

**A Study of Free Ranging Automated Guided Vehicle
Systems**

By

Anirudha Sen

**A Thesis Submitted for the Degree of
Doctor of Philosophy
of the University of London
and
The Diploma of Membership of Imperial College**

May 1990

**Department of Mechanical Engineering
Imperial College of Science, Technology and Medicine
London, United Kingdom.**

A Study of Free Ranging Automated Guided Vehicle Systems

Abstract

Free Ranging Automated Guided Vehicles are Automated Guided Vehicles (AGVs) that are not confined to physical tracks on the floor. In this thesis we examine the problems involved in implementing such a system in a structured environment. A map of the environment is downloaded into the vehicle and the vehicle is instructed to go from one point on the map to another. The most appropriate route is selected to reach the destination. No deviation from the selected route is allowed. There could be a number of vehicles in this environment, so the problems associated with fleet operation is covered. Much work has been done in simulating such a scenario. However in a simulation all parameters are known. The only way of appreciating real world problems is to make a test bed to try out the theories and hypotheses made.

The first Imperial College Automated Guided Vehicle (ICAGV-I) exhibited a number of shortcomings and was not suitable for real world applications. The present research (ICAGV-II) takes advantage of advances in technology and is a total redesign of the previous system. A top down approach has been used to ensure there is complete compatibility between the "Supervisory System" and the AGVs. The issues covered range from the selection of a supervisory system and communication system, to issues pertaining to the vehicle itself, such as guidance, motion control and obstacle detection.

The vehicle uses dead reckoning (odometry) to follow a path on a map. It uses a vision system to locate markers and compares observed marker coordinate positions with its actual coordinates on the map. Ultrasonic sensors are used to detect any obstacle in the path of the vehicle and so avoid them.

CONTENTS

Chapter 1 : Introduction - Automated Guided Vehicles	1
1.1 Automated Guided Vehicles and Their Importance to Manufacturing	1
1.2 Existing AGV Systems	3
1.3 Free Ranging Automated Guided Vehicle Systems	6
1.4 The Imperial College Automated Guided Vehicle System	7
1.4.1 ICAGV-I	7
1.4.2 Present Research (ICAGV-II)	8
1.5 Outline of Thesis	9
Chapter 2 : Supervisory System	12
2.1 Introduction	12
2.2 Inputs to Supervisory System	14
2.2.1 Transportation Request	14
2.2.2 Environment Database	15
2.2.3 Status Information	21
2.3 Scheduler	22
2.3.1 Vehicle Selection	22
2.3.2 Route Generator	22
2.3.3 Time Tabling	23
2.4 Fleet Performance and Logging	25
2.5 Simulation and Tracking	26
2.6 Conclusion	27
Chapter 3 : Communication System	29
3.1 Introduction	29
3.2 Computer Communication	31
3.2.1 Communication Protocols	31

3.2.2 Layered Communication and the OSI Reference Model	32
3.3 Communication in Industrial and Technical Environments	36
3.3.1 The MAP Standard	36
3.3.2 The TOP Standard	36
3.3.3 Interconnection between Networks	37
3.4 Communication with AGV	38
3.5 Implementation	41
3.5.1 Hardware	42
 Chapter 4 : Guidance	 44
4.1 Introduction	44
4.2 Dead Reckoning	46
4.2.1 Differential Drive	47
4.2.2 Steered Wheel Tricycle Cart	48
4.2.3 Steered Wheel Differential Drive - FLEXL	49
4.2.4 Odometric Errors	51
4.3 Inertial Guidance	52
4.4 Imaging Systems	53
4.4.1 One Dimensional Imaging	54
4.4.2 Two Dimensional Imaging	55
4.4.2.1 Offboard	55
4.4.2.2 Onboard	56
4.4.3 3-D Imaging of the Surroundings	57
4.5 Range Imaging	58
4.5.1 Laser Ranging	59
4.5.2 Ultrasonic Ranging Systems	60
4.6 Summary of Guidance Techniques	61
4.7 ICAGV Guidance System	62
 Chapter 5 : Vision System	 64
5.1 Introduction	64

5.2 Low Level Image Processing	68
5.2.1 Illumination	68
5.2.2 Surface Reflectance	69
5.3 Calibration	71
5.4 Conversion from Image to Vehicle Coordinates	73
5.5 Guidance using the Vision System	75
5.5.1 Datuming or Docking	75
5.5.2 Pathway Tracking	76
5.5.3 Update Odometry	77
5.6 Obstacle Detection	78
5.7 Implementation	79
 Chapter 6 : Obstacle Detection / Avoidance	 81
6.1 Introduction	81
6.2 Ultrasonic Ranging System	83
6.2.1 Limitations of Ultrasonic Ranging Systems	83
6.2.2 Polaroid Ultrasonic Ranging System	87
6.3 Implementation	87
6.3.1 Hardware	89
 Chapter 7 : Motion Control	 91
7.1 Introduction	91
7.2 Path Specification	92
7.2.1 Path Specification by Exit Point	92
7.2.2 Coordinate Transformation	94
7.2.3 Problems with path succession	95
7.2.4 ICAGV path succession	97
7.2.4.1 Turning around Centre	97
7.2.4.2 Straight Line Following	98
7.3 Control Strategies	99

7.3.1 ICAGV-I Control Strategy	99
7.3.2 State Feedback Control System	100
7.3.3 ICAGV - II (Present) Control System	101
7.4 Servo Control System Design	102
7.4.1 Gain Adjustment and Compensation	104
7.4.2 Motion Control Card	105
 Chapter 8 : System Integration - Implementation of the Imperial College Automated Guided Vehicle System	 107
8.1 Introduction	107
8.2 Hardware	109
8.3 ICAGV - II Control	110
8.3.1 Selection of Computer System	110
8.3.2 Block Diagram of ICAGV-II Hardware	111
8.3.3 Sensor Monitor Card	112
8.3.4 Speech Recognition Unit	113
8.4 Software	115
8.4.1 Ease of Implementation	115
8.4.2 Functional Requirements	116
8.4.3 Command Structure	118
8.5 Program Structure	121
8.5.1 Initialisation	121
8.5.2 Task Control	122
8.5.3 Command Processor	123
8.6 Example of Program	125
 Chapter 9 : Discussion of Vehicle System Performance	 128
9.1 Introduction	128
9.2 Odometry	132
9.2.1 Accuracy of Odometry	132
9.3 Vision System	133

9.4 Obstacle Detection	137
9.5 Conclusion	137
Chapter 10 : Free Ranging Automated Guided Vehicles (FRVs) and Autonomous Land Vehicles (ALVs)	139
10.1 Introduction	139
10.2 Autonomous Vehicles	140
10.3 Summary of Architecture for Mobile Robots	142
10.4 The Imperial College Free Ranging AGV	143
Chapter 11 : Conclusions and Future Work	144
11.1 Conclusions	144
11.2 Future Work	146
11.2.1 Work in Progress	147
11.2.2 Suggestions for Future Work	148
Appendix A1 : PACSLINK Radio Data Network	150
A1.1 Introduction	150
A1.2 Message Handling Protocol	150
A1.3 Message from user device to PACSLINK unit	151
A1.4 Message from the PACSLINK unit to User device	152
A1.5 Acknowledgement of reception at Remote PACSLINK unit	152
Appendix A2 : HCTL-1000 Motion Control IC Specification	154
A2.1 Introduction	154
A2.2 Theory of Operation	154
A2.3 User Accessible Registers	155

A2.4 Operating Modes	157
Appendix A3 : Image Processing Card	159
A3.1 Introduction	159
A3.2 Set up and Usage	160
A3.3 Table Scaling	161
A3.4 Video Buffer	161
A3.5 Auxiliary Buffers	162
A3.6 Memory to Screen Mapping	162
A3.7 VSC Modes of Operation	163
A3.8 Mapping Information	165
References	169

CHAPTER 1

INTRODUCTION - AUTOMATED GUIDED VEHICLES

1.1 Automated Guided Vehicles (AGVs) and Their Importance to Manufacturing

In the present manufacturing climate, with global markets bringing with it global competition it is recognizably beneficial to increase the responsiveness of the manufacturing system to prevailing market forces. To ensure fast turnover of products, total flexibility in volumes, variants and process with zero defect has become the goal. This is possible only if the response time of the system is minimised by following concepts such as Just-In-Time (JIT). The JIT philosophy requires that parts are available just when needed and the material flow time from raw material to finished goods, preferably, waiting at the customers warehouse, is minimised. Inertia in the system, such as large inventories, delays in material handling and other bottlenecks in the system must be identified fast and corrective action taken.

A key component of the Just-In-Time concept is the material handling system. The material handling system has to make sure that materials are available to the manufacturing cells or assembly cells when they are needed and the subassemblies taken away to the next process, when completed. Figure 1.1 shows the importance of material flow to any manufacturing activity.

J.A.White [8] classifies material handling into five categories : manual (1950's), mechanized (1960's), automated (1970's), integrated (1980's) and intelligent (?-1990's). Automating production lines, by connecting islands of automation by fixed conveyors, is well established. However these systems are not flexible, they are suitable for large volume productions. To meet the global challenge of flexibility, the system should be able to handle small batches (ideally a batch of one), to large batches, depending on customer orders. J.Lenoir [7] reiterates the JIT philosophy that flexibility is achievable by introducing concepts such as assembly in cells, "kitting" of materials, automating storage and handling, absolute physical parts control and a pull system where the assembly cell or manufacturing cell triggers the supply of parts (kits) according to real demand. A material handling system that ideally suits this concept is the Automated Guided Vehicle System.

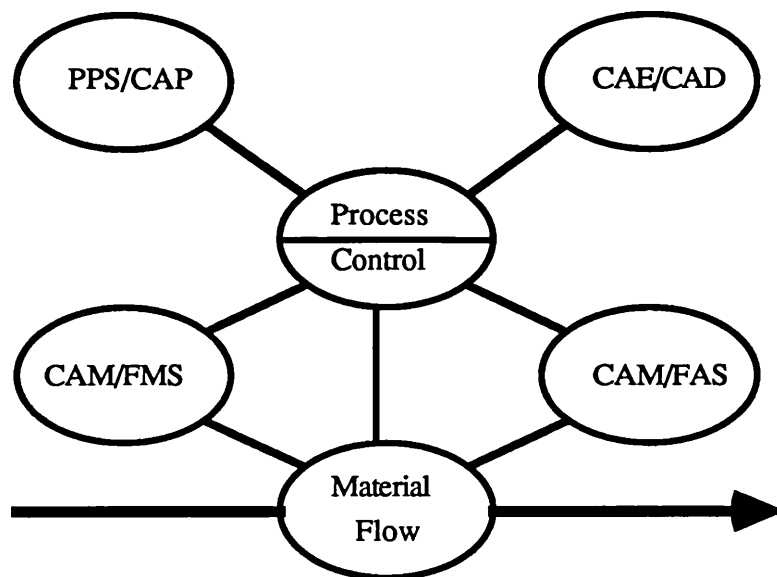


Figure 1.1 : Material Flow in a computer integrated manufacturing environment [6]

Automated Guided Vehicles (AGVs) have come a long way since their introduction in the 1950s. Taking advantage of technological developments, they have evolved from simple "fetch and carry" AGVs to "mobile work station" AGVs and to the present "intelligent" AGVs with microprocessors controlling various functions. This makes the Automated Guided Vehicle System (AGVS) a flexible transport system suitable for incorporation into Flexible Assembly Systems (FAS), Flexible Manufacturing Systems (FMS), Automatic Storage and Retrieval Systems (AS/AR) and Computer Integrated Manufacturing (CIM) Systems. In existing AGV installations, advantages such as space saving, due to better transport route planning, inventory reduction due to disciplines imposed by the system for better material resource planning, and productivity improvements have been reported. The AGV system is a well programmed material handling system that delivers the "right amount of the right materials at the right place, at the right time, in the right sequence, in the right condition, in the right orientation" [8] (and at the right cost).

1.2 Existing AGV Systems

There are basically six types of Automated Guided Vehicles. They are tow vehicle, unit load vehicles, pallet trucks, fork trucks, light load vehicles and assembly line vehicles.

Most of the installed AGV systems are wire guided systems optical track following systems in clean areas [5]. In the wire guided method, a groove is cut into the floor and a track in the form of a wire carrying alternating current is embedded into the floor. The magnetic field resulting from the current flow is picked up by inductive coils on the vehicle. This method has proven to be very robust and it is not dependent on the condition of the floor. In the optical track following system, reflective tape or high contrast paint is laid/painted on the floor. The optical method cannot be used in environments where there is chance of the track being obscured by dirt or other substances. Usually all the routes in a particular network is stored in a computer. When a journey request is received the computer checks its "route table" and downloads a conflict free route to the AGV.

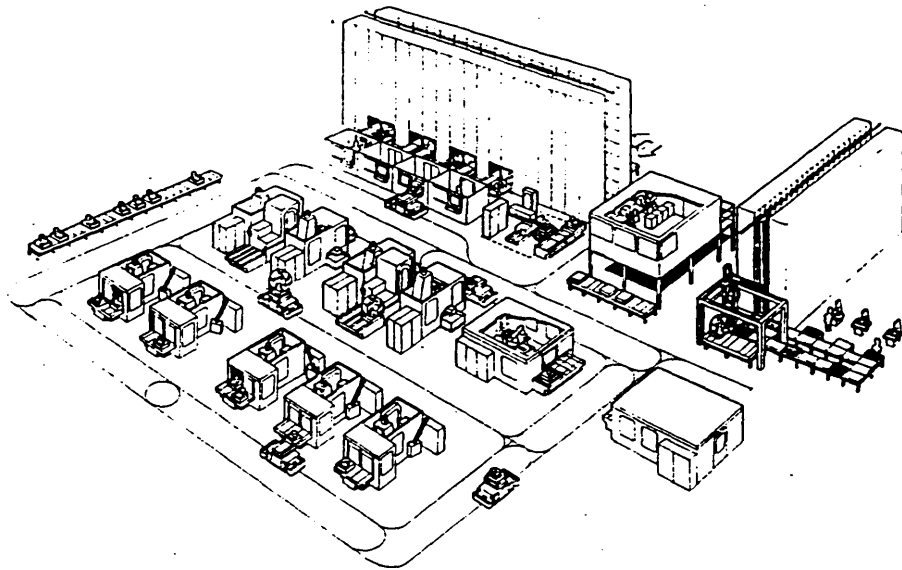
Figure 1.2 shows two AGV systems - (a) in a flexible manufacturing cell and (b) in a flexible assembly environment

Table 1.1 gives a historical evolution of AGV Control of one of the major AGV manufacturers - Wagner Indumat.

H.Challis [10] stated in 1986, that " what developments in AGVs over the past decade or so has done is to disclose potential markets - and these apparently are enormous". As if to confirm this, towards the end of 1988, T. Takahashi [9] reviewing the use of AGVs in Japan, mentions that in addition to AGV applications in the manufacturing industries, they are also being used in clean rooms and service industries including offices, hospitals, hotels, supermarkets and golf links. It is also being used as a platform for the construction industry.

In a strict sense, most existing AGV systems are not flexible transport systems. Considerable effort and cost are required to repave floor surfaces and relayout routes whenever a modification is needed. A truly flexible system would be one whose paths are software programmable and do not require any physical tracks or wire to be laid on the ground - A Free Ranging Automated Guided Vehicle System.

A.



B.

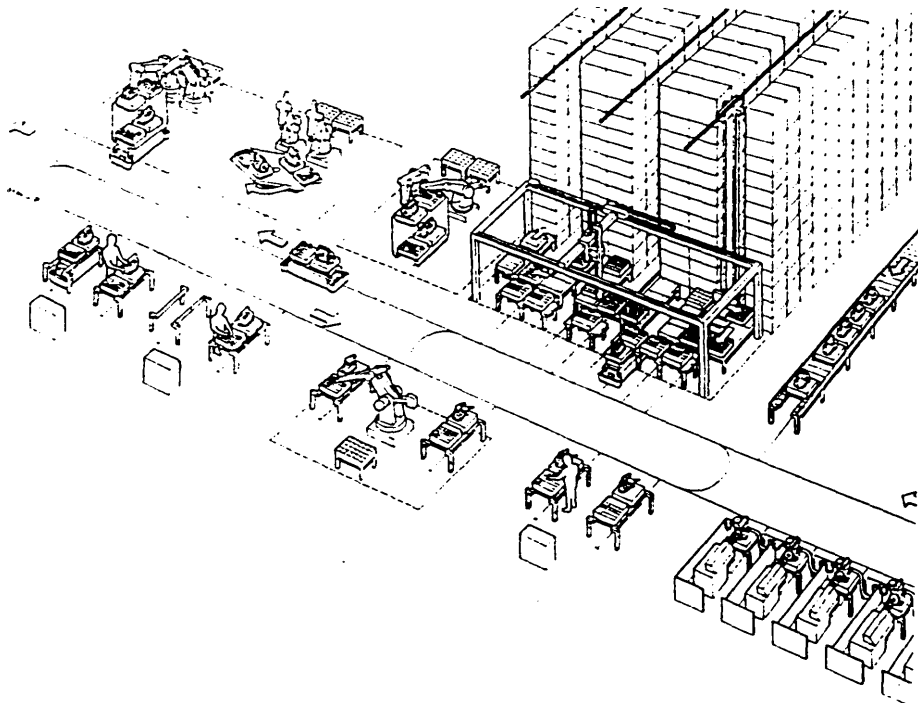


Figure 1.2 : AGVs operating within A. Flexible Manufacturing System (FMS) and B. Flexible Assembly System (FAS) [6]

- 1965 - Automatic steering vehicle with optical scanning of a floor guide line
- 1967 - First vehicle with inductive control (electronic) and integrated destination control (relay technology)
- 1981 - Simplified floor wiring with data transmission by series connected inductive loop
- 1983 - Self monitoring and automatic error diagnosis control components
- 1984 - Seven microprocessors working in parallel in an automatic narrow aisle stacker
- 1985 - Vehicles taught moves manually without guidepath
- *1987 - Caterpillar Self Guided Vehicle : Free ranging AGV based odometry and laser triangulation using retro-reflective bar charts in the surroundings

Table 1.1 : Historical evolution of AGV Control

*Caterpillar GEC Self Guided Vehicle - the only commercial free ranging AGV

1.3 Free Ranging Automated Guided Vehicle Systems

The next step in the evolution of the Automated Guided Vehicle System is the Free Ranging Automated Guided Vehicle or Free Ranging Vehicle (FRV) System [11]. From Table 1.1 we see that, since 1985, some AGVs are able to leave the track or guide and travel short distances using odometry. Odometry is the method of measuring distance by keeping track of wheel rotation. For longer distances, cumulative errors must be corrected. Extensive work is underway in universities and research institutions on sensor guided vehicles that could dispense with the physical tracks altogether.

The only free ranging AGV system in commercial operation is the Caterpillar Self Guided Vehicle System (SGVS) . This is a joint project undertaken by Caterpillar Industrial Inc. of USA and General Electric Corporation (GEC) of UK. The manufacturer's description of the system follows [12]:

"The routes of the SGVS are stored by computers, both on the SGVs themselves and on a landbased (fixed) computer system (LCS).The LCS is a DEC MicroVax II computer and includes software to control vehicle routing, traffic contention and zone policing. The LCS and SGVs carry out two way communication over a UHF/FM radio link. The SGVs use odometry as its main guidance and triangulation based on a low powered rotating laser, which scans bar codes strategically mounted in the facility. After installation the bar coded targets are accurately surveyed using a dumpy-level or theodolite and their calculated x, y coordinates are input to the LCS via the menu driven software package. The positions of these passive reference targets are therefore known very accurately by the SGVs and hence the vehicles can calculate and update their position as they move" (Figure 1.3).

"The angle of the reference targets in relation to the SGVs are measured using a bore-sight laser scanner mounted on top of the vehicle. This scanner rotates twice a second and measures the angles of, and decodes all the reference targets it "sees" within a 15 meters radius. The targets themselves are differentiated by bar codes made up of retro-reflective strips. Physically the targets consist of a metal or plastic plate 600mm x 200mm onto which self adhesive retro-reflective strips are applied to make up the targets reference code. The targets are mounted at a height of approximately 2 meters".

There are 5 onboard computers and programmable controllers to carry out the various navigational and communication tasks.

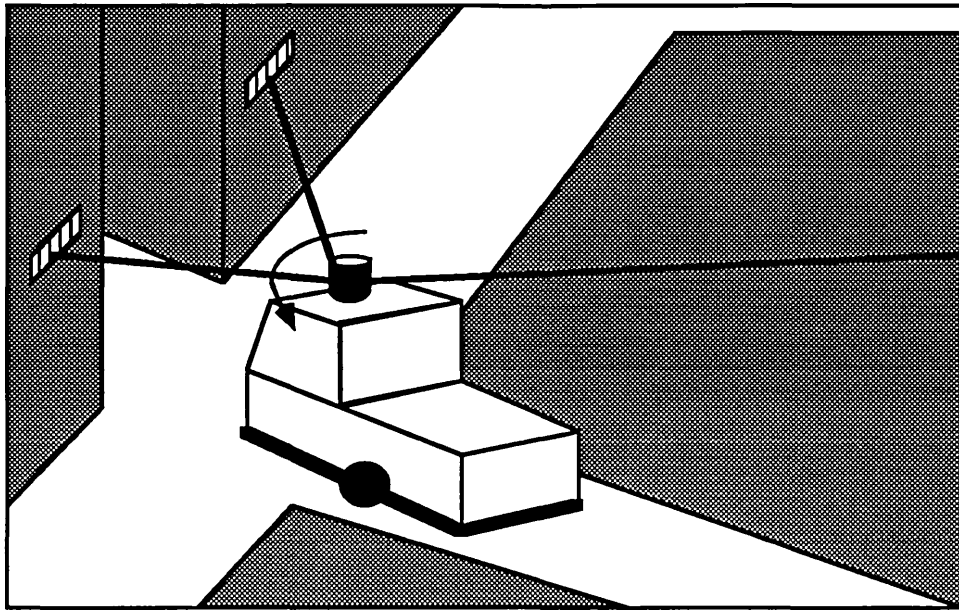


Figure 1.3 - The Caterpillar GEC Self Guided Vehicle (SGV)

1.4 The Imperial College Automated Guided Vehicle System

1.4.1 ICAGV-I

The first Imperial College AGV (ICAGV-I) was completed in 1985. It used odometry (keeping track of wheel rotations) for primary guidance and a laser triangulation system for updating odometric readings. The following observations were made from the ICAGV-I experiments :

1. Odometry yielded a lateral positional error of 10cm and a longitudinal error of 1cm for a straight 16m length path. Thus, slight heading error can cause a large lateral error. (0.1 degree error causes 2.5 cm lateral error for a 16 m path).
2. Rotating laser system based on an active position reference beacon was found to be dependent on the angular position relative to the beacon.

It was concluded that the system needed considerable improvement in its primary guidance (odometry) and secondary guidance (laser triangular) to be practically viable as a free ranging vehicle.

1.4.2 Present Research (ICAGV-II)

The emphasis of AGV research at Imperial College is for a completely "integrated" Free Ranging Automated Guided Vehicle System - a system that is flexible and could be incorporated into any manufacturing establishment, ranging from large fleet sizes to small manufacturing cells comprising of only one vehicle.

The system is divided into a "supervisory system" and "the vehicle fleet". The system must satisfy the following conditions :

1. Be easy to install and not require drastic changes in the facility or environment,
2. Operate amongst human beings and expensive machinery and
3. Handle unexpected events in a fail safe manner.

A top down approach is used in this thesis. The input to the AGV system is a transportation request from the production control system. The request is processed by the supervisory system which sends a conflict free route to the communication system which then sends it to the vehicle. A map of the environment resides in the supervisory system as well as in the vehicle. Using this approach, desirable features of the system can be arrived at and a realistic specification of the system can be developed to achieve a Free Ranging Automated Guided Vehicle System to meet present industrial needs.

Our vehicle (ICAGV-II) is a complete redesign of the previous system. To keep track of the distance moved by the vehicle, odometry is used. By improving the motion control system of the vehicle, the errors due to odometry has been reduced. This means that the vehicle can travel longer distances, before requiring an update from the environment. A vision system is used as the real world sensor for correcting any drifts in the motion of the vehicle in the world coordinate frame. The vision system becomes the main guidance system in areas where precision alignment is required such as during "daturning" or "docking". Ultrasonic sensors are used to detect obstacles in the path of the vehicle.

The main obstacle in the path of implementing any free ranging vehicle system is the general understandable public fear of the vehicle deviating from its intended route. It will be shown that our system with a combination of environment map, odometry and vision is robust enough to ensure that the specified route is followed.

1.5 Outline of Thesis

The thesis begins with the supervisory system (chapter 2) with its various options. The tasks that the supervisory system has to perform dictates to a certain extent the amount of information that will have to be sent to the vehicle. The task has to be communicated to the vehicle. Communication (Chapter 3) involves issues such as compatibility, protocol and regulations. Based on information from the Supervisory system, via the communication network, the AGV has to fulfil the transportation requests, reliably and safely.

This brings us to the vehicle and the guidance techniques (Chapter 4) available to the free ranging Automated Guided Vehicle. It is becoming apparent that quite a number of sensor guidance options are downgraded versions of vision systems and this is expanded on in the next chapter (Chapter 5). The implementation of the vision system which forms the real world sensor for the vehicle is included. The resolution obtainable from ultrasonic sensors is not as good as vision systems, however, the system is inexpensive and is suitable for obstacle detection. Obstacle detection and avoidance is described in the next chapter (Chapter 6). From planning and perception we finally come to acting - motion control (Chapter 7). The motion control task encompasses the basic command structure which transforms the information received from the supervisory system, to primitives understood by the motion control card and directs the vehicle to its goal using dead reckoning to follow the software map, vision to ensure that the software route and the actual route agree and ultrasonic sensors to detect any obstacle in its path. The hardware and software integration of the whole vehicle is now possible (Chapter 8).

The vehicle completed, we proceed to document the results of experimental trials (Chapter-9). The problems of free ranging AGVs have a lot in common with the work going on in the field of Autonomous Land Vehicles (ALV) (Chapter 10). The difference is that for an AGV the assumption is that the environment is known and that it is possible to store the map of the environment in the AGV. Autonomous Land Vehicles work in uncertain and unknown dynamic environments and must be able to create its own map. Our approach of proceeding from a known structured environment to an unknown unstructured environment appears to be a logical one. For a free ranging Automated Guided Vehicle to be viable the system has to cater to the whole gamut of industrial users ranging from small FMS to complex plants producing a range of products. The basic vehicle must be relatively inexpensive. In the conclusion (Chapter-11) we reiterate the objectives of our project and discuss the degree to which the objectives have been met. Some of the ongoing

work on the Imperial College Free Ranging Vehicle project are mentioned.

Figure 1.4 shows the basic structure of the thesis - the planning or supervisory stage first, the perception stage via the communication stage next and finally the action stage.

The field of free ranging AGVs or mobile robots is still in its infancy and as such much work is in progress to make the vehicle suitable for more varied real world situations.

The approach of considering the AGV as an integrated system involving supervisory functions, communication functions and actual vehicle hardware and software makes this an extremely interesting system, unearthing not only technical problems but also problems involved in managing even moderately complex integrated systems.

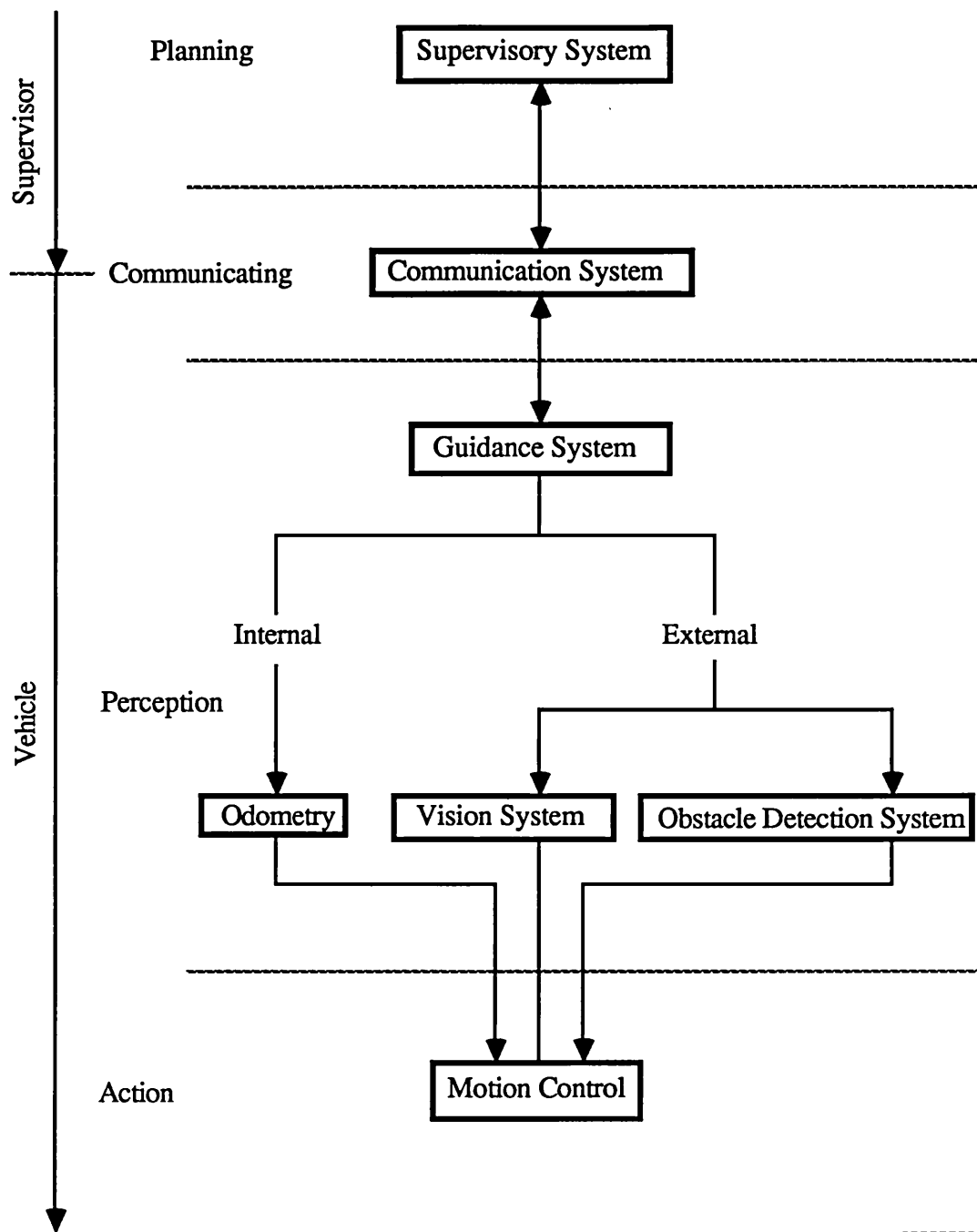


Figure 1.4 : Structure of Thesis

CHAPTER 2

SUPERVISORY SYSTEM

2.1 Introduction

The supervisory system can be visualised as the centralised "Traffic Controller" for the whole AGV fleet. It is the intermediary between the AGV fleet and the other activities taking place in the plant. The supervisory system receives transportation requests from sources such as the production controller, work centres or the vehicle itself. It schedules the request, ensuring that there is no conflict with other requests and downloads the route and time orders to the vehicles. The supervisory system monitors the performance of the fleet based on updates it receives from the environment (docking stations, sensors at intersections etc.) and information from the fleet. Any divergence from the planned or scheduled route is reported and appropriate action taken by rescheduling the effected vehicles [13], [16], [17], [15].

The necessity for a centralised supervisory system would depend on the size of the fleet. T.Takahashi [9] reported that in 1988, the average number of AGVs installed per system in Japan was 2.2 and in the United States, it was about 3.9. For small numbers such as these, the supervisory software could reside in the vehicle (if the chance of interacting with other vehicles is low) or it could reside in the Production Control Computer. For larger fleets where chance of collision is higher, a separate supervisory system computer is needed. Figure 2.1 shows the three possible scenarios.

For a free ranging vehicle fleet the supervisory system plays a very important role. The supervisory system is the only component that has complete information of all the vehicles in the fleet. The flexibility gained in scheduling and ease of route modification for free ranging systems implies that tighter controls are necessary to avoid conflicts with other vehicles and other systems in the environment.

This chapter describes the requirements of a supervisory system for a free ranging AGV fleet. A top down approach is followed, beginning with a transportation request from an external source and ending with a set of commands for the chosen AGV. These commands give an indication of the features necessary for the free ranging vehicle.

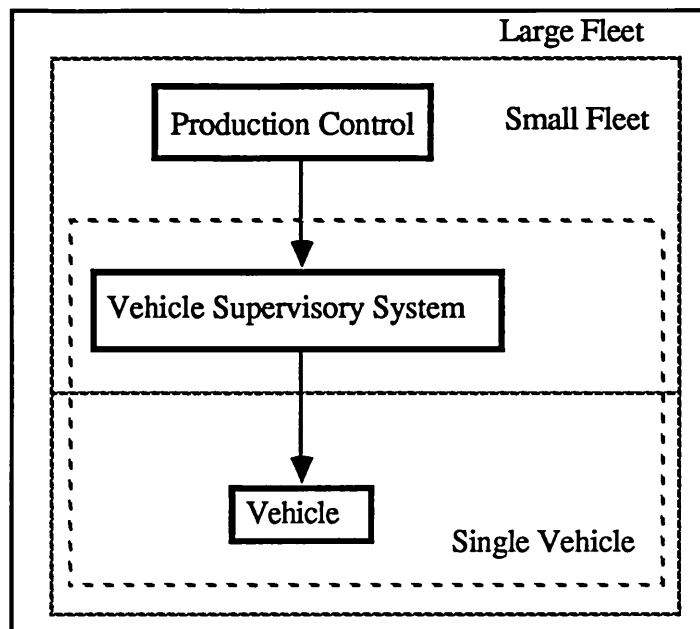


Figure 2.1 : a. Large Fleet - separate production control, separate supervisory system and separate vehicle
 b. Small Fleet - supervisory system combined in production control, separate vehicle
 c. Single Vehicle - supervisory system incorporated in vehicle

2.2 Inputs to Supervisory System

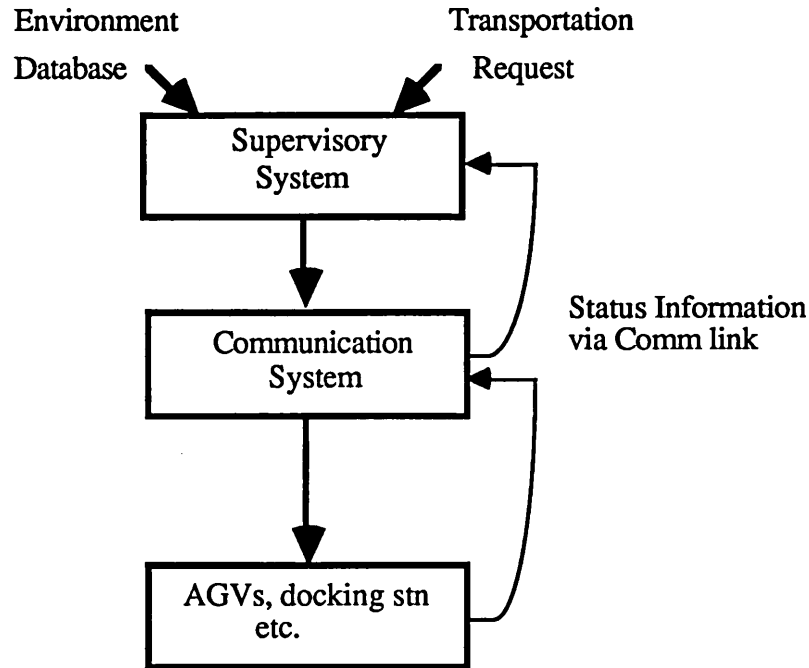


Figure 2.2 : Information flow for the AGV System

To perform its task the supervisory system needs to have the following information (illustrated in Figure 2.2) :

1. Input for journeys to be performed - the Transportation Request,
2. Knowledge of the work place - the Environment Database, and
3. Inputs from vehicles and other systems, to monitor (and correct) system behaviour - Status Information

2.2.1 Transportation Request

Transportation requests could be generated from various sources. In a factory environment, production targets are set by the "production controller". Based on the Material Resources Planning (MRPII) package , the production controller knows the requirement for materials at different stations at different stages of the manufacturing process. It is possible to

generate a Materials Transport Request (MTR) for the AGV based on this information.

A typical MTR could be of the form indicated below [15]:

Order Number	01
Priority	02
AGV Type	01
Start Node	22
Start Time	00:34:45
Task Code	01
Task Time	00:01:00
End Node	34
Arrival Time	00:36:00
Task Code	02
Task Time	00:05:00

The MTR could also be generated by a work centre (FAS or FMS) requiring material. This could be the case in centres where requirements for the day / week is not known ahead of schedule. This could also be the case for unplanned requirements, such as tool breakdown or other unforeseen events. We could envisage this situation (work centre initiated request) to be the intermediate stage towards an integrated manufacturing approach.

The transportation request may be generated by the vehicle itself. Again this request is generated only when the vehicle is unable to comply with the "planned" orders it has been given. For example, if there is an obstacle in its path or battery condition does not allow the vehicle to fulfil commands it has been given.

2.2.2 Environment Database

To undertake the "transportation request" the supervisory system has to have information of the position of all node points (ie. docking station, intersections etc.). The supervisory system must also have all the navigational information required by the vehicle, such as absolute guidance markers or beacons, line following sections etc. In addition, as the free ranging AGV is not constrained by any physical path, the supervisory system needs to

know the location of machinery and areas where the vehicle must not impinge. In effect, the supervisory system must have a complete layout of the "Floor Area" of the factory.

Rather than creating yet another drafting package, a standard Computer Aided Drafting (CAD) package is used. Most CAD packages have layer provision which allows the creation of layers of information to be put one over the other. The layers can be switched 'on' or 'off'. In Figure 2.3, the layout of an actual plant is given with the 'Machine Outline Layer' switched on. The layer information of the plant is indicated below.

Layer Information

<u>Name</u>	<u>Colour</u>	<u>Purpose</u>
Drawings	Blue	Convey drawing format information
Walls	Cyan	Display Shop boundaries
Cells	Yellow	Display machine cell boundaries
Outline	White	Display machine outlines
Areas	Green	Indicate AGV 'no-go' areas
Network	Magenta	Define the node network
Path	Red	Indicate designated AGV paths
Numbers	Yellow	Label Docking Stations
Cranes	Blue	Show position of overhead cranes

The industry standard drawing package has traditionally been AutoCAD from Autodesk. This popular drafting package is used. The layout of the factory is created with as much information as possible. This could be used for various other applications. The relevant information from the AutoCAD drawing is extracted into a DXF output file (Data Exchange Format). DXF output presents drawing entities - line, circle, text, trace, etc., in a simple defined format. The task of transferring the drawing file into the supervisory system is greatly simplified. IGES (Initial Graphics Exchange Specification) the international standard for graphics information exchange would have been a better choice, however, at the time of implementation the IGES translator for AutoCAD was not available.

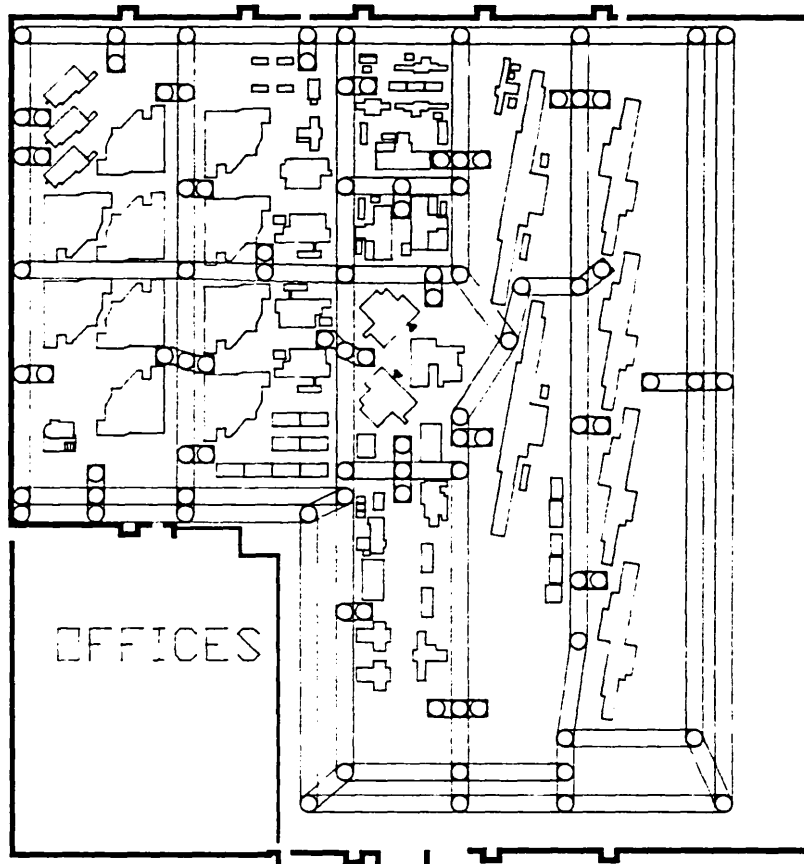


Figure 2.3 : Plot of No1Shop (Rolls Royce) with Machine Outline layer switched "on".

For a minimal system only node information is necessary. This could be in the form of x,y coordinates.

Minimum Information

matrix NODES of order $n \times 2$

where n =number of nodes

$N(i,1)$ =x position of node

$N(i,2)$ =y position of node

matrix LINK of order $n \times n$

where $L(i,j)$ = direct distance between nodes i and j

$L(j,i)$ = route availability (1=available, 0=not available)

$L(i,i)$ =zero

$L(i,j)=L(j,i)$ = infinity (or an alphabet eg. 'X') if no direct link exists

Figure 2.4 shows a plot of No1shop with only the route layout layer switched "on".

In slightly larger systems, go-no-go areas could be specified allowing modifications in the routes if required.

Figure 2.5 shows a plot of No1shop with the "go-no-go" layer "on".

By using any standard CAD package as the input, the environment database can be as large or small as the limits of the CAD package allows. If DXF or IGES translators are available for these packages, relevant information could be extracted from these packages as well.

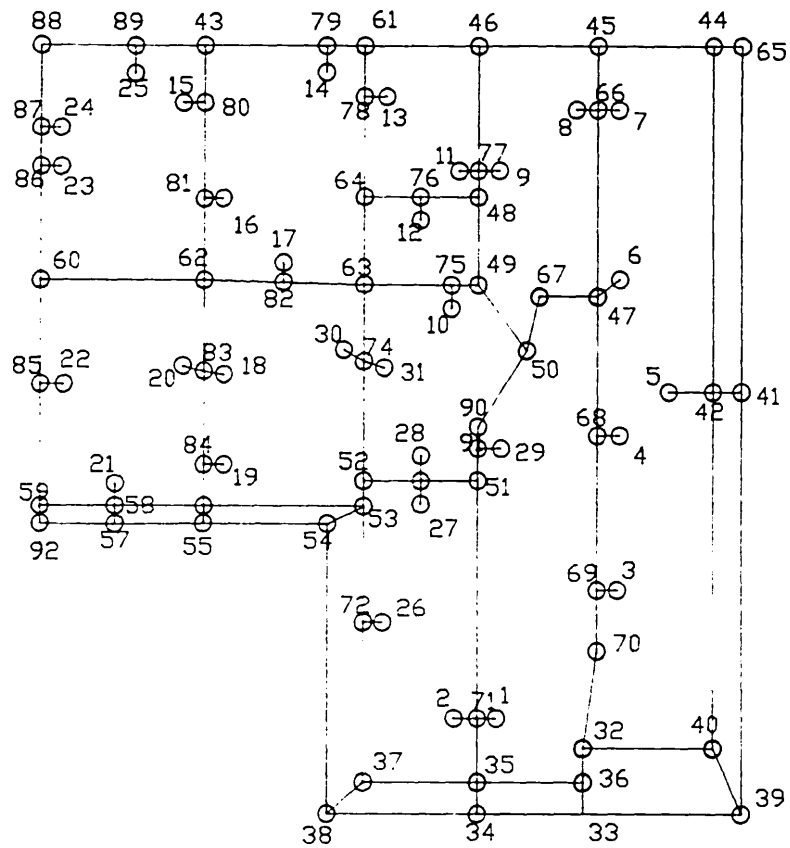


Figure 2.4 : No1shop "route" layout

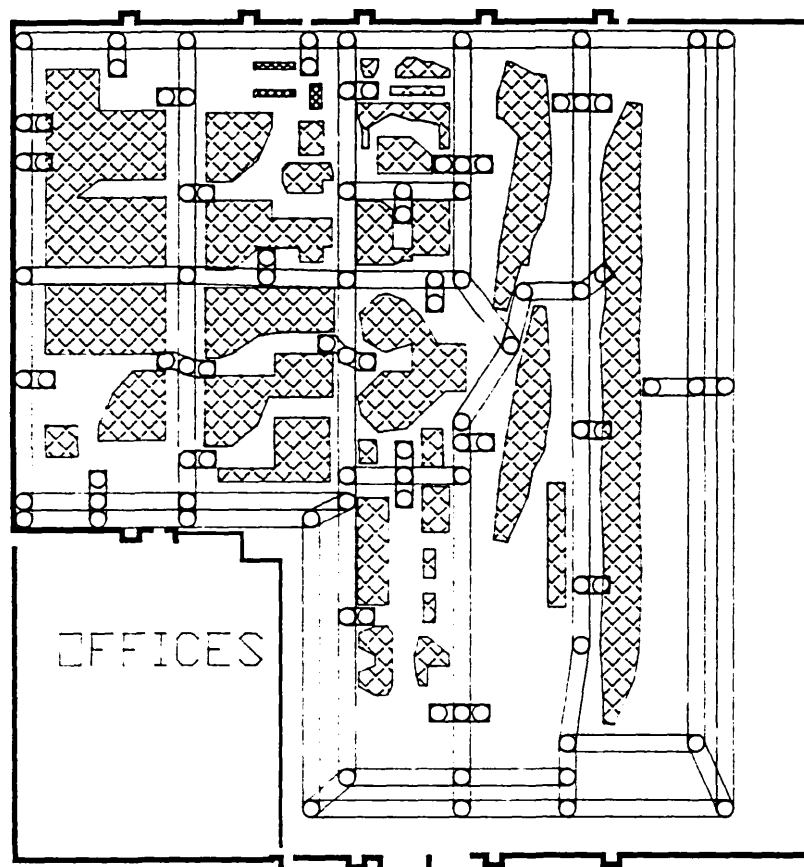


Figure 2.5 : No1shop "go-no-go" areas

2.2.3 Status Information

Based on updates from the vehicle via the communication network (and monitoring sensors placed in the environment), the supervisory system keeps track of all AGV movements. Any error in vehicle movement observed by the supervisory system can be corrected by sending correction commands to the vehicle.

Figure 2.6 shows the main components of the supervisory system.

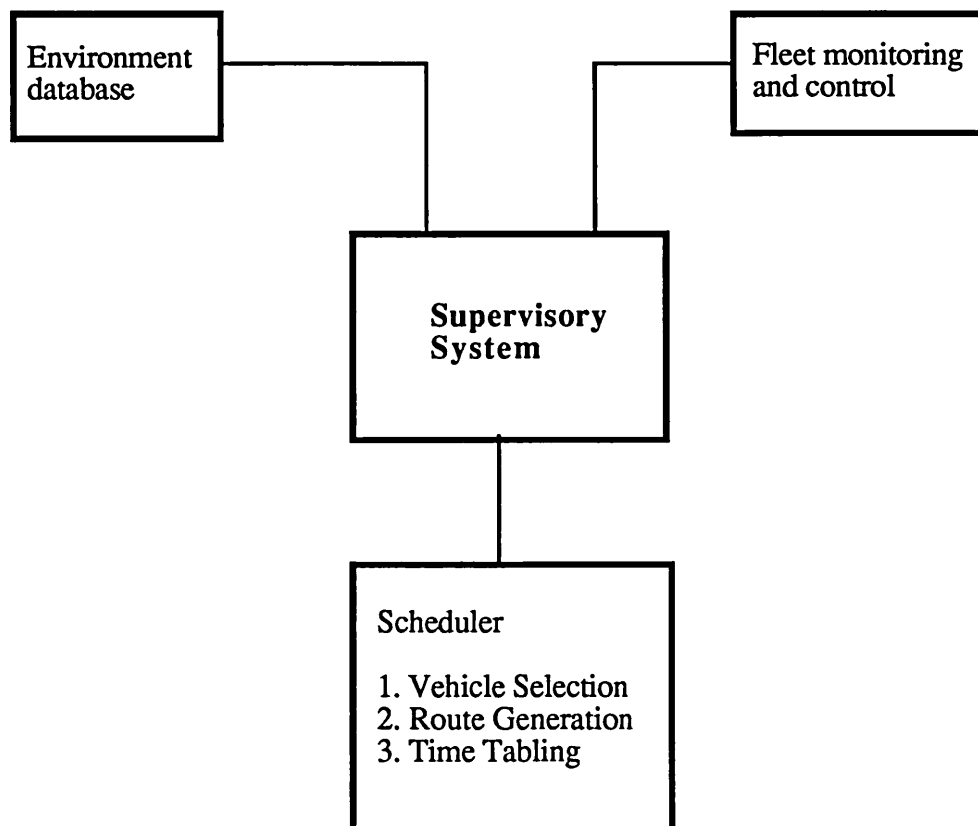


Figure 2.6 : Components of the Supervisory System

2.3 Scheduler

One of the main duties of the supervisory system is to create a schedule for the AGV journeys based on MTRs received. By scheduling a journey we imply that a proper vehicle has been selected to travel from the source node to the destination node via an optimal, conflict free route, meeting the time constraints imposed by the MTR.

Rodammer and White [21] noted that the scheduling problem encountered in production is really one of rescheduling, that is, one of modifying an existing production schedule to account for new jobs, new priorities, or unscheduled disruptions in production. Keeping this in mind any scheduling or dispatching scheme for the AGV traffic control system has to cater for various initial conditions and be able to reschedule journeys easily.

Scheduling a vehicle is made up of three steps:

1. Vehicle Selection
2. Route Generation and
3. Time-Tabling

2.3.1 Vehicle Selection

There are several static rules for the selection of the vehicle [13], [15], [17]. They are as follows:

1. First available AGV or AGV waiting longest
2. Least used AGV
3. AGV able to reach start node earliest
4. AGV 'Patrol System'

Note: the MTR specifies the type of AGV required

2.5.2 Route Generator

Depending on the Vehicle Selection routine, the route generator has to schedule a conflict

free route for the vehicle to reach the start node and then another route to reach the destination node.

Grossman [18] states that there are three alternative approaches to AGV traffic control :

1. The roads are restricted so that there is a unique route from any point to any other point. This is the case for nearly all existing systems.
2. There are multiple possible routes and each AGV has the autonomy to select its own route, in accordance with schemes intended to serve the common good.
3. There are multiple possible routes and all AGVs are under centralized traffic control. The level of control may range from simple traffic lights to global route optimization.

Our model fits into the third option. Various algorithms have been used for route generation - Travelling Salesman (TSP), Chinese Postman, Branch and Bound methods, shortest route problems, heuristic methods etc. The algorithm selected for route generation would depend on the application.

For ideal situations, well documented shortest path algorithms, such as the ones indicated below have been used [13] [16]. Heuristic rules have been added to make them suitable for AGV scheduling.

Dijkstra's Algorithm	- for shortest route
Pollack's Algorithm	- for second shortest route
Floyd's Algorithm	- all the shortest routes between pairs of nodes, stored in a lookup table

The shortest route may not be the best route to reach the destination node. Due to heavy traffic, the shortest route may actually require a longer time than a slightly longer route.

2.3.3 Time Tabling

AGVs have to be at their start node and destination nodes at particular times. This implies that when a route is generated, the time tabler of the scheduler will have to be checked to ensure that the time constraints imposed by the MTR are met. If the start and end time are

specified the AGV will have to alter its velocity along the route to meet this condition, this is the situation if the AGV is being used as an "assembly platform". If only the "end time" is specified, the AGV could travel at an optimal speed and the start time could be fixed accordingly. The time tabler will compare the node and path occupation times for the vehicle and all the active vehicles in the area. If there is any conflict, rescheduling will have to take place.

The time table has to check for three types of collision condition.

1. Node conflict - when an AGV is attempting to move to a node that is already occupied by another AGV.
2. Catching Up conflict - when there is a chance of collision between two vehicles travelling in the same path in the same direction but at different speeds.
3. Head on - collision between two vehicles travelling in the same path but in opposite directions towards one another

To avoid conflicts, the supervisory system can resort to the following methods:

1. Delay
 - a. Reduce top speed of one of the vehicles, in that way changing the time table of the vehicle.
 - b. Wait at previous node, this is similar to a traffic light situation.
 - c. Wait at starting node
2. Alternate Route - if delaying or slowing down does not solve the conflict situation, this method has to be adopted.
3. Zoning - in this method the paths are divided into zones, if an AGV is in a particular zone no other AGV can enter that zone. This method reduces the flexibility of the traffic control algorithm. The zones should be kept as small as practically possible.

If the vehicle detects an obstruction in its path, it reports to the supervisory system, the supervisory system places a blocked mark on the path and tries to reschedule.

Only if the three items - vehicle selection, route generation and time tabling are satisfied will the scheduling be successful. Otherwise the scheduler will look for an alternate route, failing which it will report failure and request for assistance.

A typical successful scheduled record would be of the form:

<u>AGV Number</u>	<u>Node</u>	<u>Arrival Time</u>
01	03	00:05:00
01	03	00:05:30
01	04	00:06:15
01	08	00:06:55
01	13	00:07:45
01	24	00:09:03
01	27	00:10:04
01	27	00:10:08
01	28	00:11:25

The table show the AGV identification number, the nodes to which the AGV must travel and the times at which the vehicle must be at the nodes. If the same node is specified twice, the vehicle should wait at the node for the time period specified by the difference of the two times.

2.4 Fleet Performance and Logging

Two databases will have to be maintained by the supervisory system to monitor system operation. They are :

1. MTR Queue - this is a list of all the MTRs that are active, that is, the MTRs that are being processed or waiting to be processed.
2. AGV Status - this list keeps track of the AGVs, whether they are available for journeys, not available for journeys, or in transit

Example of MTR Queue

<u>Position</u>	<u>Priority</u>	<u>MTR Number</u>	<u>Status</u>
01	05	00387	Scheduled
02	04	00350	Scheduled
03	04	00391	Scheduled
04	02	00388	No time table
05	01:55	00330	Scheduled
06	01:52	00331	No route
07	01:50	00332	Unscheduled

Example of AGV Status File

<u>AGV Number</u>	<u>Status</u>	<u>MTR Number</u>
01	In transit	00340
02	In transit	00345
03	Idle in node	00342
04	In transit	00346
05	Error #03	00000
06	Idle in node	00348

2.5 Simulation and Tracking

Using this package it is possible to fine tune the system to avoid obvious conflicts. This package is very useful in trying out new layouts whenever a change is needed - getting an indication of the performance of the system before any commitment to hardware.

A simulation cum tracking package is useful for monitoring individual journeys of the vehicle. A real time simulation with update information from the vehicle gives an indication of how close the vehicle is following its programmed path. The error signal is used to correct vehicle position. It is our expectation that the AGV will be following the programmed path to within acceptable limits and this feature will only be used for initial

debugging of the system. This feature also gives us an indication of proper functioning of the AGV. If there is a big error it indicates that there is a fault in the vehicle and action must be taken immediately.

2.6 Conclusion

From the supervisory system, we obtain a set of instructions that have to be sent to the AGV fleet to satisfy the transportation requests. The instruction to the vehicles could be of the following format :

Format of Instructions Sent to AGVs

{ Transportation Order Header }

{ Start Definition }

Start Node	- identifier or x, y coordinates
Start Time	- 00:03:00

{ End Definition }

End Node	- identifier or x, y coordinates
End Time	- 00:08:37
End Node Task	- 02
Task Time	- 00:00:05

{ End of Header Definition }

{ Start of Route Information }

List of nodes forming route	- identifier or x, y coordinates
Arrival time at each node	

To conserve communication bandwidth, the instruction to the vehicles should be as short as possible.

Below we see an example of an instruction sent to AGV Number 01.

Example

V01MBQ	- Vehicle No. 01 Message Begin , Queue type Command
S20T000300	- Start at Node 20 at time 00:03:00
E32T000837	- End at Node 32 at time 00:08:37
J02D000005	- End Node Job duration 5 secs.
N03T000334	- Intermediate Node 03 at time 00:03:34
N07T000413	- Intermediate Node 07 at time 00:04:13
N23T000521	- Intermediate Node 23 at time 00:05:21
N25T000642	- Intermediate Node 25 at time 00:06:42
N30T000723	- Intermediate Node 30 at time 00:07:23
V01ME	- Vehicle 01 Message End

To be able to obey the instructions, the minimum requirement of each vehicle in the fleet would be the following :

1. Know its own identification code (AGV Number)
2. Identify the type of instruction
 - a. queue type - load into batch file
 - b. immediate type - execute immediately
3. Be able to synchronise its clock with the supervisory "master clock".
4. Contain information on all the node coordinate points.
5. Vary its speed to meet the time tabled requests.
6. Wait whilst the job is being executed.

CHAPTER 3

COMMUNICATION SYSTEM

3.1 Introduction

The full potential of a computer controlled Automated Guided Vehicle system can be realised in a Computer Integrated Manufacturing environment. This can be achieved only if the computers (including the AGVs) can communicate with one another. In a free ranging Automated Guided Vehicle system, the only means for the supervisory system to know where each vehicle is located, is by having a reliable form of communication with the vehicles. The commands have to be sent from the supervisory computer to the AGVs and status information for monitoring AGV journeys must be received by the supervisory computer.

Since the AGV communication network is just one element in the overall communication network of the plant, the system has to conform to the network already in existence and contend with issues such as network access and communication protocols.

Each AGV can be treated as a computer system and the fleet can be treated as a distributed computer system networked to the AGV supervisory computer. The AGV supervisory computer is, in turn, networked to the plant wide communication network. Figure 3.1 gives an example of a possible plant wide system, showing the AGV controller and its position in the communication hierarchy.

In this chapter, we begin by giving a brief review of computer communication systems. The options available to the users are more or less as many as there are computer manufacturers. We elaborate on the two popular protocol standards - Manufacturing Automation Protocol (MAP) and Technical and Office Protocol (TOP), that were introduced by the users to force conformity on the vendors. We then proceed onto communication systems suitable for free ranging AGV Systems. The main difference between existing computer networks and the free ranging AGV network is that the physical link is wireless. The two wireless methods tried are Infrared and Radio Frequency (RF) which will be described next. Finally we give a description of the protocol used by the proprietary RF link used by our system.

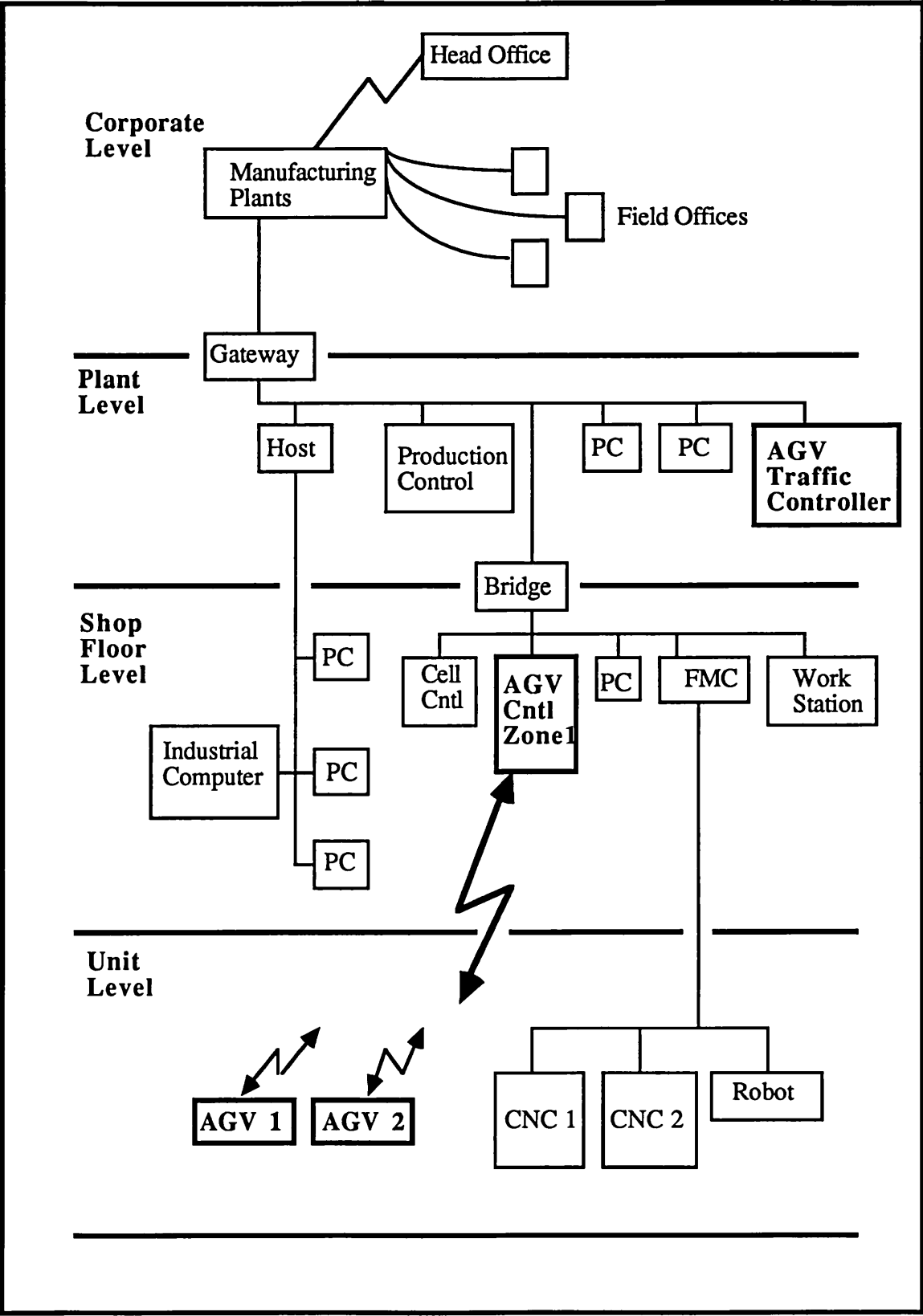


Figure 3.1: Four-tier hierarchy for industrial communications.

3.2 Computer Communication

Better control is achieved by integrating the various computing resources available in a manufacturing plant. Extensive literature is available for interconnecting computers in office environments, military environments and academic environments [27], [77], [28]. Gauthier et al. [22] give a comprehensive review of interprocess communication for distributed processes in industrial environments.

3.2.1 Communication Protocols

For computers to communicate with one another, it is necessary that they follow certain protocols. Communication protocols can be divided into two broad categories :

1. Synchronous protocols and
2. Asynchronous protocols.

Synchronous protocols use blocking techniques (eg. tokens) to regulate the flow of information from one computer to another. Once a message is transmitted, the source process is suspended or blocked until an acknowledgement is received from the destination. Blocking methods are suitable for command / response and query / status signalling. Remote Procedure Calls (RPC) is another form of synchronous protocol. RPC allows the user program to invoke "remote" procedures in a manner similar to local procedure calls and as in local calls, invocation blocks the user's program until the procedure returns. This blocking constraint implies that synchronous protocols are generally suited for tightly coupled processes (eg. bus of a computer) or in time critical processes where the tokens are held for a predetermined time period.

In asynchronous or message passing protocols, the sending process does not have to wait for the receiving process to acknowledge the message. In a "dialogue", the user "requests" service which is "granted" by the server. The mechanism could be either "transaction oriented" where a connection is established for the duration of the message (as in a "datagram") or "connection oriented" (as in a virtual circuit). In a virtual circuit, the communication link is maintained until the user terminates it. If the recipient does not send an acknowledgement within a preassigned time out period, the message is retransmitted.

Where a number of processors are involved, a "mail-service" concept can be used. A static block of shared memory (send and receive mailbox) is designated to each process. A process sends a message to a local server which transfers it to the appropriate receive mailbox. The destination processor is signalled when the message is deposited. It then responds by issuing a receive command to a local server which fetches the message. This method is suitable for parallel systems with a number of processors handling various tasks (eg. motion control, vision, ultrasonics).

3.2.2 Layered Communication and the OSI Reference Model

Due to the number of possible protocols for computer communications, interconnecting different computer systems has been difficult. To alleviate this situation, layered communication standards were introduced by various organisations (eg. IBM's Systems Network Architecture (SNA), Digital Equipment's DECNET, US Department of Defense's ARPAnet, the CCITT's X.25 standard, IEEE's 802 protocols etc.). Layering decomposes the mechanism of communication into smaller subtasks. Each subtask is independent of the subtask above and below it. Modification can be made to one layer without effecting the operation of the other layers.

To regulate the situation, the International Organization of Standardization (ISO) developed a set of Open System Interconnection (OSI) standards as a framework for defining the communications processes between systems. The OSI Reference Model includes seven hierarchical layers that define the functions involved in communicating; and definitions of services required to perform these functions. The basic model defined by ISO 7498 and adopted in 1984 is shown in Figure 3.2. Most vendors are modifying their proprietary protocols to conform to the OSI model [23].

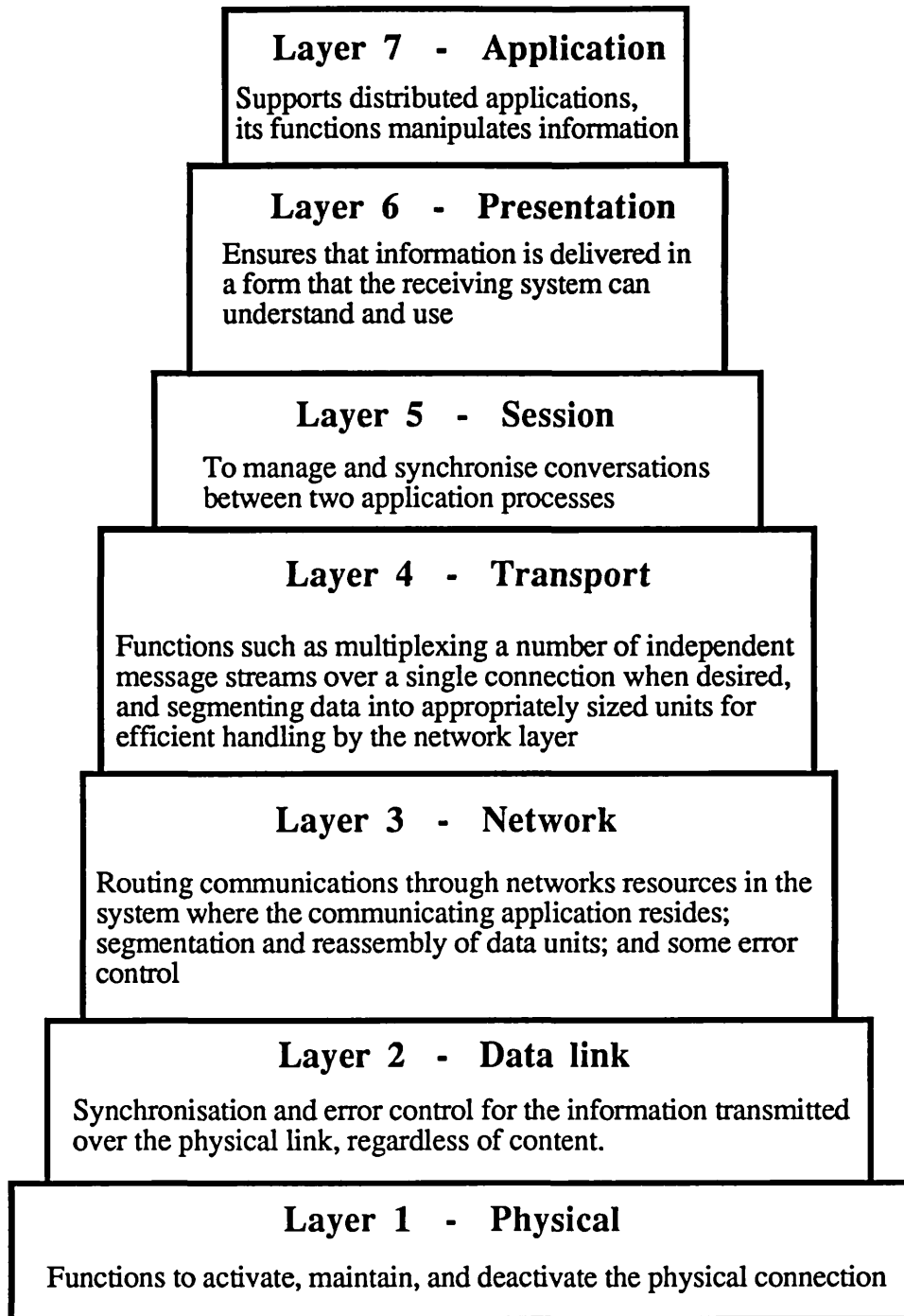


Figure 3.2. The OSI Reference Model

The physical layer is the lowest level of the reference model. It specifies the electrical and mechanical aspects of the communication hardware as well as the functional control of the data circuits. This layer activates, maintains and deactivates the physical connection. The data link layer establishes, maintains and releases data links. It also performs point-to-point error and flow control of frames. The network layer is responsible for network routing, switching, segmentation, blocking, error recovery and flow control of packets. It is in this layer that the network traffic is managed by either relaying data up to an application process or down along a physical network path. The transport layer furnishes the base for high level protocols by effecting data transfers and end-to-end reliability control. This layer provides various communication services to the session layer, multiplexes messages over logical connection and segments data to the network layer. The sessions layer establishes and terminates logical links between processes and manages the dialogue over those links. An important function of this layer is to synchronise data between the application processes. The presentation layer is responsible for the interpretation and manipulation of structured data. It ensures that data is converted to a form that the system can understand. This could include data encryption or conversion of data to device dependent codes. The highest level is the application layer. It is intended for user applications and management functions, such as file transfers, virtual file systems and virtual terminal emulators.

Figure 3.3 shows the "wineglass" of protocols. We note that even after layering the communication system there are still a number of the options available to the user at the highest (application) and lowest (physical) layer.

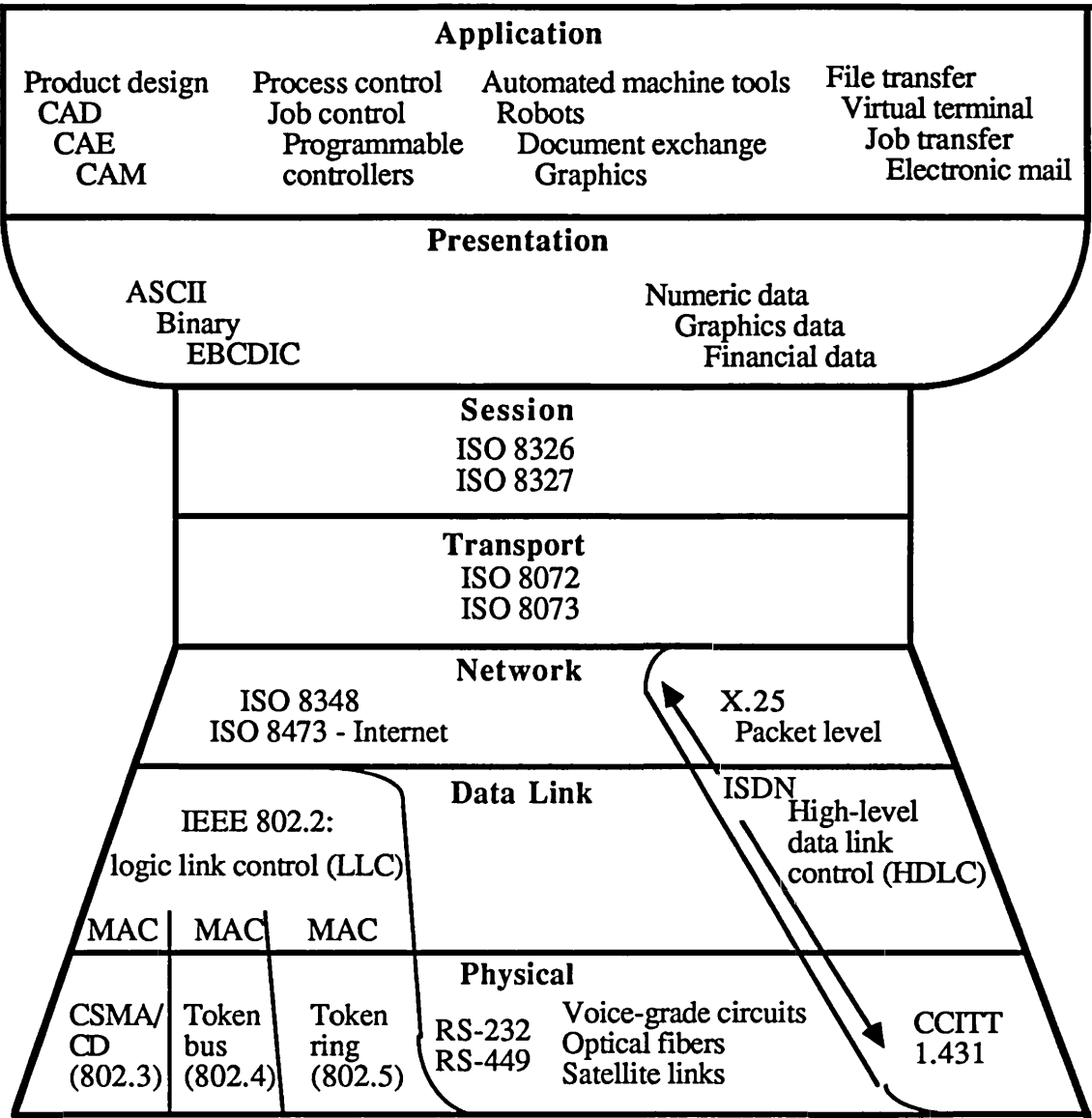


Figure 3.3 : The OSI "Wineglass" of Protocols [23]

3.3 Communication in Industrial and Technical Environments

To reduce the number of options still further, the users have sponsored two protocols :

- 1) the Manufacturing Automation Protocol (MAP) and
- 2) the Technical and Office Protocol (TOP),

to try and ensure compatibility between different equipment.

3.3.1 The MAP Standard

The Manufacturing Automation Protocol (MAP), spearheaded by General Motors, is fast becoming the de-facto standard for industrial environments. MAP is a seven layer, broadband, token-bus communications specification for the factory floor. This is a synchronous protocol.

In the MAP bus configuration, the network stations are wired together in a common bus and receive all messages. The "header" of the message contains the address of the destination station. All stations decode the "header". Token passing technique is used to ensure that all stations have a chance of sending messages. This involves passing a token (special bit pattern or packet) from one station to another. The station that holds the token has momentary control of the medium. As the token is passed from station to station a logical ring is formed. The station transmitting a message may do so for only a preset amount of time. The maximum time that any network station must wait before it receives the token and can then transmit is also precisely determined. This is important for real time process control where specific steps in the process must take place at specific time.

Broadband cabling was selected so that a variety of signals could share the same transmission medium. MAP broadband signalling technique is AM phase-shift keying specified in the IEEE802.4 standard.

3.3.2 The TOP Standard

The Technical and Office Protocol (TOP), sponsored by Boeing, aims at providing a common communication method for the exchange of information in applications, such as, electronic mail, word processing, file transfer, graphics, data management and business

analysis tools. TOP and MAP differs only in layer 1 (physical) and layer 7 (application) of the OSI Model.

For layer 1, TOP selected the IEEE 802.3 CSMA/CD (Carrier Sense Multiple Access with Collision Detect) set of protocols. This is an asynchronous protocol. Access is based on contention. When a station wants to send a message, it 'listens' to the medium to see if it is busy. If the medium is busy it waits a random amount of time and tries again. If the medium is not busy, the station begins transmission, simultaneously listening for any "collision" between its message and one from another station. If a collision is detected, the sending station backs off and jams the signal to inform the other stations. The receiving station will then discard the message received. The sending station again waits a random time period and tries to retransmit. CSMA/CD limits the range of the network size to the round trip propagation delay for collision detection. Theoretically a collision could occur every time the node tried to transmit and so block communication altogether. This drawback could disqualify IEEE 802.3 from applications where near real time performance is critical.

IEEE 802.3 was selected for its similarity to 'Ethernet' LANs. The wide base of network components available for Ethernet can then be used.

3.3.3 Interconnection between Networks

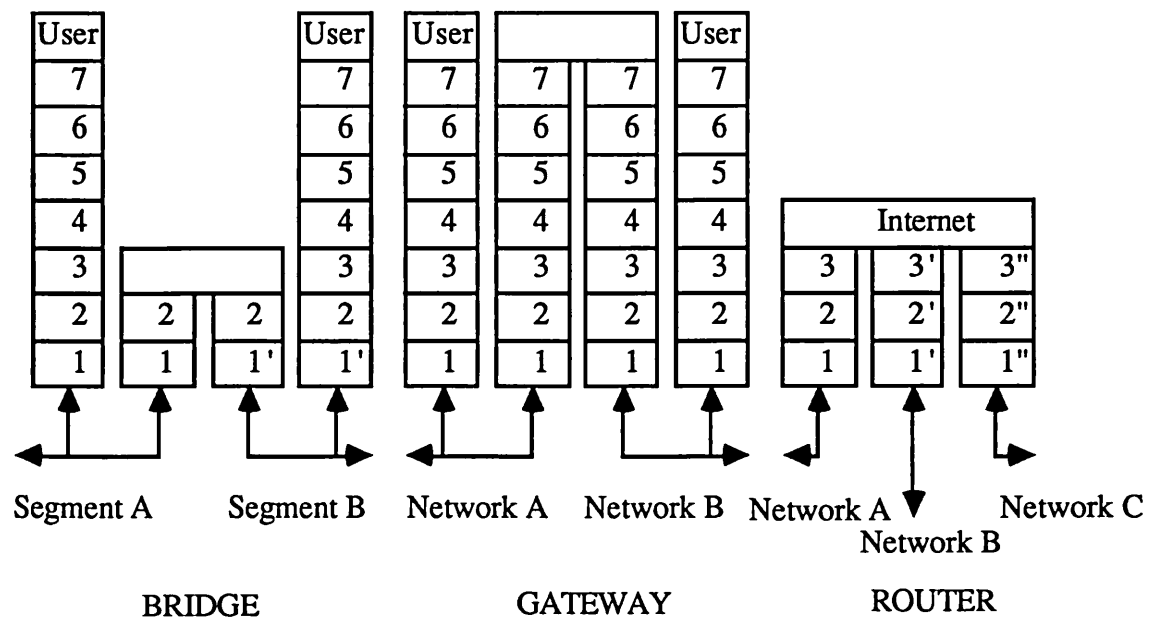


Figure 3.4 : Different methods of interconnection between networks

The various methods of interconnecting networks is shown in Figure 3.4. Figure 3.5 shows a MAP network with a gateway for non MAP networks.

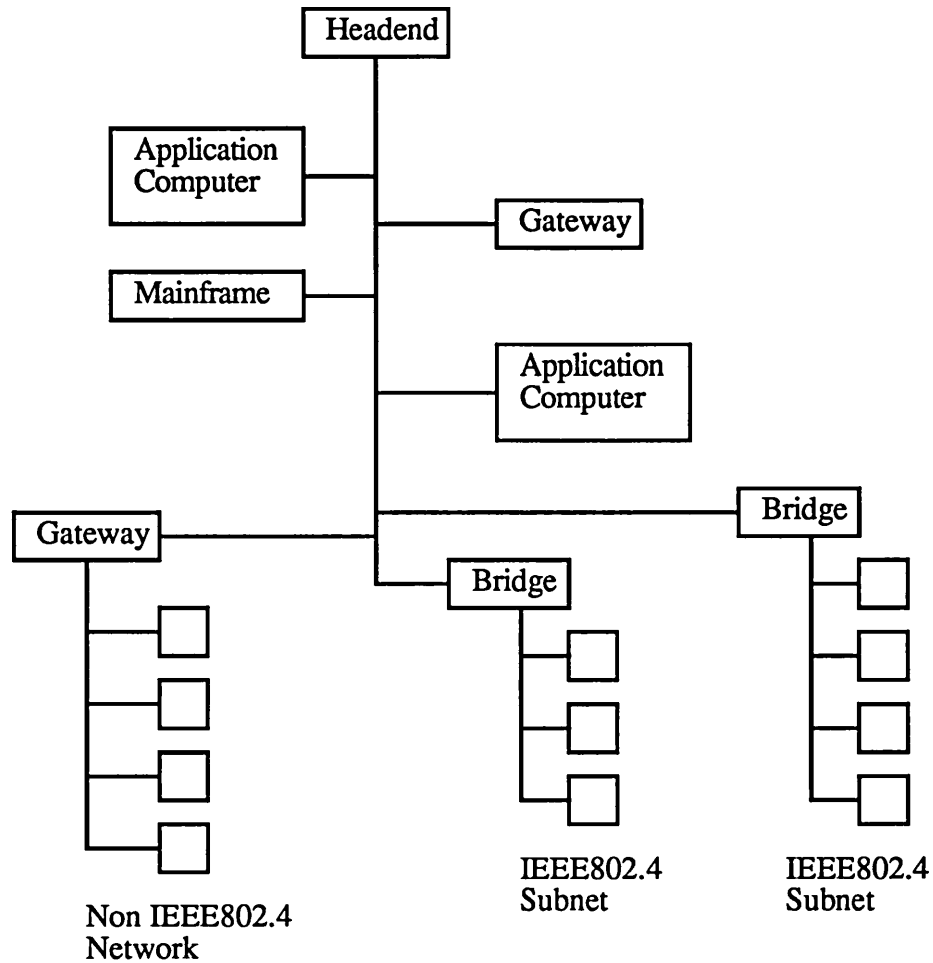


Figure 3.5 : A MAP Network with a "gateway" to a Non MAP Network [31]

3.4 Communication with AGVs

For communication between the AGV supervisory system and other systems such as production control computers and manufacturing cells, wire links, mains and network cabling can be used. The AGV supervisory system would be linked to the other systems using either MAP or TOP depending on the system in existence in the plant.

In a conventional wire guided AGV system, communication between the supervisory system and the AGV fleet is usually via an additional wire in the groove of the guide wire or by injecting a modulated signal on the guide wire itself [26]. This signal is picked up by a secondary coil in the vehicle.

For a free ranging AGV system, communication between the supervisory system and the vehicles, wireless links are desirable. The flexibility and cost savings gained by freeing the vehicle from the buried wire guidance system should not be lost in reintroducing the wires for the communication system.

There are two methods commonly used for wireless communication :

1. Infrared and
2. Radio Frequency.

Infrared communication has short range and line of sight limitation. It could be used for "local" communication in areas where these conditions are available. For example, when the AGV is about to "dock". Precise guidance information from the docking station could be beamed to the AGV. Also instructions such as loading and unloading instructions could be passed from the AGV to the station.

Radio frequency communication with its omnidirection and long range capability is ideal for "global" communication. The main advantage in having two forms of communication is that for the local form, where reliability is high, a high data rate method could be used. Infrared communication supports high data rates. However, higher the frequency, the greater the chance of absorption and so the range is limited.

By regulation, only the following bands can be used for RF communication for control applications in the UK - 173MHz and 475MHz. There is no regulation as yet on the use of infrared frequencies. However with the proliferation of infrared remote control devices, this situation may be short lived.

In Industrial environments, two factors effect the quality of radio communication :

1. interference and
2. fading.

Figure 3.6 shows the different infrared and RF communication systems for AGV systems.

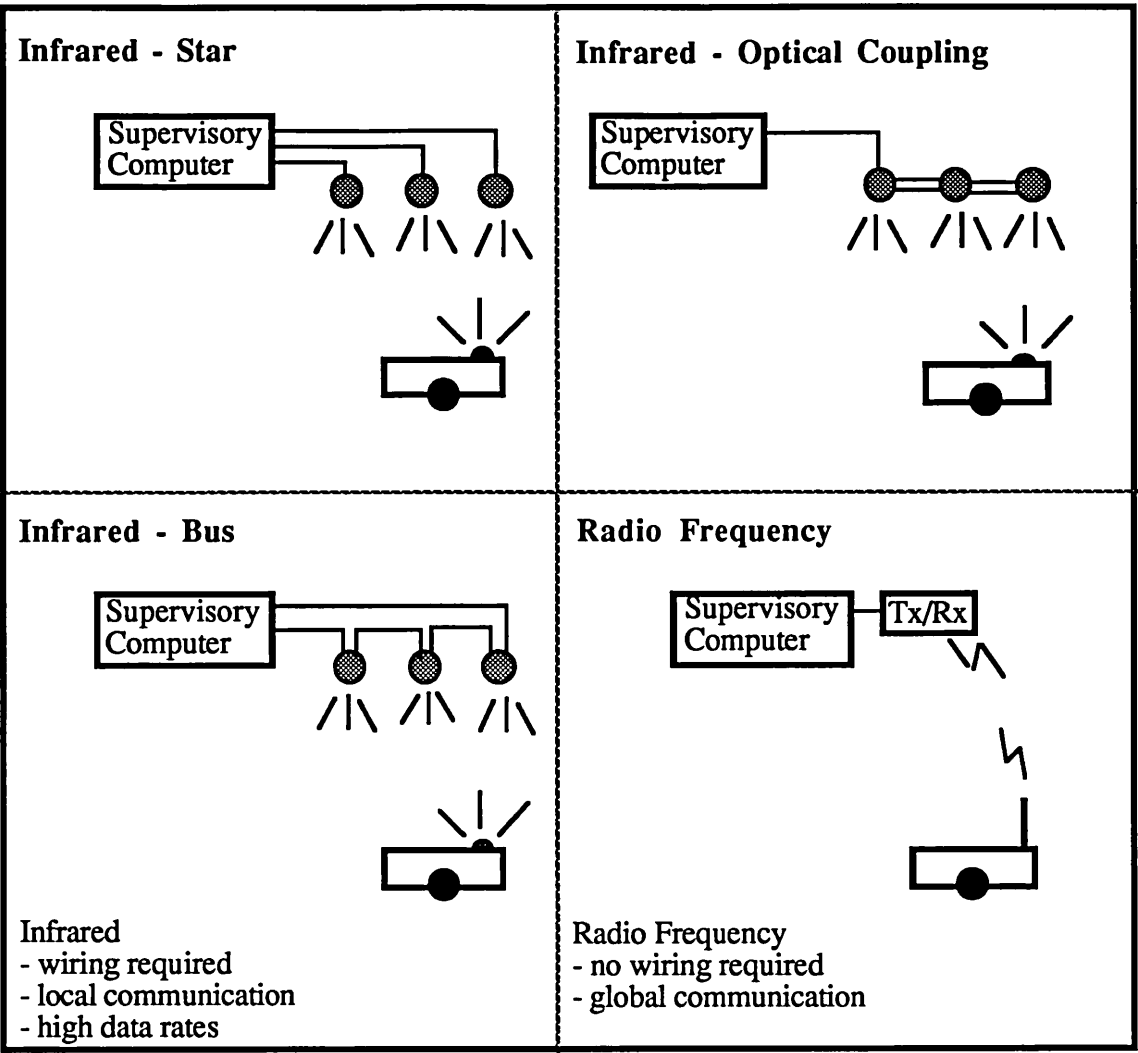


Figure 3.6. Infrared and RF Communication Systems [26]

Radio frequency noise energy generated by equipments such as arc welders, induction heaters, plastic bonders etc. causes interference in the High Frequency (HF) and the Very High Frequency (VHF) ranges. Empirical measurements [39] indicate that at frequencies above 1GHz the noise figure reduces appreciably. Radio systems operating at Ultra High Frequency (UHF) and above are not severely effected. However as the frequency increases, problems of fading, multipath and shadow effects become significant and depends on the topology of the workplace. Wide band measurements reveal that current

technology limits data rates to less than 50 - 100kbps. To solve this problem 'spread spectrum modulation' has been suggested as a possible method [1]. In the spread spectrum technique, an already modulated signal is modulated a second time in such a way as to produce a waveform which interferes in a barely noticable way with any other signals operating in the same frequency band. For example, a spread spectrum signal operating at the same carrier frequency as an AM signal with a bandwidth of 10KHz, has the same power (P_s) as the AM signal but has a bandwidth of 1MHz. The noise immunity is achieved at the expense of bandwidth.

3.5 Implementation

The RF communication system selected for the Imperial College free ranging AGV system is a proprietary system developed by PACS Limited. A proprietary system was chosen to ensure that the strict international regulations imposed for RF communication is satisfied. It operates at 173.275MHz. The PACS system caters for upto 255 AGVs (secondary nodes) controlled by a single supervisory system (primary node). It operates in half duplex mode. It has a synchronous protocol, ensuring that the messages are received within a fixed time period. So we know that the signal will be received by the fleet within a preset time period. At present this is in the range of 0.5 seconds. With features such as ACK, NAK, BCC (Block Check Character) and retransmission (if the message is not received) the protocol used is reliable and robust. A full description of the PACS Communication System can be found in Appendix A1.

3.5.1 Hardware

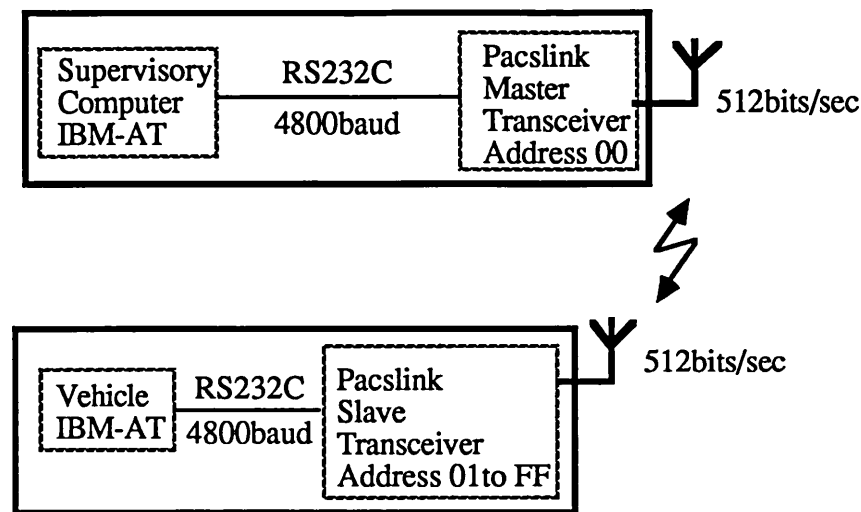


Figure 3.7 : Pacslink Communication System

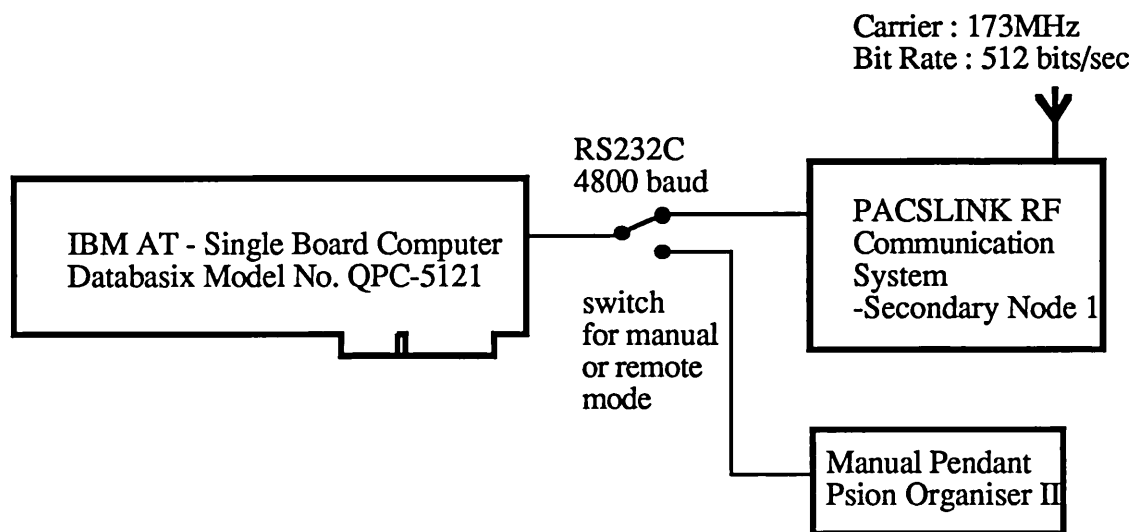


Figure 3.8 : Communication System at Vehicle End

Figure 3.7 shows the Pacslink communication system and figure 3.8 shows the connection at the vehicle end. The supervisory node or primary node has the address '0'. The vehicles have addresses from "1" to "255".

CHAPTER 4

GUIDANCE METHODS

4.1 Introduction

After selection of an appropriate time-tabled route by the supervisory system, the information is sent to the vehicle via the communication system. The vehicle will now have to fulfil the requested transport order based on its guidance system. The guidance system has to ensure that the vehicle goes from one point in the environment to another, following the specified path, safely and reliably.

There are various ways of dividing the guidance problem for free ranging vehicles. The capabilities of guidance sensors have been classified into horizons [67]. The closest horizon is just under the vehicle, providing vehicle position and orientation information. The next horizon identifies the vehicle's capability to locate and recognise nearby markers and obstacles. The final horizon defines the limit of the vehicles ability to characterize distant terrain.

Meystel [66] uses a principle called "nested hierarchical control" where the navigation problem is divided into three planning / control levels. The three hierarchical levels are Planner, Navigator and Pilot. The Planner works on a (terrain) map to generally determine the subgoals to reach an assigned task. The world representation consists of regions of constant traversability. The Navigator works on a map of higher resolution. The task of the third level of the nested hierarchy (Pilot) provides optimal motion control, obstacle avoidance and maintains continuity of motion along the path.

The software processing levels of the CMU Mobile Robot Projects [59] follow a layered architecture. Figure 4.1 shows the eight problem solving layers into which it is divided. Sensor information is integrated to create or update the real world map which is used by the navigation layer for activities such as path planning and obstacle avoidance.

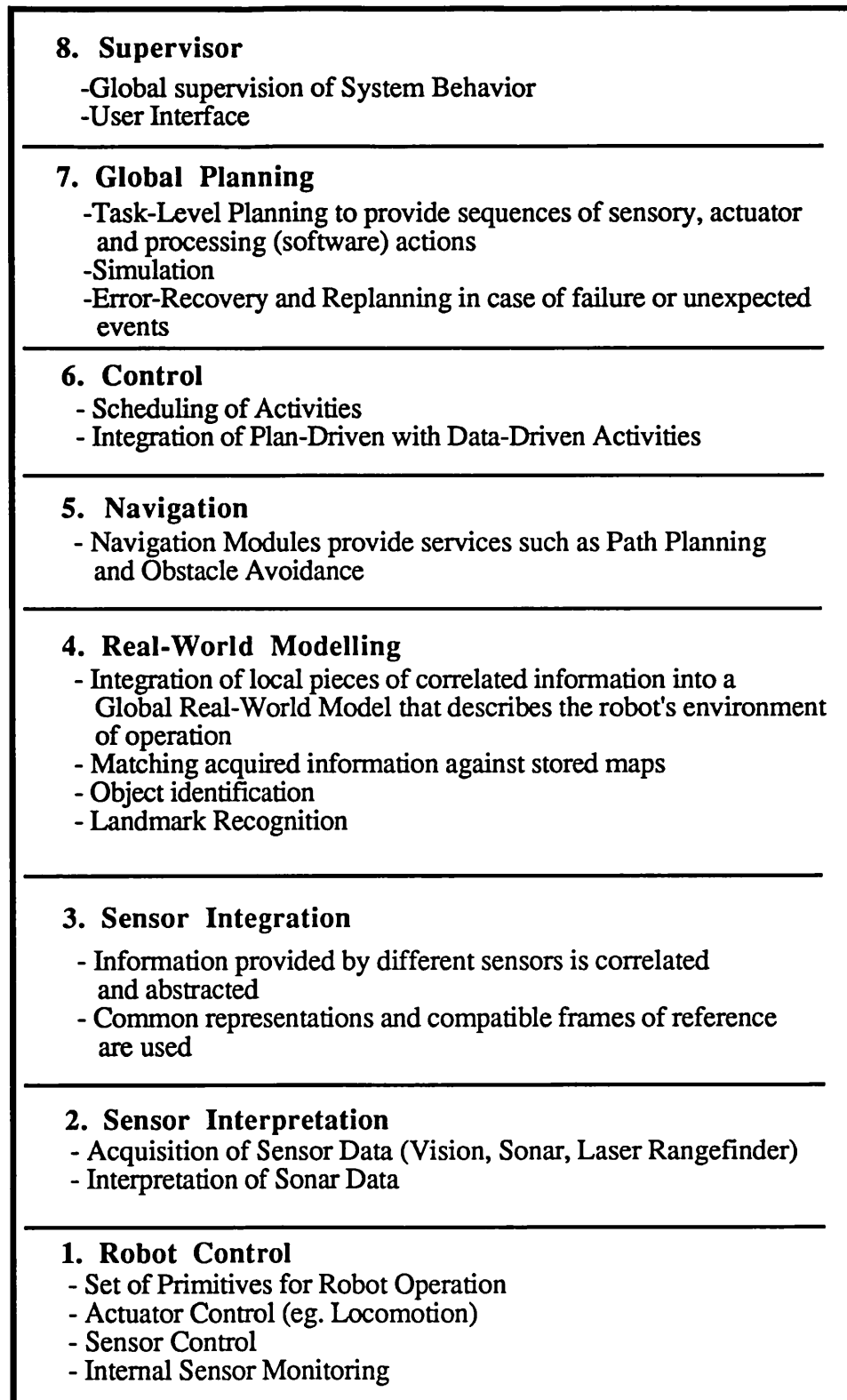


Figure 4.1 : Problem solving layers of the CMU Mobile Robot Projects.

In this chapter, we review the various guidance options available for free ranging automated guided vehicles. Unlike conventional AGVs, a free ranging vehicle does not have physical guidepaths to guide the vehicle from point to point. Therefore, for reliable guidance, a free ranging vehicle needs more than one guidance method. As in the horizon approach, we begin with guidance sensors onboard the vehicle, dead reckoning methods (odometry) and inertial guidance methods. As a check on these sensors, we proceed to the next horizon with sensors that look at the surroundings. These sensors provide the real world update information for the vehicle. Finally, we describe the guidance options available on our prototype vehicle.

4.2 Dead Reckoning

Navigation is the process of directing the movement of the vehicle from one position to another. To be able to accomplish this, the present position and velocity of the vehicle has to be known relative to a fixed frame of reference. The combination of distance travelled from a known position with an estimate of the direction of the track being followed is the basis of the dead reckoning method. Dead reckoning is one of the earliest means of navigation. In this method encoders are connected to the wheels. By measuring wheel rotations (odometry) the control system on the vehicle is able to determine the position (x,y) and heading (θ) of the vehicle. The posture of the vehicle is completely described by its position and heading (x,y, θ).

Odometric calculations for three different drive configurations are given below. The three configurations considered are :

1. Differential drive,
2. Steered wheel tricycle drive and
3. FLEXL : A steered wheel differential drive.

These three configurations cover most options available for wheeled drives.

4.2.1 Differential Drive

Two wheels are driven separately by two motors. Curved paths can be achieved by varying the differential speed of the two motors. The centre of rotation (CR) is between the two drive wheels.

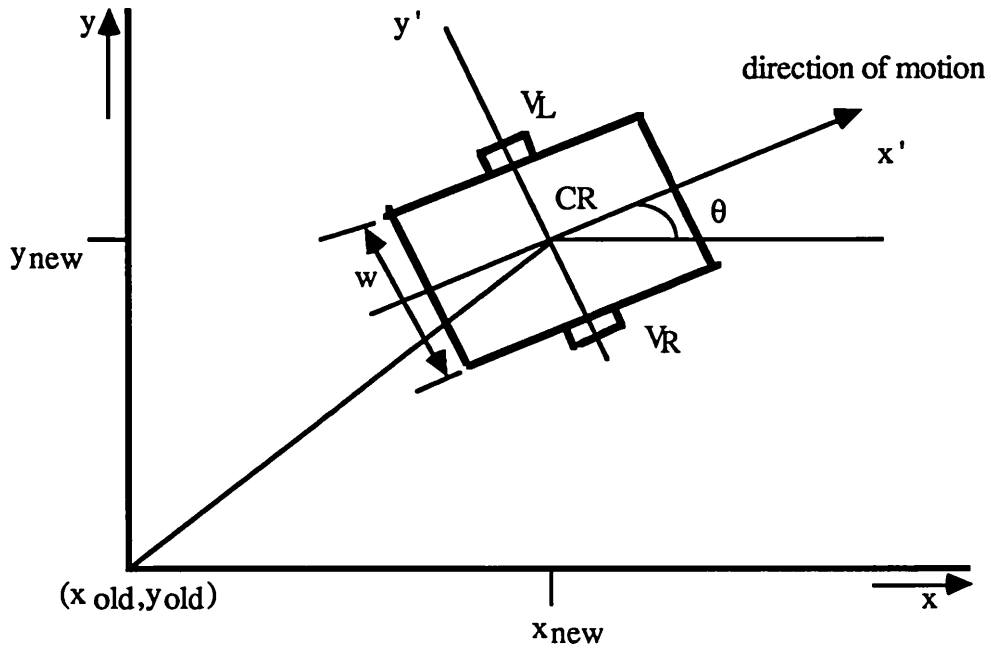


Figure 4.2 : Differential Steering

Instantaneous vehicle velocity along the path x'

$$V = (V_R + V_L) / 2$$

Rate of change of vehicle heading

$$\theta = (V_R - V_L) / w$$

where

$$\text{velocity of right wheel } V_R = (x_{R_{new}} - x_{R_{old}}) / \Delta t$$

$$\text{velocity of left wheel } V_L = (x_{L_{new}} - x_{L_{old}}) / \Delta t$$

The new coordinates of the vehicle are :

$$\begin{aligned}\theta_{\text{new}} &= \theta_{\text{old}} + \theta \\ x_{\text{new}} &= x_{\text{old}} + V\cos(\theta_{\text{new}}) \\ y_{\text{new}} &= y_{\text{old}} + V\sin(\theta_{\text{new}})\end{aligned}\tag{4.1}$$

If the wheels are situated in the center of the vehicle, the vehicle will be able to move equally well forwards and backwards. The operation of differential wheel drives and tracked drives is similar. Tracked systems follow arbitrary paths by varying the velocity of the two tracked drives. The odometric equations indicated above are valid for tracked vehicles.

4.2.2 Steered Wheel Tricycle Cart

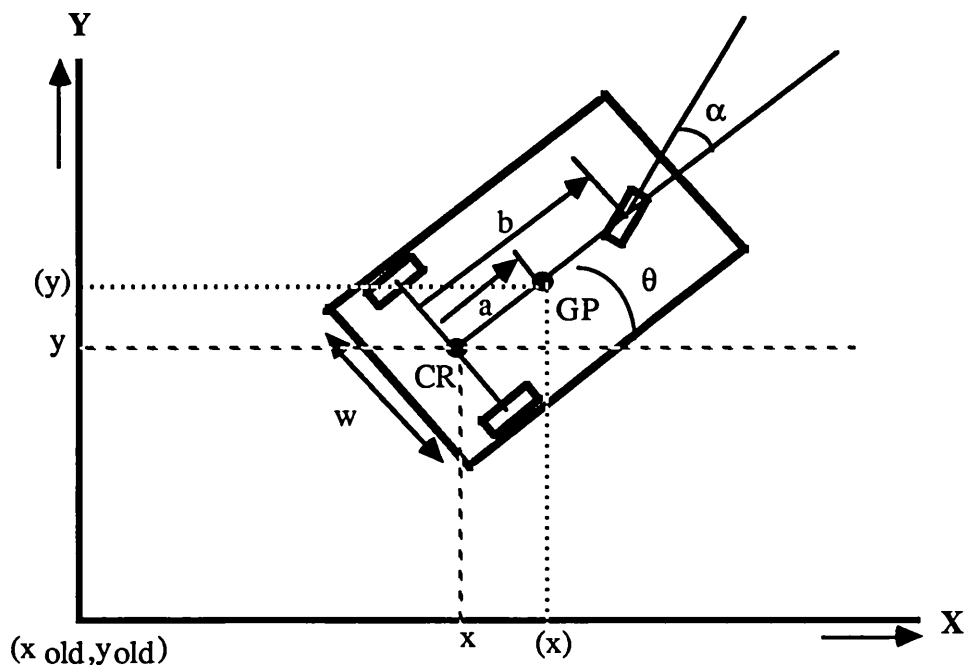


Figure 4.3 : Tricycle Cart Configuration

The tricycle cart is controlled by the front wheel steering angle α and the drive wheel

velocity ω . With respect to the center of rotation of the cart heading (the point marked "CR"), the speed v , heading θ , and position (x,y) of the cart are related to the steering and the drive angular velocity coordinates (α,ω) by the kinematic equations:

$$\begin{aligned} v &= R\omega \cos \alpha \\ d\theta/dt &= (R/b)\omega \sin \alpha \\ dx/dt &= v \cos \theta \\ dy/dt &= v \sin \theta \end{aligned} \quad [4.2]$$

where

R = radius of the wheel

θ = heading of the vehicle

GP = guide point - point specified to follow the route

a = distance from center of rotation (CR) to the guide point

b = distance from centre of rotation (CR) to steering wheel

Most of the AGVs in use or in development at the present time are steered by a single wheel (or wheel pair) or are steered by differentially driven wheels. Both of these more common types of steering have steering discontinuities for ideal path following along line-arc segments. The steered wheel configuration has discontinuities in the desired steering angle, whereas the differential drive configuration has discontinuities in the desired velocities of the drive wheels. Nelson [46] shows that by proper choice of guidepoint (the point on the vehicle that is designated to follow the reference path) these discontinuities can be eliminated. The choice of guide point effects the space swept out by the vehicle, the kinematics and dynamics of the vehicle and the desired steering and drive functions (the inverse kinematics) for path following.

4.2.3 A Steered Wheel Differential Drive : FLEXL [40]

This drive configuration has two vehicle drives and one drive for orientation. A relative movement between the centre of rotation and the vehicle frame is possible with the help of the orientation drive. With three motors, this vehicle can travel along any path in the plain with any orientation of the vehicle frame.

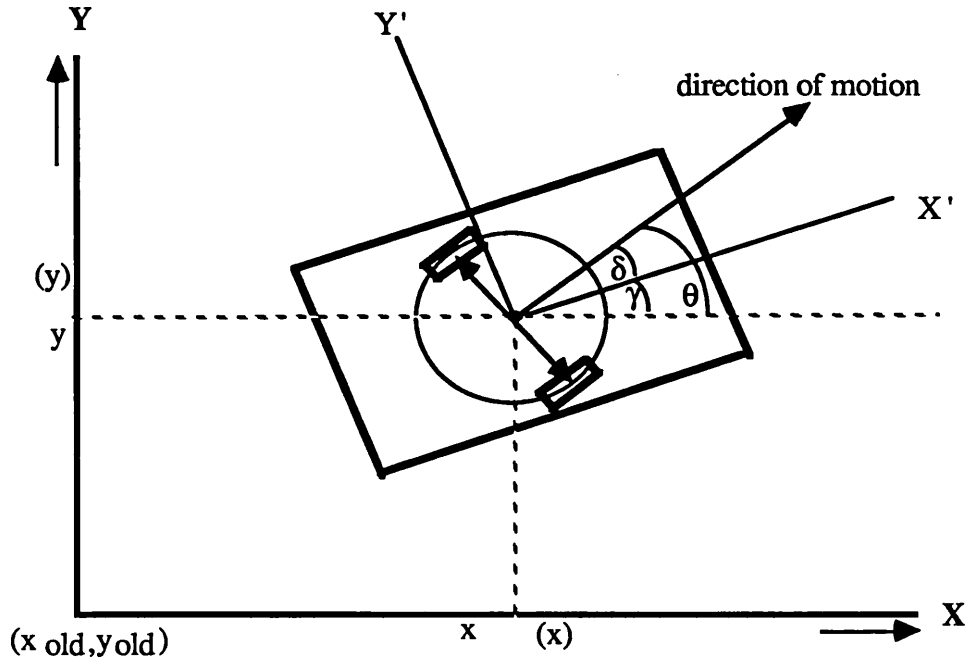


Figure 4.4 : FLEXL - Differential Drive + Orientation Drive

Instantaneous vehicle velocity along the path x' is

$$V = (V_R + V_L)/2$$

Heading (θ) of the vehicle is

$$\theta = \delta + \gamma$$

(δ is the steering angle with respect to the vehicle x' axis and γ is the angle between the vehicle x' axis and the world x axis)

Rate of change of vehicle heading

$$\Delta\theta = d\delta/dt + d\gamma/dt$$

where

$$d\delta/dt = (V_R - V_L)/W$$

(W is the distance between the two drive wheels)

velocity of right wheel : $V_R = (x_{Rnew} - x_{Rold})/\Delta t$

velocity of left wheel : $V_L = (x_{Lnew} - x_{Lold})/\Delta t$

The new posture of the vehicle is :

$$\theta_{new} = \theta_{old} + \Delta \theta$$

$$x_{new} = x_{old} + V \cos(\theta_{new})$$

$$y_{new} = y_{old} + V \sin(\theta_{new}) \quad [4.3]$$

Vehicles with omni-directional motion capability have the advantage of continuous steering functions. They also have good docking maneuverability.

4.2.4 Odometric Errors

The frame of reference of the dead reckoning method is on the vehicle. Any error in the parameters of the odometric equations will result in an incorrect posture for the vehicle with respect to the real world [39], [33]. The odometric equations for a differential drive vehicle are :

$$\Delta \theta = 2\pi R (E_R - E_L)/EW \quad [4.4]$$

$$\Delta x = (\pi R (E_R + E_L)/EW) \cos(\theta) \quad [4.5]$$

$$\Delta y = (\pi R (E_R + E_L)/EW) \sin(\theta) \quad [4.6]$$

where

R = radii of the wheels

E_R = encoder count of the right wheel

E_L = encoder count of the left wheel

E = encoder count per revolution of the wheel

W = vehicle width from centre of one wheel to another

For straight line motion, if $E_R = E_L$, the heading error is zero. If there is a slight error in the encoder readings of one of the wheels, this will result in a cumulative error in the

heading term. This error may be due to unequal radii of the two wheels. This will be reflected in the position terms. An error in the radius of the wheels will result in errors in the total distance travelled. These two errors can be minimised by proper calibration.

For turning motion, if the encoder counts are accurate, a source of error could be the vehicle width. This has to be measured precisely.

Even after calibrating and taking precise measurements of the vehicle, errors may still creep in. This is due to the following factors [47] :

1. Uncertainty in wheel rolling radius : this could alter with wear or weight of the load carried.
2. Wheel slippage : odometric wheels should make good frictional contact with the floor. However this is in conflict with the requirement above.
3. Uncertainty in separation of effective wheel contact points : floor wheel contact inevitably extends over a finite lateral distance and the lateral position of the "no slip" region thus cannot be located precisely. When cornering this uncertainty in effective wheel separation (W) produces an uncertainty in the change in the heading of the vehicle and hence ultimately in its position.
4. Small obstruction on floor : the passage of the odometric wheels over obstruction (grit or small debris), particular by only a single wheel, results in incorrect measurements.

4.3 Inertial Guidance

This method consists of setting up the axis of the gyroscope parallel to the direction of motion of the vehicle. If the vehicle deviates from its path, an acceleration will occur perpendicular to the direction of motion and is detected by the gyroscope. This acceleration is then integrated twice to yield the position deviation from the path, which can then be corrected by the servo mechanism [34]. The method is extensively used in systems where there is no external frame of reference (eg. guided missiles, aircrafts, ships etc.). Velocity and position information has to be obtained from measurements made by inertial instruments. There are four conceptual elements involved in inertial navigation :

1. a vector accelerometer
 - a device which measures a vector non gravitational acceleration of the case of the instrument. For three axis, three accelerometers are

required.

2. an attitude reference

- two or more gyros, suitably gimbaled relative to the vehicle provide a suitable element which remains in a predetermined spatial orientation inspite of vehicle movement.

3. a computer

- required for solving acceleration equation by calculating gravitational acceleration, performing integrations for velocity and position and by making coordinate transformations.

4. a clock

- needed as all other information with respect to the moving vehicle is time dependent.

The velocity and position readings depend on initial conditions. Inertial navigation instruments tend to be expensive and bulky, the axis of the gyroscope also tends to drift with time, giving rise to errors.

4.4 Imaging Systems

From sensors measuring parameters on the vehicle frame of reference, we move to sensors that locate objects and markers in the "world" reference. We begin with one dimensional problems, using photo sensors that make use of collimated light beams, progressing to painted path lines and line scan cameras. For two dimensional scene analysis, we have cameras looking down at bright point sources on the vehicle and of cameras on the vehicle looking down at lines and markers on the floor. Three dimensional image analysis comprises of image or object detection by means of stereoscopic vision. Basically all these guidance methods are simplified image analysis problems. Image processing problems can be computationally time consuming. It is necessary to reduce the amount of information to be analysed for them to be suitable for real time guidance problems. The methods described below are mainly for updating the vehicle position with respect to the real world.

4.4.1 One Dimensional Imaging

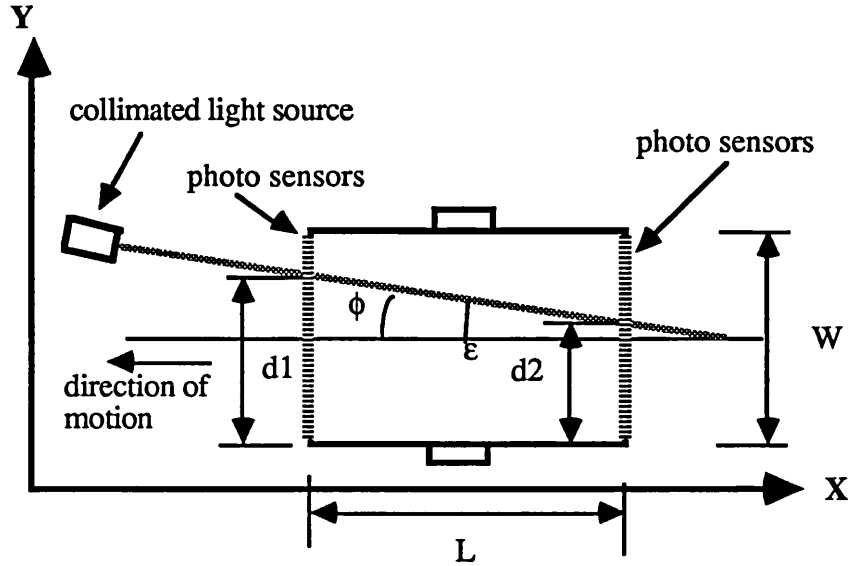


Figure 4.5 Collimated Light Source and Photo Sensors

This method suggested by Ren and Walker [47] comprises of a collimated light beam and two rows of photo sensors, one row situated in front of the vehicle and the other at the back of the vehicle. The beam angle is known with respect to world coordinates. When the vehicle crosses the beam, it is detected by the front row at $d1$ and by the back row at $d2$. From these values it is possible to calculate the lateral and heading deviation of the vehicle with respect to the beam. The vehicle could then correct any heading error that may be present.

The lateral (ϵ) and heading (ϕ) deviation with respect to the collimated beam is as follows :

$$\phi = \tan ((d1 - d2)/L)$$

$$\epsilon = (1/2)(W - d1 - d2) \cos \phi \quad [4.7]$$

The lateral deviation and heading deviation is measured with respect to the centre of rotation of the vehicle.

If the beam were emanating from the docking station, it could be used to guide the vehicle into the docking station. If the velocity of the vehicle were known it is possible to calculate

the lateral and heading errors by taking subsequent readings from a single row of photo sensors. This system was implemented for the docking system for ICAGV-I. [48], [49]

In the second method [38] a line scan camera looks at a painted line, one meter in front of the vehicle. As the velocity of the vehicle is known, it is possible to find the lateral and heading error of the vehicle.

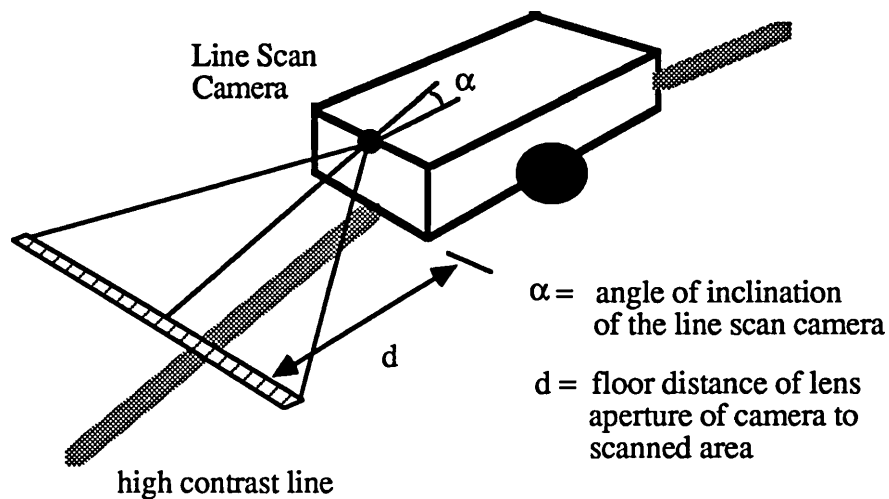


Figure 4.6 Line Scan Camera and high contrast line

Since both these methods are basically one dimensional, distance of markers cannot be computed.

4.4.2 Two Dimensional Imaging

4.4.2.1 Offboard

In this method, cameras placed in the environment track the motion of the vehicles (Figure 4.7). If the image area of the camera is calibrated with respect to the world coordinates, it is possible to calculate the coordinates of the vehicle. To simplify the image processing problem, the vehicles have bright point light sources on them. The pixel value of the high intensity spots in the image are calculated as the coordinates of the vehicle. These

coordinates are communicated to the vehicle and the vehicle corrects its posture if necessary. [19]

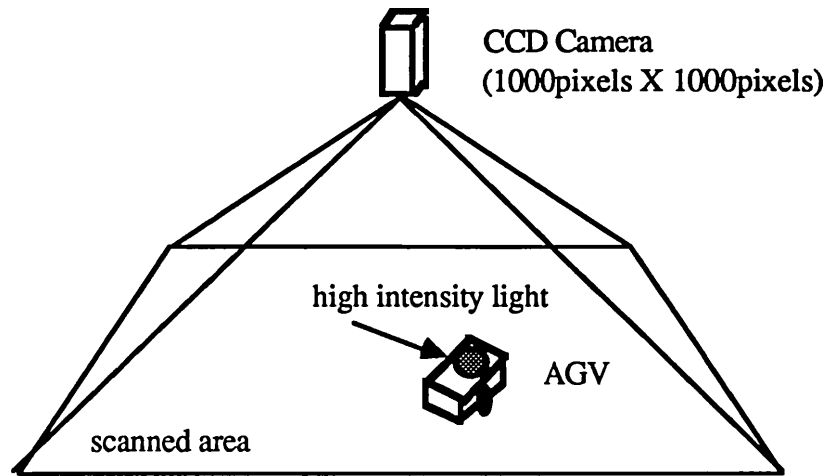


Figure 4.7 Camera looking down at AGV

This approach is part of the "intelligent environment" approach as opposed to the intelligent vehicle approach. This approach becomes difficult to implement if the area travelled by the vehicle is large or if more than one vehicle is detected by the camera. To achieve reasonable accuracy a number of cameras may be required. Each camera would have its zone of operation, there would be a slight overlapping of zones. Cameras would have to be informed when vehicles are about to enter their zone. As the system gets larger the overheads required to operate this system increases considerably.

This guidance method may be suitable for small manufacturing cells where a single vehicle has to service a few stations.

4.4.2.2 Onboard

In this case (Figure 4.8) the camera is mounted onboard the vehicle and looks at the area just in front of the vehicle. If the coordinates of the camera is known with respect to the vehicle, the image can be mapped to the vehicle coordinate system. As the camera is

looking down at a level floor (2-D surface) the coordinates, with respect to the vehicle, of any marker can be calculated. If the coordinate of the marker with respect to the real world is known, it is possible to calculate the vehicle coordinates with respect to world coordinates.

If the marker on the floor is the guidance lines, the vehicle can calculate the lateral and heading deviation and so keep within the specified path.

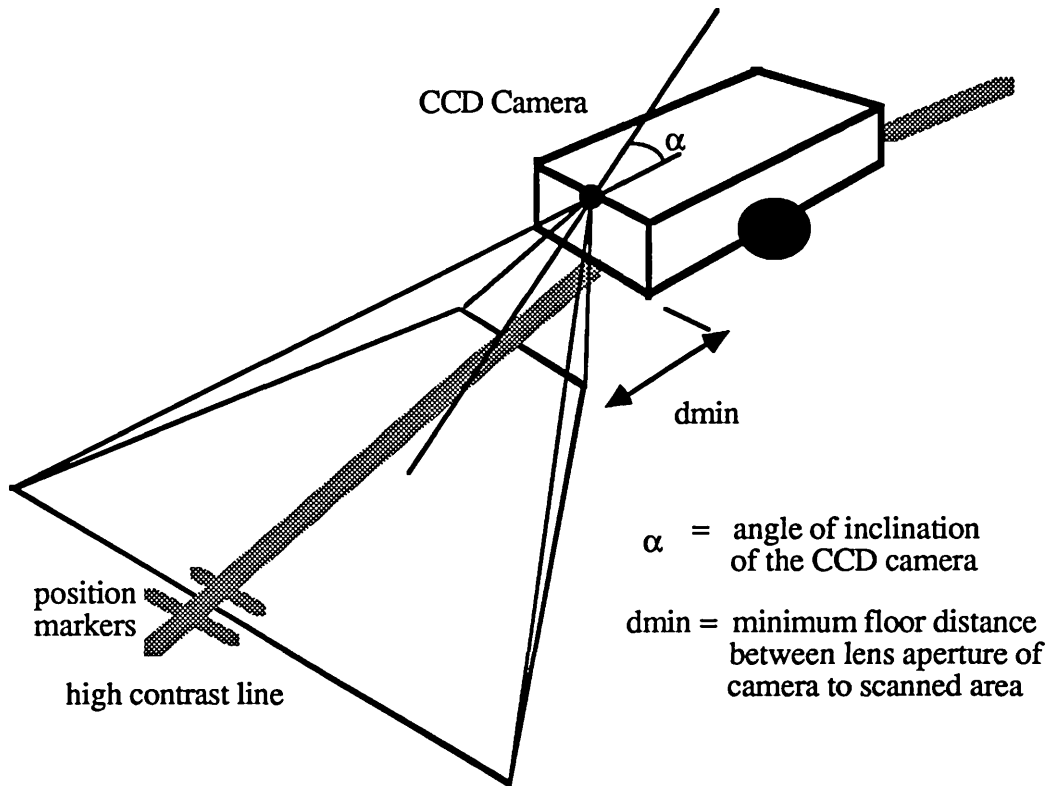


Figure 4.8 Camera Pointed at 2D Floor

4.5.3 3-D Imaging of the Surroundings

In Figure 4.9, two cameras a known distance apart (a), look at the same scene. By analysing the scene from the two cameras and by using triangulation calculations (angles α and β are known) on target points on the image, it is possible to get a position fix on the target. The computation time required to extract three dimensional information from stereo

pairs limits the number of points that can be tracked. The main problem with stereoscopic vision is the correspondence issue. It is difficult to decide on the corresponding point in the two images.

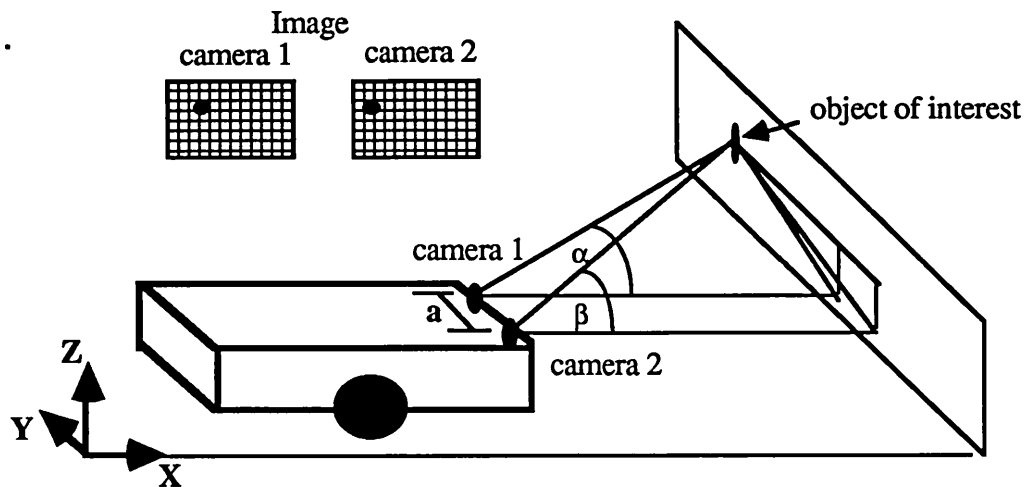


Figure 4.9 Stereoscopic Vision

4.5 Range Imaging

Range information can be obtained by two methods :

1. Triangulation

- if the coordinates of two points are known, it is possible to determine the distance of a third point if the angle subtended by the two points to the third point is known.

2. Time of flight

- if we know the velocity of the signal, it is possible to calculate the range of objects by recording the time interval between the emitted signal and return echo from the object.

4.5.1 Laser Ranging

A rotating laser on the vehicle looks for beacons (or markers) in the surrounding area. If the world coordinates of the beacons is known, from angular information, triangulation calculations can be done to determine the coordinates of the vehicle. This method is used by the only commercial free ranging AGV system in operation - Caterpillar GEC Self Guided Vehicle. The SGV system uses "bar codes" as absolute position markers.

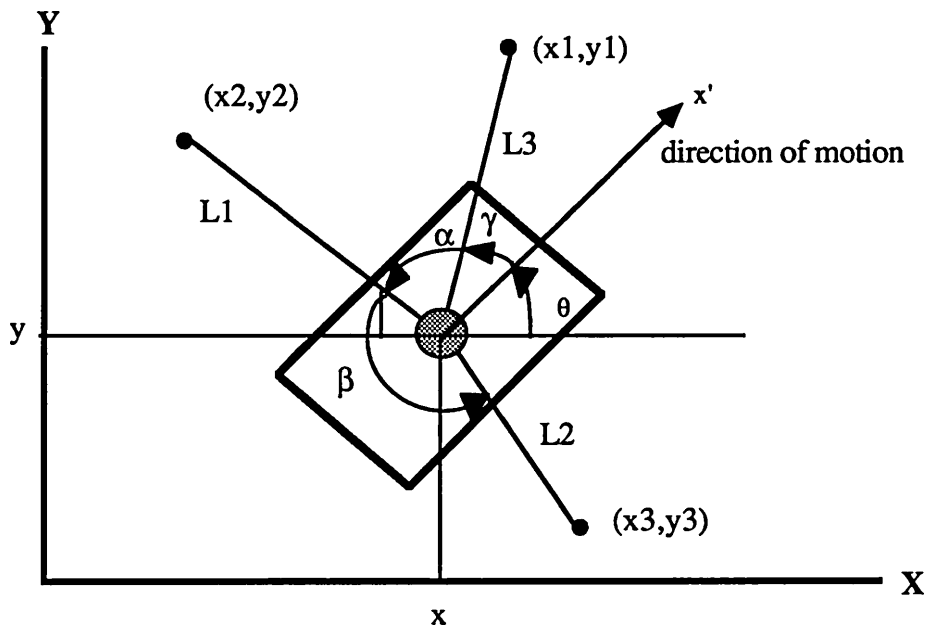


Figure 4.10 Laser Ranging

In figure 4.10, if the coordinates of three points are known (x_1, y_1) (x_2, y_2) and (x_3, y_3) and the angle γ , α and β determined by the vehicle, it is possible to calculate the coordinate of the vehicle (x, y, θ) .

$$\tan (\theta + \gamma) = (y_1 - y) / (x_1 - x)$$

$$\tan (\theta + \gamma + \alpha) = (y_2 - y) / (x_2 - x)$$

$$\tan (\theta + \gamma + \alpha + \beta) = (y_3 - y) / (x_3 - x)$$

Therefore, the coordinates x , y and θ are :

$$x = f(x_1, y_1, x_2, y_2, x_3, y_3, \alpha, \beta, \gamma)$$

$$y = f(x_1, y_1, x_2, y_2, x_3, y_3, \alpha, \beta, \gamma)$$

$$\theta = f(x_1, y_1, x_2, y_2, x_3, y_3, \alpha, \beta, \gamma)$$

[4.8]

4.5.2 Ultrasonic Ranging Systems

In this method, ultrasonic waves are bounced off objects. The time required for the echo from the object is noted, and knowing the speed of sound, the distance of the object from the ultrasonic sensor can be calculated.

$$d = s \Delta t / 2$$

where d = distance to object

s = speed of sound

Δt = time interval between sending signal and receiving echo

The speed of sound at 20°C is 343.2m/s. It varies only slightly with humidity (max. 0.35% at 20°C) and is virtually independent of pressure and, thus, of height above sea level. Only temperature has some influence : 331.3m/s at 0°C, 354.7m/s at 40°C; i.e., 7% variation from 0° to 40°. Variation with frequency is negligible.

$$s = 331.4 \sqrt{(T/273)} \text{ m/sec}$$

where s = speed of sound in air

T = Temperature in degrees Kelvin

In general laser range finders have higher resolution as compared to ultrasonic or sonar range finders.

4.6 Summary of Guidance Techniques

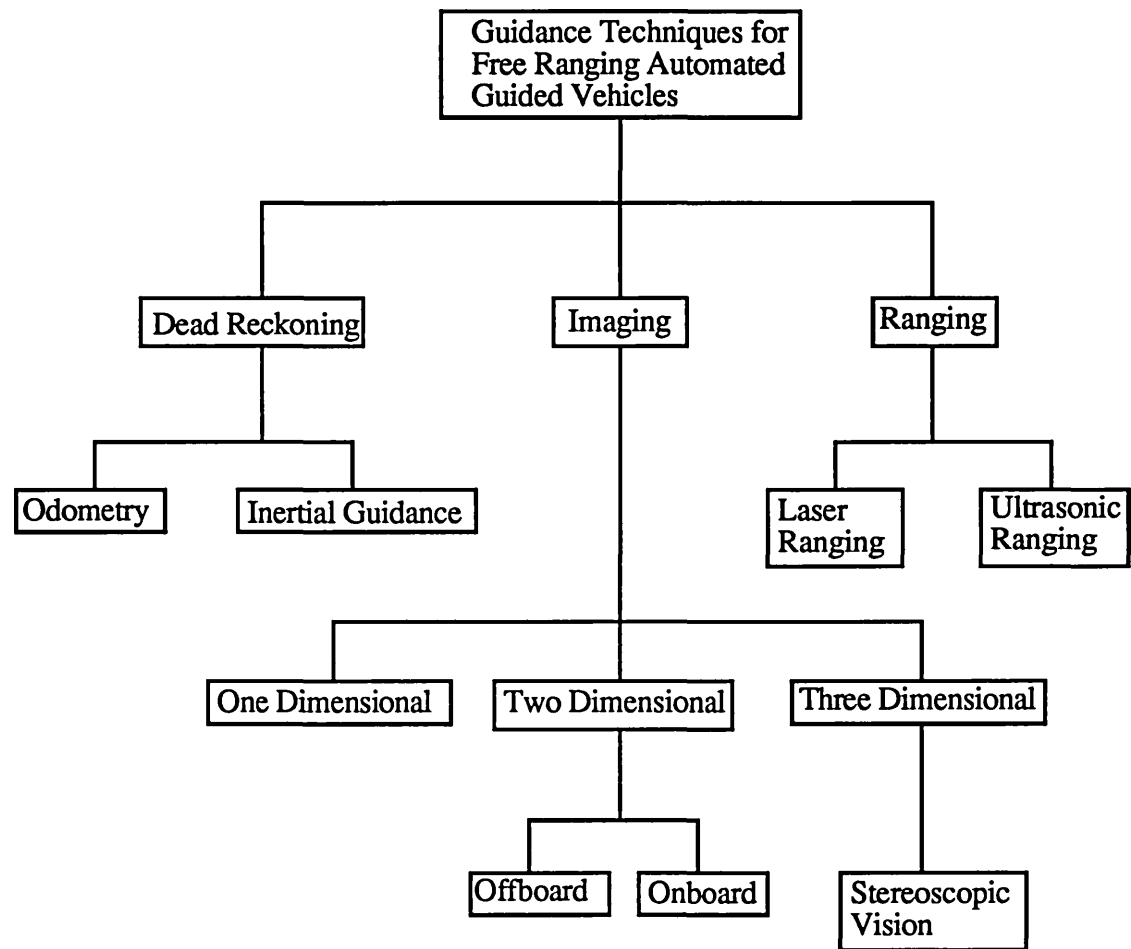


Figure 4.11 Guidance Options for Free Ranging AGVs

Figure 4.11 shows a block diagram of the various guidance techniques available for free ranging AGVs. Most free ranging vehicles use a combination of methods to guide it from one location to another.

4.7 ICAGV Guidance System

For a free ranging vehicle, a single guidance system does not ensure reliability of the system. Onboard methods such as odometry require ideal conditions for satisfactory performance, failing which they suffer from cumulative errors. Encoders on the wheels is the minimum requirement of any free ranging vehicle. Dead reckoning is the easiest and cheapest onboard guidance system to implement. If the starting posture of the vehicle is known, an estimation of the instantaneous vehicle position and heading can be obtained. This feature is necessary for any off-line programming of the vehicle.

The main error in the dead reckoning system is the heading and therefore lateral error of the vehicle with respect to world coordinates. Another system which looks at the real world is required to update the odometric calculations at regular intervals.

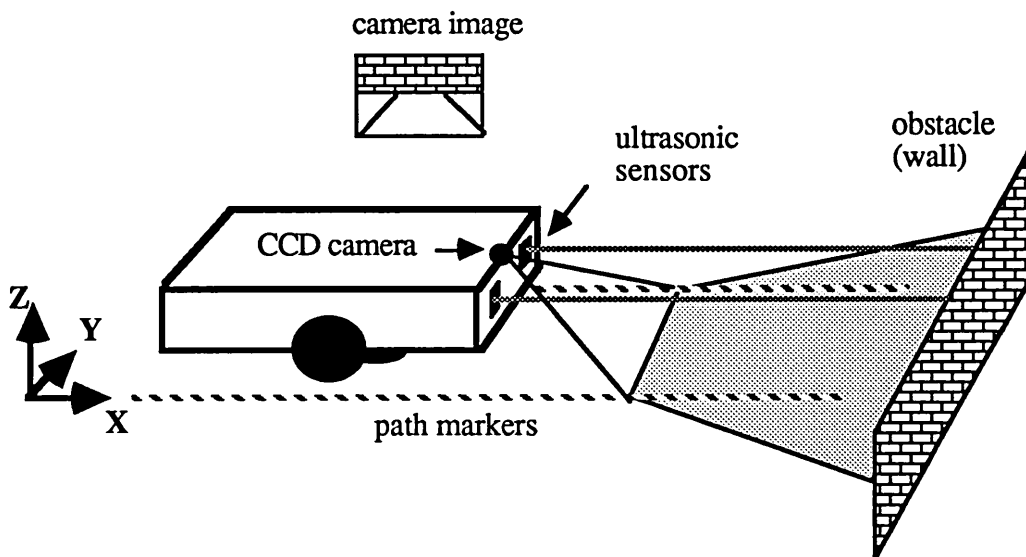


Figure 4.13 ICAGV-II Guidance System

For real world sensing, a CCD camera on the vehicle, looking down on the factory floor was chosen (Figure 4.13). By looking down at the floor, the scene has been restricted to two dimensions. Of all the guidance sensors available for free ranging AGVs, this is the most flexible. Since all the necessary hardware is on the vehicle, the method is not dependent on the size of the fleet as in the overhead camera method (Sec.4.4.2.1). The one dimensional line scan camera is a simplified version of this method. If only one horizontal

line of the image is analysed, our two dimensional situation becomes the same as the line scan situation. By restricting the scene to the two dimensional floor, information on distances to target markers, such as docking markers, can be calculated from the image. We elaborate on the vision system for our vehicle in the next chapter (chapter 5)

Ultrasonic ranging is used for obstacle detection. Two "Polaroid" ultrasonic sensors are mounted in front of the vehicle such that the emitted wave covers the path of the vehicle. If the detected obstacle is within two meters of the front of the vehicle, the vehicle should slow down. If it is within one meter, the vehicle should stop. The obstacle detection system of ICAGV-II is covered in chapter 6.

To summarise, the guidance system of ICAGV-II comprises of :

1. An environment map, created using AutoCad which has details of the vehicle network, the vehicle paths, position markers, docking nodes, intersections, junctions, go-no go areas etc.
2. Dead reckoning system based on odometry to follow the route specified on the map.
3. CCD camera (256pixels x 256pixels) to detect markers in the environment .
 - path or lane markers to reduce lateral drift of the vehicle during long straight journeys
 - position markers for real world coordinate updates and
 - docking or datum markers for precise motion control.
 - initial calibration of odometry constants
4. Ultrasonic sensors to detect obstacles in front of the vehicle.

As the vehicle is always following the software map using odometry, occasional lapses in the readings from the real world sensor (camera) is not very serious. It is reasonable to assume that large errors between actual and desired position implies that the sensor information is suspect. Subsequent readings give an indication of whether action should be taken or not. The various tasks undertaken by the vision system are triggered by software. From its map the vehicle knows when to expect markers, so detection of a marker in a particular area is a double confirmation of its presence. Adding features in the software map is relatively easy, as the industry standard drafting package AutoCAD is used. This approach ensures that the guidance method for the vehicle is robust.

CHAPTER 5

VISION SYSTEM

5.1 Introduction

Ideally, with an environment map and dead reckoning sensors, a free ranging vehicle can travel from a start to a destination point on the map. However, any fluctuation in the vehicle's odometer parameters or the environment's surface condition can result in errors and these errors can become appreciable if not corrected. For updating dead reckoning readings in the Imperial College Free Ranging AGV, a vision system is used.

Nearly all animals use vision as a short distance navigational aid. R.A.Jarvis [50] gives a comprehensive review of vision as a range finding tool. Our depth cues are received from various sources. Examples of which are : binocular convergence, stereo disparity, texture gradient, linear motion perspective, size constancy, brightness variation, aerial perspective (haziness of more distant objects), vertical position in the visual field (more distant objects are more likely to be higher in the visual field), occlusion cues and outline continuity (complete objects look closer).

For computer vision, the amount of information to be processed has to be reduced for acceptable processing speeds. With the advent of parallel processing (eg. Transputers) near real time speeds appear possible. The following methods have been used in the past by sequential systems for obtaining range information : striped lighting methods (contrived lighting to achieve triangulation based ranging), texture gradient method, relative range from occlusion cues, depth from focusing information, extracting range from time sequence imagery with the camera moving through the scene (optical flow), exploration of brightness variations for estimating surface orientation and computational methods for extracting stereo disparity and range from pair of images.

All these methods have their limitations. They are either :

- computational too complex,
- time consuming in improvement of the signal to noise ratio,
- are limited to indoor applications,
- are limited to highly textured or line structured scenes,

- have limited surface orientation or
- have limited spatial resolution.

The conclusions reached by Jarvis were as follows :

- Sheer simplicity - triangulation schemes involving one point at a time
- Greatest potential - laser time of flight could in theory eliminate all the above limitations provided an intense enough source is used.
- Anthropomorphically - stereo disparity, occlusion, photometric and texture gradient methods.

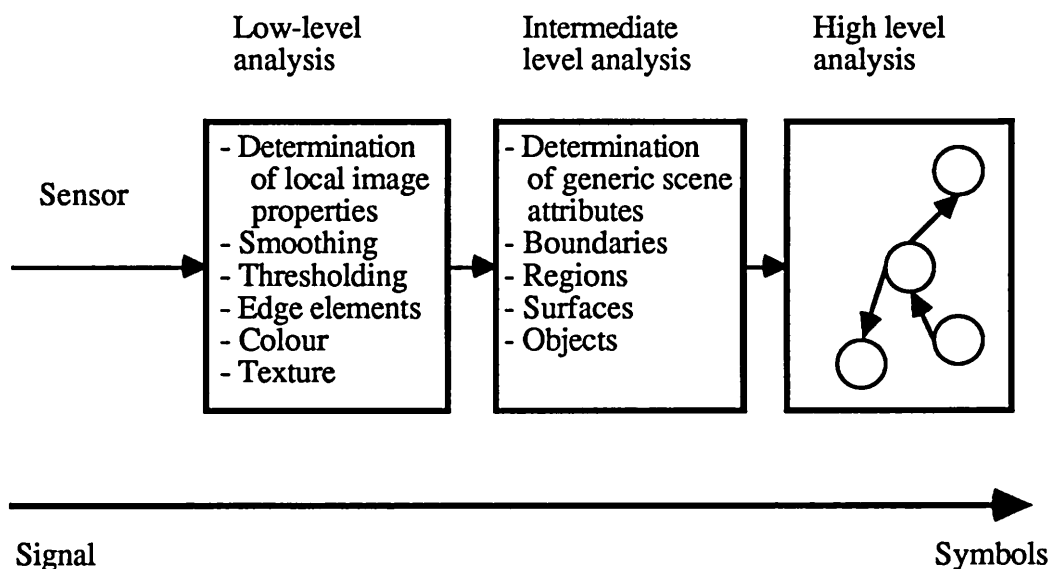


Figure 5.1 : Low level, Intermediate level and High level Image analysis [57]

Figure 5.1 shows the flow of vision analysis. Image analysis can be divided into low level, intermediate level and high level. The levels progress from raw sensor data to abstract symbols.

For a mobile platform, the issue of computer processing speed is very critical. The Caterpillar/GEC Self Guided System incorporates a low powered laser triangulation based system for updating odometer readings. The advantage of this system is that range information is obtained directly. The disadvantage is its expense and it could be a safety

hazard as the rotating laser is 2m above floor level (ie eyelevel). For flexibility a vision image based system is best. Complexity can be avoided by using one camera pointing downwards, looking for markers on the two dimensional floor. Depending on the speed of the image processing system available, the markers could be single spots or a complex pattern. For a spot, the image analysis is the simple triangulation scheme. As only one image is involved there is no disparity problem. This method is superior to the laser triangulation method because in addition to range finding, the same system can be used for a number of other functions, such as, marker location, path following and obstacle detection. All these functions can be done without additional hardware.

Figure 5.2 gives an example of the type of markers that have to be detected by the vision system - path markers and position markers.

In this chapter, the ICAGV-II Vision System is discussed. We begin with the difficulties encountered in extracting a usable floor marker or path guide in normal lighting condition. Then, proceed to converting the coordinates of the marker to real world coordinates - this involves calibration of the camera system. We show how the coordinates of the marker are used for correction of odometric errors. Finally, we describe the hardware implementation of our vision system.

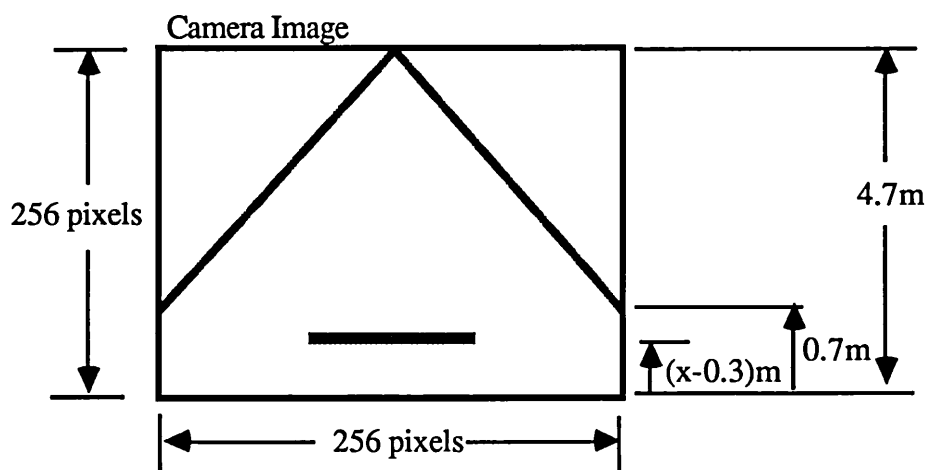
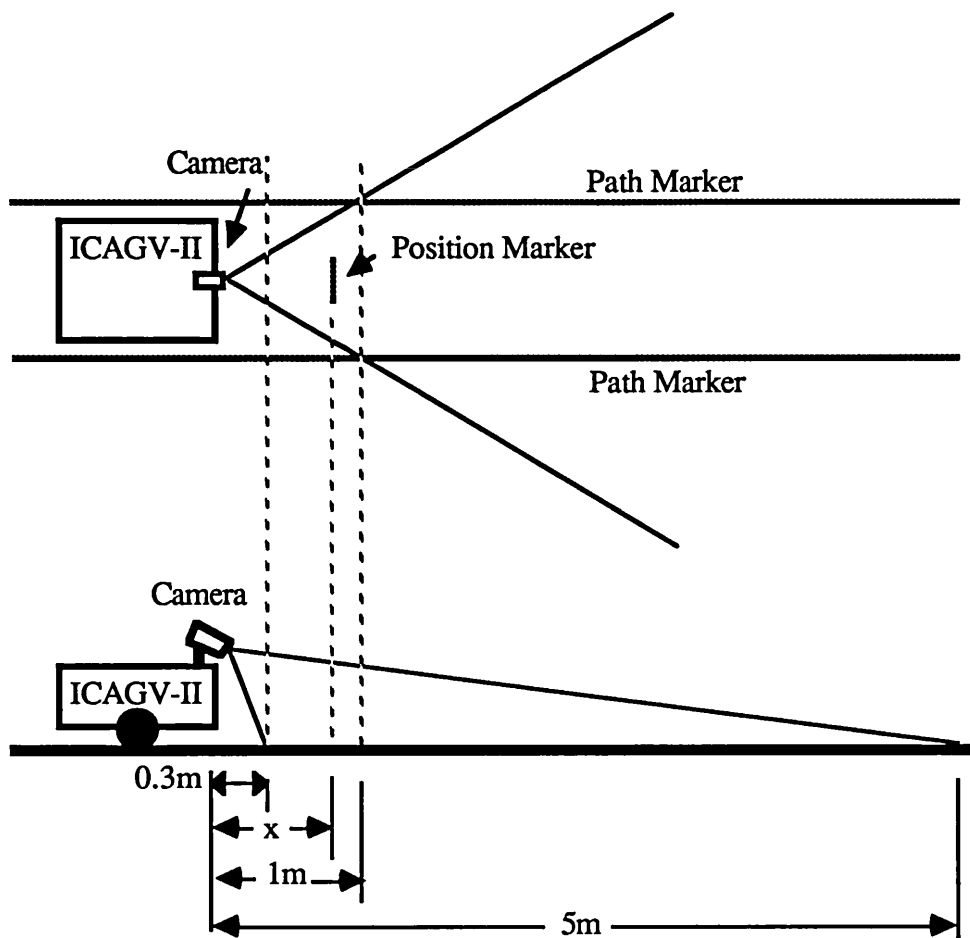


Figure 5.2 Path Markers and Position Markers

5.2 Low Level Image Processing

The first stage of the vision process is the extraction of relevant floor markers from the image. This is made difficult by the complex interaction of effects such as :

1. illumination
2. surface reflectance and
3. surface orientation.

5.2.1 Illumination

A major drawback of present day vision systems is the amount of light needed for the system to operate satisfactorily. One method of solving the illumination problem is to have a light source shining on the viewed area. This light source must function in the presence of ambient lighting - well lit or dark environment as well as sunlight. This would require a powerful light source on the vehicle. However, a powerful light source would be an unnecessary drain on the limited battery resources of the vehicle. To try and solve the ambient light problem, wavelengths in the infrared and ultraviolet have been suggested [52] - infrared for indoor applications and ultraviolet for outdoors. Unfortunately, outdoors, there is too much infrared to be effective whilst indoors, ultraviolet could prove dangerous in confined spaces.

In a structured environment, such as a factory, safety regulations dictate that pathways for vehicles be clearly marked by high contrast paint (Figure 5.2). This clearly defined pathway simplifies the problem of detection considerably. The high contrast paint ensures that the ambient light available in the factory environment is sufficient to detect the pathway markers.

A camera with an Auto-Iris lens would solve the problem of fluctuation of light level during the day.

The algorithm used in our system for detecting the markers is to a certain extent independent of the ambient light condition. The line of interest in the image is first analysed for pixels with the greatest intensity. The "threshold" for that line is then set at 90% of that value. As long as the difference of intensity between the marker and the surrounding area is greater than 10%, the mark should be detected. Also by analysing the intensity linewise,

we have avoided the problem of finding an optimum intensity level for the whole image. Since the areas to the bottom of the image are closest to the vehicle, the importance of readings vary according to their position in the image. Readings further away are more likely to have intensity problems.

5.2.2 Surface Reflectance

As opposed to insufficient light, the markers may be swamped by too much light. This is the situation if the floor surface is shiny and sunlight from a nearby window is reflected onto the camera lens. This is especially so with monochrome cameras where intensity is the only parameter being processed by the camera. Colour information could be used as another parameter to sort out this problem.

For normal factory flooring, surface reflectance may not be a major problem. Areas of the image with surface reflectance patches can be eliminated as the width of the pathway and markers are known (Figure 5.3). If high intensity patches much greater than the width of the pathway or marker is detected, the intensity patch is rejected. In this way surface reflectance patches may be detected and not considered as valid readings. The vision system is the real world feedback sensor for the vehicle and is meant to "verify" odometer readings. This means that occasional "suspect" readings obtained from the vision system can be rejected without seriously affecting the motion of the vehicle.

Since the sensor is effectively saturated by the surface reflectance patches, it is not possible to get any information from that portion of the image. The robustness of our guidance system allows the vehicle to reject any update reading that it suspects to be incorrect.

It may be possible to use polarised filters to cut out the surface reflections or glare.

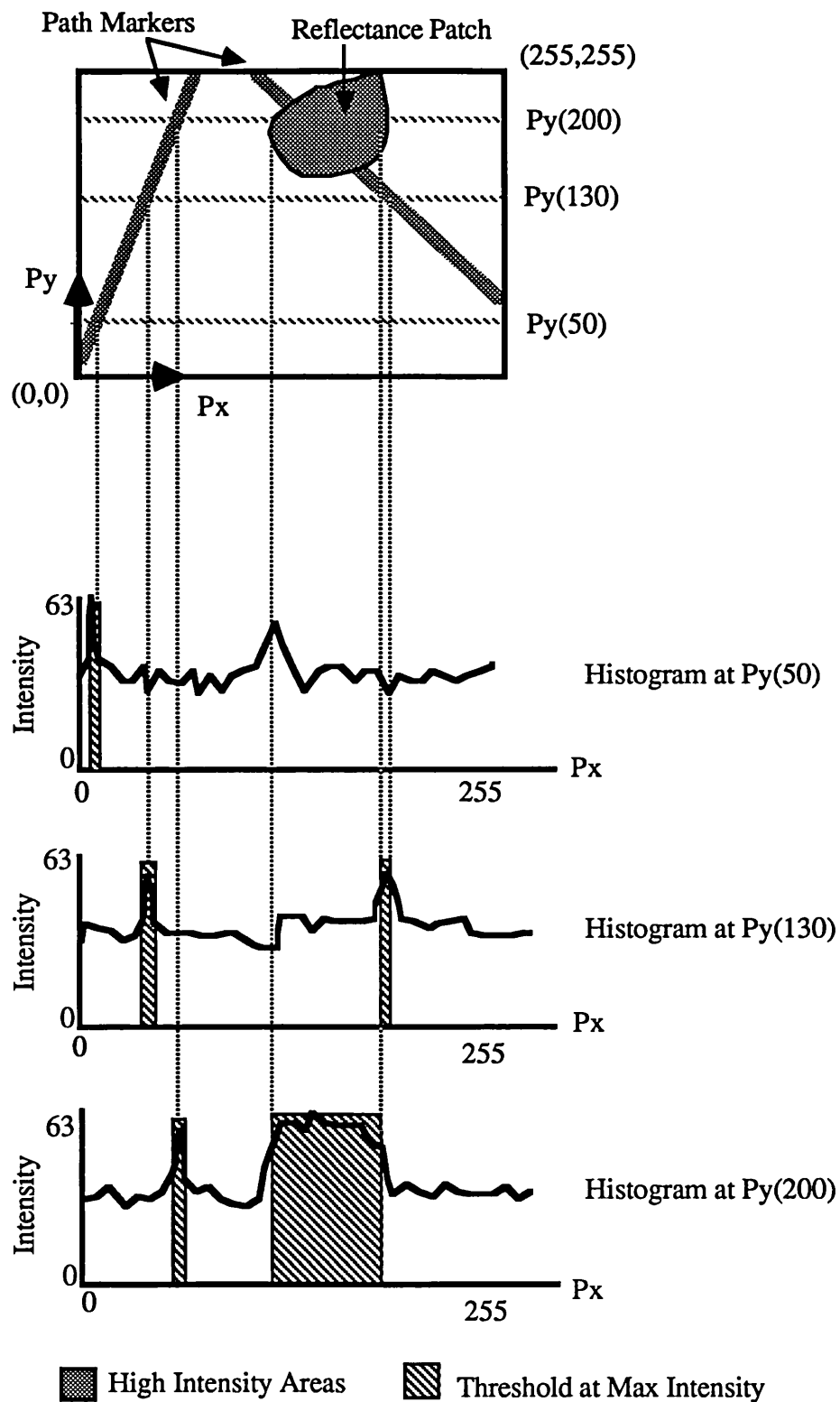
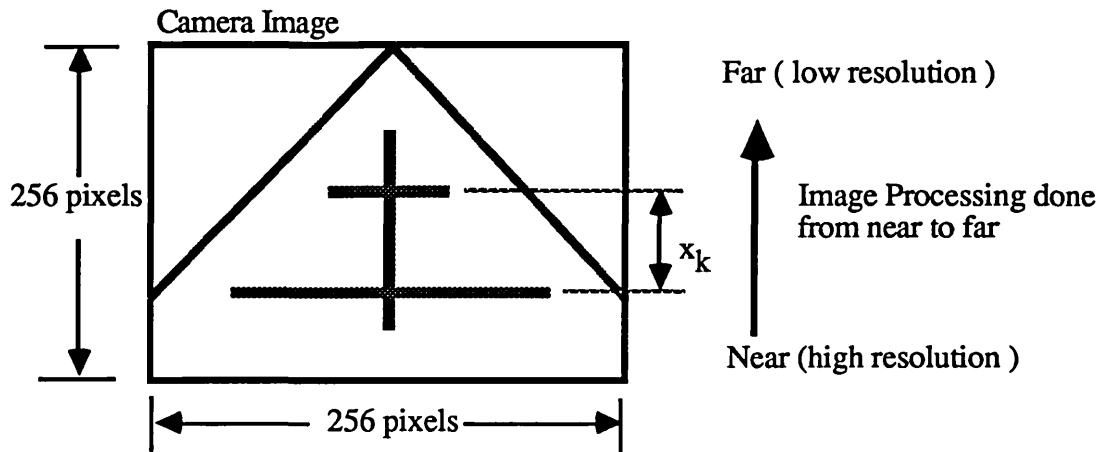


Figure 5.3 "Max" Threshold and Surface Reflectance Patch

5.3 Calibration



X_k = Known distance

Figure 5.4 Calibration Marker

Once the low level vision problem of extracting the pathway markers from the other spurious high intensity patches is solved, the next problem is to get the actual coordinates of the marker with respect to the image and so with respect to the real world. For this purpose the image space has to be calibrated with respect to the real world (ie. the bottom of the image - near point n meters, top of the image - far point f meters) must be known with respect to the centre of the vehicle. This is determined by the angle of inclination of the camera and the lens selected. The image width is fixed by the lens.

There are two elements involved in the calibration of the vision system.

1. Calibration of the Camera with respect to the vehicle
2. Calibration of the Camera with respect to the real world.

For calibrating the camera with respect to the vehicle it is necessary for the camera to focus on to something on the vehicle and the camera adjusted until the object on the vehicle is always at a particular position in the image space. An error in the calibration of the camera will result in an offset error in the image. If the above method of calibration is not possible, it is then necessary to determine the angular error by positioning the vehicle in a known location and then calibrating the camera to a known pattern on the floor. This method can only be verified if the vehicle moves forward a small distance, any angular error will result in a drifting of the line. It may be necessary to fine tune the calibration to ensure that there

is no drift. This calibration ensures that the camera axis coincides with the axis of the vehicle.

The calibration of the camera with respect to the only real world can now be attempted. This comprises of adjusting the tilt of the camera to cover a known area in the real world. The obvious method of achieving this is to once again place the vehicle in a known "calibration" area which has a pattern on the floor (as indicated in the Figure 5.5). It is desirable that the camera should be able to see a scene from 1m to 5m ahead. This preselects the lens to be used. We note that the height of the camera with respect to the floor is also involved in the equation (Equation 5.1). The two horizontal lines are a known distance away from the centre of the vehicle. The tilt of the camera will have to be adjusted so that the line passing through the 'x' axis of the vehicle is at the centre of the image (calibrated earlier) and the two horizontal lines at a preassigned position in the image space (eg 10 lines from the top and 10 lines from the bottom of the image).

Camera calibration is extremely important as the camera is the real world feedback sensor for the vehicle. Consistent false readings will result in the vehicle being in a different posture to which it should be. The camera needs to be calibrated only during the initial setup of the system.

5.4 Conversion from Image to Vehicle Coordinates

Figure 5.5 shows a camera looking down at a point on the floor. The distance of the point from the camera is d and the lateral distance of the point from the vehicle is l . With respect to the vehicle axis (x',y') , the coordinates of the point on the floor is (x_c+d,l) .

$$d = h * \tan (\phi - \alpha)$$

$$\alpha = \tan^{-1}(y/f)$$

$$l = d * \tan \theta$$

$$\theta = \tan^{-1}(x/f) \quad [5.1]$$

where

ϕ = angle of tilt of the camera with respect to the vehicle

f = focal length of the lens

h = height of camera focal point with respect to the floor

$P_x = x$ coordinate of point in the image ($x = P_x * C_x$)

$P_y = y$ coordinate of point in the image ($y = P_y * C_y$)

C_x and C_y are the terms to convert pixel values to distance

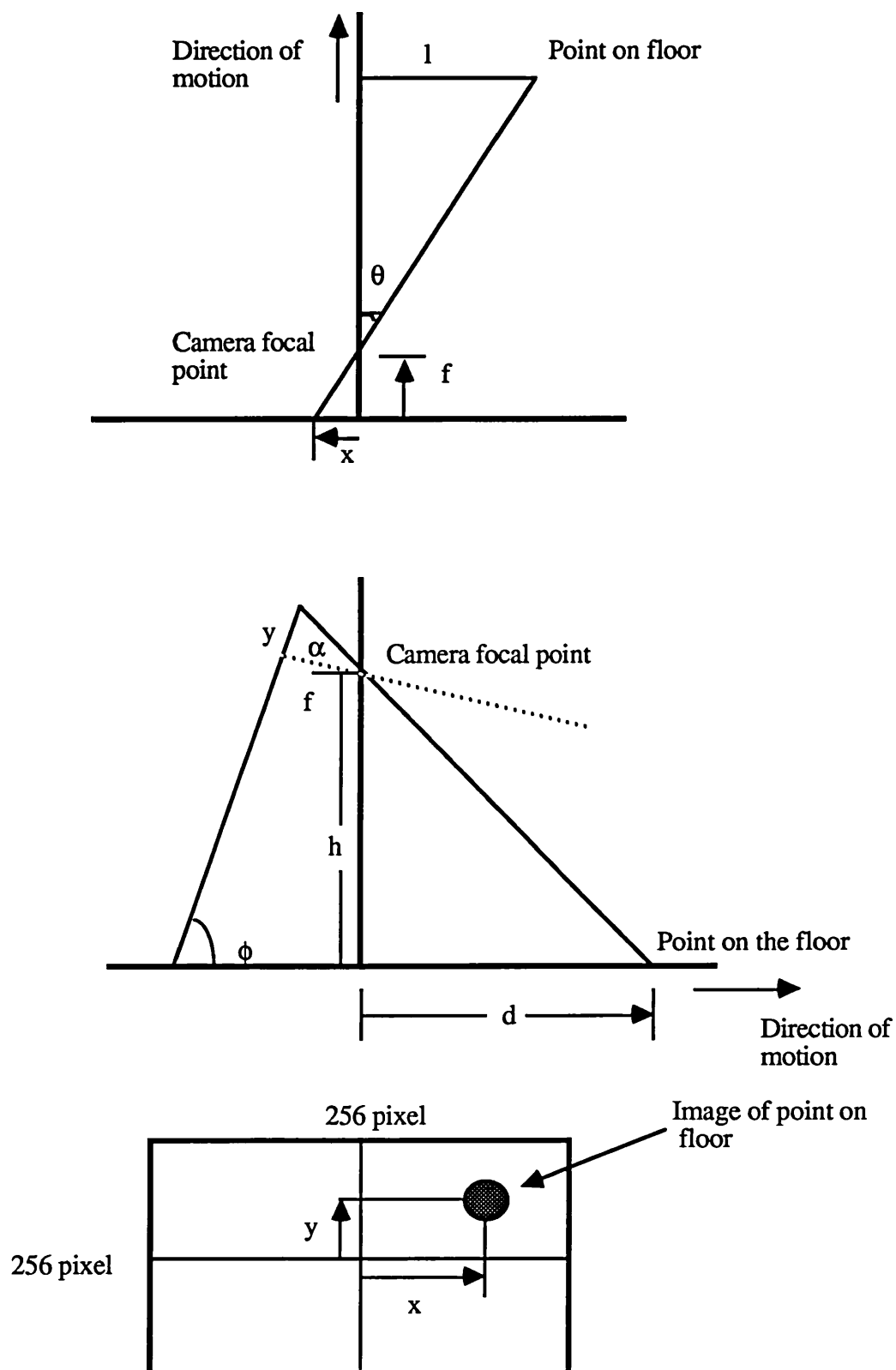


Figure 5.5 Coordinates of Point on Floor

5.5 Guidance using the vision system

The vision system is used to update the odometer readings and for navigation where real world markers or coordinates are important (eg datum position location, docking). The system also tracks the pathway to ensure that the vehicle does not drift away from its preassigned path.

5.5.1 Datuming or Docking

To fix the location of the vehicle in the real world there are three coordinates (x,y,θ) .

Before the vehicle begins on any journey its coordinates (x, y, θ) must be known.

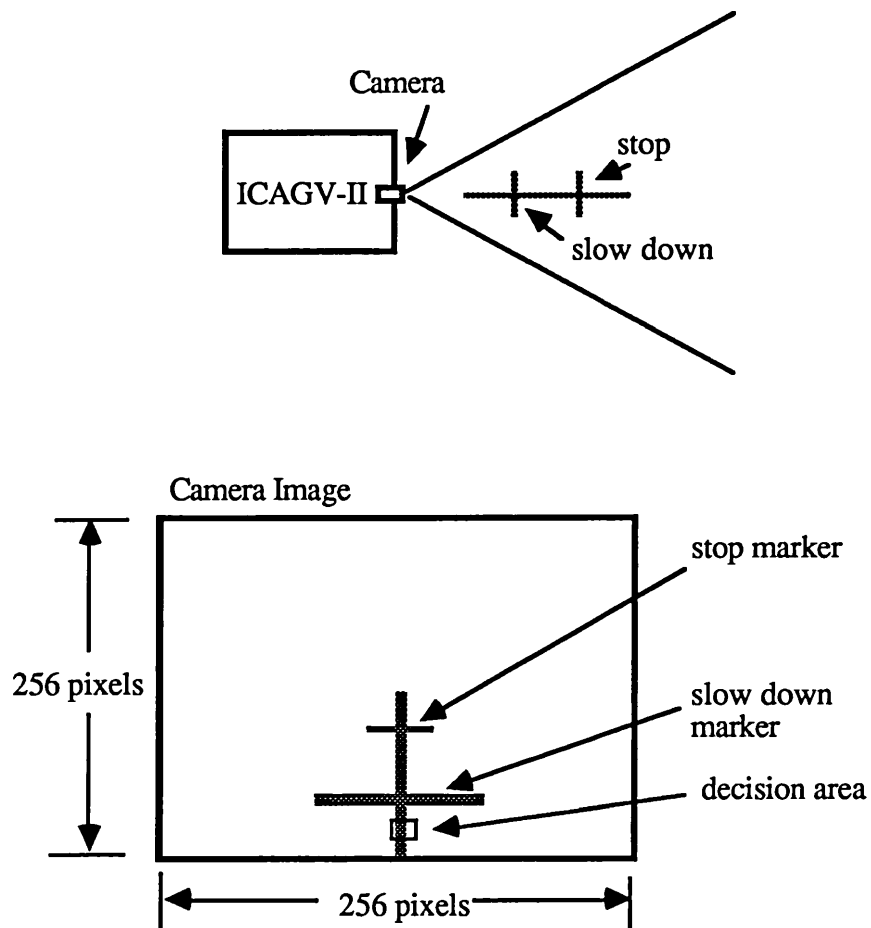


Figure 5.6 A simple docking marker.

The type of datuming or docking pattern is to a certain extent limited by the movements available to the vehicle. If the vehicle movements are confined to straight line motion and turns about its centre, the vehicle should first adjust its heading and then move forward until the line marker is in a particular position in the image. If the vehicle curves smoothly into the path, the process still involves the correction of heading and then the coordinates.

This implies (Figure 5.6) that datuming or docking markers should have a line for determining heading and a line for indicating when to stop. It may be desirable to have two lines, the first one to inform the vehicle that a docking marker is ahead and the second for stopping the vehicle.

5.5.2 Pathway Tracking

For tracking a single line of higher intensity than the surrounding background it is necessary to obtain the maximum intensity value of the horizontal lines on the screen. The average of the maximum intensity values will give us the centre of the line. As the line has a known mathematical relationship ($y=mx+c$) it is theoretically necessary to get two points at two different horizontal image lines to derive the equation of the line. However this is prone to error. To make sure that this is indeed the line, the gradient should be confirmed with another point and only if the gradients agree can the line be said to be found.

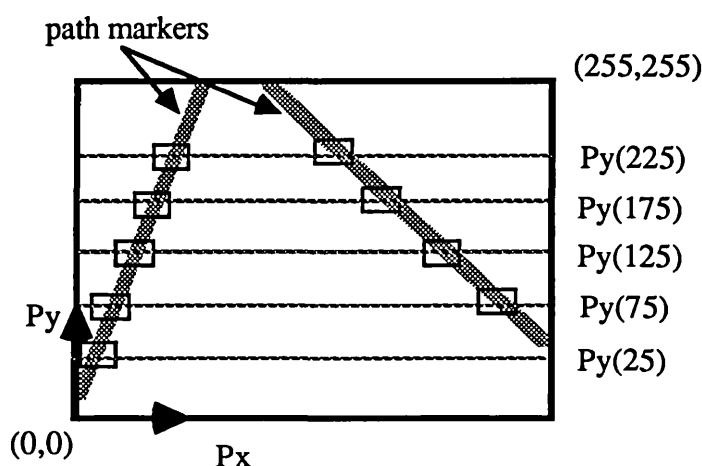


Figure 5.7 Pathway Tracking

For pathway tracking, the vehicle is moving and the speed of computation is essential. The maximum speed of the vehicle is 2m/s. A frame is captured by the vision system every

20ms which means that the vehicle would have moved 4cm within this time. As the vision system is actually looking ahead 1m to 5m from the vehicle, a point on a line 2m away and 50 frames ($2\text{m} / 4\text{cm}$) can be analysed before the vehicle reaches the point. This is absolutely necessary as the motion system requires a finite amount of time to react to any error it may see. Taking into account invalid points due to surface reflectance, the minimum number of horizontal lines to analyse is 5. If 3 out the 5 lines have the same gradient than the line is said to have been found. Since speed of computation is what is required, the more points that have to be analysed the greater the computation time. Note that once the high intensity point is found for a particular horizontal line on the image, it is not necessary to analyse the whole horizontal line for the next point. Due to continuity of the line, the next point should be within a few pixels of the previous value, so the next point is within a small window of the previous value. Only if the point is not found will the window be opened wider once again. This approach reduces the computation time even further. The heading and intercept are obtained by doing a "least squares" fit on the valid points.

Once the heading of the vehicle is correct, it is not necessary to analyse the five horizontal lines of the image. Since the vehicle is in motion, a single horizontal line could be sufficient to give an indication of whether the vehicle is diverting out of the path.

5.5.3 Update Odometry

To ensure that the distance moved is correct it is necessary for the vehicle to detect distance markers in the surroundings, via the vision system. Markers could be placed at regular intervals on the pathway. Everytime a marker is detected the vehicle knows that it has travelled a known distance. The preferred option is to have absolute markers. With absolute markers the vehicle does not have to keep tracking how far it had travelled since the last marker was detected. Also missing a marker does not confuse the system.

The vehicle gets an absolute position update when it docks. Additional absolute markers at intersection points could be introduced. Redundancy increases the reliability of the system.

5.6 Obstacle Detection

Figure 5.8 shows how it may be possible to detect obstacles using the vision system [66]. As no object should be in the pathway of the vehicle, any object detected in the path of the vehicle can be classified as an obstacle. However as the vision system is meant to look at the two dimensional floor, obstacle shapes and sizes cannot be determined. Stereoscopic vision will be required for three dimensional objects.

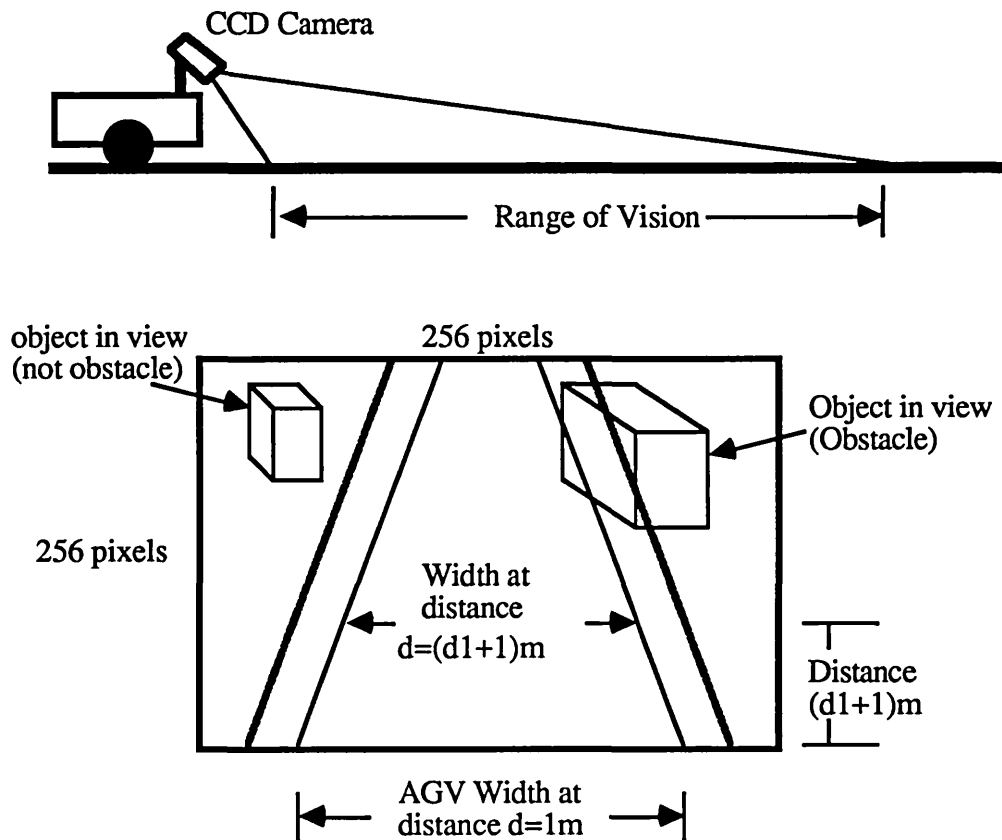


Figure 5.8 Obstacle Detection Using the Camera System

5.7 Implementation

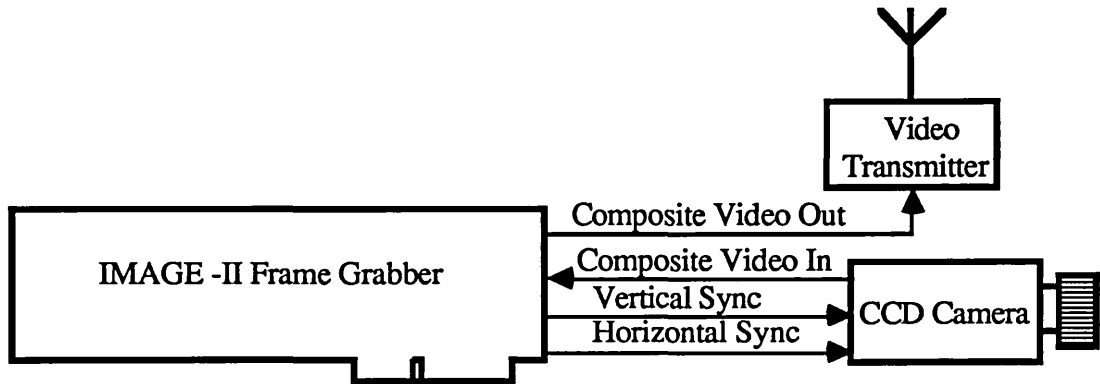


Figure 5.9

Figure 5.9 shows the hardware of our vision system. The system comprises of the following components :

1. IMAGE-II framegrabber
2. Hitachi CCD Camera Model VK-M98E
3. Ernitec Wide Angle lens, TV camera lens 1.2
4. Video Transmitter

The image processing card (IMAGE-II) from Eltime Vision is a lowcost frame grabber (256pixels x 256pixels) for the IBM PC. Each pixel can have 64 intensity levels. (ie. 6 bits). The video buffer is mapped to the IBM PC memory. The "live" image occupies memory location E0000 to EFFFF where E0000 is the top left corner of the image and EFFFF is the bottom right corner. Grabbed images occupy memory address D0000 to DFFFF, similarly D0000 is the top left corner and DFFFF the bottom right corner. Normally all image analysis is carried out on the grabbed image (D0000 to DFFFF). Complete details of the Image Processing Card (IMAGE-II) can be found in appendix A3.

As the resolution of the frame grabber is 256 x 256, no benefit is gained by having a high resolution CCD camera. If higher resolution is required, it is desirable that a controllable zoom lens is used. The Hitachi camera used has a resolution of 450H x 600V. The Ernitec

Wide Angle lens has autoiris control. This means that various lighting conditions can be catered for. A narrow angle lens was used for initial trials of the vision system.

For development purposes a video transmitter was connected to the output from the Image Processing Card. By adding this feature, the capabilities for applications such as surveillance, security, teleoperation are possible. This opens up the possibility of carrying out the image analysis on faster more powerful fixed computers

CHAPTER 6

OBSTACLE DETECTION

6.1 Introduction

In a perfectly structured environment, a primary guidance system using odometry and a secondary guidance/location system using vision is adequate to maneuver a free ranging vehicle from a source node to a destination node. However, for any practical application, the structured scenerio is usually perturbed by the presence of humans operating in the same environment. For any free ranging system to be acceptable, it must ensure that any obstacle detected (humans, material left in the path of the vehicle, another AGV etc.) must be avoided.

Much work has been done in developing algorithms for motion planning in the presence of obstacles. Basically they can be divided into two categories :

1. Where there is complete knowledge of the geometry of the vehicle and the obstacle (termed the Piano Mover Problems) - this situation could arise if there was a fixed camera looking down on the scene. In this case the obstacle size would be known and the central supervisory system could decide on an alternate path or a slight alteration of the existing path, so that the obstacle is avoided.
2. Where there is incomplete knowledge of the obstacle and this knowledge has to be complemented by sensory feedback - sensors on the vehicle would scan the obstacle, and knowing its current position and its target position, will have to feel itself around the obstacle. With incomplete information it is not possible to obtain an optimal solution and the aim is just to get a reasonable path or conclude that no path around the obstacle exists.

If space permits, the vehicle may be able to circumvent the obstacle. A very common and obvious method [61] is as follows :

Let us assume that the desired route from a source to a destination node is a straight line (Fig 6.1). This route is called the main line or M-line. If the vehicle detects an object obstructing this line, it must either move left or right to avoid it. The direction of turn would depend on the vehicles preferred direction of turn. The vehicle would walk around

the obstacle boundary until it meets the M-line again. It would then follow the M-line once again.

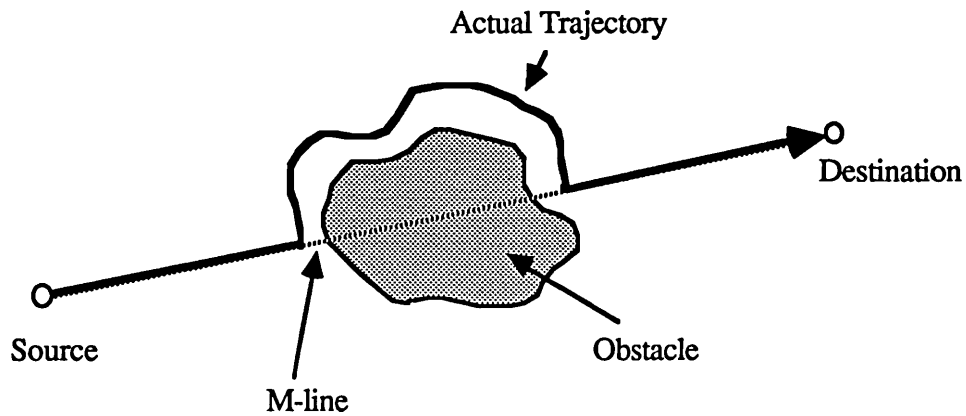


Figure 6.1 Sensor Based Obstacle Avoidance

To simplify the situation, the obstacle boundaries are expanded by shifting them parallel to the real boundaries by a distance equal to half the width of the vehicle plus a safety factor. This representation is called the "configuration space approach" and allows the vehicle to be considered shrunken to a point which may move on the expanded boundaries.

The simplest avoidance strategy would be just to "wait". If the obstacle were a human being it would most likely move away from the path of the vehicle and the vehicle could then carry on with its planned journey.

The main practical problem encountered in obstacle avoidance is the physical scanning of the surrounding area to detect the size and geometry of the obstacle. The minimum information needed is the distance (range) between the vehicle and the obstacle, so that the vehicle can initiate a "stop" procedure. Sensor based ranging is usually achieved by the following methods :

1. Stereoscopic Vision - this method is computationally complex being complicated by lighting and feature correspondence problems.
2. Infrared Range Finders - this method is ideal for range finding but can be too expensive for a low cost free ranging automated guided vehicle.
3. Ultrasonic Range Finders - this method is inexpensive (the Polaroid Ultrasonic Ranging System is approximately £20), however the resolution is not as good as the

Infrared Range Finders.

In this chapter, we describe the Ultrasonic Range Finding System used by the Imperial College Free Ranging Vehicle. We then proceed on to the hardware and software implementation of our obstacle detection system.

6.2 Ultrasonic Ranging Systems

Ultrasonic ranging systems in general have lower resolution compared to laser or infrared ranging systems. Their popularity is due to their simplicity, low cost and the fact that the distance measurements are provided directly. A sound wave (approx. 50KHz) is transmitted from the sensor and the time for the first reflected echo is noted. The range can be calculated from the simple equation shown below :

$$r = (v * t)/2$$

where r = distance to obstacle

v = velocity of sound

t = time interval sending and receiving reflected signal

6.2.1 Limitations of Ultrasonic Ranging Systems

There are fundamental drawbacks which make ultrasonic ranging unsuitable for applications that require high accuracies.

1. The velocity of sound varies very slightly with humidity and pressure but appreciably with ambient temperature and is given by the following equation.

$$v = 331.4\sqrt{(T/273)} \text{ m/s}$$

where v = speed of sound

T = temperature in degrees Kelvin

2. The validity of the range readings depends on the nature of the obstacle. Figure 6.2 shows three situations that may arise as the wave is bounced off an obstacle. If the

obstacle is perpendicular to the waveform emitted, most of the sound energy is reflected back to the sensor. However if the angle of incidence is greater than about 25° , only a small amount is reflected back and the obstacle could remain undetected.

Multiple surfaces could result in false range readings.

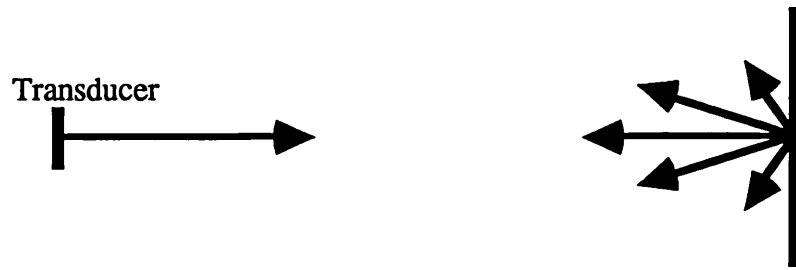
The amount of energy reflected also depends on the surface structure of the obstacle. To obtain a defused reflection, the reflecting surface of the obstacle should be much greater than the wavelength of the emitted sound signal.

$$\begin{aligned}\lambda &= v/f \\ &= 340 / 50000 \\ &= 6.8 \text{ mm}\end{aligned}$$

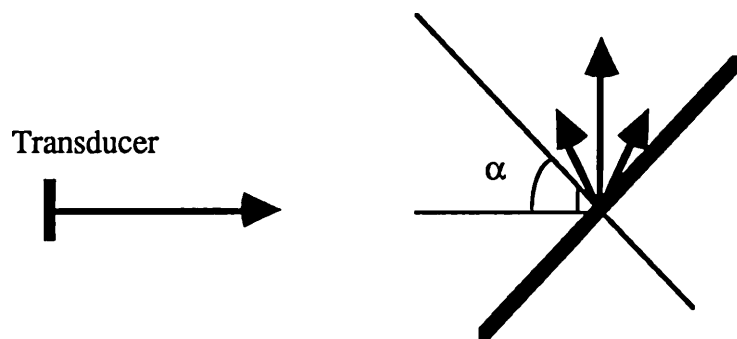
where λ = wavelength

v = velocity of sound at room temperature (340 m/s)

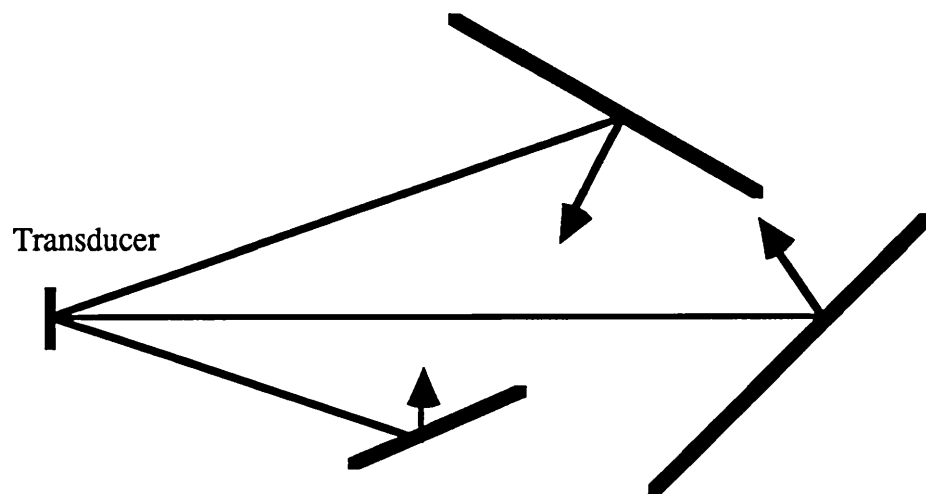
f = frequency of transmitted ultrasonic wave (50 KHz)



a) Reflections of sound waves from smooth surface perpendicular to acoustic axis



b) Reflected sound waves are not received by transducer when angle α is large



c) Reflection of waves from multiple surfaces

Figure 6.2 Problems faced by Ultrasonic Systems

3. The emission cone of the sound wave from the sensors (Polaroid) have an angle of about 20° - 30° . This causes uncertainty in the range measurement (Figure 6.3). The beam width can be narrowed by using devices such as acoustic lenses.

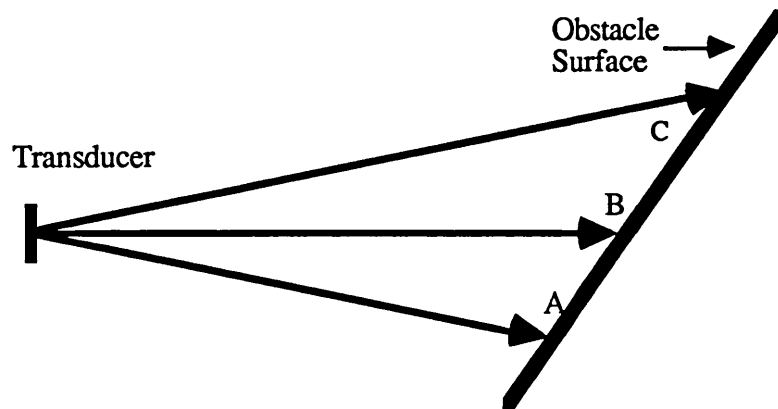


Figure 6.3 Uncertainty in range reading due to ultrasonic beam width

For a wide field of view, a number of narrow beamed transceivers can be used, each pointing in a different direction. Designs with 7, 14, 24, and 36 sensors have been built. To build a sonar map of the world multiple readings are required to reduce the probability of false readings. Elfes [59] using a 24 sensor system, categorised the range information into "probably empty" and "somewhere occupied" probability profiles and created a rasterized 2D horizontal map of the environment. Multiple views are taken to improve the accuracy of the map. The final map showed probably occupied, probably empty and unknown regions and dealt effectively with noisy data and the intrinsic limitation of the sensors.

6.2.2 Polaroid Ultrasonic Ranging System

Most researchers (including the Imperial College group) have used the Polaroid Ultrasonic Ranging System for implementing their system. The system comprises of the following modules:

1. Polaroid Instrument Grade Electrostatic Transducer
2. Ultrasonic Ranging Module - this module contains all the components necessary to generate the drive signal for the transducer, to control the timing functions, to receive, amplify and filter the return echo and to process the signal, providing a step function output at the time of the received echo.
3. Range Display Module - this module times the return echo and converts this to distance in feet and displays it on three multiplexed 7-segment displays.

The specifications of the system are as follows :

Measurement Range	- 0.9 ft to 35 ft
Nominal Resolution	- ± 0.12 inches to 10 ft $\pm 1\%$ over entire range
Beam Width	- 20° to 30° .
Polaroid Electrostatic Transducer	- 50KHz, 300V
Temperature Range	- 0° - 40°C

The lower limit of 0.9 ft is due to the recovery time of the transducer.

6.3 Implementation

The obstacle detection system of the Imperial College Free Ranging Vehicle was designed to satisfy the following conditions :

1. If the obstacle is within 2m but greater than 1m in front of the intended travel path of the vehicle, the vehicle must slow down.
2. If the obstacle is within 1m the vehicle must immediately decelerate and stop.
3. If the obstacle is removed within a specified time limit, the vehicle should carry on with its previous time tabled journey.
4. If the obstacle is not removed within the specified time limit, the vehicle should request permission from the supervisory system to ensure that it is safe to carry on with its

previous time tabled journey or to follow a new route specified by the supervisory system.

Two Polaroid Ultrasonic Ranging Systems are mounted in front of the vehicle. From figure-6.4, we note that there are certain "blind spots" not covered by the ultrasonic beam (cone in the centre and two areas at the sides of the vehicle). The obstacle would most likely be detected at a distance and not appear suddenly in front of the vehicle. The position of the sensors are located such that the beam width just covers the path markers of the vehicle at a distance of 2m. Since the threshold of the range readings are set at 2m, any object detected beyond 2m (ie. not within the path markers) is not considered as an obstacle.

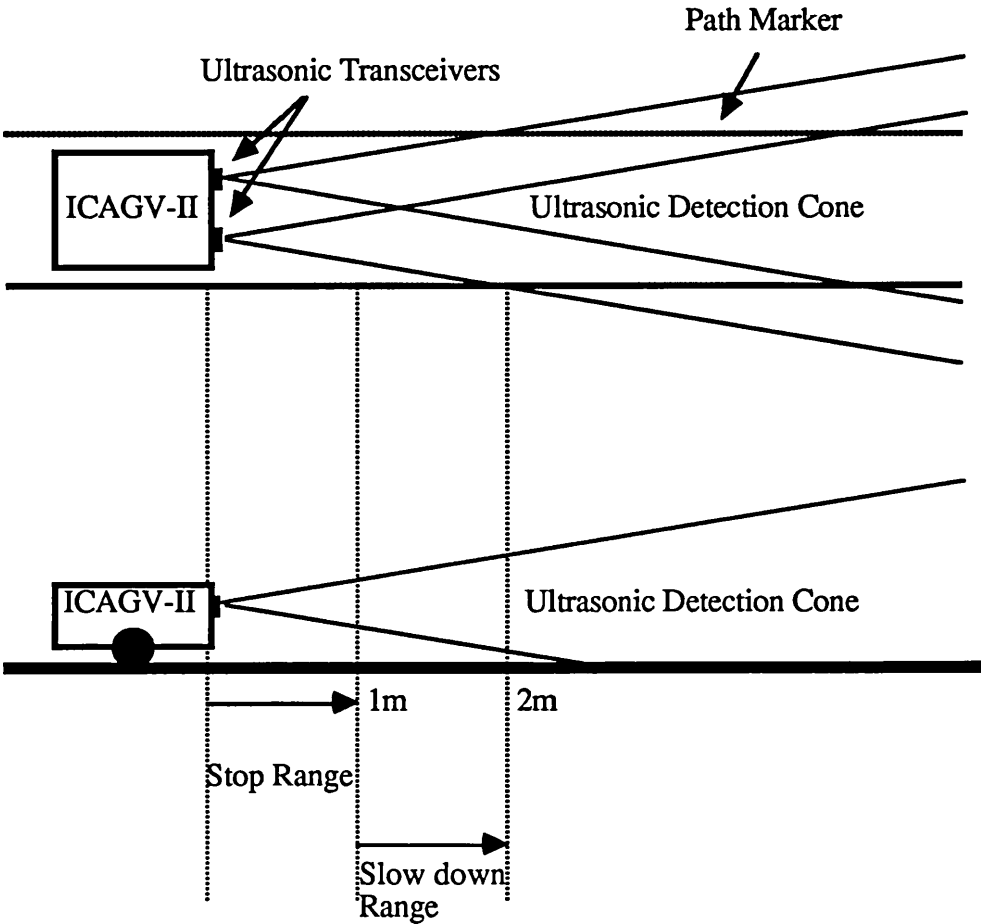


Figure 6.4 Requirements of ICAGV-II Ultrasonic System

6.3.1 Hardware

The range information obtained from the Ultrasonic Ranging System is interfaced to the IBM AT onboard the vehicle (Fig. 6.5).

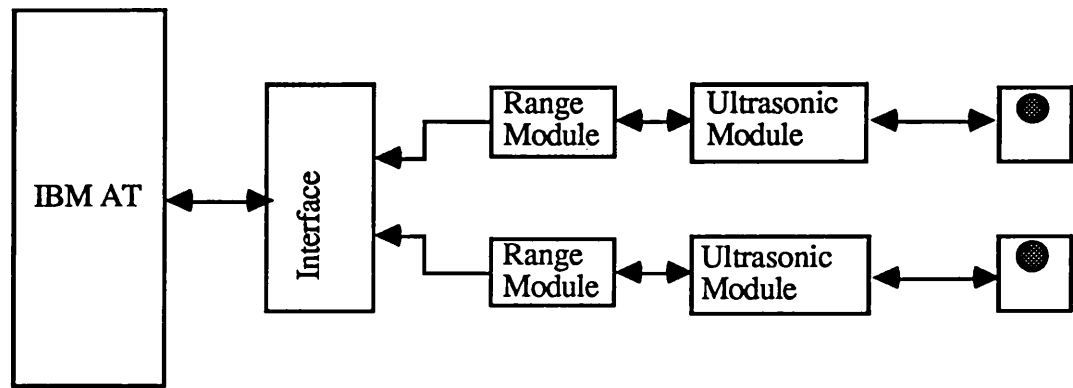


Figure 6.5 Block Diagram of Ultrasonic System

The interface card [60] converts the distance readings multiplexed on the 3 digit display of the "Ultrasonic Range module" to latched BCD numbers for the 8255 I/O card of the IBM AT. The card has the provision of sending an interrupt to the IBM AT if the range is below a presetable software range threshold.

Figure 6.6 shows the schematic of one channel of the interface card. The card has two channels for the two ultrasonic ranging units. It consists of the two sets of three 4 bit latches which provide the data necessary for the computer to read the binary coded decimal numbers as and when required. The circuit is designed in two halves, one half for each transducer, when one half is being read, latches of the other half remain in a high impedance state, thus avoiding a bus clash of data. When a data latch is being read by a computer, the read line prevents the second set of data latches from being updated and so avoid corruption of the data.

A comparator is used to compare the lower 8 bits of the range to detect the presence of objects within a preset value of the vehicle. The comparator carries out a comparison of data only when a full package of data is present in the first three 4 bit latches. As only the lower two digits are fed to the comparator, if an object is detected 10 ft or more from the

vehicle, the comparator is inhibited preventing a false interrupt from being sent to the computer.

An interrupt is sent to the computer when an object is within the preset distance from the AGV.

The design uses 14 connections of the 8255 I/O card to carry out its full operations, however an extended version of the interface using upto 8 transducers could be made requiring no extra I/O lines to be used. This could be done by adding a multiplexer to the read lines in the circuit and just duplicating the rest of the circuit.

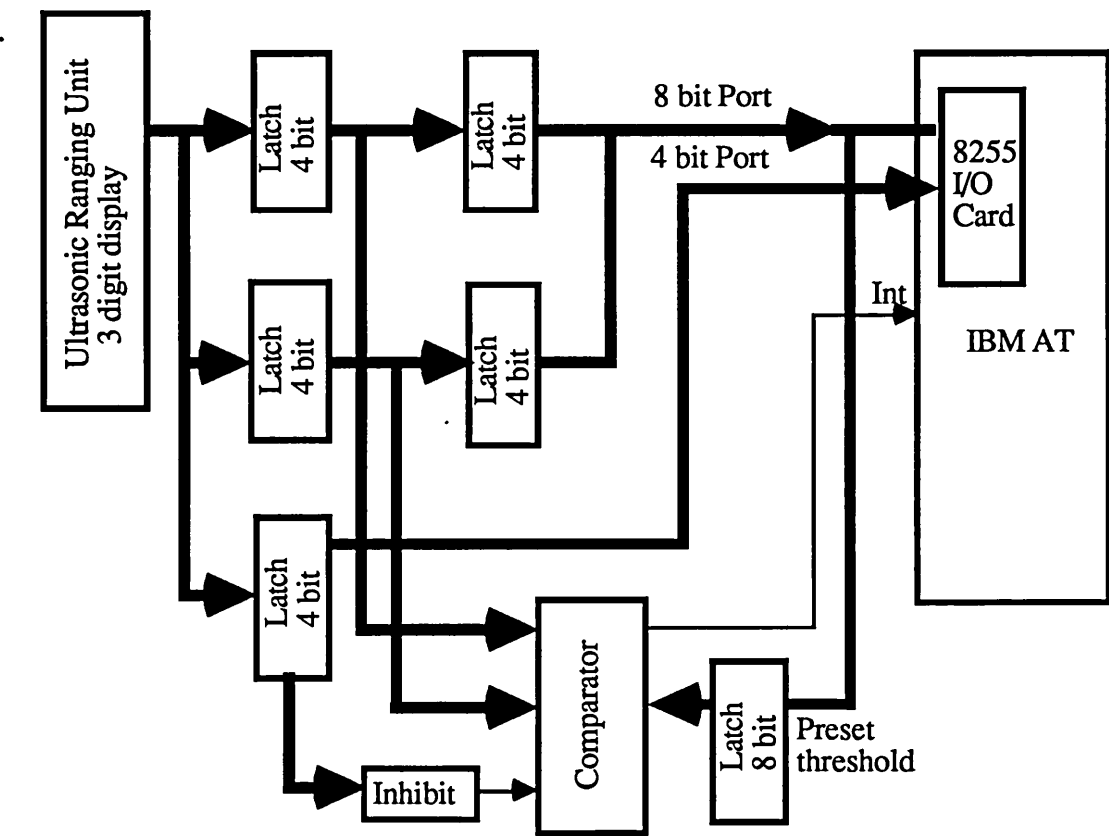


Figure 6.6 Schematic of One Channel of Ultrasonic Interface Card

CHAPTER 7

MOTION CONTROL

7.1 Introduction

Using odometry the vehicle has a dead reckoning method to follow a path on a map. Using the vision system, it can detect markers, comparing observed marker coordinate positions with its actual coordinates on the map. Using the ultrasonic detectors, any obstacle in the path of the vehicle can be detected and so avoided. Information from these three sensors must be utilised in the motion control system to ensure smooth, reliable and safe motion of the vehicle.

However before we start moving the vehicle, the decision on the actual path to be followed by the vehicle must be taken. There are two ways of specifying the path of the vehicle [37]. They are the continuous path (CP) method and the point to point (PTP) method. In the CP control method the vehicle moves on a continuous line whilst in the PTP control method, transient target points are specified. For the point to point method, complex paths can be executed by generating a series of transient target points. The continuous path method would theoretically produce smoother movement but computationally this method is complex, there are many variables involved in the control of the vehicle. Continuous path systems have been used only where physical guidepaths are present. It should be noted that in the limiting case when the target points are sufficiently small, the PTP method becomes an approximation of the CP method.

Most free ranging vehicles use the point to point method. The vehicle uses its current position and heading to determine its course to its target point.

In this chapter we begin by discussing the method of path specification for free ranging vehicles. We then proceed to the control strategies required to execute the path specified. Finally we come to the implementation of the motion control system of the Imperial College Automated Guided Vehicle. It will be shown that by integrating the odometric, vision and ultrasonic sensors and coupling it with our "in-house built motion control card", an extremely flexible and robust motion control system can be achieved.

7.2 Path Specification

The shortest path between two fixed points is a straight line. However due to restricted areas (walls, machines, etc.) the two points may have to be connected by a sequence of straight lines. Figure 7.1 shows possible straight line path sequences from a start point A to destination point B.

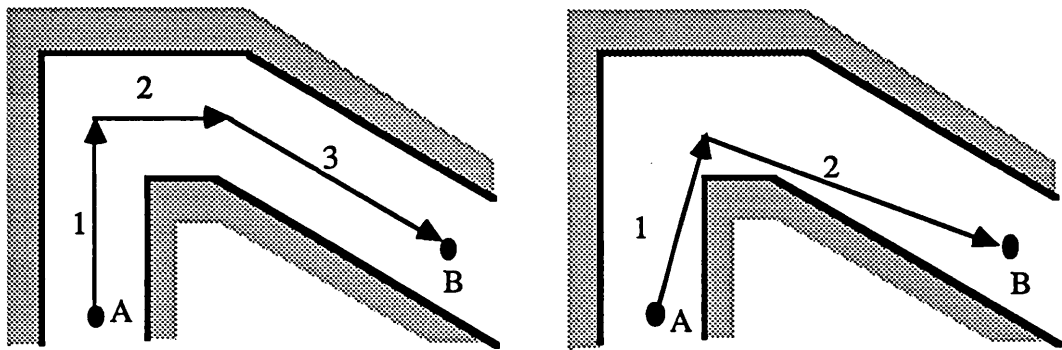


Figure 7.1 Path Sequence from A to B

To execute the path sequence shown, the motion control system of the vehicle should be able to :

1. Guide the vehicle precisely along any requested straight line segment (path following).
2. Guide the vehicle from one segment to the next segment (path succession).

7.2.1 Path Succession by Exit Point

Kanayama and Yuta [63] for their "Yamabico" vehicles, have specified paths by a sequence of straight lines (path following) and transient portions determined by a combination of user "exit" points and feedback control, for path succession. When a vehicle wants to take a new directed straight line X_1 , it specifies an exit E_1 on the old straight line X_0 , and allows the feedback system to track it on to the new line (Figure 7.2). The posture of the vehicle is determined by its coordinates (x,y) and its heading (θ) measured counterclockwise from the x axis.

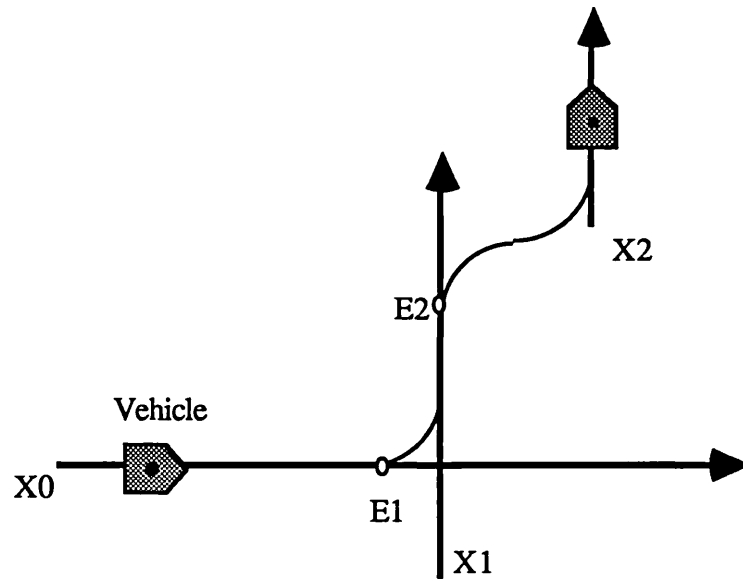


Figure 7.2 Path Succession by Exit Point

The feedback controller ensures that the vehicle travels along the x-axis in the positive direction with a given velocity v , satisfying the following equations :

$$\begin{aligned} y &= 0 \\ dy / dt &= 0 \\ dx / dt &= v \end{aligned} \quad [7.1]$$

Each directed straight line is a reference local path that the vehicle should follow momentarily.

7.2.2 Coordinate Transformation

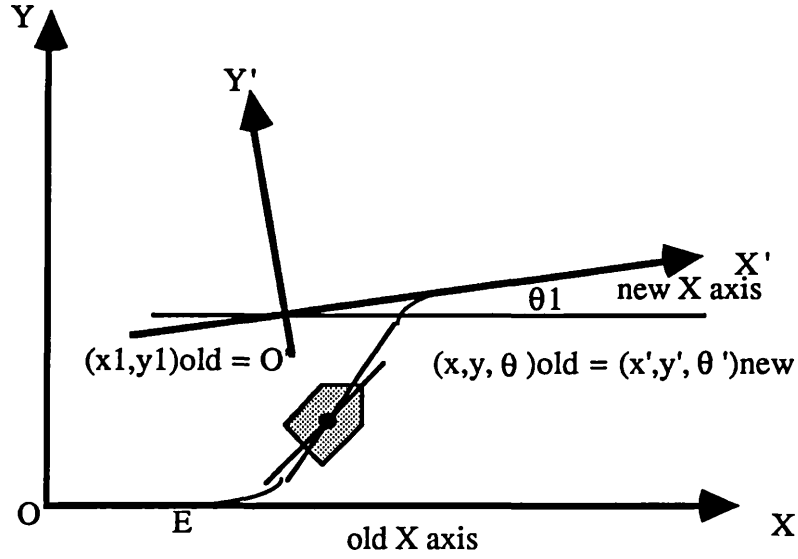


Figure 7.3 Change of Vehicle Coordinate System

Figure 7.3 shows a vehicle traveling with posture (x, y, θ) with respect to the XY coordinate system. Let the posture of the vehicle with respect to the $X'Y'$ coordinate system be (x', y', θ') . The relationship between the two coordinate system XY and $X'Y'$ is as follows :

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

$$\theta = \theta' + \theta_1$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

$$\theta' = \theta - \theta_1$$

[7.2]

In the method described by Kanayama and Yuta [63], at the exit point E, the vehicle suddenly updates its posture to the new coordinate system $X'Y'$. The feedback control

system of the vehicle detects an error and corrects the posture until it rides on the new X' axis.

The actual trajectory followed from the exit point E to the steady state value along the X' axis is dependent on the feedback control system. Proper exit points must be selected to ensure acceptable motion of the vehicle during path succession.

This method can also be used for correcting the posture of the vehicle if the local (vehicle) coordinate system and the real world coordinate systems do not agree.

7.2.3 Problems with path succession

The free ranging vehicle must travel from one line segment to the other without unacceptable deviation from the specified route.

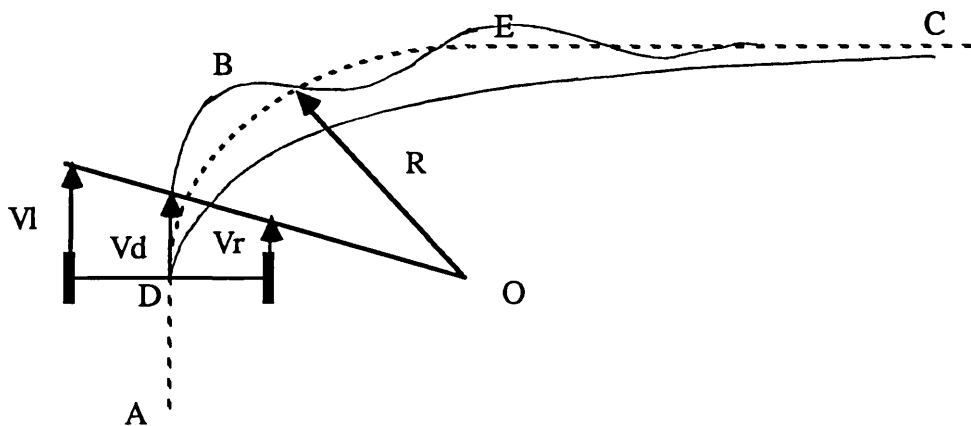


Figure 7.4 Desired and Actual Trajectory

In Figure 7.4 we note that the vehicle will have to travel along an arc DE to go from one line segment to the other. To execute this arc, the wheels will have to be given velocities V_r and V_l where the velocities are specified by the geometry of the arc.

Unfortunately, depending on the dynamics (loading of the vehicle) of the system, the actual

trajectory may be one of the alternate curves shown in the Figure 7.4. The actual final position E could be unacceptable for the specified route.

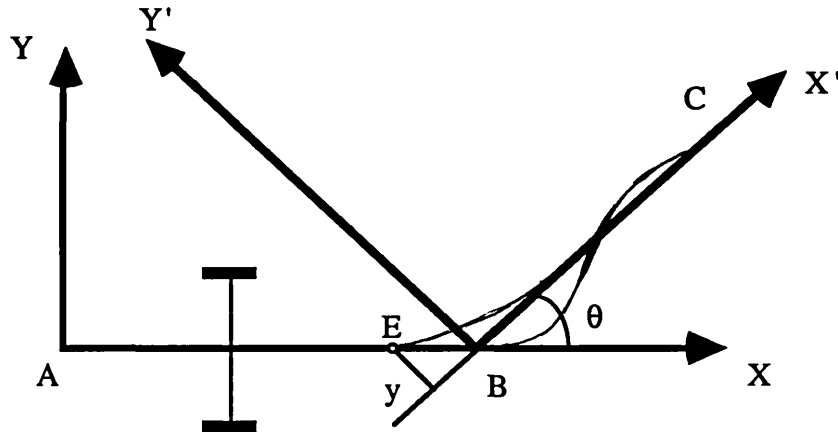


Figure 7.5 Critical Nature of Selection of Exit Point

In the exit point approach (Fig. 7.5), the selection of the exit point becomes critical and would depend on the size of the vehicle as well as the load carried by the vehicle.

7.2.4 ICAGV Path Specification

In a factory environment, the AGV cannot be left to follow arbitrary trajectories depending on its load. Due to space limitation, routes must be precisely defined and vehicles must follow the routes diligently. To optimise the usage of space, the area swept by the vehicle during path succession should not be excessive. The area swept out would depend on the minimum turning radius of the vehicle.

The Imperial College Automated Guided Vehicle with its wheels equidistant from the front and rear of the vehicle, can turn around its geometric centre. Using this feature, the vehicle can stop at one line segment, turn around its centre θ degrees and then follow the next line segment (Fig. 7.6).

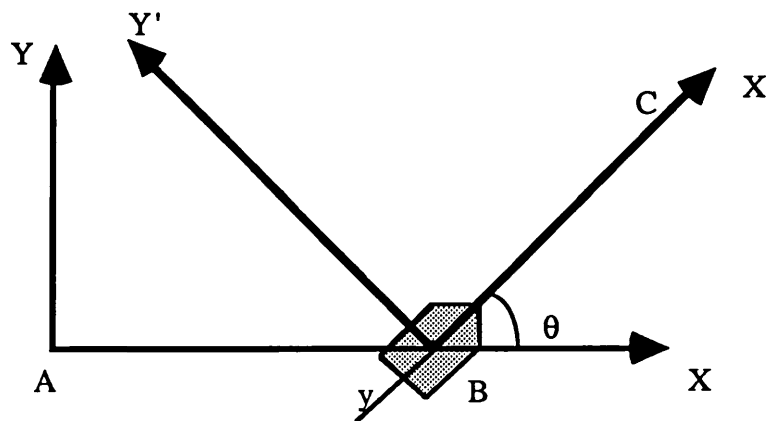


Figure 7.6 Turn Around Centre

7.2.4.1 Turning Around Centre

If the two wheels were controlled such that one wheel moved clockwise and the other anticlockwise with the same velocity, the vehicle would turn around its centre. Typically the velocity profile would comprise of an acceleration portion, a constant velocity portion and a deceleration portion, as illustrated in Figure 7.7.

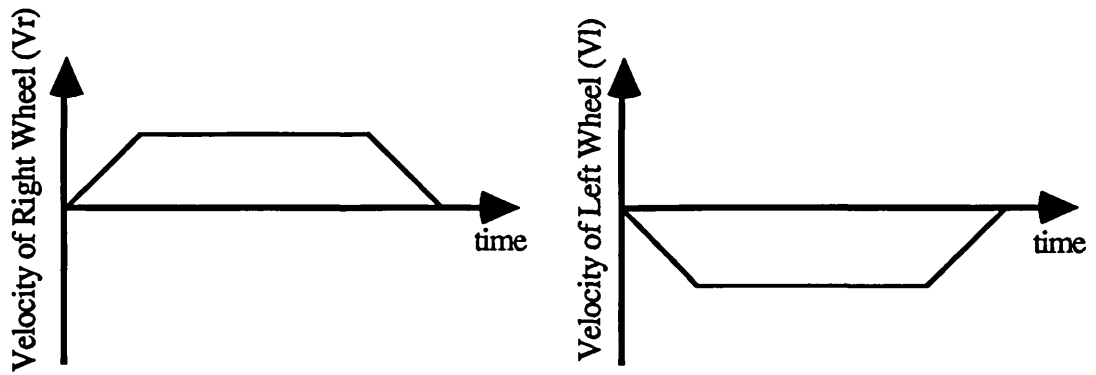


Figure 7.7 Velocity Profiles for Turn

7.2.4.2 Straight Line Following

If the wheels had identical velocity profiles, the vehicle would be travelling in a straight line. This is shown in Figure 7.8.

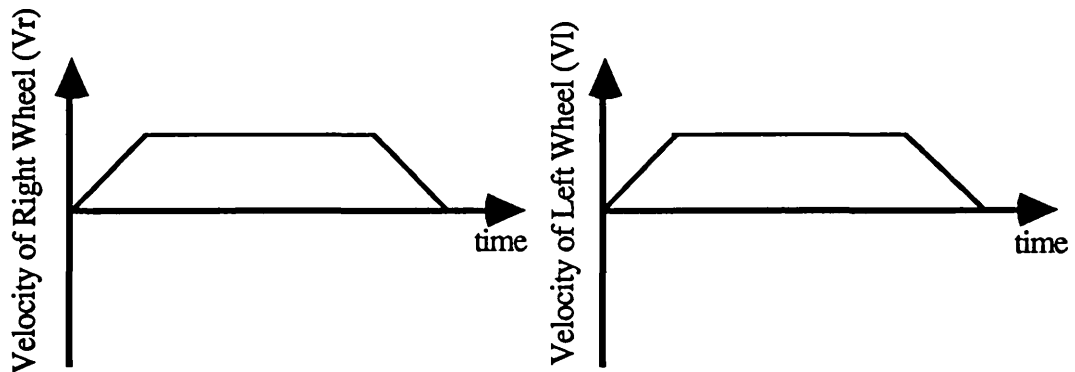


Figure 7.8 Velocity Profiles for Straight Line Motion

If the tracking errors which drive the wheel servos are not equal, the vehicle will follow a straight line but at an angle with the demanded X axis. Let us assume that the radius of the left wheel is slightly greater than the right wheel. At a time Δt , the left wheel will have travelled a greater distance than the right wheel. Hence the vehicle will have developed a heading $\Delta\theta$ with reference to the original heading, and the actual trajectory would be an arc.

$$\Delta\theta = (El - Er)/2w \quad [7.3]$$

where $2w$ = wheel distance

Constant errors can be compensated for by proper calibration of the servo system.

7.3 Control Strategies

The motion control system of the vehicle has to ensure that the route specified for the vehicle is adhered to strictly. In this section we describe various control system options considered for the ICAGV. The system implemented in ICAGV-II is a combination of these systems.

7.3.1 ICAGV-I Control System

In the first Imperial College AGV (ICAGV-I), the path control system was arranged as in Figure 7.9. Briefly the operation of the system is as follows : Using odometry, the vehicle, calculated its lateral deviation y and its heading θ , at regular time intervals Δt . Both these values were multiplied by proportionality constants to produce corrective signals which were fed with inverse signs to wheels velocity servo. If a positive y was detected then the left wheel would accelerate while the right would decelerate such that y decreased. Similarly, for heading errors. In a steady state situation y and θ would be zero and the vehicle would travel along x axis with velocity V_d .

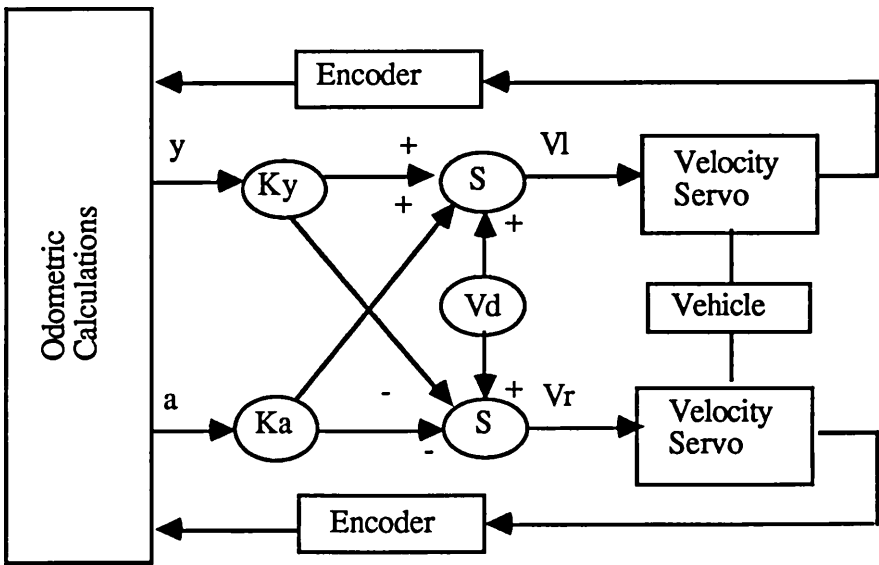


Figure 7.9 ICAGV-I Control Strategy

The control system in Figure 7.9 involves two interconnected motors. The selection of proportionality constants must be such that the vehicle conforms to the desired line precisely exhibiting non oscillatory behaviour.

7.3.2 State Feedback Control System

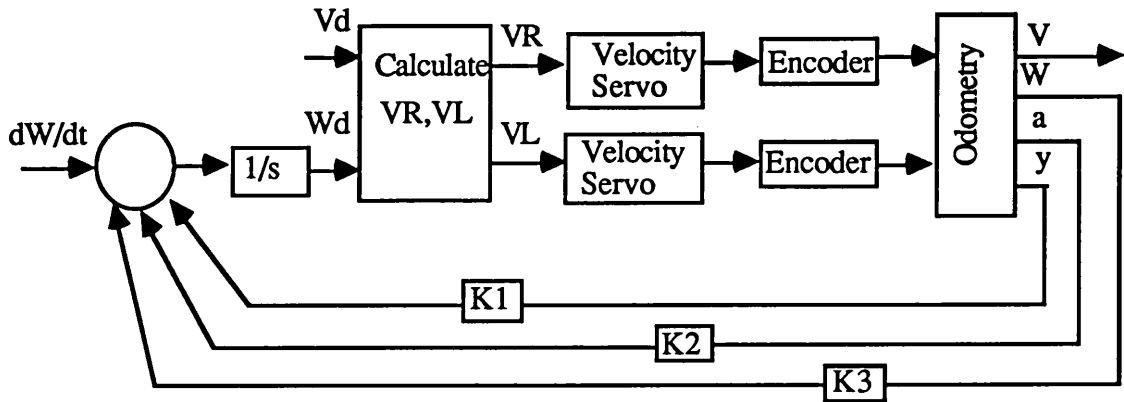


Figure 7.10 State Feedback Method

In this method [63], the translational velocity (V) and rotational velocity (W) of the vehicle are controlled by controlling their derivatives.

For a desired set of (V, W) the required velocities V_r , V_l of the wheels are computed using the expressions:

$$V_r = (V + W * w) / (2 \pi r)$$

$$V_l = (V - W * w) / (2 \pi r) \quad [7.4]$$

where

$2w$ = wheel distance,

r = wheel radius

The required velocities V_r and V_l are then fed to the servo amplifiers. The states, rotational velocity, lateral deviation and heading deviation, are fed back to close the loop. Gains K_1 ,

K2, K3 are calculated using optimal control theory.

The main difference with the previously described approach is that the actual rotational velocity is fed back as well. This contributes to the stability of the system.

7.3.3 ICAGV-II (Present) Control System

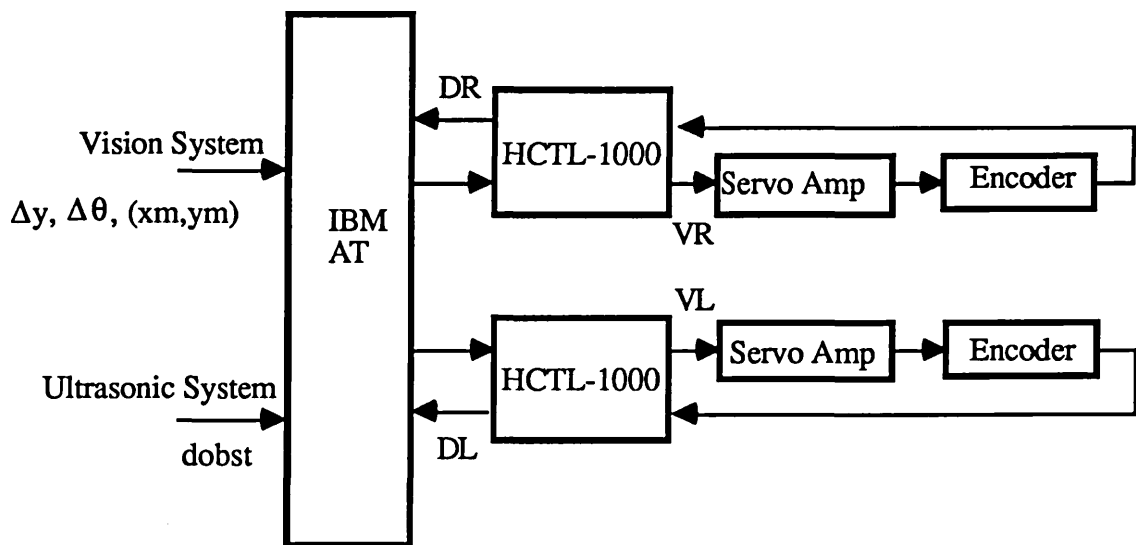


Figure 7.11 ICAGV-II Control Strategy

Figure 7.11 shows the motion control system implemented on ICAGV-II, our present vehicle. A special card (motion control card) is dedicated for servo control of the two AGV motors. This card is interfaced to an IBM AT Single Board Computer. The main components of the motion card are two "General Purpose Motion Control ICs from Hewlett Packard : HCTL-1000", one for each motor. These motion control ICs perform all the time intensive tasks of digital control, freeing the host processor for other tasks.

Unlike the previous two systems, the motion control chips ensure that the velocities specified to the two wheels are controlled to a tight tolerance. Vehicle coordinates are calculated by the host processor (IBM AT) based on distance readings from the motion control ICs. The primary guidance system for the vehicle is odometry. Velocity profiles are sent to the two motors via the HCTL-1000 based on the route distance information and the journey time table.

During straight line motion, the real world sensor, the vision system, measures any drift in the lateral (Δy) and heading ($\Delta \theta$) posture of the vehicle. It also detects marker positions (x_m, y_m) and compares the coordinates with its coordinates in its map. If the coordinates do not agree, the IBM AT interrupts the motion control ICs and sends the velocity profile of the two wheels required to correct the posture of the vehicle to the ICs. After correction the IBM AT returns the two HCTL-1000s to their original task.

The Ultrasonic system detects obstacles that are in front of the vehicle. If the obstacle distance (d_{obst}) is between 1m and 2m the vehicle decelerates. If it is less than 1m the vehicle stops. If the obstacle is removed from the path of the vehicle, it carries on with its previous journey. The Ultrasonic System informs the IBM AT whenever an obstacle is within the "action" zone. The IBM AT will compare the obstacle distance with the remaining distance to be moved by the vehicle and interrupt the motion control card if necessary.

7.4 Servo Control System Design

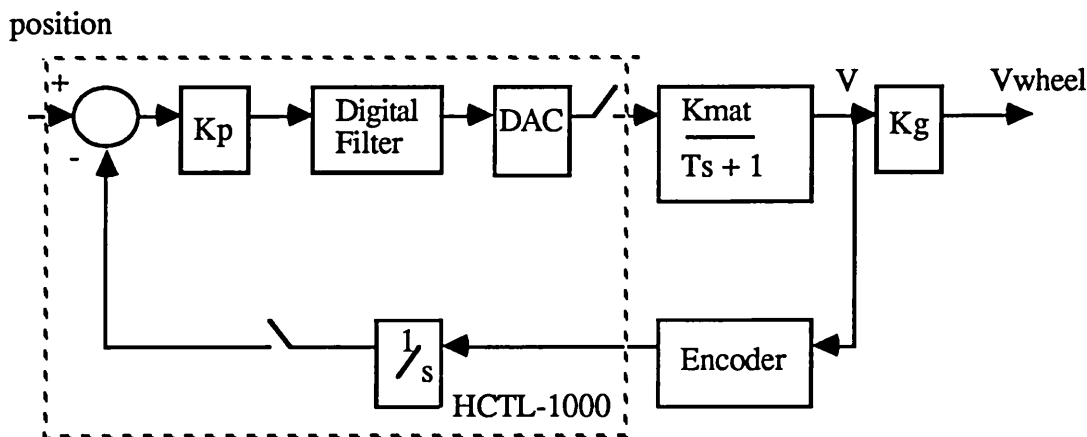


Figure 7.12 HCTL-1000 Motion Control System

Figure 7.12 shows the block diagram of the position servo. The amplifier, motor, tachogenerator are modelled with a first order transfer function with gain K_{mat} and time constant T . The motion control IC (HCTL-1000) performs the loop closure. It provides a programmable gain K_p as well as a digital filter for compensation. The filter is of the form :

$$D(z) = \frac{z - A}{z + B}$$

Both A (zero) and B (pole) are programmable and contribute phase lead and gain to the system. Complete information about phase and gain can be found in Ref [65].

The system excluding the digital filter is the classic 2nd order position servo with an overall gain of :

$$K = K_p * K_{mat} * K_{enc} * K_{dac} \quad [7.5]$$

For its natural frequency ω_n and damping ratio ζ , we know that

$$\omega_n = \sqrt{\frac{K}{T}} \quad , \quad \zeta = \frac{1}{2}\sqrt{KT} \quad [7.6]$$

The time constant T of the actuation mechanism is directly proportional to the moment of inertia J reflected to the shaft of the motor

$$T = C * J \quad [7.7]$$

The moment of inertia J therefore directly effects the damping ratio of the system. Kavoussanos [62] showed that in the simplified case :

1. For straight line motion - moment of inertia (J_s)

$$J_s = (m/2) * r^2 * (1/K_g)^2 \quad [7.8]$$

2. For turning around its centre - moment of inertia (J_t)

$$J_t = (J_v/2) * (r/w) * (1/K_g)^2 \quad [7.9]$$

where m = mass of vehicle

r = radius of the wheel

K_g = gear ratio (from motor shaft to wheel)

J_v = moment of inertia of the vehicle about its centre

w = distance between the two wheels

From equation [7.8] and [7.9]

$$J_s/J_t = m * w^2/J_v \quad [7.10]$$

If the mass of the vehicle could be considered distributed in the periphery of the circle $(0,w)$ (Fig.7.13), then J_v would be equal to mw^2 and J_s equal to J_t .

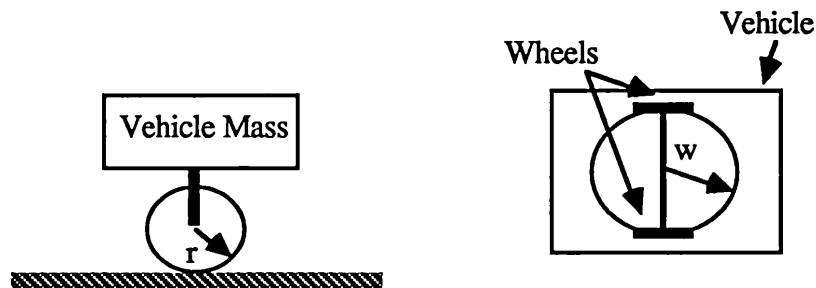


Figure 7.13 Concentration of Mass of Vehicle

In the Imperial College AGV, the greater portion of the mass is actually distributed outside the above circle. Thus $J_v > mw^2$ implying $J_t > J_s$. The inertia which the motors face is considerably greater during turning than during straight line motion. The damping system will be less during turning.

7.4.1 Gain Adjustment and Compensation

The worst case for the vehicle in terms of stability (directly related to damping) is when turning around its centre. Ideally for this case we would like to calculate the gain as well as the compensation parameters in such a way that :

1. The servo possesses the highest possible gain - for high stiffness and
2. Adequate damping (0.7) for stability.

Since the moment of inertia of the vehicle about its centre is difficult to evaluate considering the variable nature of the load the vehicle will be carrying, gain adjustment and compensation were actually performed by experimental turns of the vehicle, keeping in mind the following:

- increasing the gain worsens stability

- increasing the digital filter zero (A) improves stability
- increasing the digital filter pole (B) slightly improves stability

During initial setup, the gain is increased from a low value, to a value where there is no observable oscillation in the wheels velocity. The sampling time of the digital controller is chosen to be 1.5msec which is fast sampling considering that the time constant is not likely to be less than 20msec.

- ✓ If the vehicle does not oscillate during turns, then in straight line motion it will behave even better.

7.4.3 Motion Control Card

The motion control card was developed in-house. The main component in this card is the HP Motion Control IC - HCTL1000. It is possible to program the HCTL1000 in the following modes.

- | | |
|----------------------------------|---|
| 1. Position Control | - the user has to specify the final position |
| 2. Proportional Velocity Control | - the user has to specify the velocity |
| 3. Integral Velocity Control | - the user has to specify the velocity and acceleration |
| 4. Trapezoidal Velocity Control | - the user has to specify the acceleration, maximum velocity and final position. The trapezoidal profile needed to achieve the conditions are generated by the HCTL-1000 IC |

The output is fed to a 8-bit DAC which is connected to a servo amplifier. There are two identical channels on the card for the two motors of the AGV, see Figure 7.14.

The complete specifications of the HCTL-1000 General Purpose Motion Control IC can be found in Appendix A2

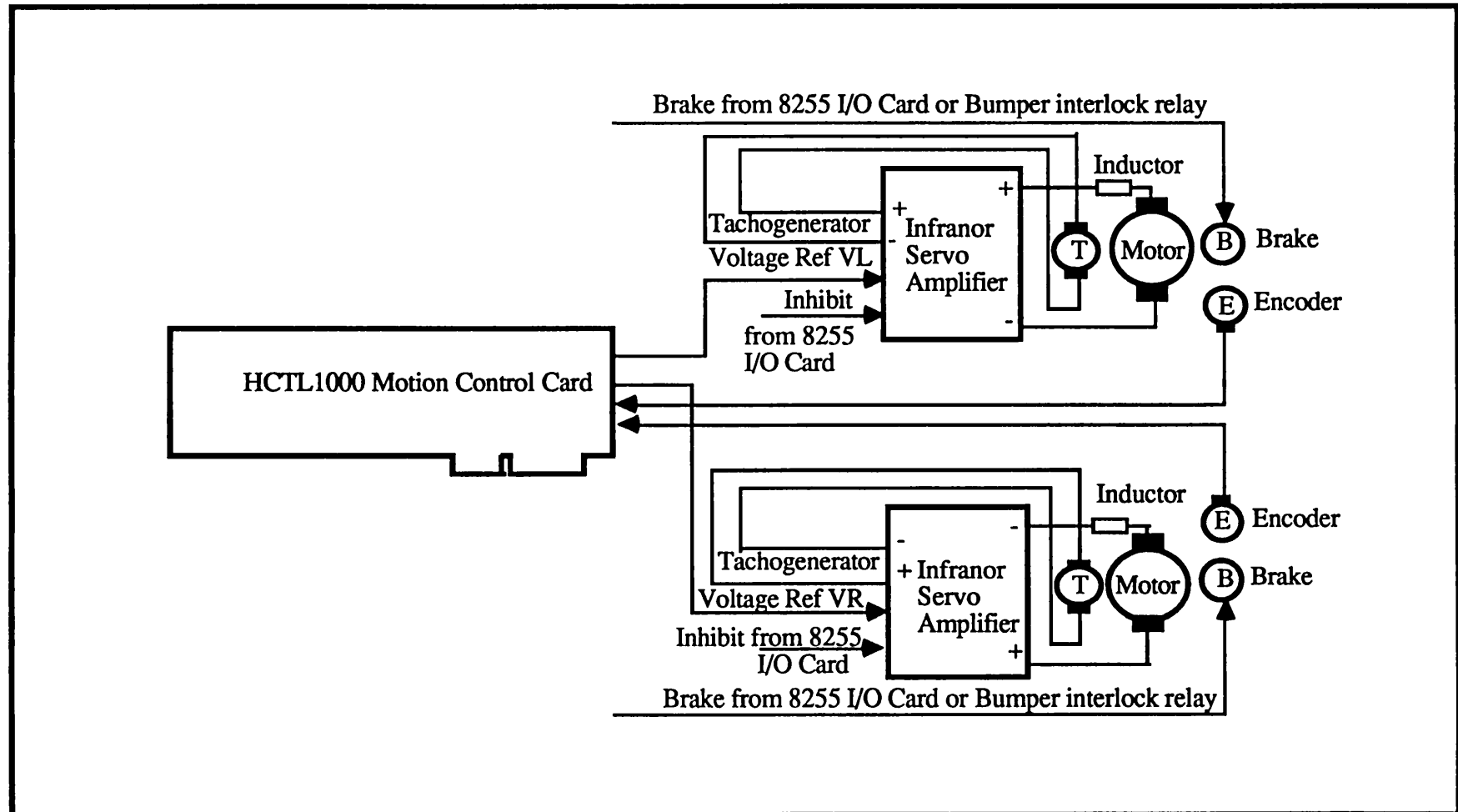


Figure 7.14

CHAPTER 8

SYSTEM INTEGRATION - IMPLEMENTATION OF THE IMPERIAL COLLEGE AUTOMATED GUIDED VEHICLE

8.1 Introduction

Practical difficulties involved in implementing any moderately complex system necessitates that an overall methodology is established before any commitment is made to hardware. It is commonly accepted [68], [69], [70], [71] that to ensure robustness of the hardware for real time, real world applications, the free ranging system needs to fuse or integrate the information from various sensors (information fusion), as far as possible, executes its tasks in parallel (parallel execution) and have adequate scope for expansion or extension (extensibility).

Information fusion - To get a reliable picture of the surroundings, it is necessary to have an array of different types of sensors, maps and markers to complement and reinforce one another to guarantee that there is no chance of the vehicle diverting from its programmed path and to continually scan the path to ensure that there is no obstacle in front of it.

Parallel Execution - There are a number of real time duties that the vehicle performs. They include odometry, line following/marker detection using vision, obstacle detection using the ultrasonic ranging transducers and communication between vehicle and the supervisory system. To ensure near real time response, each function, wherever possible, should be executed concurrently, an interrupt being generated if other functions need to be involved.

Extensibility - The field of free ranging vehicles (or mobile robots) is still in its infancy. Real world problems cannot be planned completely in advance. An open architecture allows for future additions to take care of any unforeseen requirement.

In this chapter, we integrate all the building blocks, supervisory system (chapter-2), communication system (chapter-3), odometry (chapter-4), vision system (chapter-5), ultrasonic ranging system (chapter-6) and motion control system (chapter-7) into our complete free ranging automated guided vehicle system. We will show that by using the top down approach of considering supervisory issues first and moving down via the

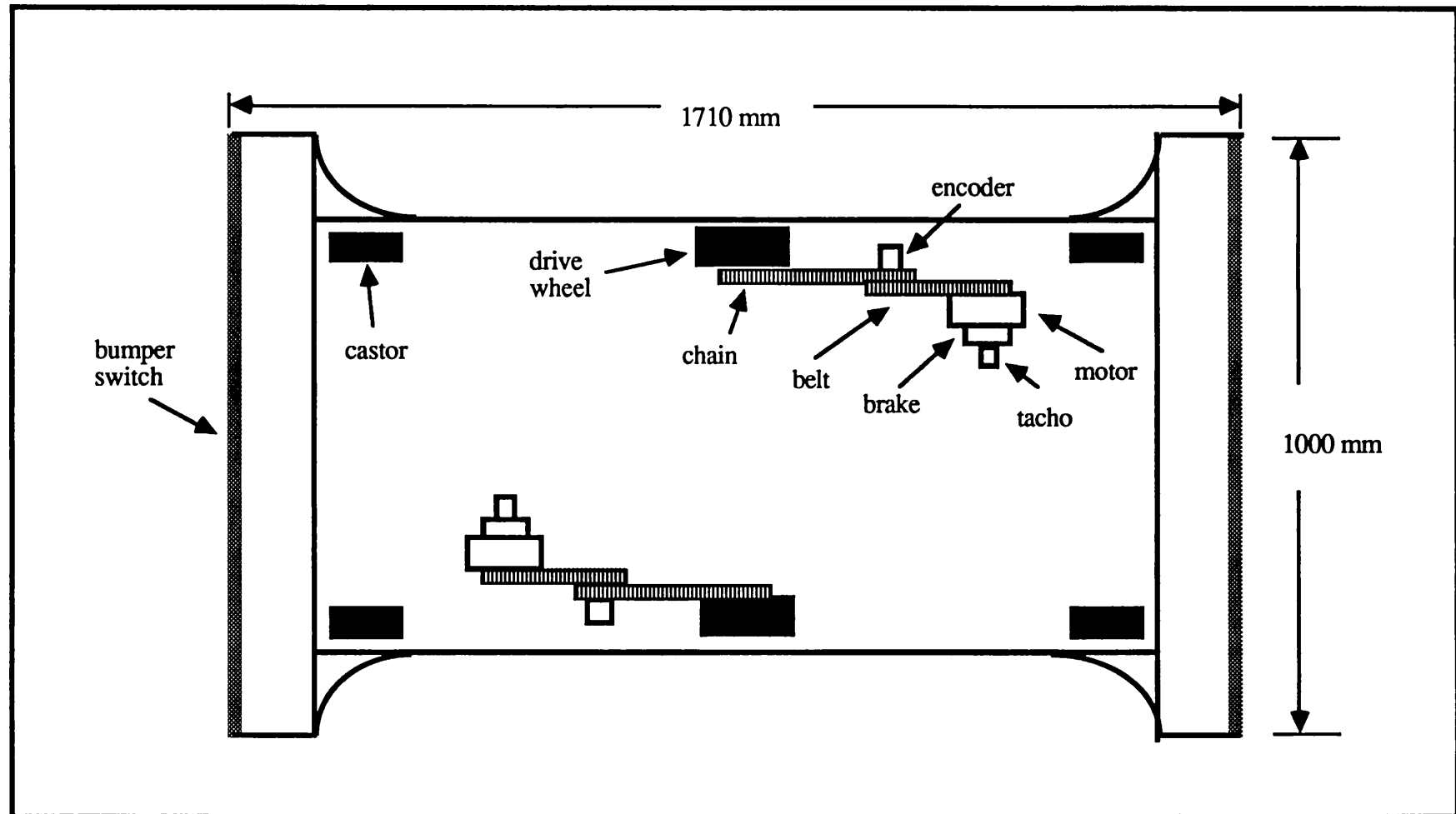


Figure 8.1

communication system to issues pertaining to the vehicle , the system implemented (ICAGV-II System) is extremely flexible and its potential applications seem limitless.

8.2 Hardware

The Imperial College Automated Guided Vehicle (ICAGV-II) shown in Fig.8.1, is approximately 1.7m by 1m in size and is constructed with 5cm square steel sections. It is designed to transport a maximum load of 1000kgs. This rugged construction ensures that the vehicle is suitable for load carrying tasks in an industrial environment. This is in direct contrast to small "toy" vehicles used for most prototype development. The vehicle is differentially steered by two independently driven wheels located equidistant from the front and rear of the vehicle. Thus the vehicle is equally controllable in the forward and reverse direction and can turn around its geometric center.

There are four supporting castors at each corner of the vehicle. All castors are equipped with springs to provide suspension.

Two DC motors (Vactric-Mavilor Model No. 601, 56V, 3000rpm, 600W) with integrated brakes and tachogenerators drive the two wheels. The motors are driven by two "Infranor Series 90/20" servo amplifiers modified for 48V operation. The transmission mechanism (Figure 8.2) consists of a belt, sprocket, chain, sprocket and wheel combination and provides a total speed reduction of 1 : 19 (ie. $(12/48)*(12/57)$). Optical shaft incremental encoders (Muirhead type 25, 1000 pulses/rev or 4000 quadrature counts / rev) are coupled to the intermediate sprocket of the drive mechanism. They are used for servo loop closure and provide the control system with a positional resolution of about 0.0413mm (ie $(12/57)*2\pi*(1/4000)$).

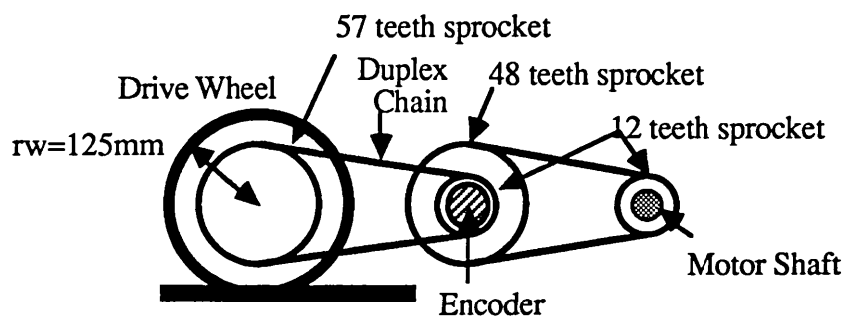


Figure 8.2 ICAGV-II Drive Chain

The energy source comprises of four 12V, 75Ah batteries connected in series to provide the 48V DC Power for traction and one 12V, 75Ah battery for the controls. This battery is connected to a dc-dc power supply which supplies +5V, +12V, -12V, +15V, -15V and +24V for the various systems on the vehicle.

- With a maximum of 48V available for traction, theoretically the maximum speed attainable by the vehicle is 1.77 m/sec (ie. $(3000\text{rpm} / 19) * (1/60) * 2\pi * (48\text{V}/56\text{V})$).

For safety, touch sensitive bumper pad switches are connected to the front and rear of the vehicle to ensure that the vehicle stops immediately it hits any object .

8.3 ICAGV-II Control

8.3.1. Selection of Computer System

IBM PC ATs were selected for the supervisory computer and for the vehicle computer. The IBM AT on board the vehicle comprises of a chassis with a passive back plane (10 IBM AT bus slots) and a single board computer (Databasix Model No. QPC 5121) that fits in one of the slots. A major problem with all computer systems is its obsolescence rate. The advantages of the selection is as follows :

- 1) The large user base for IBM PC ATs guarantees its maintainability for the next few years.
- 2) There is complete hardware and software interchangeability between the two systems ensuring that interfacing problems are kept to a minimum.
- 3) Numerous add-on-cards are available for the IBM AT. As all these cards are aimed at the personal computer user market, the cost is not prohibitive.
- 4) The passive backplane implies that if more computing power is required, it is possible to replace the computer card with an Intel 80386 base computer or any other system with the same bus configuration. Transputer cards are also available in the IBM AT bus format.
- 5) The abundance of software developed for the IBM PC can also be utilised resulting in faster development cycles.

8.3.2 Block Diagram of ICAGV-II Control Hardware

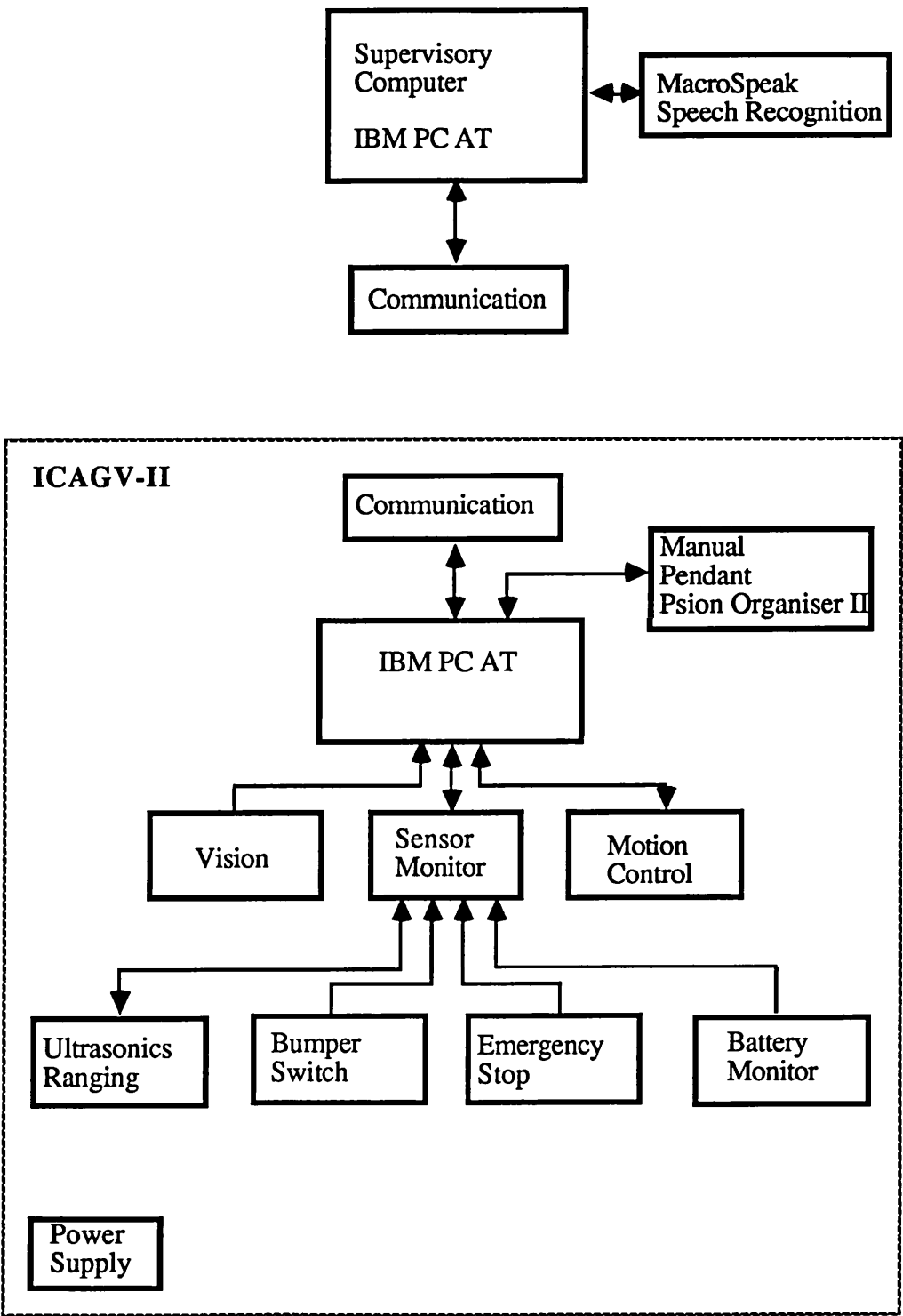


Figure 8.3 Block Diagram of ICAGV-II

Fig. 8.3. shows the block diagram of the hardware of the ICAGV- II controls.

The main cards in the passive backplane are as follows:

- 1) IBM AT Compatible Single Board Computer with one serial and one parallel port - Databasix Model No. QPC 5121
- 2) Vision card - Eltime Framegrabber IMAGE-II (Ref. Chapter 5, Section 5.7).
- 3) Motion control card - HCTL-1000 Motion Control Card (Ref. Chapter 7, Section 7.5).
- 4) Sensor monitor card - 8255 I/O Card.

The description of the vision card can be found in chapter 5 and the description of the motion control card can be found in chapter 7. The ultrasonic obstacle detection system is interfaced to the IBM AT via the sensor monitor card.

8.3.4 Sensor Monitor Card

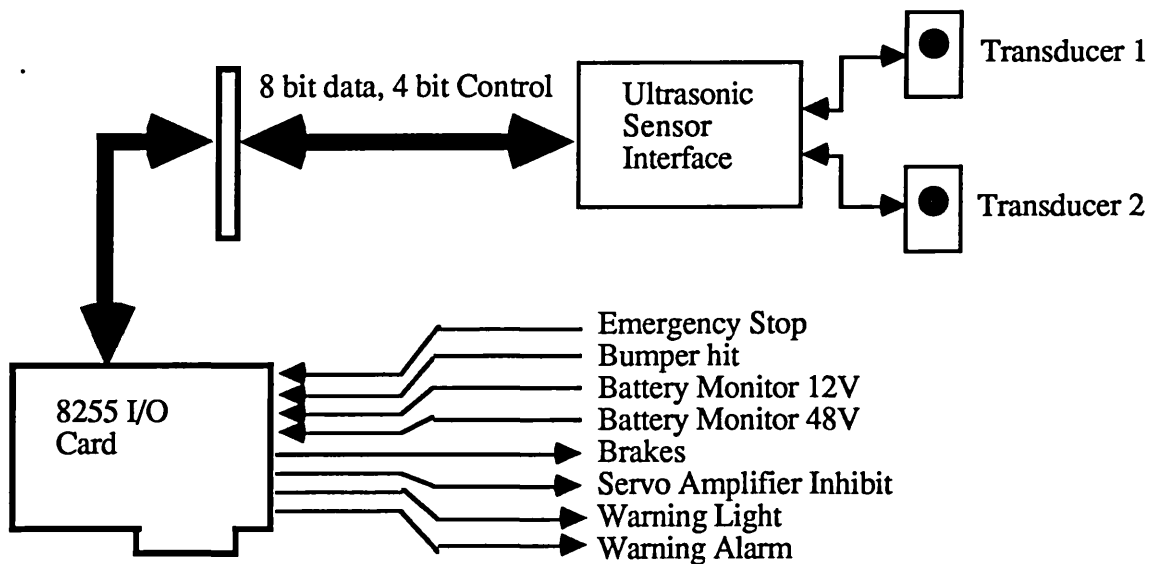


Figure 8.4 Sensor Monitor Card

This card is an 8255 I/O card for the IBM AT. Altogether 6 ports of 8 bits are available for digital input / output connections. The card is used for the following purposes:

- 1) Battery voltage monitor - there are five 75Ah batteries, four of which are used for traction (48V) and one for the controls (12V). Using DC to DC converters the 12V generates +5V, +12V and -12V. Both the 48V and the 12V are monitored. If the voltage reduces below a certain limit, a warning alarm is raised.
- 2) Ultrasonic Sensor - the threshold is set to detect objects upto a distance of 2m away. If an object is detected a slow down routine is executed. If the object is within 1m of the vehicle, it will decelerate smoothly to a halt. The vehicle will start moving if the object is removed.
- 3) Bumper Switches - If the bumper is hit, the AGV stops immediately. The AGV can be restarted only if it is manually reset. The bumper switches are hardware and software interlocked to the inhibit signal going to the servo amplifiers.
- 4) Emergency Stop Buttons - if pressed the AGV stops instantly. The servo amplifiers are inhibited and power to the vehicle is cut off.

8.3.5 Speech Recognition Unit

This unit, MacroSpeak (from Marconi Speech and Information Systems) was added to the system for demonstration purposes. By sending simple commands by speech to the vehicle, the dimension of speech is added to the system and increases its "user friendliness".

This feature makes our system suitable for applications where hands are occupied or cannot be used as for the physically handicapped. By simple spoken commands the vehicle can be directed to perform various movements. It is possible to visualise a mobile robot with a robot arm on it picking up objects or switching on appliances for a bedridden patient.

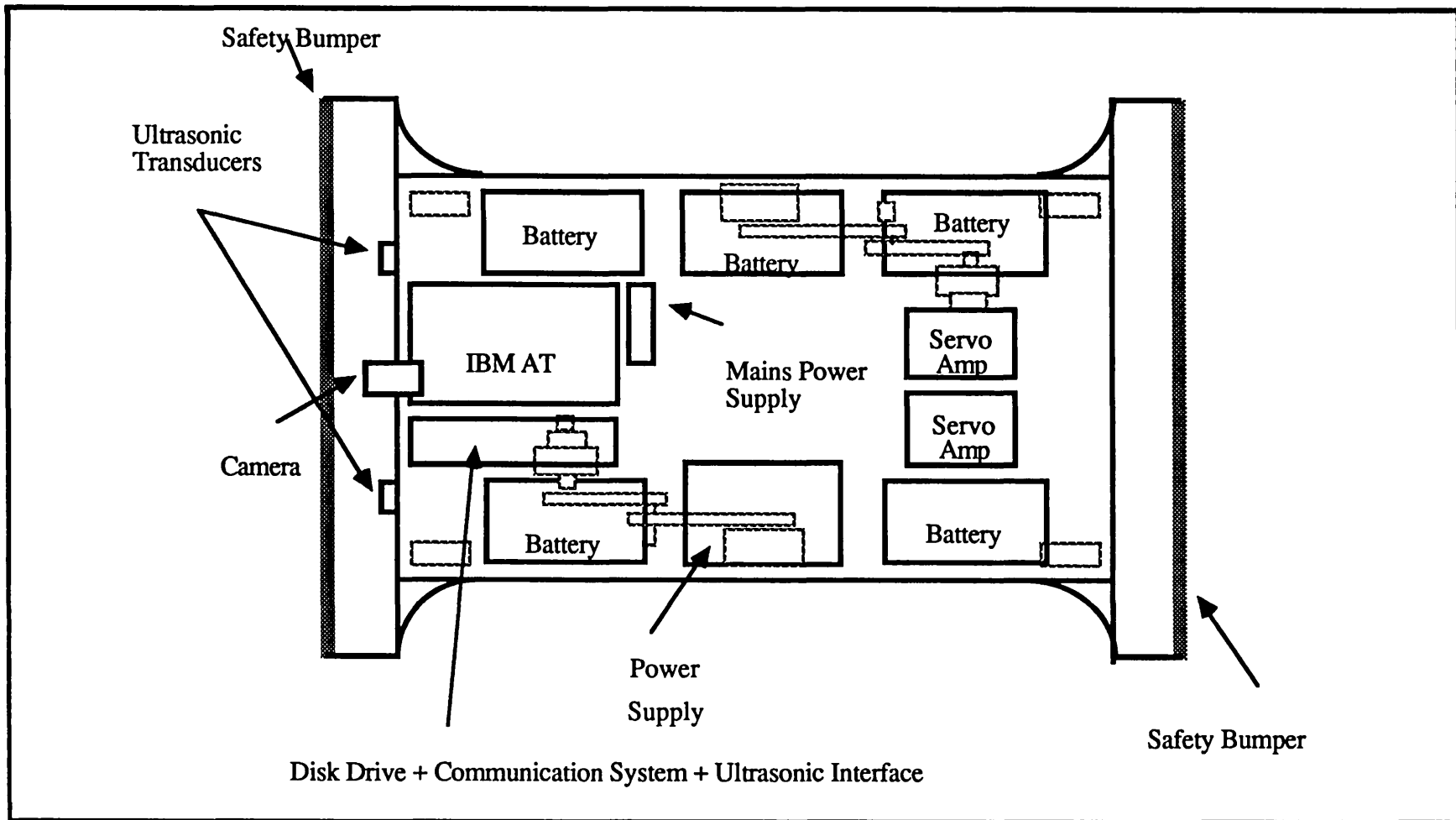


Figure 8.5

8.4 Software

With the hardware in place, we now turn to the software design phase. Birrell and Ould [75] state that : "The principal aim of the system design phase should be the same no matter what design method is used ; namely, to produce a system design which

- is easily implemented

and which when implemented

- will provide all required functions
- will operate within any required constraint
- will be easily maintained".

To a certain extent the software choices available were dictated by the hardware selection - IBM AT. MS DOS was selected as the operating system.

In a free ranging system it is difficult to think of "all required functions" - the best that can be achieved is to create a software environment where future or unforeseen requirements can be easily incorporated. The system would then ultimately satisfy all the required functions and constraints.

8.4.1 Ease of Implementation

Not all high level languages are suitable for real time programming. Even if the language itself is, a particular compilers implementation may not be. The following factors may need to be considered :

- Whether the language is interpreted or compiled
- ROMability of the code
- Support for separate compilation
- Debugger support

The initial choice of languages was between Pascal, C and Modula 2. Since the project involved a number of people, a necessary requirement was the readability of the code. Ultimately Borland's Turbo Pascal was chosen. Turbo Pascal with its low level commands for the IBM PC makes it ideal for hardware control

8.4.2 Functional Requirements

To recapitulate, to be able to understand the commands given by the supervisory system (chapter 2), the vehicle has to have the following information:

1. Have complete information on all the node coordinate points
2. Know its own identification code (AGV No.)
3. Identify the type of instruction it has received
 - a. queue type
 - b. immediate type
4. Be able to synchronise its clock with the "master clock".
5. Vary its speed to meet the time tabled requests
6. Wait whilst the job is being executed.

From the above points, we see that the software requirements are :

1. Graphic Commands - for displaying layout of environment with AGV routes
2. Control Commands - AGV identification, type of command etc.
3. Timing Commands - clock synchronisation commands, wait commands etc
4. Movement Commands - for controlling the movement of the AGVs.
5. Image Commands - for verifying actual node positions

Figure 8.6 shows the commands available in the software package.

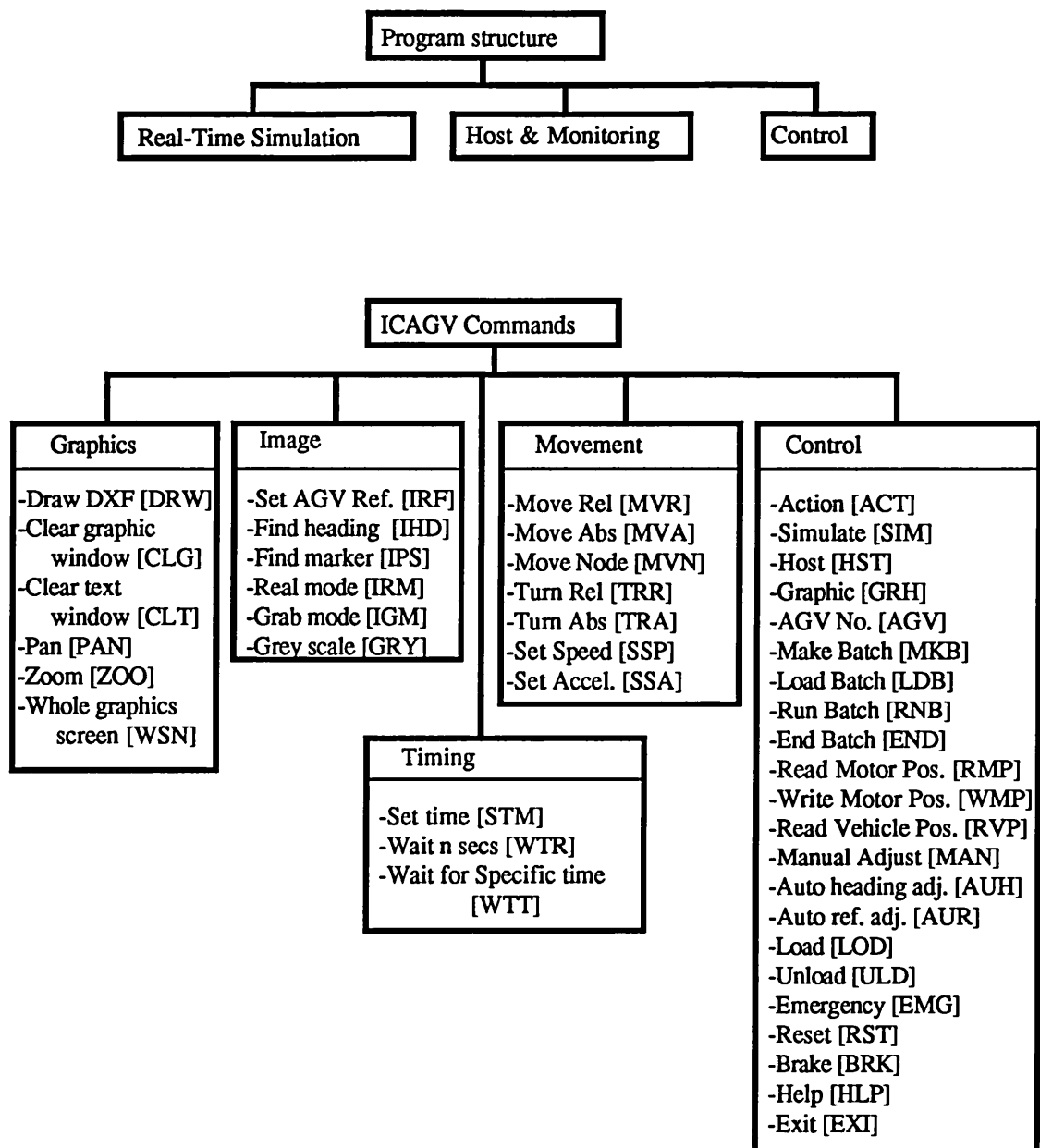


Figure 8.6 Command Structure

8.4.3 Command Structure

Basically the commands have been divided into five categories - graphics, control, timing, movement and image.

1. Graphics

The screen is divided into two windows, a graphics window and a text window. This unit allows us to display a previously saved layout (in DXF) on the graphics screen. The following commands are available in this category :

- DRW { draw DXF drawing in graphics window }
- CLG { clear graphics window }
- CLT { clear text window }
- PAN { Pan }
- ZOO { Zoom }
- WSN { use whole screen for graphics }

2. Control - These commands handle the general operation of the system. To ensure that the system is as flexible as possible - the same software resides in both the supervisory system and the vehicle. The system can be set to any of the three following modes - simulation mode, host and monitor mode or action mode.

The following commands are available in this category :

a. Mode command

- ACT { set action mode }
- SIM { set simulation mode }
- HST { set host mode }
- GRH { set graphic mode - this mode is for development purposes only }

b. AGV Identification

- AGV { AGV identification / call command }

c. Batch

- MKB { make batch file }
- LDB { load batch file }

RNB { run batch file }
END { end command for batch file }

c. Status Commands

RMP { read motor position }
WMP { write motor position }
RVC {read vehicle coordinates (x,y,θ) }

d. Adjust Commands

MAN { manual adjust }
AUH { automatic heading adjust }
AUR { automatic reference adjust }

e. Commands to other systems

LOD { load pallet }
ULD { unload pallet }

f. Emergency

EMG { Emergency Stop }
RST { Reset HCTL Motion Control IC }
BRK { set brake on / off }

g. Miscellaneous

HLP { help - switches on "help" screen }
EXI { Exits program }

3. Timing

As the schedule is based on "time-tabled journeys", we must ensure that the whole system is synchronised to the same clock. The commands available in this category are :

STM { set time }
WTR { wait for n secs }
WTT { wait for specific time }

Unfortunately due to the inherent delay in the communication system, there is always about

0.5 sec delay between sending and receiving the signal.

4. Movement

The movements are mainly straight line movements and turn movements. Move commands are either relative movements or absolute movements. For relative movements, a distance or degree is specified and the vehicle executes the motion specified. For absolute movement, world coordinates are specified. the vehicle calculates the easiest way to fulfil this request and then executes the command. Node points can also be specified instead of world coordinates. the vehicle has a list of nodes and their corresponding coordinate positions.

The following commands are available in this category :

- MVR { move relative }
- MVA { move absolute }
- TRR { turn relative }
- TRA { turn absolute }
- MVN { move to node }

5. Image

The camera is the real world sensor of the vehicle. The following commands are available in this category :

- IRH { set AGV reference }
- IHD { find heading and lateral error }
- IPS { find marker position }
- IRM { Image real mode }
- IGM { Image grab mode }
- GRY { draw grey scale to monitor }

8.5 Program Structure

The program structure of the ICAGV-II system is shown below. After "initialising " the whole system, the program gets into a loop and remains in the loop until it receives an "exit" command which signals the end of process.

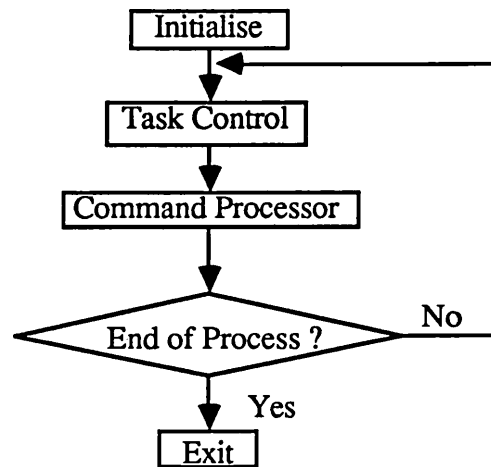


Figure 8.7 Program Flowchart

8.5.1 Initialisation

The initialisation step comprises of initialising the following items :

Vehicle

- vehicle coordinates
- vision system
- communication system
- I/O port
- control parameters

Supervisory System

- parameters
- graphics parameters
- simulation parameters

8.5.2 Task Control

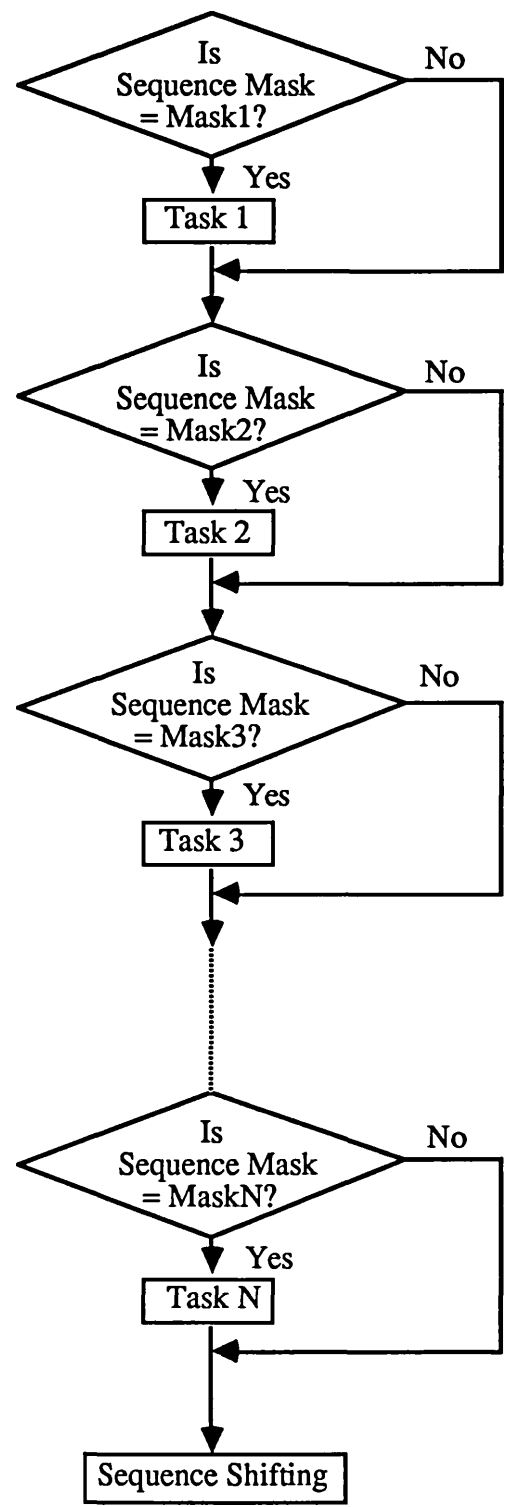


Figure 8.8 : Task Control Flowchart

There are a number of operations that have to be done simultaneously by the free ranging AGV system. Some of these are :

keyboard operation, asynchronous communication, read position, display operations, display time, display vehicle, display register, speed detect

As DOS is a nonreentrant operating system, when a DOS function is in progress, it is not possible to interrupt it and execute another. Multitasking is possible only by cycling through a list of tasks, mapping task data into active memory and passing control to the task for a discrete time slice. Figure 8.8 shows the "Task Control" scheme of the ICAGV-II systems. Priorities can be specified by presetting 8 bit masks for each of the tasks - 0000 0001 (01) indicates that the task will be executed ones every 256 cycles of the repeat loop whilst 1111 1111 (FF) indicates that the task will be executed every cycle.

Example

Task1	{ Keyboard	0000 0001 }
Task2	{ Async	1111 1111 }
Task3	{ Position calc.	0101 0101 }
Task4	{ Display status	0101 0101 }
Task5	{ Clock	0000 0010 }
Task6	{ Vehicle image	0100 0100 }
Task7	{ Register display	1010 1010 }
Task8	{ Safety factor	0000 0000 }

8.5.3 Command Processor

The command processor reads each command that it receives via the keyboard or the communication channel. It checks the syntax of the command and if correct executes it according to the flowchart shown in Figure 8.9.

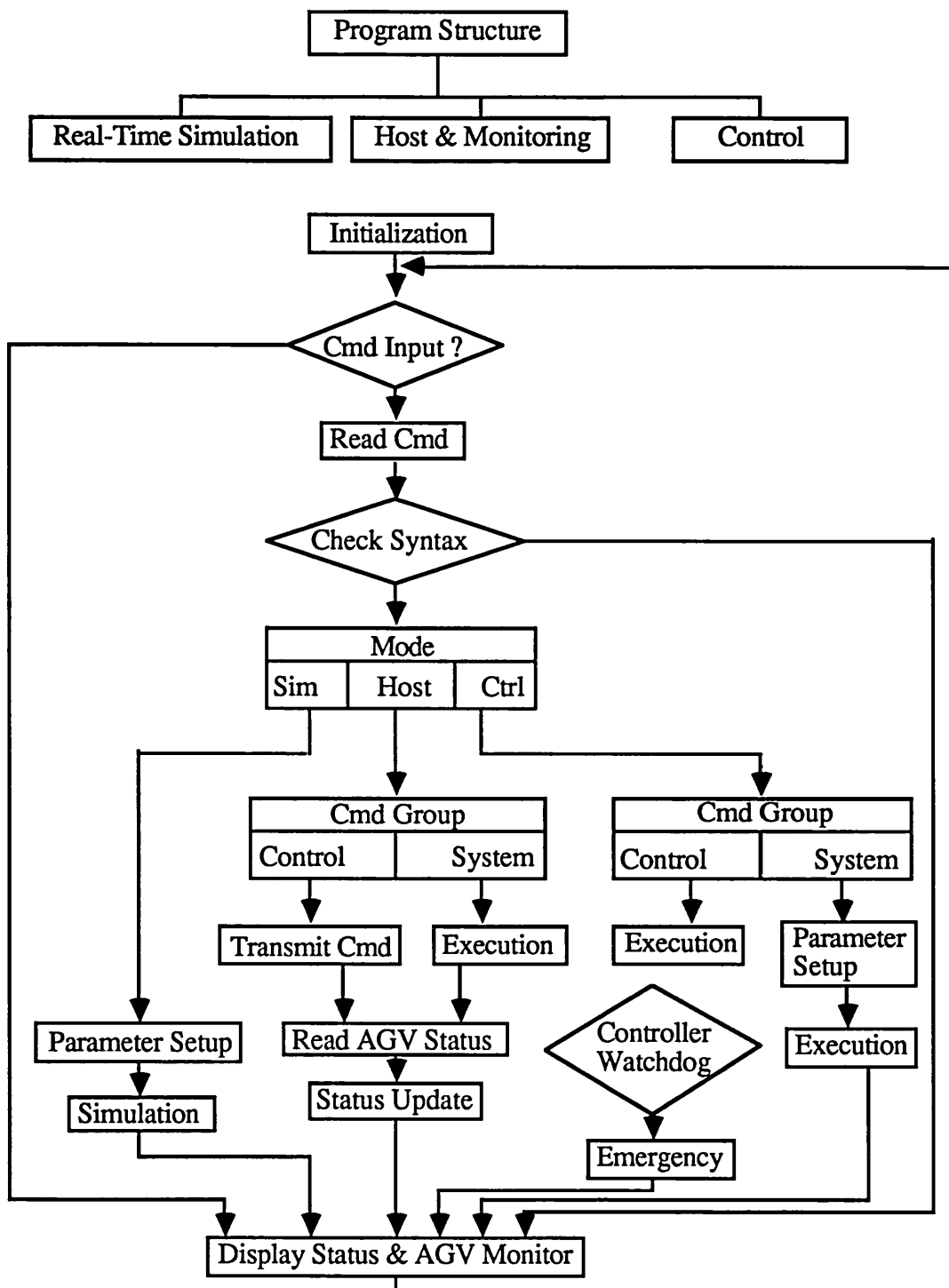


Figure 8.9 : Command Processor

8.7 Example of Program

Below we have a file (Section 2.6) sent to the vehicle by the supervisory system :

V01MBQ	- Vehicle No. 01 Message Begin , Queue type Command
S20T000300	- Start at Node 20 at time 00:03:00
E32T000837	- End at Node 32 at time 00:08:37
J02D000005	- End Node Job duration 5 secs.
N03T000334	- Intermediate Node 03 at time 00:03:34
N07T000413	- Intermediate Node 07 at time 00:04:13
N23T000521	- Intermediate Node 23 at time 00:05:21
N25T000642	- Intermediate Node 25 at time 00:06:42
N30T000723	- Intermediate Node 30 at time 00:07:23
V01ME	- Vehicle 01 Message End

Below we see the same file rewritten, using the commands available for the AGV.

```
AGV 01
MVN 20
WTT 00,03,00
SSP a
MVN 03
WTT 00,03,34
SSP b
MVN 07
WTT 00,04,13
SSP c
MVN 23
WTT 00,05,21
SSP d
MVN 25
WTT 00,06,42
SSP e
MVN 30
WTT 00,07,23
```

SSP f
MVN 32
WTR 5
END.

a,b,c,d,e,f are speeds calculated by the vehicle to meet the time constraints imposed by the traffic control system.

CHAPTER 9

DISCUSSION OF VEHICLE SYSTEM PERFORMANCE

9.1 Introduction

The Imperial College Automated Guided Vehicle (ICAGV - II) is shown in figure 9.1 and in figure 9.2. To evaluate the performance of the complete free-ranging system, we must first evaluate the performance of the individual components that make up the system. For this purpose, a layout of our laboratory was created using "AutoCAD" and this was then transferred via the DXF format into our supervisory package written in Turbo Pascal. Figure 9.3A shows the layout created using "AutoCAD" and Figure 9.3B the same layout in our supervisory package. With this information, it is possible to simulate the movement of a vehicle travelling from one position in the laboratory to another.

If the present posture of the vehicle (x,y,θ) is known, the vehicle can be requested to go from one known position in the laboratory to another, for example if the vehicle is at node 5 it could be requested to travel to node 14. The supervisory system will select the shortest route from the start node to the destination node. The other option is for the user to specify the complete route including all intermediary nodes that have to be crossed.

Here we present the performance of the various components of the Imperial College free ranging Automated Guided Vehicle system. We start with the dead-reckoning system - odometry and proceed to the vision system and the obstacle detection system. The supervisory system sends instructions to the vehicle and receives status information from the vehicle via the RF communication system.

The free ranging concept can be claimed to be a success only if it is possible to precisely navigate the vehicle in the laboratory using odometry and vision ensuring that the specified software track is followed.

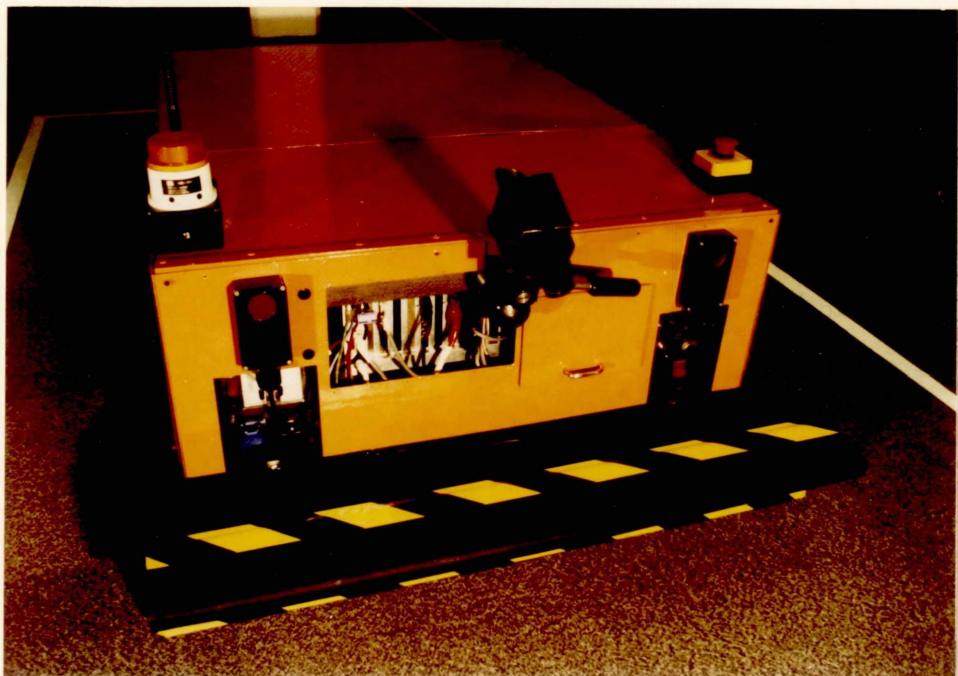
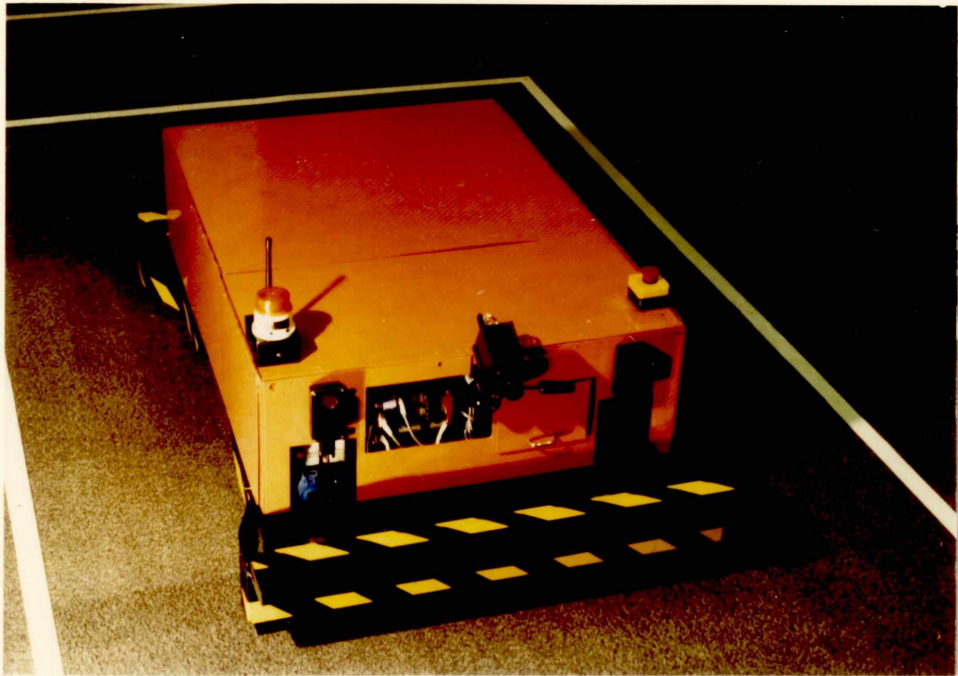


Figure 9.1 The Imperial College Automated Guided Vehicle (ICAGV-II)

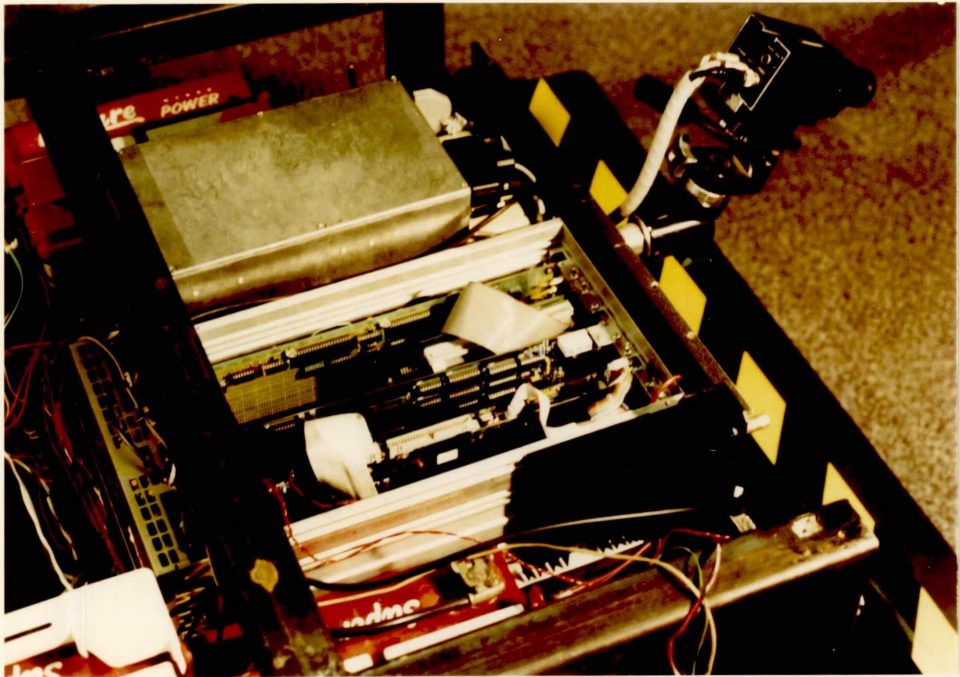


Figure 9.2 Imperial College Automated Guided Vehicle (ICAGV-II) without the cover

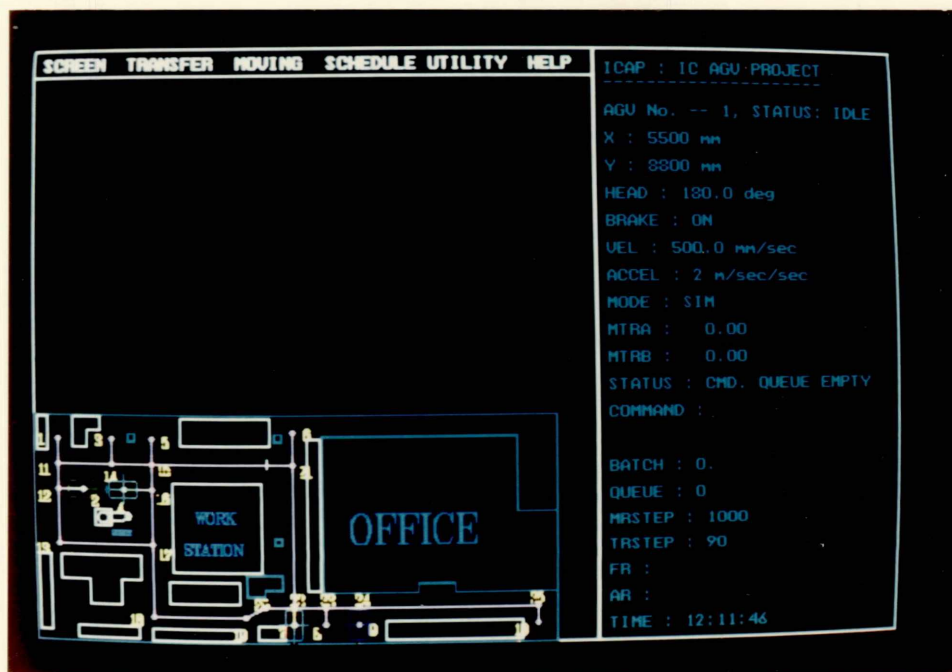
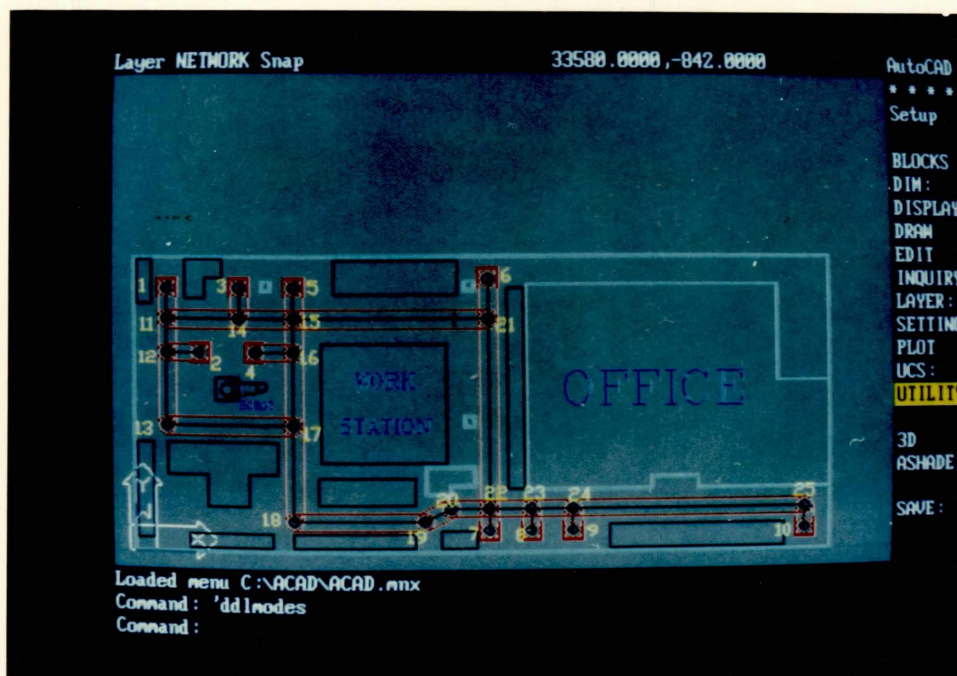


Figure 9.3 Layout of CAE Laboratory
A. Created using AutoCAD
B. Transferred to Supervisory System

9.2 Odometry

The motion control task has been delegated to the Hewlett Packard Motion Control ICs-HCTL1000. The advantages to this selection is :

1. the IC performs the servo loop closure as well as generates position profiles, jobs which otherwise had to be done by the host processor or a general purpose microprocessor.
2. it provides programmable gain as well as a digital filter for compensation - no control law has to be implemented.
3. the incremental encoder is interfaced directly to the HCTL 1000 IC. The IC employs an internal filter that removes any noisy spikes from the encoder. No tachogenerator is needed. The velocity term is calculated from the encoder readings.
4. it can be easily interfaced with the amplifier. The drive signal is available either as a 8bit word for a DAC to drive a linear amplifier or as a "pulse" and "sign" output to drive a pulse width modulated (PWM) amplifier.

Usually a servo amplifier is driven by the analog signal from a DAC. Making use of the "pulse" and "sign" output available, the expensive servo amplifiers can be replaced by cheaper "H Bridge Amplifiers".

In our system servo amplifiers were used and since the motors have integrated tachogenerators, the tachogenerator signals were fed to the servo amplifiers.

9.2.1 Accuracy of Odometry

The accuracy of the odometric system is directly related to the condition of the floor surface. For our initial experiments, the floor was covered by PVC tiles. The accuracy of the ICAGV-II with the HCTL-1000 Motion Control ICs on this surface is as follows :

For a 12m straight line path : the vehicle travels along the path and comes back to the starting point having a lateral error of approximately 1cm and longitudinal error of approximately 3 mms.

For carpeted surfaces (polyamide carpet tiles) the value is as follows :

For a 4m straight line path : the vehicle travels along the path and comes back to the starting point having a lateral error of approximately 2cm and a negligible longitudinal error.

We note that the results were considerably worse for the carpeted surface. These results were expected.

The following points must be noted:

- a. Odometry is continually being used to monitor the vehicles position and heading. The errors observed from odometric calculations based on encoder readings were of the magnitude of some mms. That means that the actual errors are caused either due to wheel slippage or due to changes in wheel radius or due to unevenness of the floor.
- b. Very small deviation in the begining of the line segment will cause considerable lateral error. For an initial heading error of 0.1 degrees, the lateral error will be 17mm after 10 meters run. With 1 degree error the lateral error will become the unacceptable 17cm in 10m path.

Heading deviations can be caused by :

- Misalignment of the vehicle with the desired line in the begining of a route.
- Errors during turning for path succession.

Both these errors can be corrected by the vision system.

9.3 Vision System

The vision system is the only real world sensor of the vehicle. To ensure that cumulative errors do not build up due to slip, uneven surface or other external factors, updates must be sent to the position reading of the vehicle. From the discussion above, we note that the lateral error is much greater than the longitudinal error. The duties of the vision system are:

1. Correction of heading errors which is the main contributor to the lateral error
 - this can be achieved by line tracking or pathway tracking
2. Marker detection - for datum location and precise docking

Figure 9.4 shows the line observed by the CCD camera on board the vehicle. Figure 9.4A shows the image after processing and with a least squares line fitted onto the valid points. Figure 9.4B shows the robustness of the algorithm used. The clutter did not confuse the line finding routine.

For keeping track of the pathway markers during motion only one line of the image needs to be analysed. If the line is in the centre of the path, it should always have pixel "x" coordinate value 128 (frame grabber 256pixels by 256pixels). The deviation is $P_x(x) - 128$. During acceptable straight line motion, only one line or a segment of the line about the 128 pixel position is monitored.

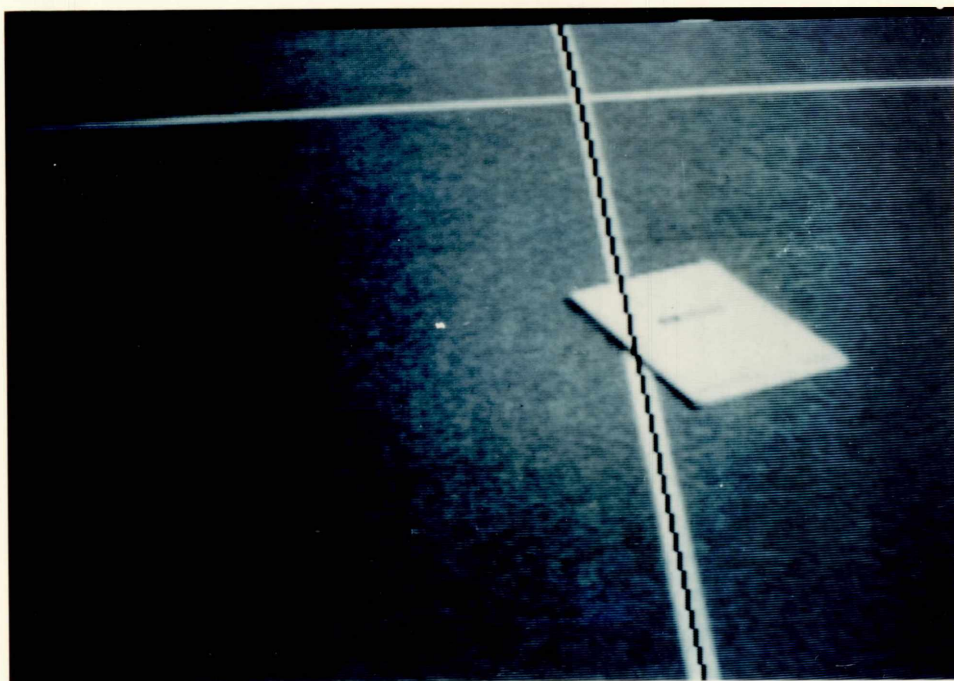
Figure 9.5 shows the marker observed by the CCD camera on board the vehicle. Figure 9.5A shows the image after processing and with a cross drawn on the detected marker. Figure 9.5B shows that the image can be detected in the presence of clutter.

The resolution of the vision system depends on the scanned area of the camera. This is dependent on the lens selected, the height of the camera and the angle of inclination of the camera. Closer objects have a higher resolution than distant objects. In figure 9.4 and figure 9.5, a TV lens 16mm, 1:1.6 is used. The bottom of the image is 1m in front of the vehicle and the scanned width is 41cm. The scanned width at the top is 82cm. Therefore the resolution of each pixel at the bottom of the image is 1.6 mm (41cm/256 pixels) and the resolution at the top of the image is 3.2 mm (82cm/256pixels).

For the wide angle lens that will ultimately be used for our prototype vehicle, the image of the area 1m in front of the vehicle must cover the pathway markers. The pathway markers are 1.5m apart. Therefore the pixel resolution at the point on the image where both the pathway markers are visible is 5.85mm (1.5m/256pixels).

The width constraint on the line was sufficient to distinguish the "clutter" from the line. The main difficulty encountered was surface reflection of sunlight on shiny floors. These surface reflectance patches are of greater intensity than the markers or lines. The width constraint on the line will reject any false readings obtained in this situation. Under this condition the vehicle may not get an update until the surface reflectance patch is crossed.

A.



B.

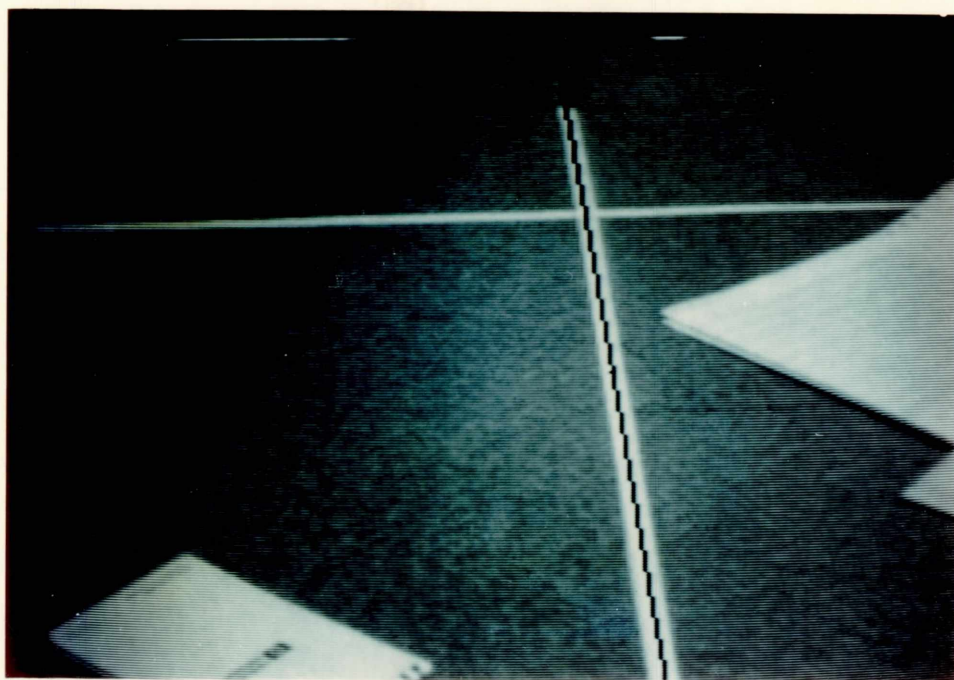
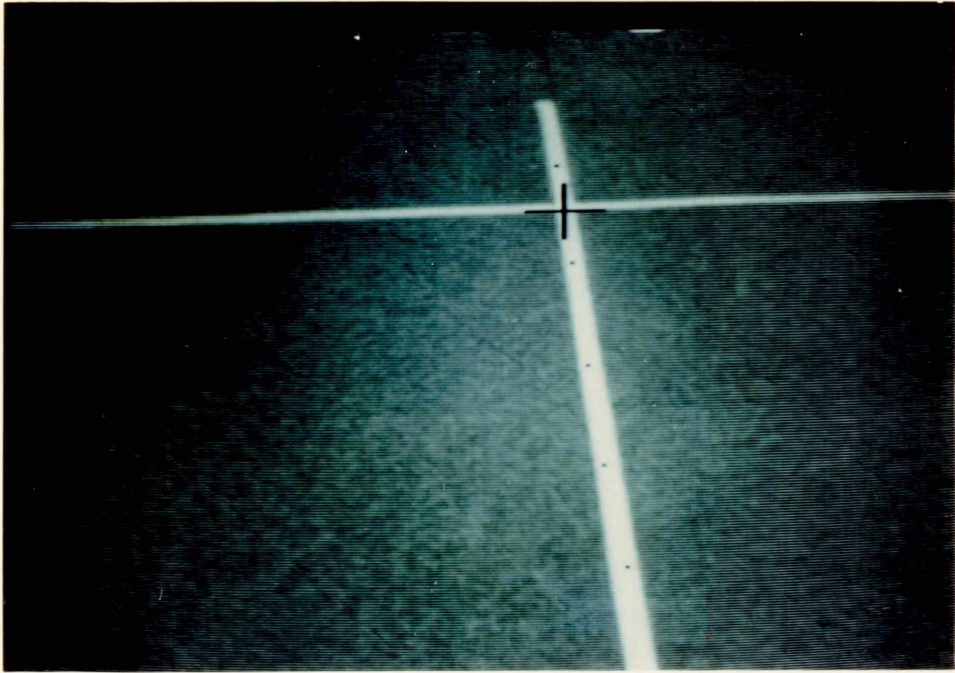


Figure 9.4 Line observed by the camera onboard the vehicle.

A. Image with a least squares line fitted onto the valid points.

B. Least squares line fit in the presence of clutter

A.



B.

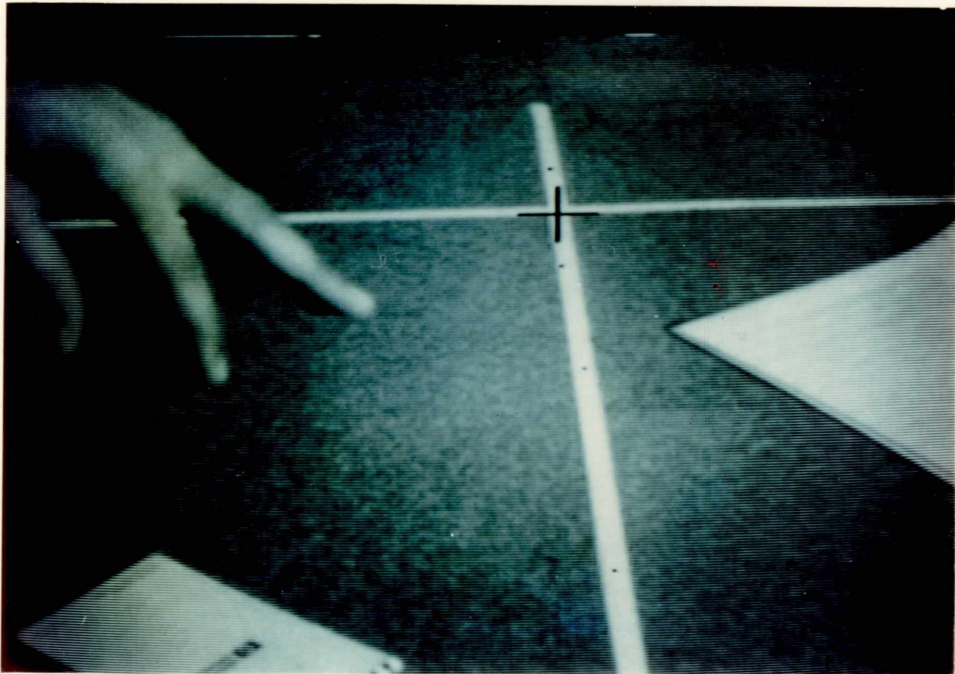


Figure 9.5 The marker observed by the camera onboard the vehicle.
A. The image with a cross drawn on the detected marker.
B. Marker detection in the presence of clutter

9.4 Obstacle Detection

The ultrasonic obstacle detection system was designed to scan the area in front of the vehicle. Considering the known problems with ultrasonic range finding, no attempt was made to actually determine the shape of the obstacle. If an object is within 1m to 2m in front, in the path of the vehicle, the vehicle will slow down. If it is within 1m, the vehicle stops. The transducers were placed such that the scanned area just covered the pathway 2m in front. This ensured that spurious reflections from walls, machines and other objects in the side of the vehicle did not get detected.

The obstacle detection system operates as specified. The main problem encountered was during turns and when there is an object within 2m in front but the vehicle is not required to travel that distance. Therefore when an object is detected it is always necessary to check whether, the distance to be travelled by the vehicle is greater or less than the range of the object. It is also necessary to disregard the values from the ultrasonic system, when the vehicle is executing a turn.

9.5 Conclusion

The Imperial College free ranging Automated Guided Vehicle System involved the following subsystems :

1. the supervisory system
2. the RF communication system
3. the vision system
4. the ultrasonic range finding system and
5. the HCTL1000 motion control system

All the subsystems are operational. The supervisory system can send a set of instructions to the vehicle via the communication system. The command is stored as a "batch" file in the AGV. The vehicle will execute each instruction in turn. At present the only command that can be executed directly is the "EMG" or emergency command which stops the AGV and resets the HCTL-1000 motion control IC.

Commands are available for tracking lines. When the line is found - the gradient and offset of the line is calculated. Marker location command is also available.

It is necessary for the vehicle to first check its position via the vision system. The marker should be at the centre of the image (ie. it should have pixel coordinates 128,128) and there should be no heading error observed by the vehicle.

After identifying its own position, the vehicle can execute the batch file sent by the supervisory system and thus fulfil the transportation order.

9.6 Results of System Performance

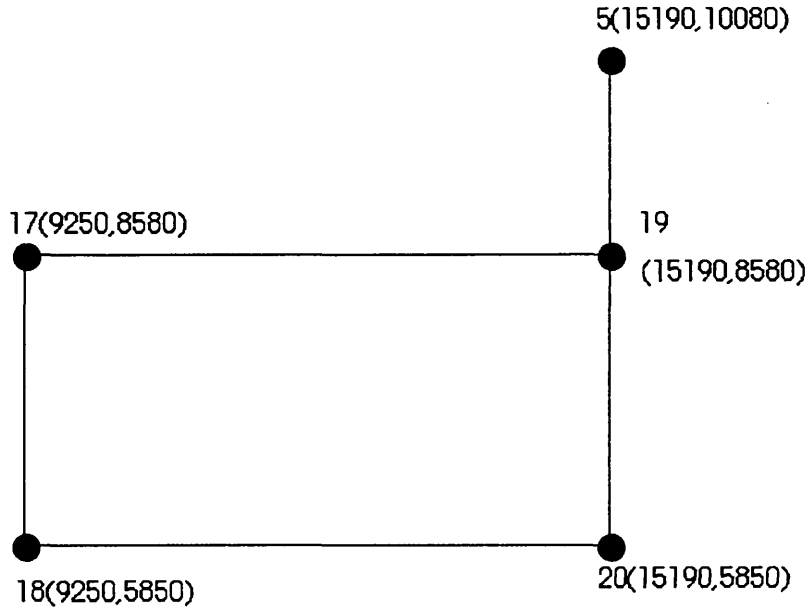


Figure 9.6 : Network of "Test" route

To test the performance of the free ranging vehicle, the vehicle was requested to move from the docking node (Node 5) (Ref. Fig 9.6) to Node 18 using the "move to node" command and return to Node 19 using a "batch file". It was then "docked" at Node 5 using the "automatic heading" command and the "automatic reference" command. The trials were carried out on carpeted surface (polyamide carpet tiles). The following instructions were given to the vehicle from the supervisory system via the communication system:

1. MVN 18 (move to node 18)
2. LDB DEMO (load batch file "DEMO.AGV")
3. RNB (run batch file "DEMO.AGV")
4. AUH (Adjust Heading Automatically)
5. AUR (Adjust Reference Automatically - ie. dock)

where the batch file "DEMO.AGV" is

TRR 90	(turn relative 90 degrees)
AHD	(adjust heading based on observed marker)
MVR 5940	(move relative 5940 mm)
TRR 90	(turn relative 90 degrees)
AHD	(adjust heading based on observed marker)
MVR 2730	(move relative 2730 mm)
END.	(end of batch file)

It was observed that the vehicle had to make a correction of the order of 1 degree at each turn. With the two heading corrections at Node 19 and 17, the vehicle lateral position at Node 18 was consistently within 5 mm of the node position. Note - without heading correction (Sec 9.2.1) the error was in the range of 20 mm for a 4m to and fro straight line path.

After "running" the batch file, it was observed that the vehicle developed a heading error at Node 20. This translated to a lateral error in the order of 3 cms at Node 19. This heading error was due to an error in the positioning of the heading correction marker on the floor. This error was corrected by the automatic heading adjust (AUH) command and the vehicle docked to within 5mm of the desired docking position.

CHAPTER 10

Free Ranging Automated Guided Vehicles (AGVs) and Autonomous Land Vehicles (ALVs)

10.1 Introduction

The present trend in robotics is towards more and more autonomy. In robotics, autonomy has been categorised by Iyengar and Kashyap [76] into four levels :

1. Teleoperation
 2. Telesensing
 3. Telerobotics and
 4. Intelligent Autonomous Robots
- Teleoperation - the extension of a persons sensing and manipulating capabilities to a remote location under his or her direct and continuous control.
 - Telesensing - involves sensing additional information about the teleoperator and task environment and then communicating the information to the human operator in a sufficiently natural way that he or she feels physically present at the remote site.
 - Telerobotics - in a telerobotic environment the human operator acts merely as a supervisor, intermittently communicating to a computer information about goals, constraints, plans, contingencies, assumptions, suggestions and orders relative to a limited task and getting back information about accomplishments, problems, concerns and raw sensor data. The telerobot executes the task based on information received from the human operator and augmented by its own sensing and "intelligence".
 - Intelligent Autonomous Robots - the operator supplies a single high level command, in response to which the robot does all the necessary task, sequencing and planning to satisfy the command. This robot can perform a variety of tasks under unpredictable conditions.

Free ranging automated guided vehicles fall under the third category. In our system the operator is the supervisory system. Based on commands received from the supervisory system, the vehicle refers to its software map and navigates from one node in the environment to another. It continually looks for markers in the environment (which are specified in the software map) to ensure that it is not lost. Since the vehicle would most

likely have to operate between expensive equipment and human operators, the vehicle cannot be allowed to deviate from the pathways specified in its software map.

Research on autonomous vehicles on the other hand, is for a vehicle that operates in uncertain and unknown dynamic environments.

In this chapter we will show that the issues and factors relevant to the implementation of free ranging automated guided vehicles are similar to those for autonomous mobile robots. We will review some of the existing autonomous vehicle projects and compare the approach followed in those systems with the approach that we have followed.

10.2 Autonomous Vehicles

The foundation of autonomous vehicle development was prepared by the following projects:

- Shaky - vision and tactile sensing
- JASON - combined acoustic and infrared proximity sensors
- RPI Rover
- JPL Rover
- Stanford Cart - acoustic, infrared sensors and stereoscopic vision
- CMU Rover - acoustic, stereoscopic vision
- Hilare - contact, acoustic, two dimensional vision and laser range finding sensors
- Yamabico mobile robots - mainly concentrated on path planning problems
- LAAS robot
- Purdue robots
- MIT's MOBOT-1 and MOBOT-2

These were relatively small vehicles and were designed to operate in-doors or in controlled outdoor environments. The issues dealt with by these vehicles were

1. perception
2. planning
3. motion control and
4. obstacle avoidance.

The chief problems were recognised to be perception (particularly visual perception) ,

world modelling and problem solving. Based on experiences of the above vehicles, larger second generation vehicles were developed. These vehicles increased their perception capability by incorporating more sensors, thus introducing redundancy in sensor data which they hope will result in the creation of a more reliable world model. Also enhancing the problem solving capability by taking advantage of the phenomenal growth in computing power in the last few years. Examples of some of these projects are :

- the FMC Autonomous Vehicle - Road follower colour camera, sonic sensors, inertial navigational system, pitch and roll sensors, remote drive colour cameras (180° front view, 60° near front view, 60° rear view), remote commander (54°/5.5° colour camera, 40°/7° FLIR), laser range finder, ultrasonic sensors.
- the DARPA Autonomous Land Vehicle (ALV)
- the CMU mobile robots - (Navlab and Terregator) colour camera, laser range finder and sonar sensors
- Ground Surveillance Robot (GSR) - vision and acoustic sensors for obstacle detection, laser range finder together with a high resolution gray -level camera and low resolution colour camera used for distant terrain and landmark recognition.
- Hermies IIB - laser range finders, sonar sensors and cameras

To summarise, the design goals of nearly all autonomous vehicles are as follows [70]:

1. Map driven mission execution - the system drives the vehicle to a given position (goal).
2. On and off road navigation - navigation environments include roads and open terrain.
3. Landmark recognition - Landmark sightings are essential when correcting for drift in the vehicles "dead reckoning system"
4. Obstacle Avoidance - absolute necessity for safety
5. Continuous motion in real time - stop-and-go motion is inadequate. Perception, planning and control should be carried out while the vehicle is moving at a reasonable speed.

10.3 Summary of Architecture for Mobile Robots

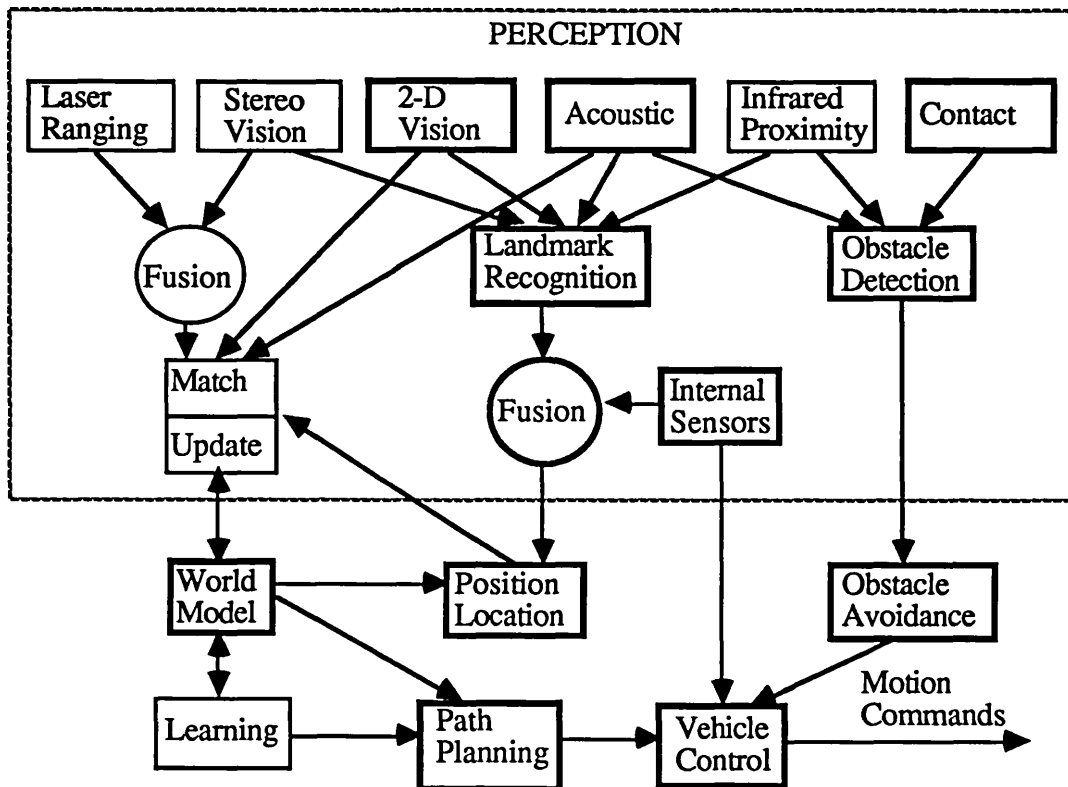


Figure 10.1 : Role of perception function in hypothetical architecture for mobile robot [72]

In Figure 10.1, Ren et al. [72] illustrates the block architecture of any autonomous vehicle project. The generic functions needed for intelligent autonomous navigation are :

1. perception, 2. vehicle control, 3. obstacle avoidance, 4. position location,
5. path planning, and 6. learning.

The perception function comprises of external sensors as inputs. The role of the perception function is :

- to match sensor data with a world model or map present in its internal database.
- update the model if there is any deviation
- recognise landmarks / objects in the environment for determining its location or to avoid obstacles.

The degree of integration and fusion of the sensor data required for each of these tasks can differ.

10.4 The Imperial College Free Ranging AGV

We believe that it is necessary to be able to precisely navigate a vehicle in a fixed structured environment before attempting uncertain and unknown environments. However since the environment cannot be known completely, it is necessary for any free ranging vehicle to plan for some degree of uncertainty. Comparing our vehicle (highlighted blocks) with the hypothetical mobile robot architecture in Figure 10.1, we note that the following elements are common to the two approaches :

- 2D vision - camera looking down on the floor,
- acoustic - two ultrasonic detectors,
- contact - touch sensitive bumper switches
- landmark recognition - pathways and simple markers
- world model - AutoCAD layout
- position location - markers in the environment
- obstacle avoidance - using the ultrasonic transducers
- vehicle control

There are only two major components are not available in our vehicle :

- 1.world model updating and
2. learning.

World model updating is necessary for creating a realistic map. Minor changes in the map can then be incorporated and taken into consideration during path planning. For intelligent behaviour learning is necessary. Without this feature the system will only be able to handle situations that the user could foresee. Both these features require range sensors. Since we are using 2D vision for marker location and ultrasonic sensing to detect objects within 2m in front of the vehicle, we do not have sufficient information to update the map.

However it is only necessary to extend the hardware to scan a larger area to obtain the required information.

The approach followed in this thesis is one of incremental changes. This ensures that all intermediate systems perform satisfactorily before going on to the next function.

CHAPTER 11

CONCLUSION AND FUTURE WORK

11.1 Conclusions

The only free-ranging Automated Guided Vehicle system in commercial operation is the Caterpillar - GEC Self Guided Vehicle. The vehicle uses a bar code seeking rotating laser system to update odometry, unfortunately, this gives rise to its inherent problems. The bar codes are two metres above floor level, which means that obstructive mounting posts must be placed in flat areas of the factory floor. The presence of a rotating laser beam at approximately eye-level, creates a hazardous work environment for its staff.

The objective of this research project was to implement a free ranging Automated Guided Vehicle system that was :

- Easy to install or reinstall in a factory environment
- Operate amongst humans and machines and
- Handle unexpected events

This thesis presents a free-ranging system that not only meets all the above requirements, but offers greater flexibility and robustness than any other system in operation. The system does not require obstrusive installations, it makes use of markers already present in its working environment. The ease with which layouts are created and modified using the industry standard drafting software package AutoCAD, further emphasises its flexibility.

We have incorporated in our data base a complete map of the environment, with clearly defined pathways for the vehicle and go -no-go areas which means the vehicle always knows where it should and should not travel. The passive camera system ensures that the vehicle is confined to its pathways and the ultrasonic obstacle detection system detects any objects two metres in front, in its pathway.

The ability to handle unexpected events has always been seen as one of the most difficult problems to solve. In our system, provisions have been made to incorporate as yet unforeseen requirements. By using a passive backplane IBM AT system, we have allowed ample scope for future changes. Raw sensor data from wheel encoder readings, the CCD

camera and the Polaroid ultrasonic transducers, are all available and can be manipulated by the software.

The Imperial College Free Ranging Automated Guided Vehicle system has involved the implementation of the following:

- Supervisory system
- Communication system
- Vision system
- Obstacle detection system
- Motion control system

All these units had to be integrated to form a complete system. The present work may thus be viewed as a successful example of " system integration."

To get a reliable picture of the surroundings, we had to consider information from the encoder, the camera, and the ultrasonic transducers. This can thus be termed as "sensor integration" or "sensor fusion."

From a desire to free the AGV from physical tracks or buried wires, the humble AGV has been transformed into a test bed for mobile robot problems.

"Research in mobile robots provides a very rich environment for the development and testing of advanced concepts in a variety of areas such as robotics, artificial intelligence, sensor understanding and integration, real world modelling, planning and high level control" [59] .

Autonomous land vehicles and mobile robots have usually been developed as part of a single major project, for example, the " Mars Rover", CMU's ALV - NAVLAB and Terregator, and DARPA's Ground Surveillance Robot (GSR). Each of these projects have had substantial financial backing. Our free- ranging vehicle system was initially conceived as a fleet of load - carriers in an industrial environment.. Each vehicle has to be relatively inexpensive. The Imperial College AGV System uses the ubiquitous IBM AT and other standard hardware, keeping costs extremely low. It is projected that the cost of each vehicle is in the range of ten to fifteen thousand pounds.

We believe that our free ranging vehicle system can be applied to a wide range of transportation applications. The system can be used in any structured environment with various levels of control depending on the number of vehicles involved. Not only can it be used in factory environments but also in office and hospital environments. It can also be used as a sentry for guarding establishments and as an aid for the invalid. The system's principle advantages over other AGV systems are in its flexibility in use, ease of installation and its potentially low manufacturing cost.

As a research topic the free ranging vehicle project is an open ended problem. With further research, the system can be constantly upgraded and improved. The project was first started as a load carrying vehicle for the factory environment and the topic has expanded to cover quite a number of issues.

11.2 Future Work

The field of free-ranging AGV's and mobile robots is still in its early stages of development. One definition of robotics could be the linking of computers to the real world to achieve a desired motion on a manipulator. The emphasis of robotics has shifted from manipulators on stationary platforms to movable platforms. For a fixed manipulator, the environment remains unchanged and thus can be ordered for ease of operation. In a movable platform, the environment is continually changing. In spite of the presence of a map of the factory layout, conditions such as different floor surfaces, lighting variations, poor communications and sudden appearance of obstacles, will affect the performance of the platform.

To handle more real world problems successfully, we have initiated a number of changes to the Imperial College system. These changes are discussed next.

11.2.1 Work in Progress

a) Development of V25 card.

To achieve a true parallel hardware architecture, it is necessary to decouple the real world functions from the planning functions. It is desirable to have the IBM AT deliver task level commands to interface cards which carry out the task. The V25 card under development in our laboratory, is a microcontroller card, based on NEC's uPD70322 single chip microcomputer, piggybacked on to a " IBM prototype card." This card communicates with the IBM AT bus via a 128 byte FIFO memory buffer. For the envisaged motion control card, the V25 card will be connected to two HCTL 1000 Motion Control ICs on the prototype card. Task level motion commands for the vehicle will be given by the IBM AT, and this will be translated into commands for the individual wheels by the V25 card. The card will perform the odometer calculations, and ensure that the specified task is fulfilled.

The "piggyback" concept implies that the V25 card can be used with any prototype card allowing any task to be delegated to this card to ease the computational burden of the main processor. The V25 microcomputer has features such as "programmable priority", "vector interrupt function", "register bank switching function" and "macro service function" which make this controller suitable for real time applications. It also accepts Intel 8086 assembly codes which means that the IBM PC is software compatible with it.

b) Video Signal Conditioning Circuit

One of the major problems with all mobile robots is the extraction of relevant information from the camera image. High intensity on the image array can be controlled by using an auto iris lens. The auto iris lens limits the maximum light level as seen by the array. Unfortunately this process also reduces the information in the lower grey levels. The envisaged video conditioning circuit will automatically compensate the video signal to maintain the level of the original grey scale image areas. Further, where the image area light level is greatly reduced below the normal video operational level (iris fully open) the circuit will maintain the image grey scale levels.

The video signal conditioning circuit will process the image before it is fed to the frame-grabber. The circuit will be able to determine the average of the maximum intensity area of any specified line. This can be used for keeping track of the high contrast pathway

markers. This circuit will work on the full resolution of the camera which is 577 by 388 pixels, this is in contrast to the 256 by 256 pixels of the frame - grabber. This information can be used to correct the lateral deviation of the vehicle in close to real time. This information is independent of the line detection information from the frame grabber.

c) Third Imperial College Free Ranging Vehicle (ICAGV-III)

This vehicle is under development at present. It is hoped that the shortcomings of the previous vehicle will be solved in this vehicle. The vehicle is smaller than ICAGV -II and will be able to negotiate standard doorways. This second vehicle will be used to study the interaction of more than one vehicle in a particular environment. All future experiments (colour vision, stereoscopic vision etc.) will be conducted on this vehicle.

11.2.2 Suggestions for Future Work

a) Vision system

In the present system, intensity is the only vision parameter that is being analysed. This can be a problem if there is insufficient contrast between the marker and surrounding pathway. Colour information can be used as an additional parameter to differentiate the marker from its surroundings. This is specially so for road following in open terrain. Using colour information the roads can be easily distinguished from the surroundings.

It is not possible to distinguish shapes reliably with monoscopic vision. It would be desirable to have a second camera so that range information could be obtained. This will allow us to use the vision system for analysing three-dimensional objects. Which would imply that objects with known coordinates in the environment could be used as location markers.

b) Map Creation

The main features differentiating our free-ranging vehicle from autonomous land vehicles is the map update and learning capability of the latter. The vehicle should be able to update its software map based on information received by its sensors. Generally range sensors (infrared range finder or ultrasonic range finder) are used for this purpose. Information

from the range finders could be superimposed on to the existing map. With this feature, it would be possible to navigate the vehicle in unknown surroundings.

c) Uneven terrain

At present, our vehicle is only suitable for use on flat surfaces. It would be useful to study different wheel configurations for different terrains.

d) Artificial Intelligence

Mobile robots are ideal test beds for "artificial intelligence" research. Sensor based obstacle avoidance problems can be tried out.

APPENDIX A1

PACSLINK RADIO DATA NETWORK

A1.1 Introduction

The PACSLINK System is a Data Acquisition and Control Network capable of handling both real time data and message packets. It is a radio based system utilising fm radio transmission (173.275MHz) in half duplex mode.

A1.2 Message Handling Protocol

PACSLINK may be operated as a simple LAN linking a number of devices together each of which is communicating over an RS232 Serial Link. This system is generally referred to as PACSNET. It is capable of configuration from a simple point to point link, upto a full network of 256 nodes, where messages may be transmitted between any two nodes on the network.

The PACSNET system operates with a single primary node and multiple secondary nodes. The Primary node is used to control the network, but the serial protocol interface to all nodes is identical. The radio system provides a fully error checked system which utilises both forward error correction and ARQ techniques to check for successful reception of all messages.

The PACSNET system operates by assembling the asynchronous characters received from the host and formatting these into a synchronous message packet with error encoding. The packet is then transmitted to the required node which will then acknowledge valid reception of the packet. If the transmission fails and the acknowledgement is not returned the packet can be retransmitted a configurable number of times. Higher levels of the interface protocol allow the PACSNET node to notify the host whether or not the message was correctly received by the remote PACSNET node.

A1.3 Messages from user device to PACSLINK unit

The format of messages from the user device to the PACSLINK unit is as follows:

<SOH><ADDR><LEN><STX><DATA><ETX><BCC><EOT>

Upon reception of a complete message as defined above the PACSLINK unit will respond with an acknowledgement. If the message has been correctly received and the BCC can be recalculated the character <ACK> (ASCII_6) will be returned. If it is not possible to interpret the message or the BCC cannot be recalculated, the PACSLINK unit will respond with the character <NAK> (ASCII_21). If the response is NAK the message should be retransmitted by the user device.

The following definitions apply :

<SOH>	(ASCII_1) Start of Header is used to denote the beginning of a message header.
<ADDR>	The Device ADDRESS consists of two ASCII characters which together provide the address of the PACSLINK unit which is to receive the message. The number will be in the range 0..FF. Both characters must be present so address numbers less than 10 will require the leading zero. Communication between secondary nodes is currently reserved for future expansion so all messages from secondary nodes will contain a device address of zero.
<LEN>	The LENGTH field consists of two ASCII characters which together provide the number of characters in the data field. The number will be in the range 0-FF. Both characters must be present so lengths less than 10 will require the leading zero.
<STX>	(ASCII_2) Start of Text provides a delimiter between the four byte message header and the information that is to be actually transmitted.
<DATA>	The data part of the message is that which is required to be transmitted to the receiving node. Messages from the main node to the secondary will be limited to a maximum of 255 characters.
<ETX>	(ASCII_3) End of Text is used to denote the end of the current message.
<BCC>	The Block Check Character provides insurance against corruption of data on the RS232 link between controlling node and PACSLINK unit. It is calculated by the Exclusive-Oring addition of all characters in the message

from the <SOH> up to and including <ETX>. <BCC> is a single character representing this value.

<EOT> (ASCII_4) End of Transmission will be used as the message terminator.

A1.4 Message from the PACSLINK Unit to User device

The format of messages from the PACSLINK to the user device is as follows:

<SOH><ADDR><LEN><STX><DATA><ETX><BCC><EOT>

Upon reception of a complete message as defined above the user device should respond with an acknowledgement. If the message has been correctly received and the BCC can be recalculated the character <ACK> (ASCII_6) should be returned. If it is not possible to interpret the message or the BCC cannot be recalculated the user device should respond with the character <NAK> (ASCII_21). If a NAK is returned to the PACSLINK unit, or there is no response within the timeout period, it will retransmit the message. The message will be retransmitted a total of three times at which point it will be assumed that communication with the user device is not possible and the message will be discarded.

This format provides for a variable length message which is delimited by a number of control codes. The definition of the fixed format then allows full data transparency so that no limitation is put on the codes that may be used by the user devices in the Data part of the message.

A1.4 Acknowledgement of reception at Remote PACSLINK unit

The simple acknowledgement described in the preceding sections provide an effective message exchange between the user device and the PACSLINK unit. It is however important that the user device is informed whether or not the message actually reaches the destination at a second PACSLINK unit. This is achieved with a secondary acknowledgement.

The format of the secondary acknowledgement is as follows:

<SOH><ADDR><ACK><ETX><EOT>

or

<SOH><ADDR><NAK><ETX><EOT>

The use of the fixed format and enclosing the acknowledgement within the SOH and ETX delimiters should prevent confusion with the simple RS232 acknowledgements.

The use of the device address to identify the source of the remote acknowledgements requires the limitation that only a single message can be unacknowledged by a particular device. If a second message can be unacknowledged by a particular device. If a second message is sent to a particular device prior to a remote acknowledgement from that device for the previous message the PACSLINK unit will refuse to accept the message and notify the user device by using a primary NAK in response to the message.

APPENDIX A2

HCTL1000 MOTION CONTROL IC SPECIFICATION

A2.1 Introduction

The HCTL-1000 is a high performance, general purpose motion control IC fabricated in Hewlett-Packard NMOS technology. It performs all the time-intensive tasks of digital motion control, thereby freeing the host processor for other tasks. The simple programmability of all control parameters provides the user with maximum flexibility and quick design of all control systems with a minimum number of components. All that is needed for a complete servo system is a host processor to specify commands, an amplifier and motor with an incremental encoder. No analog compensation or velocity feedback is necessary (Figure A2.1).

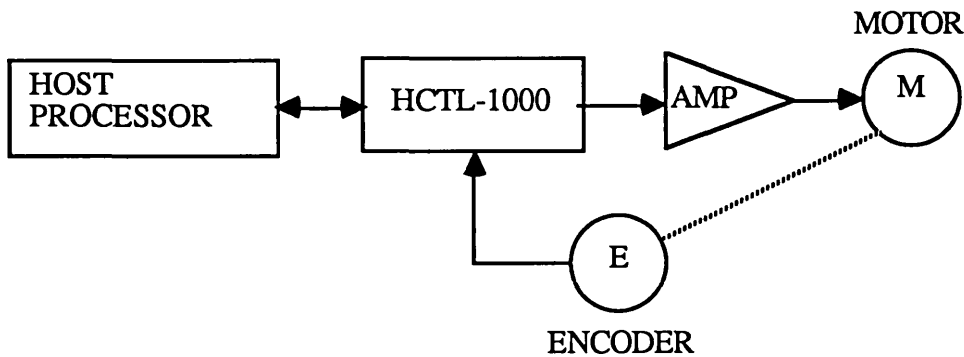


Figure A2.1 HCTL-1000 System Block Diagram

A2.2 Theory of Operation

The HCTL-1000 is a general purpose motor controller which provides position and velocity control for dc, dc brushless and stepper motors. The HCTL-1000 receives its input commands from a host processor and position feedback from an incremental encoder with quadrature output. An 8-bit bidirectional multiplexed address/data bus interfaces the HCTL-1000 to the host processor. The encoder feedback is decoded into quadrature counts

and a 24-bit counter keeps track of position. The HCTL-1000 executes any one of four control algorithms selected by the user. The four control modes are:

- Position Control
- Proportional Velocity Control
- Trapezoidal Profile Control for point to point moves
- Integral Velocity Control with continuous velocity profiling using linear acceleration.

The resident Position Profile Generator calculates the necessary profiles for Trapezoidal Profile Control and Integral Velocity Control. The HCTL-1000 compares the desired position (or velocity) to compute compensated motor commands using a programmable digital filter $D(z)$. The motor command is externally available at the Motor Command Port as an 8-bit byte and at the PWM Port as a Pulse Width Modulated (PWM) signal.

The HCTL-1000 has the capability of providing electronic commutation for dc brushless and stepper motors. Using the encoder position information, the motor phases are enabled in the correct sequence. The commutator is fully programmable to encompass most encoder combinations. In addition, phase overlap and phase advance can be programmed to improve torque ripple and high speed performance. the HCTL-1000 contains a number of flags including, two externally available flags, Profile and Initialization, which allow the user to see and check the status of the controller. It also has two emergency flags Limit and Stop, which allow operation of the HCTL-1000 to be interrupted under emergency conditions

The HCTL-1000 controller is a digitally sampled data system. While information from the host processor is accepted asynchronously with respect to the control functions, the motor command is computed on a discrete sample time basis. The sample timer is programmable.

A2.3 User Accessible Registers

The HCTL-1000 operation is controlled by a bank of 64 8-bit registers, 32 of which are user accessible. These registers contain command and configuration information necessary to properly run the controller chip. A functional block diagram showing the role of the user accessible registers is shown in Figure-A2.2. The register number is also the address. The other 32 registers are used by the internal CPU as scratch registers and should not be accessed by the user. Information on programming the registers can be found in [79].

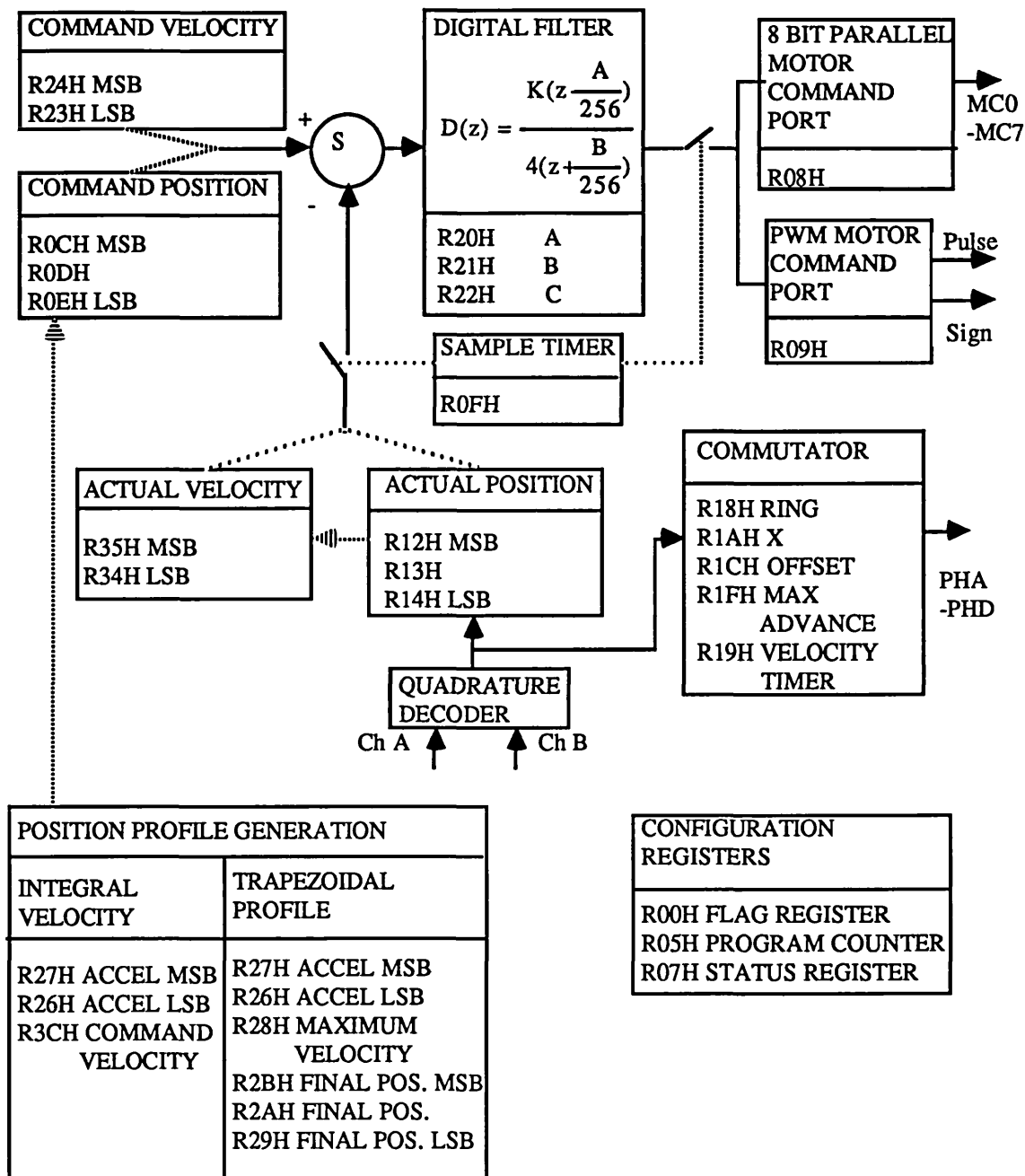


Figure A2.2 Register Block Diagram

A2.4 Operating Modes

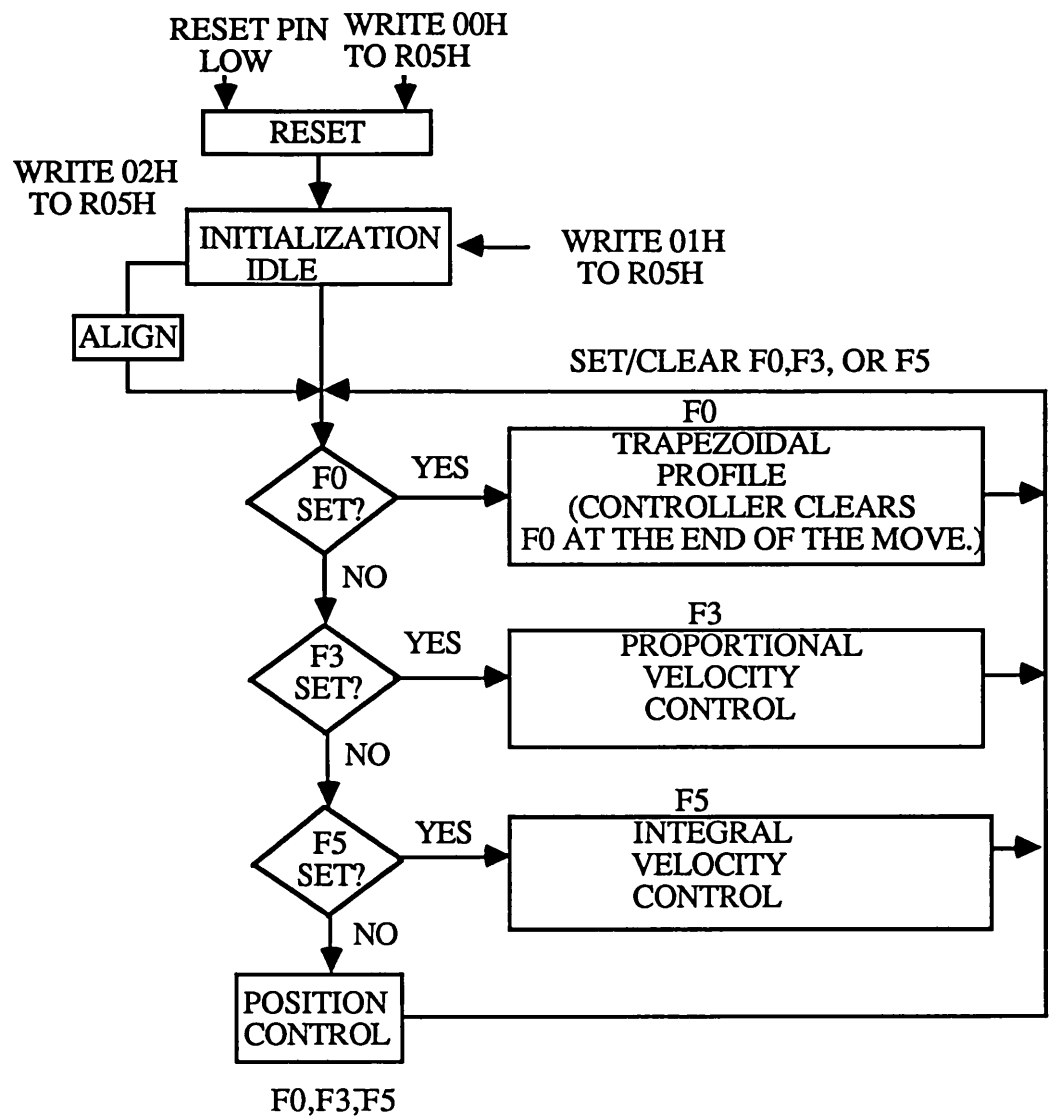


Figure A2.3 Operating Mode Flowchart

The HCTL-1000 executes any one of 3 set up routines or 4 control modes selected by the user. The three setup routines include :

- Reset
- Initialization / Idle
- Align.

The four control modes available to the user include :

- Position Control
- Proportional Velocity Control
- Trapezoidal Profile Control
- Integral Velocity Control

The HCTL-1000 switches from one mode to another as a result of one of the following three mechanisms :

1. The user writes to the Program Counter.
2. The user sets / clears flags F0, F3, or F5 by writing to the Flag Register (R00H).
3. The controller switches automatically when certain initial conditions are provided by the user.

Figure A2.3 shows a flowchart of the set up routines and control modes, and shows the commands required to switch from one mode to another.

APPENDIX A3

IMAGE PROCESSING CARD

A3.1 Introduction

The image processing card (IMAGE II) is a low cost frame grabber from Eltime Vision Systems Pte Ltd. The IMAGE II frame grabber is an IBM PC XT/AT compatible card that occupies a slot in the computer. It resides completely in the host's memory map (no input/output map requirement), and can be split into three major memory areas: the video buffer (64K bytes), the auxiliary buffers (64K bytes), and the control area (4K bytes), the total memory requirement being 132K bytes. Particular memory addresses are allocated to the three major memory areas outlined above at manufacture. The IMAGE II framegrabber has one video frame buffer and four auxiliary frame buffers, each of which is capable of resolving a video into 256 pixels (picture elements) horizontally by 256 pixels vertically, thus dividing the video frame into a total of 65536 pixels. Quantisation is to a level of 6 bits, allowing each pixel to assume one of 64 grey levels or colours. Each buffer is actually 8 bits deep, thus allowing the implementation of text or graphic overlays in any of 3 shades of grey or colours.

The single input channel and the three output channels have allocated to them four software selectable, 8 bit, programmable look up tables.

The IMAGE-II framegrabber card requires a driven sync video camera as input and a monochrome or RGB (analog signal) monitor as output.

A3.2 Set up and Usage

The IMAGE-II framegrabber card has five jumper links which must be set to suit a particular system requirement prior to installation. The links and their purposes are listed below :

LINK	POSITION	PURPOSE
LK1	A*	Video and auxialiary buffers located in contiguous segments, video buffer in highest segment.
	B	Video and auxialiary buffers located in contiguous segments, auxiliary buffer in highest segment.
LK2	A*	Control area starts at address C0000H
	B	Control area starts at address CE000H
LK3	A	Socket SK4 output is mixed sync signal derived from video camera input.
	B*	Socket SK4 output is mixed sync signal derived from video system controller.
LK4	A*	Socket SK2 output is green channel with mixed sync.
	B	Socket SK2 output is green channel without mixed sync.
LK5	A*	Socket SK6 output is separated horizontal (line) sync drive.
	B	Socket SK6 output is mixed sync signal derived from the video system controller.

Starred options indicate factory settings.

The IMAGE-II framegrabber, once installed in the host computer chassis, will have seven sockets accessible at the rear of the chassis for the connection of a camera and a monitor.

The purpose of each socket is listed below:

<u>SOCKET</u>	<u>PURPOSE</u>
SK1	Blue channel monitor output.
SK2	Green channel monitor output, with or without mixed sync

	depending on LK4 setting. This channel should be used for display.
SK3	Red channel monitor output.
SK4	Monitor mixed sync output, derived either from the video camera input or the video system controller, depending on the setting of LK3.
SK5	Separate camera vertical (field) sync output.
SK6	Separate camera horizontal (line) sync output or camera mixed sync output derived from the video system controller. Dependant on setting of LK5.
SK7	Video camera input.

Signals on sockets SK4, SK5 and SK6 are 4 Volts peak to peak (TTL compatible).

The first 64 bytes of each channel's tables make up output look up table 0. The second 64 bytes of each channel's tables make up output lookup table 1. The third 64 bytes of each channel's tables make up output look up table 2. The fourth 64 bytes of each channel's tables make up output look up table 3.

A3.3 Table Scaling

As the frame buffers are configured for 6 bits of grey level plus two overlay planes, and the card uses 8 bit output look up tables, it is important to realise that in order to utilise the full video output signal the data in the output tables will require scaling.

Typically, for a frame buffer output (which forms the address to the output lookup table data) of 63, the data in the output look up table at address 63 will be $4 \times 63 = 252$.

A3.4 Video Buffer

One video buffer is available and is configured as 256 pixels x 256 pixels x 8 bits per pixel (8 planes).

Bits 5 to 0 are for grey level data and bits 6 and 7 are overlay plane bits. Overlay data can be loaded into the video buffer overlay planes from the input look up tables at the same time as grey level data is being loaded into planes 5 thru 0; that is, as the video data is being

captured.

The data loaded into the overlay planes depends upon what data resides in the upper two bits of the input look up tables.

Upon display, if overlays are enabled, the data in the overlay planes select one of three overlay colours loaded into the output look up table's overlay registers as indicated in the table overleaf.

	<u>OVERLAY BITS</u>	
	<u>7</u>	<u>6</u>
Video buffer data	0	0
Overlay colour 0	0	1
Overlay colour 1	1	0
Overlay colour 2	1	1

A3.5 Auxiliary Buffers

Four auxiliary buffers are available. All four buffers reside in the same 64K of host memory space and are paged in and out as required using a two bit code for control latch 1. Each buffer is configured as 256 pixels x 256 pixels x 8 bits per pixel with bits 7 thru 0 of each auxiliary buffer, mapped directly to bits 7 thru 0 of the video buffer.

Unlike the video buffer, data cannot be displayed on the screen directly from the auxiliary buffers.

A3.6 Memory to Screen Mapping

<u>Screen Position</u>	<u>Video Buffer Address</u>	<u>Auxiliary Buffer Address</u>
Top left	E0000	D0000
Top right	E00FF	D00FF
Bottom left	EFF00	DFF00
Bottom right	EFFFF	DFFFF

The above mapping details apply only to the host direct addressing mode.

A3.7 VSC Modes of Operation

All modes of VSC memory access are supported. See the appendices and supplied mapping information for more details.

CONTROL LATCH 1 FUNCTIONS

IT0	-	Input table select code bit 0.
IT1	-	Input table select code bit 1.
OT0	-	Output table select code bit 0.
OT1	-	Output table select code bit 1.
RA8	-	Auxiliary buffer page select bit 0.
CA8	-	Auxiliary buffer page select bit 1.

FUNCTION	BIT:	OT1	OT0	IT1	IT0	NU	NU	CA8	RA8
		7	6	5	4	3	2	1	0
Select auxiliary buffer 0		x	x	x	x	x	x	0	0
Select auxiliary buffer 1		x	x	x	x	x	x	0	1
Select auxiliary buffer 2		x	x	x	x	x	x	1	0
Select auxiliary buffer 3		x	x	x	x	x	x	1	1
Select input look up table 0		x	x	0	0	x	x	x	x
Select input look up table 1		x	x	0	1	x	x	x	x
Select input look up table 2		x	x	1	0	x	x	x	x
Select input look up table 3		x	x	1	1	x	x	x	x
Select output look up table 0		0	0	x	x	x	x	x	x
Select output look up table 1		0	1	x	x	x	x	x	x
Select output look up table 2		1	0	x	x	x	x	x	x
Select output look up table 3		1	1	x	x	x	x	x	x

At system power up, control latch 1 is set to 00

CONTROL LATCH 2 FUNCTIONS

- SC** - Serial gate control
- LIVE/** - Live or stored image display control
- OEN** - Overlay enable
- RESET /** - Video system controller reset

FUNCTION	BIT:	NU		RESET/ OEN		LIVE/ NU		NU	SC
		7	6	5	4	3	2	1	0
VSC reset active		x	x	0	x	x	x	x	x
VSC reset inactive		x	x	1	x	x	x	x	x
Overlays disabled		x	x	x	0	x	x	x	x
Overlays enabled		x	x	x	1	x	x	x	x
Live input data enabled		x	x	x	x	0	x	x	x
Live input data disabled		x	x	x	x	1	x	x	x

With live data enabled (LIVE/ = 0), and SC = 0, the video buffer outputs are disabled allowing live data to be displayed. In this situation a write mode control command can be issued by the host computer which will configure the video buffer to its input mode allowing a frame to be captured. Once the write mode control mode has been issued by the host computer the SC control line is switched to 1 (maintaining LIVE / = 0), allowing live data to enter the video buffer.

With live data disabled (LIVE/ = 1), the video buffer outputs are enabled regardless of the state of SC and the stored frame is displayed.

Bit 7 of this control latch is a read only line that indicates whether an even field or an odd field is in progress; 0 indicates an even field an 1 an odd field.

At system power up, control latch 2 is set to 00.

A3.8 Mapping Information

Host Direct Access

[illegible][illegible]

Four Auxiliary Buffers :	CA8	RA8
Buffer 0	0	0
Buffer 1	0	1
Buffer 2	1	0
Buffer 3	1	1

RA8 - Control latch 1 bit 0

CA8 - Control latch 1 bit 1

Shift Register to Memory Access

C04xx xx = Row to be transferred (00 <= xx <= FF)

Memory to Shift Register Access

C05xx xx = Row to be transferred (00 <= xx <= FF)

X - Y Indirect Access

<u>Address</u>	<u>Pointer Adjustment</u>	
	X	Y
C0100	Non	Non
C0102	Inc	Non
C0104	Dec	Non
C0106	Clr	Non
C0108	Non	Inc
C010A	Inc	Inc
C010C	Dec	Inc
C010E	Clr	Inc
C0110	Non	Dec
C0112	Inc	Dec
C0114	Dec	Dec
C0116	Clr	Dec
C0118	Non	Clr
C011A	Inc	Clr
C011C	Dec	Clr
C011E	Clr	Clr

VSC Register Access

Address	Register Function
C0000	Horizontal end sync low byte
C0002	Horizontal end sync high byte
C0004	Horizontal end blank low byte
C0006	Horizontal end blank high byte
C0008	Horizontal start blank low byte
C000A	Horizontal start blank high byte
C000C	Horizontal total low byte
C000E	Horizontal total high byte
C0010	Vertical end sync low byte
C0012	Vertical end sync high byte

C0014	Vertical end blank low byte
C0016	Vertical end blank high byte
C0018	Vertical start blank low byte
C001A	Vertical start blank high byte
C001C	Vertical total low byte
C001E	Vertical total high byte
C0020	Display address update
C0022	Not implemented
C0024	Display start low byte
C0026	Display start high byte
C0028	Vertical interrupt low byte
C002A	Vertical interrupt high byte
C002C	Control register 1 low byte
C002E	Control register 1 high byte
C0030	Control register 2 low byte
C0032	Control register 2 high byte
C0034	Status register
C0036	Not implemented
C0038	X-Y Offset register low byte
C003A	X-Y Offset register high byte
C003C	X-Y Address register low byte
C003E	X-Y Address register high byte
C0040	Display address register low byte
C0042	Display address register high byte
C0044	Vertical count register low byte
C0046	Vertical count register high byte

Input Look Up Tables

C0800 - C08FF Write only

Output Look Up Tables and Overlays

C0900 - C0903 Read / Write

Control Latch 1

C0904 Read / Write

Bit 0 ----- RA8
Bit 1 ----- CA8
Bit 2 ----- not used
Bit 3 ----- not used
Bit 4 ----- IT0
Bit 5 ----- IT1
Bit 6 ----- OT0
Bit 7 ----- OT1

Control Latch 2

C0905 Read / write (except where noted)

Bit 0 ----- SC
Bit 1 ----- not used
Bit 2 ----- not used
Bit 3 ----- LIVE /
Bit 4 ----- OEN
Bit 5 ----- RESET /
Bit 6 ----- not used
Bit 7 ----- FIELD (read only)

REFERENCES

- [1] Hollier R.H., "Automated Guided Vehicle Systems", IFS Publications Ltd, 1987.
- [2] Hammond G., "AGVs at work - Automated Guided Vehicle System", IFS Publications Ltd, 1987.
- [3] "AGV Promotional Literature", Wagner-Indumat, 1987.
- [4] "Self Guided Vehicle needs no wires (SGVS)", Automation, October 1987.
- [5] Graham G.A., "Encyclopedia of Industrial Automation", Longman Scientific and Technical, 1988.
- [6] Krieg W., "Future trends in production and distribution logistics", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 3-23, October 1988.
- [7] Lenoir J., "Financial justification for automation requires a new type of management", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 25-31, October 1988.
- [8] White J.A. , "The evolution of materials handling in warehousing", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 33-47, October 1988.
- [9] Takahashi T., "State of art of AGVs in Japan", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 51-60, October 1988.
- [10] Challis H., "Directing the future of AGVs in Japan", Automation 1986.
- [11] Rooks B., "Freeing the automated guided vehicle", The FMS Magazine, 87-89, April 1989.
- [12] "Caterpillar Material Handling - Self Guided Vehicle System", Promotional

Literature, 1987

- [13] Broadbent A.J. , "The Scheduling of Automated Guided Vehicles", Ph.D Thesis, Dept. of Mechanical Engineering, Imperial College, London, 1987.
- [14] Sen A., "Free Ranging Automated Guided Vehicle System", Internal Report, Dept. of Mechanical Engineering, Imperial College, London, Dec 1987.
- [15] Hill A., "A Supervisory System for AGVs", Final Year B.Sc Project Report, Dept. of Mechanical Engineering, Imperial College, London, June 1988.
- [16] Weng X. , "A Scheduling System for a Free Ranging Automated Guided Vehicle System", Internal Report, Dept of Mechanical Engineering, Imperial College, London, Dec. 1987.
- [17] Taghaboni F. and Tanchoco J.M.A., "A Lisp based controller for Free Ranging Automated Guided Vehicle Systems", School of Industrial Engineering, Purdue University, March 1987.
- [18] Grossman D.D., "Traffic Control of Multiple Robot Vehicles", IEEE Journal of Robotics and Automation, Vol 4, No. 5, October 1988.
- [19] Goodhead T.C., Hines E.L., Nwufoh C.N. and Stevens M., "AGV control using an intelligent environment approach", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 241-251, October 1988.
- [20] Besant C.B. and Ristic M., "A Free Ranging Automated Guided Vehicle System", Internal Report, Dept. of Mechanical Engineering, Imperial College, London, December 1987.
- [21] Rodammer F.A. and White Jr. K.P., "A Recent Survey of Production Scheduling", IEEE Trans. on Systems, Man and Cybernetics, Vol. 18, No.6, November/December 1988.
- [22] Gauthier D., Freedman P., Carayannis G. and Malowany A.S., "Interprocess Communication for Distributed Robotics", IEEE Journal of Robotics and

Automation, Vol RA-3, No.6, December 1987.

- [23] Voelcker J., "Helping Computers Communicate", IEEE Spectrum, March 1986.
- [24] Kaminski Jr. M.A., "Protocols for Communicating in the factory", IEEE Spectrum, April 1986.
- [25] Farowich S.A., "Communicating in the technical office", IEEE Spectrum, April 1986.
- [26] Schwager J., "New AGV developments offering larger flexibility to the user", Proc. 6th Int. Conf. Automated Guided Vehicle Systems, 89-99, October 1988.
- [27] Cole R., "Computer Communications", 2nd ed. Macmillan Education Ltd. 1986.
- [28] Halsall F., "Introduction to Data Communications and Computer Networks", Addison-Wesley Publishing Ltd 1985.
- [29] Coates R.F.W., "Modern Communication Systems", Macmillan Press Ltd., 1982.
- [30] Taub H. and Shilling D.L., "Principles of Communication Systems", McGraw Hill Inc. 1986.
- [31] Erickson I.L., "Implementing a Carrier-Band Node Using VLSI", IEEE Transactions on Industrial Electronics, Vol. 35, No. 1, February 1988.
- [32] Schutz H.A. , "The Role of MAP in Factory Integration", IEEE Transactions on Industrial Electronics, Vol. 35, No. 1, February 1988.
- [33] Premi S.K., "The design of a free-ranging Automated Guided Vehicle (AGV) System", PhD Thesis, Department of Mechanical Engineering, Imperial College, University of London, pp. 108-112, April 1985.
- [34] Premi S.K. and Besant C.B. , "A Review of Various Guidance Techniques that can be used by Mobile Robots or AGVs", Proc. 2nd Int. Conf. on

AGVs, pp195 - 209, June 1983.

- [35] Boegli P., "A comparative evaluation of AGV navigation techniques", Proc. 3rd Int. Conf.on Automated Guided Vehicles, pp 169 - 179, October 1985.
- [36] Kanayama T. and Yuta S. , "Vehicle path specifications by a sequence of straight lines". IEEE Jour. of Robotics and Automation, Vol.4, No.3, pp 265 - 276, June 1988.
- [37] Hongo T., Arakawa H., Sugimoto G., Tampe K. and Yamamoto Y., "An Automated Guided System of a Self Controlled Vehicle", IEEE Trans. on Industrial Electronics, Vol.34, No.1, pp 5 - 10, Feb 1987.
- [38] Julliere M. and Bazin E., "Video Guiding of an Industrial Cart". Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 171 - 178, Oct 1988.
- [39] McGillem C.D., Callison J.D. and Rappaport T.S., "Autonomous roving navigation, control and communication". Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 179 - 188, Oct 1988.
- [40] Pritschow G., Jantzer M. and Schumacher H., "A control system for free ranging robots in production lines". Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 189 - 197, Oct 1988.
- [41] Wiklund U., Andersson U., and Hyypa K., " AGV navigation by angle measurements". Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 199 -212, Oct 1988.
- [42] Alimenti R., Bogaert M. and Delaunoy B., " Knowledge based navigation system for locally free ranging vehicle". Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 213 - 222, Oct 1988.
- [43] Van Brussel H., Van Helsdingen C.C. and Machiels K., " FROG - free ranging on grid : new perspectives in automated transport", Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 223 - 232, Oct 1988.

- [44] Wakaumi H., Tsukagoshi T., Miura T. and Yamauchi F., "A magnetic lattice lane AGV system with semi- autonomous movement", Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 233 - 240, Oct 1988.
- [45] Goodhead T.C., Hines E.L., Nwufoh C.N. and Stevens M., "AGV control using an intelligent environment approach", Proc. 6th Int. Conf. Automated Guided Vehicle System, pp 241 - 251, Oct 1988.
- [46] Nelson W.L., "Continuous Steering-Function Control of Robot Carts", IEEE Trans. on Industrial Electronics, Vol.36, No.3, Aug.1989.
- [47] Ren Y.T. and Walker S.P., "Position Determination of Automatically Guided Vehicles", The International Journal of Advanced Manufacturing Technology, Vol.1, No.5, pp 45-53, November 1986.
- [48] Marchington P., "A Simulation Program for the ICAGV Docking Operation", Third Year Project Report, Department of Mechanical Engineering, Imperial College, London, 1987.
- [49] Widden D.E., "The Design and Implementation of a Docking System for the ICAGV", M.Sc Project Report, Department of Mechanical Engineering, Imperial College, London, 1987.
- [50] Jarvis R.A., "A Perspective for Range Finding Techniques for Computer Vision", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.PAMI-5, No.2, March 1983.
- [51] Sharma U.K. and Davis L.S., "Road Boundary Detection in Range Imagery for an Autonomous Robot", IEEE Journal of Robotics and Automation, Vol.4, No.5, October 1988.
- [52] Le Moigne J.J. and Waxman A.M., "Structured Light Patterns for Robot Mobility", IEEE Journal of Robotics and Automation, Vol.4, No.5, October 1988.
- [53] Waxman A.M., Le Moigne J.J., Davis L.S., Srinivasan B., Kushner T.R.,

- Liang E. and Siddalingaiah T., "A Visual Navigation system for Autonomous Land Vehicles", IEEE Journal of Robotics and Automation Vol.RA-3, No.2, April 1987.
- [54] Dawson B.M., "Introduction to Image Processing Algorithms", Byte, March 1987.
- [55] Fu K.S., Gonzalez R.C. and Lee C.S.G., "Robotics, Control, Sensing, Vision and Intelligence", McGraw Hill, 1987.
- [56] Ishikawa S., Kuwamoto H., and Ozawa S., "Visual Navigation of Autonomous Vehicle Using White Line Recognition", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.10, No.5, September 1988.
- [57] Fishler M.A. and Firschen O., "Intelligence : the eye, the brain, and the computer", Addison-Wesley Publishing Company Inc., 1987.
- [58] Borenstein J., and Koren Y., "Obstacle Avoidance with Ultrasonic Sensors", IEEE Journal of Robots and Automation, Vol.4, No. 2, April 1988.
- [59] Elfes A., "Sonar Based Real World Mapping and Navigation", IEEE Journal of Robotics and Automation, Vol. RA-3, No.3, June 1987.
- [60] Kelly P., "Obstacle Detection and Navigation Systems for Automated Guided Vehicles", M.Sc. Thesis, Centre for Robotics and Automation Systems, Imperial College of Science Technology and Medicine, September 1989.
- [61] Lumelsky V.J., "Effect of Kinematics on Motion Planning for Planar Robot Arms Moving Amidst Unknown Obstacles", IEEE Journal of Robotics and Automation, Vol. RA-3, No.3, June 1987.
- [62] Kavoussanos E., "The Design of a new controller for the ICAGV", MSc Thesis, Department of Mechanical Engineering, Imperial College, London, 1988.
- [63] Kanayama Y. and Yuta S., "Vehicle Path Specification by a Sequence Straight

- Lines", IEEE Journal of Robotics and Automation, Vol.4, No.3, June 1988.
- [64] "The General Purpose Motion Control IC : HCTL-1000", Technical Data Sheet, Hewlett-Packard, November 1985.
 - [65] "Design of the HCTL-1000's Digital Filter Parameters by the Combination Method", Application Note 1032, Hewlett-Packard, November 1987.
 - [66] Isik C. and Meystel A.M., "Pilot Level of a Hierarchical Controller for an Unmanned Mobile Robot", IEEE Jour. of Robotics and Automation, Vol.4, No.3, June1988.
 - [67] Harmon S.Y., "The Ground Surveillance Robot (GSR): An Autonomous Vehicle Designed to Transit Unknown Terrain", IEEE Jour. of Robotics and Automation, Vol. RA-3, No.3, June 1987.
 - [68] Burks B.L., deSaussure D., Weisbin C.R., Jones J.P. and Hamel W.R., "Autonomous Navigation, Exploration and Recognition using the Hermies-IIB", IEEE Expert, Winter 1987.
 - [69] Crowley J.L., "Coordination of Action and Perception in a Surveillance Robot", IEEE Expert, Winter 1987.
 - [70] Goto Y. and Stentz A., "Mobile Robot Navigation : The CMU System", IEEE Expert, Winter 1987.
 - [71] McTamaney, "Mobile Robots : Real Time Intelligent Control", IEEE Expert, Winter 1987.
 - [72] Ren C.L. and Kay M.G., "Multisensor Integration and Fusion in Intelligent Systems", IEEE Trans. on Systems, Man, and Cybernetics, Vol.19, No.5, September/October 1989.
 - [73] Duncan R., "A Survey of Parallel computer Architecture", IEEE Computer, February 1990.

- [74] Eglowstein H., and Diehl S., " The Multiuser Solution ", Byte, September 1989.
- [75] Birrell and Ould M.A., " A Practical Handbook for Software Development ", Cambridge University Press, 1985.
- [76] Iyengar S.S. and Kashyap R.L., " Autonomous Intelligent Machines ", IEEE Computer, June 1989.
- [77] Stallings W., " Data and Computer Communications ", Macmillan Publishing Company, 1985.
- [78] "Pacslink Radio Data Network - Serial Interface Manual Revision 3.00", PACS Ltd, Green Lane, Maybush, Hampshire S01 9FP, October 1989.
- [79] "HCTL-1000, General Purpose Motion Control IC", Hewlett-Packard Technical Data, November 1985.
- [80] Design of the HCTL-1000's Digital Filter Parameters by the Combination Method", Hewlett-Packard Application Note 1032.