,J,6)	00070800
DT(K,2,DP(K))=TIME-IROAD(I,J,6)	00070900
DP(K)=DP(K)+1	00071000
RETURN	00071100
END	00071200
C	00071300
C	00071400
C	00071500
SUBROUTINE PRINT2(JW,J1)	00071600
C	00071700
INTEGER JW,J1	00071800
INTEGER IROAD(2,400,6),TIME,IROADC(2,400,7),ICOMZN(400,2)	00071900
CHARACTER DSPR(2,400)	00072000
COMMON /RD/IROAD,/CRD/DSPR,/T/TIME,/SIG/NSIG,	00072100
+ /COM/IROADC,ICOMZN	00072200
C	00072300
INTEGER ICK1,ICK2,ICK3	00072400
CHARACTER DD(2,400),C(400),CHSIG(3)	00072500
C	00072600
CHSIG(1)='G'	00072700
CHSIG(2)='Y'	00072800
CHSIG(3)='R'	00072900
CALL RCARCK(ICK1,ICK2,ICK3)	00073000
DO 10 I=1,2	00073100
DO 10 J=1,400	00073200

DD(I, J)=DSPR(I, J)	00073300
IF(IROAD(1, J, 1).EQ. 1.AND. J.EQ. ICK1)DD(1, J)=' ('	00073400
IF(IROAD(1, J, 1).EQ. 1.AND. J.EQ. ICK2)DD(1, J)=' *'	00073500
IF(IROAD(1, J, 1).EQ. 1.AND. J.EQ. ICK3)DD(1, J)=')'	00073600
IF(IROAD(I, J, 1).EQ. 1.AND. IROAD(I, J, 3).EQ. 1)DD(I, J)=' +'	00073700
IF(IROAD(I, J, 1).EQ. 1.AND. IROAD(I, J, 3).EQ. 2)DD(I, J)=' -'	00073800
10 CONTINUE	00073900
DO 20 I=1, 400	00074000
IF(ICOMZN(I, 1).EQ. 1)THEN	00074100
C(I)=' Z'	00074200
ELSE	00074300
C(I)=' .'	00074400
ENDIF	00074500
20 CONTINUE	00074600
WRITE(6, 101)	00074700
WRITE(6, 102) TIME, CHSIG(NSIG)	00074800
DO 21 K=1, J1	00074900
K1=JW*(K-1)+1	00075000
K2=JW*K	00075100
WRITE(6, 100) (DD(1, J), J=K1, K2)	00075200
WRITE(6, 100) (DD(2, J), J=K1, K2)	00075300
WRITE(6, 100) (C(I), I=K1, K2)	00075400
21 CONTINUE	00075500
WRITE(6, 101)	00075600
100 FORMAT('@', 100A1)	00075700
101 FORMAT(1H0, '*****')	00075800
102 FORMAT(1H0, ' TIME=', I5, ' NOW SIGNAL=', A1)	00075900
RETURN	00076000
END	00076100
C	00076200
C	00076300
C	00076400
INTEGER FUNCTION NCTT(L, K)	00076500
C	00076600
INTEGER L, K	00076700
C	00076800
INTEGER F(0:17)	00076900
COMMON /FT/F	00077000
C	00077100
INTEGER NC, QS, AVVEL	00077200
C	00077300
IF(L.GT. K)THEN	00077400
NC=NQC2(1, K, L)+NQC2(2, 1, L)	00077500
ELSE	00077600
NC=NQC2(2, 1, L)	00077700
ENDIF	00077800
IF(NC.EQ. 0)THEN	00077900
QS=17	00078000
ELSE	00078100
QS=L/NC	00078200
ENDIF	00078300
IF(QS.GT. 17)QS=17	00078400
AVVEL=F(QS)	00078500
IF(AVVEL.LE. 8)THEN	00078600
NCTT=2*NC	00078700

ELSE	00078800
NCTT=4*L/AVVEL	00078900
ENDIF	00079000
RETURN	00079100
END	00079200
C	00079300
C	00079400
C	00079500
INTEGER FUNCTION IVEL3(I, J)	00079600
C	00079700
INTEGER I, J	00079800
C	00079900
INTEGER IROAD(2, 400, 6), NSIG, LYTL, ENDSGT, TIME, F(0:17), BR(0:16)	00080000
COMMON /RD/IROAD, /SIG/NSIG, ENDSGT, LYTL, /T/TIME, /FT/F, BR	00080100
C	00080200
IF(IROAD(I, J, 3).EQ.0)THEN	00080300
IVEL3=IVEL0(I, J)	00080400
ELSEIF(IROAD(I, J, 3).EQ.1)THEN	00080500
IVEL3=IVEL1(I, J)	00080600
ELSE	00080700
IVEL3=IVEL2(I, J)	00080800
ENDIF	00080900
IF(IVEL3.GT. IROAD(I, J, 4))THEN	00081000
IVEL3=MAX(IROAD(I, J, 4), BR(IROAD(I, J, 5)))	00081100
ENDIF	00081200
RETURN	00081300
END	00081400
C	00081500
C	00081600
C	00081700
INTEGER FUNCTION IVEL0(I, J)	00081800
C	00081900
INTEGER I, J	00082000
C	00082100
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16), AC(0:16)	00082200
COMMON /RD/IROAD, /FT/F, BR, AC	00082300
C	00082400
IVF=F(IFD(I, J, 1))	00082500
ISP=IROAD(I, J, 5)	00082600
IVG=AC(ISP)	00082700
IVEL0=MIN(IVF, IVG)	00082800
RETURN	00082900
END	00083000
C	00083100
C	00083200
C	00083300
INTEGER FUNCTION IVEL1(I, J)	00083400
C	00083500
INTEGER I, J	00083600
C	00083700
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16)	00083800
COMMON /RD/IROAD, /FT/F, BR	00083900
C	00084000
IV=IVEL0(I, J)	00084100
K=MOD(I, 2)+1	00084200

IFG=IFD(I, J, K)	00084300
IF(F(IFG).LT. IV. AND. F(IFG).GT. 0) THEN	00084400
IVEL1=MAX(F(IFG), BR(IROAD(I, J, 5)))	00084500
ELSE	00084600
IVEL1=IV	00084700
ENDIF	00084800
RETURN	00084900
END	00085000
C	00085100
C	00085200
C	00085300
INTEGER FUNCTION IVEL2(I, J)	00085400
C	00085500
INTEGER I, J	00085600
C	00085700
INTEGER IROAD(2, 400, 6), F(0:17), BR(0:16)	00085800
COMMON /RD/IROAD, /FT/F, BR	00085900
C	00086000
IV=IVEL0(I, J)	00086100
K=MOD(I, 2)+1	00086200
IFG=IFDC(I, J, K, 1)	00086300
IF(F(IFG).LT. IV) THEN	00086400
IVEL2=MAX(F(IFG), BR(IROAD(I, J, 5)))	00086500
ELSE	00086600
IVEL2=IV	00086700
ENDIF	00086800
RETURN	00086900
END	00087000
C	00087100
C	00087200
C	00087300
INTEGER FUNCTION IFD(I, J, K)	00087400
C	00087500
INTEGER I, J, K	00087600
C	00087700
INTEGER IROAD(2, 400, 6)	00087800
COMMON /RD/IROAD	00087900
C	00088000
IF(I.NE. K. AND. IROAD(K, J, 1).NE. 0) THEN	00088100
IF((IROAD(I, J, 3).NE. 1. AND. IROAD(K, J, 3).EQ. 1).OR.	00088200
+ (IROAD(I, J, 3).EQ. 1. AND. IROAD(K, J, 3).EQ. 1. AND. I.EQ. 2)) THEN	00088300
IFD=0	00088400
RETURN	00088500
ENDIF	00088600
ENDIF	00088700
L=1	00088800
DO 10 WHILE(IROAD(K, J-L, 1).EQ. 0. AND. J.NE. L. AND. L.NE. 17)	00088900
L=L+1	00089000
10 CONTINUE	00089100
IF(J.EQ. L) L=17	00089200
IFD=L	00089300
RETURN	00089400
END	00089500
C	00089600
C	00089700

C	INTEGER FUNCTION IBD(I, J, K)	00089800
		00089900
C	INTEGER I, J, K	00090000
		00090100
C	INTEGER IROAD(2, 400, 6)	00090200
	COMMON /RD/IROAD	00090300
		00090400
C	IF (I. NE. K. AND. IROAD(K, J, 1). NE. 0) THEN	00090500
	IF ((IROAD(I, J, 3). EQ. 1. AND. IROAD(K, J, 3). NE. 1). OR.	00090600
+	(IROAD(I, J, 3). EQ. 1. AND. IROAD(K, J, 3). EQ. 1. AND. I. EQ. 1)) THEN	00090700
	IBD=0	00090800
	RETURN	00090900
	ENDIF	00091000
	ENDIF	00091100
	L=1	00091200
	DO 10 WHILE (IROAD(K, J+L, 1). EQ. 0. AND. J+L. NE. 400. AND. L. NE. 17)	00091300
	L=L+1	00091400
10	CONTINUE	00091500
	IF (J+L. EQ. 400) L=17	00091600
	IBD=L	00091700
	RETURN	00091800
	END	00091900
		00092000
C		00092100
C		00092200
C		00092300
	INTEGER FUNCTION IFDC(I, J, K, L)	00092400
C		00092500
	INTEGER I, J, K, L	00092600
C		00092700
	INTEGER IROAD(2, 400, 6)	00092800
	COMMON /RD/IROAD	00092900
C		00093000
	M=IFD(I, J, K)	00093100
	DO 10 WHILE (IROAD(K, J-M, 3). NE. L. AND. M. LT. 17)	00093200
	M=IFD(K, J-M, K)+M	00093300
10	CONTINUE	00093400
	IF (M. GT. 17) M=17	00093500
	IFDC=M	00093600
	RETURN	00093700
	END	00093800
C		00093900
C		00094000
C		00094100
	INTEGER FUNCTION IBDC(I, J, K, L)	00094200
C		00094300
	INTEGER I, J, K, L	00094400
C		00094500
	INTEGER IROAD(2, 400, 6)	00094600
	COMMON /RD/IROAD	00094700
C		00094800
	M=IBD(I, J, K)	00094900
	DO 10 WHILE (IROAD(K, J+M, 2). NE. L. AND. M. LT. 17)	00095000
	M=IBD(K, J+M, K)+M	00095100
10	CONTINUE	00095200

IF (M. GT. 17) M=17	00095300
IBDC=M	00095400
RETURN	00095500
END	00095600
C	00095700
C	00095800
C	00095900
SUBROUTINE COMINI(I, J)	00096000
INTEGER I, J	00096100
INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2), TIME	00096200
COMMON /RD/IROAD, /COM/IROADC, ICOMZN, /T/TIME	00096300
C	00096400
IROADC(I, J, 1)=ICOMZN(400, 2)+1	00096500
IROADC(I, J, 2)=I	00096600
IROADC(I, J, 3)=IROAD(I, J, 5)	00096700
IROADC(I, J, 4)=I	00096800
IROADC(I, J, 5)=TIME	00096900
IROADC(I, J, 6)=I	00097000
IROADC(I, J, 7)=16	00097100
C	00097200
RETURN	00097300
END	00097400
C	00097500
C	00097600
C	00097700
SUBROUTINE CHCAR2	00097800
INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2)	00097900
COMMON /RD/IROAD, /COM/IROADC, ICOMZN	00098000
C	00098100
CALL CHLCNC(1, 400)	00098200
DO 10 I=1, 2	00098300
DO 10 J=11, 400	00098400
IF (IROAD(I, J, 1). EQ. 1) THEN	00098500
IF (IROADC(I, J, 2). EQ. 0) THEN	00098600
CALL CHLVEL(I, J)	00098700
ELSEIF (IROADC(I, J, 2). NE. 1) THEN	00098800
IROAD(I, J, 3)=1	00098900
ENDIF	00099000
ENDIF	00099100
10 CONTINUE	00099200
C	00099300
RETURN	00099400
END	00099500
C	00099600
C	00099700
C	00099800
SUBROUTINE COMMN	00099900
INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2)	00100000
COMMON /RD/IROAD, /COM/IROADC, ICOMZN	00100100
C	00100200
DO 10 I=1, 2	00100300
DO 10 J=1, 400	00100400
IF (IROAD(I, J, 1). EQ. 1) THEN	00100500
IF (IROADC(I, J, 1). NE. ICOMZN(J, 2)) THEN	00100600
CALL COMEXE(I, J)	00100700

	ENDIF	00100800
	ENDIF	00100900
10	CONTINUE	00101000
C		00101100
	RETURN	00101200
	END	00101300
C		00101400
C		00101500
C		00101600
	SUBROUTINE COMEXE(I, J)	00101700
	INTEGER I, J	00101800
	INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2), TIME	00101900
	COMMON /RD/IROAD, /COM/IROADC, ICOMZN, /T/TIME	00102000
C		00102100
	IROADC(I, J, 1)=ICOMZN(J, 2)	00102200
	IROADC(I, J, 2)=IROADC(I, J, 6)	00102300
	IROAD(I, J, 4)=IROADC(I, J, 7)	00102400
	IROADC(I, J, 3)=IROAD(I, J, 5)	00102500
	IROADC(I, J, 4)=I	00102600
	IROADC(I, J, 5)=TIME	00102700
	IF(J.GT.10.AND.IROAD(I, J, 2).EQ.2)THEN	00102800
	CALL YUDLN(I, J)	00102900
	ELSE	00103000
	IROADC(I, J, 6)=I	00103100
	ENDIF	00103200
	CALL YUDVEL(I, J)	00103300
C		00103400
	RETURN	00103500
	END	00103600
C		00103700
C		00103800
C		00103900
	SUBROUTINE YUDLN(I, J)	00104000
	INTEGER I, J	00104100
	INTEGER IROADC(2, 400, 7), ICOMZN(400, 2), NSIG	00104200
	COMMON /COM/IROADC, ICOMZN, /SIG/NSIG	00104300
C		00104400
	ILIM=2	00104500
C		00104600
	CALL RCARCK(ICK1, ICK2, ICK3)	00104700
	IF(ICK1.NE.400)THEN	00104800
	IF(IROADC(I, J, 1).LT.(IROADC(1, ICK1, 1)-ILIM))THEN	00104900
	IROADC(I, J, 6)=I	00105000
	ELSEIF(IROADC(I, J, 1).EQ.(IROADC(1, ICK1, 1)-ILIM))THEN	00105100
	IF(NSIG.NE.2)THEN	00105200
	CALL CHLNUM(I, J, ICK1)	00105300
	ELSE	00105400
	IROADC(I, J, 6)=I	00105500
	ENDIF	00105600
	ELSEIF(ICK3.NE.0)THEN	00105700
	IF(IROADC(I, J, 1).LE.(IROADC(1, ICK3, 1)+1))THEN	00105800
	IROADC(I, J, 6)=2	00105900
	ELSE	00106000
	IROADC(I, J, 6)=0	00106100
	ENDIF	00106200

ELSE	00106300
IF(IROADC(I, J, 1). LE. (IROADC(1, ICK1, 1)+1)) THEN	00106400
IROADC(I, J, 6)=2	00106500
ELSE	00106600
IROADC(I, J, 6)=0	00106700
ENDIF	00106800
ENDIF	00106900
ELSEIF(ICK3. NE. 0) THEN	00107000
IF(IROADC(I, J, 1). LE. (IROADC(1, ICK3, 1)+1)) THEN	00107100
IROADC(I, J, 6)=2	00107200
ELSE	00107300
IROADC(I, J, 6)=0	00107400
ENDIF	00107500
ENDIF	00107600
C	00107700
RETURN	00107800
END	00107900
C	00108000
C	00108100
C	00108200
SUBROUTINE CHLNUM(I, J, K1)	00108300
INTEGER I, J	00108400
INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2), TIME	00108500
COMMON /RD/IROAD, /COM/IROADC, ICOMZN, /T/TIME	00108600
C	00108700
NC1=0	00108800
NC2=0	00108900
KM1=K1	00109000
DO 30 WHILE(ICOMZN(KM1, 2). EQ. ICOMZN(K1, 2). AND. KM1. NE. 400)	00109100
KM1=KM1+1	00109200
30 CONTINUE	00109300
KM2=K1	00109400
DO 40 WHILE(ICOMZN(KM2, 2). EQ. ICOMZN(K1, 2). AND. KM2. NE. 0)	00109500
KM2=KM2-1	00109600
40 CONTINUE	00109700
KR1A=KM1-(IROADC(1, K1, 3)*(TIME-IROADC(1, K1, 5))/4)	00109800
IF(KR1A. LT. KM2)KR1A=KM2	00109900
DO 10 L=1, 2	00110000
K=1	00110100
DO 20 WHILE(ICOMZN(K, 2). LT. ICOMZN(K1, 2). AND. K. NE. 400)	00110200
IF(IROAD(L, K, 1). EQ. 1. AND. IROAD(L, K, 2). EQ. 2) THEN	00110300
IF(IROADC(L, K, 2). EQ. 1) THEN	00110400
NC1=NC1+1	00110500
ELSEIF(IROADC(L, K, 2). EQ. 2) THEN	00110600
NC2=NC2+1	00110700
ELSEIF(IROADC(L, K, 2). EQ. 0) THEN	00110800
IF(IROADC(L, K, 4). EQ. 1) THEN	00110900
NC1=NC1+1	00111000
ELSE	00111100
NC2=NC2+1	00111200
ENDIF	00111300
ENDIF	00111400
ENDIF	00111500
K=K+1	00111600
20 CONTINUE	00111700

DO 21 WHILE (ICOMZN(K, 2). EQ. ICOMZN(K1, 2). AND. K. NE. 400)	00111800
IF (IROAD(L, K, 1). EQ. 1. AND. IROAD(L, K, 2). EQ. 2) THEN	00111900
KA=KM1-(IROADC(L, K, 3)*(TIME-IROADC(L, K, 5))/4)	00112000
IF (KA. LT. KR1A) THEN	00112100
IF (IROADC(L, K, 2). EQ. 1) THEN	00112200
NC1=NC1+1	00112300
ELSEIF (IROADC(L, K, 2). EQ. 2) THEN	00112400
NC2=NC2+1	00112500
ELSEIF (IROADC(L, K, 2). EQ. 0) THEN	00112600
IF (IROADC(L, K, 4). EQ. 1) THEN	00112700
NC1=NC1+1	00112800
ELSE	00112900
NC2=NC2+1	00113000
ENDIF	00113100
ENDIF	00113200
ENDIF	00113300
ENDIF	00113400
K=K+1	00113500
21 CONTINUE	00113600
10 CONTINUE	00113700
IF (NC1. GT. NC2) THEN	00113800
IROADC(I, J, 6)=2	00113900
ELSEIF (NC1. LT. NC2) THEN	00114000
IROADC(I, J, 6)=1	00114100
ELSE	00114200
IROADC(I, J, 6)=I	00114300
ENDIF	00114400
C	00114500
RETURN	00114600
END	00114700
C	00114800
C	00114900
C	00115000
SUBROUTINE YUDVEL(I, J)	00115100
INTEGER I, J	00115200
INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2),	00115300
+ NSIG, LYTL, ENDSGT, TIME	00115400
COMMON /RD/IROAD, /COM/IROADC, ICOMZN, /SIG/NSIG, ENDSGT, LYTL,	00115500
+ /T/TIME	00115600
C	00115700
IF (IROAD(I, J, 2). EQ. 3. OR. IROAD(I, J, 2). EQ. 4) THEN	00115800
JM=J	00115900
DO 10 WHILE (ICOMZN(JM, 1). NE. 1. AND. JM. LT. 400)	00116000
JM=JM+1	00116100
10 CONTINUE	00116200
IF (IROAD(I, J, 2). EQ. 3) THEN	00116300
IF (NSIG. EQ. 3) THEN	00116400
IROADC(I, J, 7)=MAX(1, MIN(4*JM/(ENDSGT-TIME+30), 16))	00116500
ELSEIF (NSIG. EQ. 1) THEN	00116600
IF (ENDSGT-TIME. LE. 0) THEN	00116700
IROADC(I, J, 7)=16	00116800
ELSE	00116900
IROADC(I, J, 7)=MAX(1, MIN(4*JM/(ENDSGT-TIME), 16))	00117000
ENDIF	00117100
ELSE	00117200

	IROADC(I, J, 7)=16	00117300
	ENDIF	00117400
	ELSEIF(IROAD(I, J, 2).EQ.4)THEN	00117500
	IF(NSIG.EQ.3)THEN	00117600
	IROADC(I, J, 7)=MAX(1, MIN(4*JM/(ENDSGT-TIME+90+2*LYTL), 16))	00117700
	ELSE	00117800
	IF(ENDSGT-TIME+60+2*LYTL.LE.0)THEN	00117900
	IROADC(I, J, 7)=16	00118000
	ELSE	00118100
	IROADC(I, J, 7)=MAX(1, MIN(4*JM/(ENDSGT-TIME+60+2*LYTL), 16))	00118200
	ENDIF	00118300
	ENDIF	00118400
	ENDIF	00118500
	ELSE	00118600
	IROADC(I, J, 7)=16	00118700
	ENDIF	00118800
C	RETURN	00118900
	END	00119000
C		00119100
C		00119200
C		00119300
C		00119400
	INTEGER FUNCTION NQC2(I, K1, K2)	00119500
	INTEGER I, K1, K2	00119600
	INTEGER IROAD(2, 400, 6), IROADC(2, 400, 7), ICOMZN(400, 2), TIME	00119700
	COMMON /RD/IROAD, /COM/IROADC, ICOMZN, /T/TIME	00119800
C		00119900
	NQC2=0	00120000
	IF(K1.LT.K2)THEN	00120100
	LM1=K1	00120200
	DO 10 WHILE(ICOMZN(LM1, 1).NE.1.AND.LM1.NE.0)	00120300
	LM1=LM1-1	00120400
10	CONTINUE	00120500
	LM1=LM1+1	00120600
	LM2=K2	00120700
	DO 20 WHILE(ICOMZN(LM2, 1).NE.1.AND.LM2.NE.400)	00120800
	LM2=LM2+1	00120900
20	CONTINUE	00121000
	LMT1=K1	00121100
	DO 30 WHILE(ICOMZN(LMT1, 1).NE.1.AND.LMT1.NE.400)	00121200
	LMT1=LMT1+1	00121300
30	CONTINUE	00121400
	LMT2=LM2	00121500
	DO 40 J=1, 2	00121600
	DO 40 K=LM1, LM2	00121700
	IF(IROAD(J, K, 1).EQ.1.AND.IROAD(J, K, 2).EQ.2)THEN	00121800
	IF(IROADC(J, K, 2).EQ.1.OR.	00121900
+	(IROADC(J, K, 2).EQ.0.AND.IROAD(J, K, 4).EQ.1))THEN	00122000
	IF(ICOMZN(K, 2).EQ.ICOMZN(LM1, 2))THEN	00122100
	KA=LMT1-(IROADC(J, K, 3)*(TIME-IROADC(J, K, 5))/4)	00122200
	IF(KA.GE.K1)THEN	00122300
	IF(ICOMZN(K, 2).EQ.ICOMZN(LM2, 2))THEN	00122400
	KA=LMT2-(IROADC(J, K, 3)*(TIME-IROADC(J, K, 5))/4)	00122500
	IF(KA.LE.K2)NQC2=NQC2+1	00122600
	ELSE	00122700

NQC2=NQC2+1	00122800
ENDIF	00122900
ENDIF	00123000
ELSEIF(ICOMZN(K, 2).EQ. ICOMZN(LM2, 2)) THEN	00123100
KA=LMT2-(IROADC(J, K, 3)*(TIME-IROADC(J, K, 5))/4)	00123200
IF(KA.LE. K2)NQC2=NQC2+1	00123300
ELSE	00123400
NQC2=NQC2+1	00123500
ENDIF	00123600
ENDIF	00123700
40 CONTINUE	00123800
ENDIF	00123900
C	00124000
RETURN	00124100
END	00124200
	00124300

A. 4. ハイブリッドシミュレーション用プログラム

A. 4. 1. 進入実験用プログラム

A. 4. 1. 1. 交通流シミュレータ用プログラム

```
#define BACKGROUND 1
#define VEHICLE 2
#include <stdio.h>
#include <usercore.h>
#include <sys/types.h>
#include <sys/timeb.h>

static float vehdx[] = {0.0, 4.0, 1.0, -1.0, -4.0};
static float vehdy[] = {3.0, 0.0, -1.5, -1.5, 0.0};
static float zonedx[] = {0.0, 10.0, 0.0, -10.0};
static float zonedy[] = {14.0, 0.0, -14.0, 0.0};
static float lanedx[] = {40.0};
static float lanedy[] = {0.0};

int pixwindd();
struct vwsurf vwsurf = DEFAULT_VWSURF(pixwindd), vwsurf_save;

static long int mill_ini;

main()
{
    FILE *fpa, *fpb, *fopen();
    char *outporta="/dev/ttya", *outportb="/dev/ttyb", str[10];
    float x1, x2, v1, v2, d2;
    int t, dt, pt, s1_in, s2_in, s1_out, s2_out, tc, oc, df, w1_in, w1_out, w2_in, w2_out, l;
    int idemand1, idemand2;
    int ndemand1, ndemand2;

    if ((fpa=fopen(outporta, "w"))==NULL)
        printf("Can't open outputport_A!\n");
    else {
        if ((fpb=fopen(outportb, "w"))==NULL)
            printf("Can't open outputport_B!\n");
        else {
            vwsurf_save=vwsurf;
re_input:
            printf("input speed(km/h) of vehicle-1 ="); scanf("%f", &v1);
            printf("input number of data for upwards of vehicle-1 =");
            scanf("%d", &ndemand1); idemand1=ndemand1+0x30;
            printf("input speed(km/h) of vehicle-2 ="); scanf("%f", &v2);
            printf("input number of data for upwards of vehicle-2 =");
            scanf("%d", &ndemand2); idemand2=ndemand2+0x30;
            printf("input distance(m) from vehicle-1 ="); scanf("%f", &d2);
            printf("select display mode (no-0 yes-1) ="); scanf("%d", &df);
            if(df==1) {
                printf("input difraction of time(ms) ="); scanf("%d", &dt);
            }
        }
    }
}
```

```

initialize_core(BASIC, NOINPUT, TWOD);
initialize_view_surface(&vwsurf, FALSE);
select_view_surface(&vwsurf);
set_viewport_2(0.125, 0.875, 0.125, 0.75);
set_window(0.0, 40.0, 0.0, 20.0);

```

```

initialize_device(KEYBOARD, 1);
set_echo(KEYBOARD, 1, 1);
set_echo_position(KEYBOARD, 1, 0.2, 0.7);
set_echo_surface(KEYBOARD, 1, &vwsurf);

```

re_start:

```

        create_retained_segment(BACKGROUND);
            move_abs_2(0.0, 5.0);
            polyline_rel_2(lanedx, lanedy, 1);
            move_abs_2(0.0, 10.0);
            set_linestyle(DASHED);
            polyline_rel_2(lanedx, lanedy, 1);
            move_abs_2(0.0, 15.0);
            set_linestyle(SOLID);
            polyline_rel_2(lanedx, lanedy, 1);
            move_abs_2(15.0, 3.0);
            polyline_rel_2(zonedx, zonedy, 4);
        close_retained_segment(BACKGROUND);

        sl_in=0;   s2_in=0;
        sl_out=0;  s2_out=0;
        create_retained_segment(VEHICLE);
            move_abs_2(5.0, 11.0);
            polyline_rel_2(vehdx, vehdy, 5);
            move_abs_2(5.0-d2, 6.0);
            polyline_rel_2(vehdx, vehdy, 5);
        close_retained_segment(VEHICLE);

        set_keyboard(1, 10, "Hit space bar to start!", 7);
        l=0;
        while((l==0) || (str[0]!=' '))
            await_keyboard(5000, 1, str, &l);

        if(df==1) {
            t=0;   tc=0;   oc=0;
            while(1) {
                tim_ini();
                x1=10.0+(v1*t/3600.0);
                x2=10.0-d2+(v2*t/3600.0);

                delete_retained_segment(VEHICLE);
                create_retained_segment(VEHICLE);
                    move_abs_2(x1-5.0, 11.0);
                    polyline_rel_2(vehdx, vehdy, 5);
                    move_abs_2(x2-5.0, 6.0);
                    polyline_rel_2(vehdx, vehdy, 5);
                close_retained_segment(VEHICLE);
            }
        }

```

```

        if((s1_in==0) && (x1>15.0)) {
            putc('A', fpa);
            putc(idemand1, fpa);
            putc('Yn', fpa);
            s1_in=1;
        }
        if((s2_in==0) && (x2>15.0)) {
            putc('A', fpb);
            putc(idemand2, fpb);
            putc('Yn', fpb);
            s2_in=1;
        }

        if((s1_out==0) && (x1>25.0)) {
            putc('B', fpa);
            putc('Yn', fpa);
            s1_out=1;
        }
        if((s2_out==0) && (x2>25.0)) {
            putc('B', fpb);
            putc('Yn', fpb);
            s2_out=1;
        }
        if((s1_out==1) && (s2_out==1)) break;

        if((pt=tim_val())<dt) {
            tim_wait(dt-pt);
            pt=dt;
        }
        else    oc=oc+1;
        tc=tc+1;
        t=t+pt;
    }
}
else {
    tim_ini();
    w1_in=5.0/(v1/3600.0);
    w2_in=(5.0+d2)/(v2/3600.0);
    w1_out=15.0/(v1/3600.0);
    w2_out=(15.0+d2)/(v2/3600.0);
    while(1) {
        if((s1_in==0) && (tim_val())>w1_in)) {
            putc('A', fpa);
            putc(idemand1, fpa);
            putc('Yn', fpa);
            s1_in=1;
        }
        if((s2_in==0) && (tim_val())>w2_in)) {
            putc('A', fpb);
            putc(idemand2, fpb);
            putc('Yn', fpb);
            s2_in=1;
        }
        if((s1_out==0) && (tim_val())>w1_out)) {
            putc('B', fpa);

```

```

        putc('Yn', fpa);
        s1_out=1;
    }
    if((s2_out==0) && (tim_val()>w2_out)) {
        putc('B', fpb);
        putc('Yn', fpb);
        s2_out=1;
    }
    if((s1_out==1) && (s2_out==1)) break;
}
if(w1_out>w2_out) {
    x1=25.0;
    x2=10.0-d2+(v2*w1_out/3600.0);
}
else {
    x1=10.0+(v1*w2_out/3600.0);
    x2=25.0;
}
}
delete_retained_segment(VEHICLE);
create_retained_segment(VEHICLE);
    move_abs_2(x1-5.0, 11.0);
    polyline_rel_2(vehdx, vehdy, 5);
    move_abs_2(x2-5.0, 6.0);
    polyline_rel_2(vehdx, vehdy, 5);
close_retained_segment(VEHICLE);

set_keyboard(1, 10,
"Hit r(restart) ,i(reset condition) or q(quit)!", 7);

key_in:

l=0;
while(l==0)
    await_keyboard(5000, 1, str, &l);
switch(str[0]) {
    case 'i':
        deselect_view_surface(&vwsurf);
        terminate_device(KEYBOARD, 1);
        terminate_core();
        vwsurf=vwsurf_save;
        goto re_input;
    case 'r':
        delete_retained_segment(VEHICLE);
        delete_retained_segment(BACKGROUND);
        goto re_start;
    case 'q':
        goto quit;
    default:
        goto key_in;
}

quit:

deselect_view_surface(&vwsurf);
terminate_device(KEYBOARD, 1);
terminate_core();

if(df==1)

```



```

        printf("over count of difraction = %5d/%5d\n", oc, tc);

        fclose(fpb);
    }
    fclose(fpa);
}

tim_wait(w_tim)
    int w_tim;
{
    struct timeb buf;
    long int limtim, nowtim;

    ftime(&buf);
    limtim=buf.millitm+buf.time*1000+w_tim;
    while(1) {
        ftime(&buf);
        nowtim=buf.millitm+buf.time*1000;
        if(nowtim >= limtim) break;
    }
}

tim_ini() {
    struct timeb buf;
    ftime(&buf);
    mill_ini=buf.millitm+buf.time*1000;
}

tim_val() {
    struct timeb buf;
    int s;
    long int t;
    ftime(&buf);
    t=(buf.time*1000+buf.millitm);
    s=t-mill_ini;
    return(s);
}

```

A. 4. 1. 2. 車載機用プログラム

/* VUMAIN1S.C

Main program for VU Made by S.SAKAI */

```

#include <stdio.h>
#include <bios98.h>
struct{
    char zname[2];
    int drrc;
    int ngecom;
    int nrecieved;
    int ntransmitted;
    int vdrdown;

```

```

    int vdrup;
    char drdown[60];
    char drup[60];
    }zone[10];
int ngecom, nrecieved, nupleft, nupdemand, vgedown[20], vgeup[20],
    cycle, zno, recno, recloss[1000][2], recrc[1000], upnew;
char gedown[20][60], geup[20][60], token[4];
extern time_out;
extern int reack;

```

```

void com_init(unsigned char *buf) {
    COM_INIT cint;
    cint.cmmd=1;
    cint.transfer=0x07;
    cint.mode_inf=0x7b;
    cint.cmmd_inf=0x16;
    cint.recv_buf=buf;
    cint.recv_siz=256;
    cint.time_snd=0x10;
    cint.time_rcv=0xff;
    bios98com_init(&cint);
    if(cint.stus != 0)
        printf("init_status=%2x.%n", cint.stus);
}

```

```

unsigned com_recvsize() {
    COM_INFO cinf;
    cinf.cmmd=2;
    bios98com(&cinf);
    return cinf.data_siz;
}

```

```

unsigned com_recv(void) {
    COM_INFO cinf;
    cinf.cmmd=4;
    bios98com(&cinf);
    return cinf.data;
}

```

```

record(rc, loss)
    int rc, loss[];
    {
        recrc[recno]=rc;
        recloss[recno][0]=loss[0];
        recloss[recno][1]=loss[1];
        recno++;
    }

```

```

recprn()
    {
        int i;
        for(i=0; i<recno; i++)
            {
                printf("%02d loss1=%04d ", i, recloss[i][0]);
                printf("loss2=%04d ", recloss[i][1]);
                printf("return code = %02x.%n", recrc[i]);
            }
    }

```

```

    }
}

trrer(trrc)
{
    int trrc;
    {
        unsigned int rc;
        switch(trrc)
        {
            case -1:rc=0x10;break;
            case -2:rc=0x11;break;
            case -3:rc=0x12;break;
            default:rc=0x13;
        }
        return(rc);
    }
}

trdaanal(trpo, trrc)
{
    int trrc;
    unsigned char trpo[];
    {
        unsigned int rc, pano;
        rc=0;
        if((trpo[0]=='t')&&(trpo[1]==token[0]))
        {
            strcpy(trpo, "");
            rc=1;
        }
        if(strncmp(trpo, "ge", 2) == 0)
        {
            rc=2;
        }
        if((trpo[0]=='p')&&(trpo[1]==token[0]))
        {
            rc=3;
        }
        if((trpo[0]=='d')&&(trpo[1]==token[0]))
        {
            pano=trpo[2]-0x30;
            strcpy(gedown[pano], trpo);
            vgedown[pano]=trrc;
            nrecieved=pano;
            ngecom++;
            zone[zno].ngecom++;
            zone[zno].nrecieved++;
            strcpy(trpo, "");
            rc=4;
        }
        if((trpo[0]=='y')&&(trpo[1]==token[0]))
        {
            vgeup[--nupleft]=0x00;
            strcpy(trpo, "");
            rc=5;
        }
        if(trpo[0]=='a')

```

```

        {
            rc=6;
        }
        if(strncmp(trpo,"finish",6)==0)
        {
            rc=7;
        }
        return(rc);
    }

main()
{
    int trrc,ctoken,idemand,itrans=0x30,irecv=0x30,ireack=0x30,iupnew=1;
    unsigned int i,j,rpt,k,l,m,rc,loss[10];
    unsigned char trpo[200],cond[4],condup[30],nrqpa[4],condw[100],ans[10];
    unsigned char buf[300];

    vuinitinfo();
    printf("input token assignment="); ctoken=getchar(); getchar();
    rpt=100;
    LSlopen();
    disable_rx_A();
    com_init(buf);
    printf("Now waiting start signal!\n");
    while(com_recv()!='A');
    while(1) if(com_recvsize())>0 {idemand=com_recv(); break;}
    vuindata(ctoken,idemand,itrans,irecv,ireack,iupnew);
    printf("demand of data for upwards=%d\n",nupdemand);
    enable_rx_A();
    printf("Executing! CTRL+C to Quit\n");

    j=0;
    while((recno<rpt)&&(j<rpt))
    {
        i=0;
        while(i<1)
        {
            if(com_recvsize())>0
                if(com_recv()=='B') {
                    LSfclose();
                    printf("Gone out from a zone!\n");
                    goto gone_out;
                }

            trrc=rx_data_A(trpo);
            trpo[trrc]=0x00;
            if(trrc>=0)
            {
                i++;
            }
            else
            {
                rc=trer(trrc);
                loss[0]=0;
            }
        }
    }
}

```

```

        loss[1]=0;
        record(rc, loss);
    }
}
switch(trdaanal(trpo, trrc))
{
    case 1:start_timer(2458);      /* time out 1ms */
        cycle=0;
        rc=vuspdr_a(loss);
        if(zone[zno].drrc!=0x1f)
        {
            zone[zno].drrc=rc;
        }
        break;
/* case 2:rc=vugedr_a();
    cycle=0;
    zone[zno].drrc=rc;
    break;
    case 3:start_timer(2458);      Entranse zone    */
        cycle=0;                      /* time out 1ms */
        rc=vuupda_a(trpo, loss);
        break;
    case 4:start_timer(2458);      /* time out 1ms */
        cycle=0;
        rc=vudwda_a(loss);
        break;
    case 5:recno--;
        rc=0x9f;
        break;
    case 6:if((cycle==1)&&(zone[zno].drrc!=0x1f)
        &&(zone[zno].zname[0]!=trpo[1]))
        {
            start_timer(2458);      /* time out 1ms */
            zno++;
            rc=midvugedr_a(loss);
            zone[zno].zname[0]=trpo[1];
            zone[zno].drrc=rc;
            cycle=0;
        }
        else
        {
            cycle=1;
            rc=0;
            loss[0]=0;
            loss[1]=0;
        }
        break;
    case 7:j+=rpt;
default:rc=0x30;
    loss[0]=0;
    loss[1]=0;
    start_timer(1259);      /* wait 0.5msec */
    while(!time_out)
    {

```

```

    }
    record(rc, loss);
}

```

```

gone_out:
    LSIclose();
    printf("remaint of data for upwards=%d\n", nupleft);
    printf("remaint of data for downwards=%d\n", nrecieved);
    printf("Hit return key to see detailed results!\n");
    getchar();
    recprn();
    displayinfo();
    savedatavu(ctoken, idemand, (idemand-itrans), irecv);
}

```

A. 4. 2. 追跡実験用プログラム

A. 4. 2. 1. 交通流シミュレータ用プログラム

```

#define BACKGROUND 1
#define VEHICLE 2
#include <stdio.h>
#include <usercore.h>
#include <sys/types.h>
#include <sys/timeb.h>

static float vehdx[] = {0.0, 4.0, 1.0, -1.0, -4.0};
static float vehdy[] = {3.0, 0.0, -1.5, -1.5, 0.0};
static float zonedx[] = {0.0, 10.0, 0.0, -10.0};
static float zonedy[] = {14.0, 0.0, -14.0, 0.0};
static float lanedx[] = {56.0};
static float lanedy[] = {0.0};

int pixwindd();
struct vwsurf vwsurf = DEFAULT_VWSURF(pixwindd), vwsurf_save;

static long int mill_ini;

main()
{
    FILE *fpa, *fpb, *fopen();
    char *outporta="/dev/ttya", *outportb="/dev/ttyb", str[10];
    float x, v;
    int t, dt, pt, s1_in, s2_in, s1_out, s2_out, tc, oc, df, w1_in, w1_out, w2_in, w2_out, l;

    int idemandl, ndemandl;

    if ((fpa=fopen(outporta, "w"))==NULL)
        printf("Can't open output_A!\n");
    else {
        if ((fpb=fopen(outportb, "w"))==NULL)
            printf("Can't open output_B!\n");
        else {
            vwsurf_save=vwsurf;

```

re_input:

```
printf("input speed(km/h) of vehicle ="); scanf("%f",&v);
printf("input number of data from vehicle =");
scanf("%d",&ndemand1);
idend1=ndemand1+0x30;
printf("select display mode (no-0 yes-1) ="); scanf("%d",&df);
if(df==1) {
    printf("input diffraction of time(ms) ="); scanf("%d",&dt);
}
```

```
initialize_core(BASIC, NOINPUT, TWOD);
initialize_view_surface(&vw surf, FALSE);
select_view_surface(&vw surf);
set_viewport_2(0.125, 0.875, 0.125, 0.75);
set_window(0.0, 56.0, 0.0, 20.0);
```

```
initialize_device(KEYBOARD,1);
set_echo(KEYBOARD,1,1);
set_echo_position(KEYBOARD,1,0.2,0.7);
set_echo_surface(KEYBOARD,1,&vw surf);
```

re_start:

```
create_retained_segment(BACKGROUND);
    move_abs_2(0.0,7.5);
    set_linestyle(SOLID);
    polyline_rel_2(lanedx,lanedy,1);
    move_abs_2(0.0,12.5);
    set_linestyle(SOLID);
    polyline_rel_2(lanedx,lanedy,1);
    move_abs_2(10.0,3.0);
    polyline_rel_2(zonedx,zonedy,4);
    move_abs_2(36.0,3.0);
    polyline_rel_2(zonedx,zonedy,4);
close_retained_segment(BACKGROUND);

s1_in=0; s2_in=0;
s1_out=0; s2_out=0;

x=6.0;
create_retained_segment(VEHICLE);
    move_abs_2(x-5.0,8.5);
    polyline_rel_2(vehdx,vehdy,5);
close_retained_segment(VEHICLE);

set_keyboard(1,10,"Hit space bar to start!",7);
l=0;
while((l==0) || (str[0]!=' '))
    await_keyboard(5000,1,str,&l);

if(df==1) {
    t=0; tc=0; oc=0;
    while(1) {
        tim_ini();
        x=6.0+(v*t/3600.0);
```

```

delete_retained_segment(VEHICLE);
create_retained_segment(VEHICLE);
    move_abs_2(x-5.0, 8.5);
    polyline_rel_2(vehdx, vehdy, 5);
close_retained_segment(VEHICLE);

if((s1_in==0) && (x>10.0)) {
    putc('A', fpa);
    putc(idemand1, fpa);
    putc('Yn', fpa);
    s1_in=1;
}
if((s1_in==1) && (s1_out==0) && (x>20.0))

    putc('B', fpa);
    putc('Yn', fpa);
    s1_out=1;
}
if((s1_out==1) && (s2_in==0) && (x>36.0))

    putc('A', fpb);
    putc('Yn', fpb);
    s2_in=1;
}
if((s2_in==1) && (s2_out==0) && (x>46.0))

    putc('B', fpb);
    putc('Yn', fpb);
    s2_out=1;
}
if((x>50.0) && (s2_out==1)) break;

if((pt=tim_val())<dt) {
    tim_wait(dt-pt);
    pt=dt;
}
else    oc=oc+1;
        tc=tc+1;
        t=t+pt;
    }
}
else {
    tim_ini();
    w1_in=4.0/(v/3600.0);
    w1_out=14.0/(v/3600.0);
    w2_in=30.0/(v/3600.0);
    w2_out=40.0/(v/3600.0);
    while(1) {
        if((s1_in==0) && (tim_val())>w1_in)) {
            putc('A', fpa);
            putc(idemand1, fpa);
            putc('Yn', fpa);
            s1_in=1;
        }
        if((s1_in==1) && (s1_out==0) && (tim_val

```



```

()>w1_out)) {

                                putc('B', fpa);
                                putc('Yn', fpa);
                                sl_out=1;
                                }
                                if((sl_out==1) && (s2_in==0) && (tim_val

()>w2_in)) {

                                putc('A', fpb);
                                putc('Yn', fpb);
                                s2_in=1;
                                }
                                if((s2_in==1) && (s2_out==0) && (tim_val

()>w2_out)) {

                                putc('B', fpb);
                                putc('Yn', fpb);
                                s2_out=1;
                                }
                                if((sl_out==1) && (s2_out==1)) break;
                                }
                                x=46.0;
                                }
                                delete_retained_segment(VEHICLE);
                                create_retained_segment(VEHICLE);
                                move_abs_2(x-5.0, 8.5);
                                polyline_rel_2(vehdx, vehdy, 5);
                                close_retained_segment(VEHICLE);

                                set_keyboard(1, 10,
                                "Hit r(restart) ,i(reset condition) or q(quit)!", 7);

key_in:

                                l=0;
                                while(l==0)
                                        await_keyboard(5000, 1, str, &l);
                                switch(str[0]) {
                                        case 'i':
                                                deselect_view_surface(&vwsurf);
                                                terminate_device(KEYBOARD, 1);
                                                terminate_core();
                                                vwsurf=vwsurf_save;
                                                goto re_input;
                                        case 'r':
                                                delete_retained_segment(VEHICLE);
                                                delete_retained_segment(BACKGROUND);
                                                goto re_start;
                                        case 'q':
                                                goto quit;
                                        default:
                                                goto key_in;
                                }

quit:

                                deselect_view_surface(&vwsurf);
                                terminate_device(KEYBOARD, 1);
                                terminate_core();

                                if(df==1)

```

```

        printf("over count of diffracton = %5d/%5d¥n", oc, tc);
        fclose(fpb);
    }
    fclose(fpa);
}

}

tim_wait(w_tim)
    int w_tim;
{
    struct timeb buf;
    long int limtim, nowtim;

    ftime(&buf);
    limtim=buf.millitm+buf.time*1000+w_tim;
    while(1) {
        ftime(&buf);
        nowtim=buf.millitm+buf.time*1000;
        if(nowtim >= limtim) break;
    }
}

tim_ini() {
    struct timeb buf;
    ftime(&buf);
    mill_ini=buf.millitm+buf.time*1000;
}

tim_val() {
    struct timeb buf;
    int s;
    long int t;
    ftime(&buf);
    t=(buf.time*1000+buf.millitm);
    s=t-mill_ini;
    return(s);
}

```

A. 4. 2. 2. 車載機用プログラム

(前ゾーン設置用)

/* VUMAIN2A.C

Main program for VU Made by S.SAKAI */

#include <stdio.h>

#include <bios98.h>

struct{

char zname[2];

int drrc;

int ngecom;

int nrecieved;

int ntransmitted;

```

    int vdrdown;
    int vdrup;
    char drdown[60];
    char drup[60];
    }zone[10];
int ngecom,nrecieved,nupleft,nupdemand,vgedown[20],vgeup[20],
    cycle,zno,recno,recloss[1000][2],recrc[1000],upnew;
char gedown[20][60],geup[20][60],token[4];
extern time_out;
extern int reack;

```

```

void com_init(unsigned char *buf) {
    COM_INIT cint;
    cint.cmmd=1;
    cint.transfer=0x07;
    cint.mode_inf=0x7b;
    cint.cmmd_inf=0x16;
    cint.recv_buf=buf;
    cint.recv_siz=256;
    cint.time_snd=0x10;
    cint.time_rcv=0xff;
    bios98com_init(&cint);
    if(cint.stus != 0)
        printf("init_status=%2x.\n",cint.stus);
}

```

```

unsigned com_recvsize() {
    COM_INFO cinf;
    cinf.cmmd=2;
    bios98com(&cinf);
    return cinf.data_siz;
}

```

```

unsigned com_recv(void) {
    COM_INFO cinf;
    cinf.cmmd=4;
    bios98com(&cinf);
    return cinf.data;
}

```

```

record(rc,loss)
    int rc,loss[];
    {
        recrc[recno]=rc;
        recloss[recno][0]=loss[0];
        recloss[recno][1]=loss[1];
        recno++;
    }

```

```

recprn()
    {
        int i;
        for(i=0;i<recno;i++)
            {
                printf("%02d loss1=%04d ",i,recloss[i][0]);
                printf("loss2=%04d ",recloss[i][1]);
            }
    }

```

```

        printf("return code = %02x\n", recrc[i]);
    }

}

trcr(trrc)
{
    int trrc;
    {
        unsigned int rc;
        switch(trrc)
        {
            case -1:rc=0x10;break;
            case -2:rc=0x11;break;
            case -3:rc=0x12;break;
            default:rc=0x13;
        }
        return(rc);
    }
}

trdaanal(trpo, trrc)
{
    int trrc;
    unsigned char trpo[];
    {
        unsigned int rc, pano;
        rc=0;
        if((trpo[0]=='t')&&(trpo[1]==token[0]))
        {
            strcpy(trpo, "");
            rc=1;
        }
        if(strncmp(trpo, "ge", 2) == 0)
        {
            rc=2;
        }
        if((trpo[0]=='p')&&(trpo[1]==token[0]))
        {
            rc=3;
        }
        if((trpo[0]=='d')&&(trpo[1]==token[0]))
        {
            pano=trpo[2]-0x30;
            strcpy(gedown[pano], trpo);
            vgedown[pano]=trrc;
            nrecieved=pano;
            ngecom++;
            zone[zno].ngecom++;
            zone[zno].nrecieved++;
            strcpy(trpo, "");
            rc=4;
        }
        if((trpo[0]=='y')&&(trpo[1]==token[0]))
        {
            vgeup[--nupleft]=0x00;
            strcpy(trpo, "");
            rc=5;
        }
    }
}

```

```

        if(trpo[0]=='a')
        {
            rc=6;
        }
        if(strncmp(trpo,"finish",6)==0)
        {
            rc=7;
        }
        return(rc);
    }

main()
{
    int trrc,ctoken,idemand,itrans=0x30,irecv=0x30,ireack=0x30,iupnew=1;
    unsigned int i,j,rpt,k,l,m,rc,loss[10];
    unsigned char trpo[200],cond[4],condup[30],nrqpa[4],conddw[100],ans[10];
    unsigned char buf[300],bufb[100];

    vuinitinfo();
    printf("input token assignment ="); ctoken=getchar(); getchar();
    rpt=100;
    LSlopen();
    inst_timer();
    disable_rx_A();
    com_init(buf);
    printf("Now waiting start signal!\n");
    while(com_recv()!='A');
    while(1) if(com_recvsize())>0 {idemand=com_recv(); break;}
    vuindata(ctoken,idemand,itrans,irecv,ireack,iupnew);
    printf("demand of data for upwards=%d\n",nupdemand);
    enable_rx_A();
    printf("Executing! CTRL+C to Quit\n");

    j=0;
    while((recno<rpt)&&(j<rpt))
    {
        i=0;
        while(i<1)
        {

            if(com_recvsize())>0
            if(com_recv()=='B') {
                ireack=ireack+0x30;
                iupnew=iupnew;
                bufb[0]='I';
                bufb[1]=token[0];
                bufb[2]=nupdemand+0x30;
                bufb[3]=nupdemand-nupleft+0x30;
                bufb[4]=nrecieved+0x30;
                bufb[5]=ireack+2*iupnew;
                bufb[6]=0x00;
                if(tx_data_B(6,bufb)<0)
                    printf("tx error in channel B!\n");
                goto gone_out;
            }

        }
    }
}

```

```

trrc=rx_data_A(trpo);
trpo[trrc]=0x00;
if(trrc>=0)
{
    i++;
}
else
{
    rc=trcr(trrc);
    loss[0]=0;
    loss[1]=0;
    record(rc, loss);
}
}
switch(trdaanal(trpo, trrc))
{
    case 1:start_timer(2458);      /* time out lms */
        cycle=0;
        rc=vuspdr_a(loss);
        if(zone[zno].drrc!=0x1f)
        {
            zone[zno].drrc=rc;
        }
        break;
/* case 2:rc=vugedr_a();
    cycle=0;
    zone[zno].drrc=rc;
    break;
    Entranse zone */
    case 3:start_timer(2458);      /* time out lms */
        cycle=0;
        rc=vuupda_a(trpo, loss);
        break;
    case 4:start_timer(2458);      /* time out lms */
        cycle=0;
        rc=vudwda_a(loss);
        break;
    case 5:recno--;
        rc=0x9f;
        break;
    case 6:if((cycle==1)&&(zone[zno].drrc!=0x1f)
        &&(zone[zno].zname[0]!=trpo[1]))
        {
            start_timer(2458);      /* time out lms */
            zno++;
            rc=midvugedr_a(loss);
            zone[zno].zname[0]=trpo[1];
            zone[zno].drrc=rc;
            cycle=0;
        }
        else
        {
            cycle=1;
            rc=0;
            loss[0]=0;

```

```

                                loss[1]=0;
                                }
                                break;
                                case 7:j+=rpt;
                                default:rc=0x30;
                                    loss[0]=0;
                                    loss[1]=0;
                                    start_timer(1259);          /* wait 0.5msec */
                                    while(!time_out)
                                        {
                                        }
                                }
                                record(rc, loss);
                                }
gone_out:
    rest_timer();
    LSIclose();
    printf("Gone out from a zone!\n");
    printf("last data \n");
    printf("remaint of data for upwards=%d\n",nupleft);
    printf("remaint of data for downwards=%d\n",nrecieved);
    printf("\n");

    printf("Hit return key to see detailed results!\n");
    getchar();
    recprn();
    displayinfo();
    savedatavu(ctoken, idemand, (idemand-itrans), irecv);
}

```

(後ゾーン設置用)

/* VUMAIN2B.C

Main program for VU Made by S.SAKAI */

```

#include <stdio.h>
#include <bios98.h>
struct {
    char zname[2];
    int drrc;
    int ngecom;
    int nrecieved;
    int ntransmitted;
    int vdrdown;
    int vdrup;
    char drdown[60];
    char drup[60];
    } zone[10];
int ngecom, nrecieved, nupleft, nupdemand, vgedown[20], vgeup[20],
    cycle, zno, recno, recloss[1000][2], recrc[1000], upnew;
char gedown[20][60], geup[20][60], token[4];
extern time_out;
extern int react;

void com_init(unsigned char *buf) {

```

```

    COM_INIT cint;
    cint.cmmd=1;
    cint.transfer=0x07;
    cint.mode_inf=0x7b;
    cint.cmmd_inf=0x16;
    cint.recv_buf=buf;
    cint.recv_siz=256;
    cint.time_snd=0x10;
    cint.time_rcv=0xff;
    bios98com_init(&cint);
    if(cint.stus != 0)
        printf("init_status=%2x.%Yn",cint.stus);
}

unsigned com_recvsize() {
    COM_INFO cinf;
    cinf.cmmd=2;
    bios98com(&cinf);
    return cinf.data_siz;
}

unsigned com_recv(void) {
    COM_INFO cinf;
    cinf.cmmd=4;
    bios98com(&cinf);
    return cinf.data;
}

record(rc,loss)
    int rc,loss[];
    {
        recrc[recno]=rc;
        recloss[recno][0]=loss[0];
        recloss[recno][1]=loss[1];
        recno++;
    }

recprn()
    {
        int i;
        for(i=0;i<recno;i++)
            {
                printf("%02d loss1=%04d ",i,recloss[i][0]);
                printf("loss2=%04d ",recloss[i][1]);
                printf("return code = %02x%Yn",recrc[i]);
            }
    }

trrer(trrc)
    int trrc;
    {
        unsigned int rc;
        switch(trrc)
            {
                case -1:rc=0x10;break;
                case -2:rc=0x11;break;
            }
    }

```



```

        case -3:rc=0x12;break;
        default:rc=0x13;
    }
    return(rc);
}

```

```

trdaanal(trpo, trrc)
{
    int trrc;
    unsigned char trpo[];
    {
        unsigned int rc, pano;
        rc=0;
        if((trpo[0]=='t')&&(trpo[1]==token[0]))
        {
            strcpy(trpo, "");
            rc=1;
        }
        if(strncmp(trpo, "ge", 2) == 0)
        {
            rc=2;
        }
        if((trpo[0]=='p')&&(trpo[1]==token[0]))
        {
            rc=3;
        }
        if((trpo[0]=='d')&&(trpo[1]==token[0]))
        {
            pano=trpo[2]-0x30;
            strcpy(gedown[pano], trpo);
            vgedown[pano]=trrc;
            nrecieved=pano;
            ngecom++;
            zone[zno].ngecom++;
            zone[zno].nrecieved++;
            strcpy(trpo, "");
            rc=4;
        }
        if((trpo[0]=='y')&&(trpo[1]==token[0]))
        {
            vgeup[--nupleft]=0x00;
            strcpy(trpo, "");
            rc=5;
        }
        if(trpo[0]=='a')
        {
            rc=6;
        }
        if(strncmp(trpo, "finish", 6)==0)
        {
            rc=7;
        }
        return(rc);
    }
}

```

```
main()
```

```

{
int trrc, ctoken, idemand, itrans, irecv, ireack, iupnew;
unsigned int i, j, rpt, k, l, m, rc, loss[10];
unsigned char trpo[200], cond[4], condup[30], nrqpa[4], conddw[100], ans[10];
unsigned char buf[300], bufb[100];

vuinitinfo();
rpt=100;
LSlopen();
inst_timer();
disable_rx_A();
com_init(buf);
printf("Now waiting initial data!\n");
while(1) if((rx_data_B(bufb)>0) && (bufb[0]!='I')) {
    ctoken=bufb[1];
    idemand=bufb[2];
    itrans=bufb[3];
    irecv=bufb[4];
    ireack=(bufb[5] & 0x31);
    iupnew=(bufb[5] & 0x02)/2;
    break;
}
vuindata(ctoken, idemand, itrans, irecv, ireack, iupnew);
printf("initial data\n");
printf("token=%c\n", token[0]);
printf("demand of data for upward=%d\n", nupdemand);
printf("remaint of data for upwards=%d\n", nupleft);
printf("remaint of data for downwards=%d\n", nrecieved);
printf("\n");

printf("Now waiting start signal!\n");
while(com_recv()!='A');
enable_rx_A();
printf("Executing! CTRL+C to Quit\n");

j=0;
while((recno<rpt)&&(j<rpt))
{
    i=0;
    while(i<1)
    {
        if(com_recvsize())>0
            if(com_recv()=='B') {
                goto gone_out;
            }

        trrc=rx_data_A(trpo);
        trpo[trrc]=0x00;
        if(trrc>=0)
        {
            i++;
        }
        else
        {

```

```

        rc=trcr(trrc);
        loss[0]=0;
        loss[1]=0;
        record(rc, loss);
    }
}
switch(trdaanal(trpo, trrc))
{
    case 1:start_timer(2458);    /* time out 1ms */
        cycle=0;
        rc=vuspdr_a(loss);
        if(zone[zno].drrc!=0x1f)
        {
            zone[zno].drrc=rc;
        }
        break;
/* case 2:rc=vugedr_a();
    cycle=0;
    zone[zno].drrc=rc;
    break;
    case 3:start_timer(2458);    Entranse zone    */
    cycle=0;    /* time out 1ms */
    rc=vuupda_a(trpo, loss);
    break;
    case 4:start_timer(2458);    /* time out 1ms */
    cycle=0;
    rc=vudwda_a(loss);
    break;
    case 5:recno--;
    rc=0x9f;
    break;
    case 6:if((cycle==1)&&(zone[zno].drrc!=0x1f)
        &&(zone[zno].zname[0]!=trpo[1]))
        {
            start_timer(2458);    /* time out 1ms */
            zno++;
            rc=midvugedr_a(loss);
            zone[zno].zname[0]=trpo[1];
            zone[zno].drrc=rc;
            cycle=0;
        }
        else
        {
            cycle=1;
            rc=0;
            loss[0]=0;
            loss[1]=0;
        }
        break;
    case 7:j+=rpt;
default:rc=0x30;
    loss[0]=0;
    loss[1]=0;
    start_timer(1259);    /* wait 0.5msec */
    while(!time_out)

```

```

        {
        }
    }
    record(rc, loss);
}
gone_out:
    rest_timer();
    LSfclose();
    printf("Gone out from a zone!\n");
    printf("last data\n");
    printf("demand of data for upward=%d\n", nupdemand);
    printf("remaint of data for upwards=%d\n", nupleft);
    printf("remaint of data for downwards=%d\n", nrecieved);
    printf("\n");
    printf("Hit return key to see detailed results!\n");
    getchar();

    recprn();
    displayinfo();
    savedatavu(ctoken, idemand, (idemand-itrans), irecv);
}

```