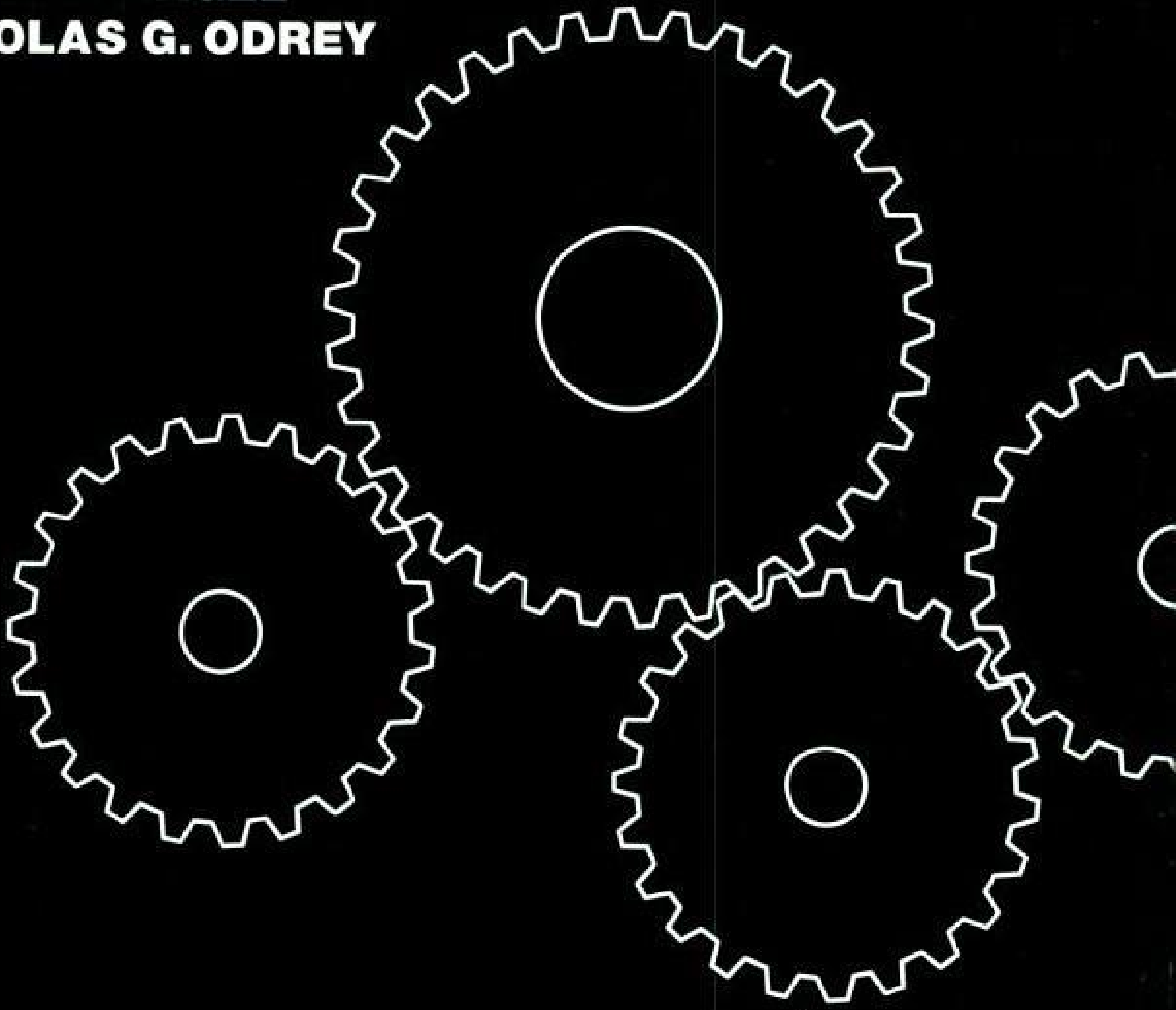
INDUSTRIAL ROBOTICS

TECHNOLOGY, PROGRAMMING, AND APPLICATIONS

MIKELL P. GROOVER
MITCHELL WEISS
ROGER N. NAGEL
NICHOLAS G. ODREY



For sale in India, Pakistan, Nepal, Bangladesh, Sri Lanka and Bhutan only

TATA McGRAW-HILL EDITION

Tata McGraw-Hill

INDUSTRIAL ROBOTICS
Technology, Programming, and Applications

Tata McGraw-Hill Edition 2008

Third reprint 2008
RAXLCDDXRCRCA

Reprinted in India by arrangement with The McGraw-Hill Companies, Inc., New York

Sales territories: India, Pakistan, Nepal, Bangladesh, Sri Lanka and Bhutan.

Library of Congress Cataloging-in-Publication Data

Groover, Mikell P., 1939—
Industrial robotics.

1. Robots, Industrial. I. Title.
TS191.8.G76 1986 629.8'92 85-18453
ISBN 0-07-024989-X

ISBN-13: 978-0-07-026509-7
ISBN-10: 0-07-026509-7

Published by Tata McGraw-Hill Publishing Company Limited, 7 West Patel Nagar, New Delhi 110 008, and printed at Sheel Print-N-Pack, Noida 201 305

The McGraw-Hill Companies

CONTENTS

Part 3 Robot Programming and Languages

Part 4 Applications Engineering for Manufacturing

Part 6 Implementation Principles and Issues

Part 7 Social Issues and the Future of Robotics

FOREWORD

In 1946 when I was a senior at Columbia University one course I would have revelled in is the one whose textbook might have been *Industrial Robotics: Technology, Programming, and Applications*. But, of course, in 1946 such a course would have been impossible. So much of the technology was not at hand and there was hardly any motivation, other than sheer fun, to build robots. Fun had already prompted the ingenious automatons that appear herein as background history.

Yet the beast was stirring. My friend, Isaac Asimov, was busy at Boston College telling us with prophetic science fiction how it could be; nay, how it should be. None of that doomsday view owing to Capek.

Surely Asimov left a subliminal message with me as I put my Columbia training to industrial control problems. Servo theory was born in World War II, an esoteric discipline called boolean algebra became digital logic and the transistor was invented after I graduated.

The story of my fortuitous association with inventor Devol is recounted herein. Collectively it all welled up, as Victor Hugo would have had it, "an idea whose time had come."

Devol and I started in 1956. We had a first robot in the field in 1961 when the all-in cost of an automotive worker was $3.50/hour. Ever since the cost of labor has increased to the current level of $21.00/hour in the automotive industry. Meanwhile, the cost of manufacturing a robot has tumbled from $60,000 in 1961 dollars to $25,000 in 1984 dollars. And a robot's capabilities today are so vastly superior to those of the first lumbering machines!

Enter *Industrial Robotics: Technology, Programming, and Applications*. Here is the whole spectrum of the state of the art. My mind boggles to think how an engineering senior armed with this background would have been greeted by my original employer's personnel office. Probably he or she would have been burned at the stake!

The authors bring it all together from design to use. And there is no shirking from peripheral activities such as CAD-CAM and Artificial Intelligence.

A would-be robot designer will learn the basic criteria. He or she will not have to "reinvent the wheel." His or her counterpart on the factory floor will be able to pick intelligently from the range of robots on the market.

My only problem in recommending this book is a selfish one. Since my colleagues and I sold Unimation, Inc., I have become a consultant. Will my clients still pay me dearly if they can read all about it in this exhaustive tutorial treatise?

Joseph F. Engelberger

PREFACE

This book is intended to provide a comprehensive survey of the technical topics related to industrial robotics. The field of robotics is emerging to become one of the important automation areas for the 1980s and 1990s. Engineers, technicians, and managers must be educated and trained in order to realize the full potential of this technology. It is our hope that this book might help to satisfy the need for text materials to develop these technically educated people.

Our book is designed principally as a text for use in undergraduate and first year graduate engineering programs. It should be suitable for courses in several departments, including mechanical, industrial, manufacturing, and electrical engineering. The book includes mechanical joint-link analysis, control systems, sensors, machine vision, end effector design, and other topics of interest to these engineering disciplines. The text would also be appropriate for courses in computer science since a substantial portion of the book is devoted to robot programming. We have also designed the book for industrial training courses, and it contains much material that is relevant to those who must install robot systems. In short, it is a book on the technology, programming, and applications of industrial robots that should serve the student of robotics making the transition from the classroom and laboratory environment of academia into the applied and practical world of industry.

We began developing the outline for this text in 1981. The contract with McGraw-Hill to write the book was signed in summer 1982. A great deal has happened in the field of robotics since then. The robotics industry has changed dramatically from one dominated by small companies to one consisting of a significant number of large corporations. We are beginning to see the fallout of the weaker companies in the industry. The technology has also developed dramatically during these several years. Computer control has become pervasive; machine vision and other sensors have captured much of the spotlight in robotics; and other advances have made robots a more sophisticated yet easier to use technology.

Also during the last several years we (the authors) have learned a great deal about the field of robotics. We have taught a course at Lehigh in the subject several times; we have developed new text materials and problem sets suitable for such a course; we have designed new robot systems; and we have developed a wide variety of industrial robot applications. The reader of this book is the beneficiary of these developments.

Something else that has happened since 1982 is that Lehigh University has hired two new faculty members whose expertise includes robotics: Roger Nagel in Fall 1982 and Nick Odrey in Fall 1983. Accordingly, we have seen our way to invite them to add their expertise to this book. Roger's education and interests are in computer science, and his professional experience includes research at the National Bureau of Standards in robot programming and machine vision. He was also Director of Automation at International Harvester where he managed projects in robotics before coming to Lehigh. Nick is an aerospace engineer turned industrial engineer. His background is heavily oriented toward the mathematical analysis of control systems and mechanical linkages (such as a robot's mechanical manipulator). Combining their knowledge with that of the two original coauthors we have a team whose expertise in robotics is substantial. The coauthors' educational backgrounds and affiliations include mechanical engineering, industrial engineering, computer science and electrical engineering. Their professional backgrounds include both academe and industry. We believe that the breadth and depth of knowledge and experience of this team has permitted us to provide a more complete and comprehensive coverage of industrial robotics than exists in any other available text on this subject.

The book contains 20 chapters, many of them technical with engineering problem sets at the end. Even the most ambitious and work-oriented instructor will have difficulty in packing all twenty chapters into a single semester. Accordingly, what must be done is to cover the chapters that are most appropriate for the particular course being offered, and send the students on their way with the hope that they will read the other chapters if the need to do so subsequently arises in their work in robotics.

In the *Instructor's and Solutions Manual* for the book (available from McGraw-Hill Book Co.), we provide suggestions about how the book might best be used to complement a course in the following disciplines: mechanical engineering, industrial or manufacturing engineering, computer science, and electrical engineering. We also recommend an outline of chapters that might be used in industrial training courses whose emphasis is on applications.

ACKNOWLEDGMENTS

There are many people and organizations to be acknowledged for their contributions and assistance in publishing this book. Our fear is that we may overlook some whose contributions were significant enough to merit inclusion. To those we have overlooked, if there are any, we apologize in advance. For their technical input and/or review of portions of the manuscript, we are indebted to the following: Robert Alexander of the Lord Corp; Jack Basiago, graduate student in Manufacturing Systems Engineering (MSE) at Lehigh; Geneen Budreau of the Lord Corp; Mark Dane of Cincinnati Milacron; Scott Dickenson, a student of robotics who questioned things; Joseph F. Engelberger, for providing us with a very nice Foreword to our book; Vernon E. Estes of the General Electric Company who, in addition to providing material for the book, was helpful during initial startup of our Robotics Laboratory at Lehigh; David Fitzpatrick of ORSI; Ray Floyd of IBM Corporation; Joseph Fromme of Cincinnati Milacron; David Hanan, MSE student; Don Hillman, Computer Science Professor at Lehigh and specialist in artificial intelligence;

Joan Juzwiak, advertising specialist for GMF Robotics; John Keefe, loyal graduate student and project engineer for special assignments; Art Kendall of IBM Corporation, good friend and advocate for Lehigh's MSE Program; Peter Kerstens, MSE student; Dana Morgan, MSE student who stimulated our thinking on a joint classification scheme; Paul J. Nelson of the IBM Corporation; John Ochs, Director of Lehigh's Computer-Aided Design Laboratory; Jim Reid, on leave from Cincinnati Milacron, currently a Lehigh MSE student; Chester A. Sadlow of Westinghouse (Unimation Inc.); Armand Small of Grumman Aerospace; Ramadan Taher, IE graduate student; Gordon Vanderbrug of Automatix; West Vogel, a graduate student in IE who prompted our thinking on robot economic analysis.

In addition, we acknowledge with gratitude the reviews of our academic peers whose comments were helpful in shaping the final version of the text: Rashpal S. Ahluwalia (Ohio State University), Stephen J. Derby (Rensselaer Polytechnic Institute), Steven Dickerson (Georgia Institute of Technology), Lyman L. Francis (University of Missouri, Rolla), Herbert Freeman (Rutgers University), Ernest L. Hall (University of Cincinnati), R. T. Johnson (University of Missouri, Rolla), Donald J. McAleece and Edward E. Messal (Indiana University and Purdue University), Daniel Metz (University of Illinois), Wolfgang Sauer (University of Massachusetts), Holger J. Sommer (Pennsylvania State University), Allen Tucker (Colgate University), Richard A. Wysk (Pennsylvania State University).

We are also indebted to B. J. Clarke, Rodger Klas, and Anne Murphy of McGraw-Hill Book Co. for their wisdom and perception in selecting us to do the book, and their patience and tolerance with us during manuscript preparation.

For their help in preparing our solutions manual for the book we would like to thank Cemal Doyden, IE graduate student; Thomas Grycan, IE undergraduate; and W. Scott Sendel, IE undergraduate. We must also acknowledge Fern Sotzing, for her secretarial assistance and friendly disposition during manuscript preparation.

Finally, we would like to express our appreciation to our wives, respectively, Bonnie, Nancy, Arlene, and Sandy, for their understanding and encouragement during the many hours that their husbands spent on the book.

Mikell P. Groover
Mitchell Weiss

ABOUT THE AUTHORS

MIKELL P. GROOVER received his B.A. in Applied Science, B.S. in Mechanical Engineering, and M.S. and Ph.D. in Industrial Engineering from Lehigh University. He is Professor of Industrial Engineering and Director of the Manufacturing Technology Laboratory at Lehigh. He is author and coauthor of two previous books on automation and CAD/CAM, respectively (published by Prentice-Hall). His areas of specialization include manufacturing technology, automation, and robotics.

MITCHELL WEISS received his B.S. in Mechanical Engineering from the Massachusetts Institute of Technology. He was employed as the applications engineer for the PUMA robot by Unimation, Inc., prior to cofounding United States Robots in 1980. He was involved in the design of robots for the company. He has started another company, ProgramMation, and is currently its President.

ROGER N. NAGEL received his B.S. from Stevens Institute of Technology and his M.S. and Ph.D. in Computer Science from the University of Maryland. His professional experience includes the National Bureau of Standards and International Harvester, Inc., as Corporate Director of Automation Technology. He is currently Director of the Institute for Robotics at Lehigh University, and Professor of Computer Science and Electrical Engineering.

NICHOLAS G. ODREY is currently the Director of the Robotics Laboratory within the Institute for Robotics at Lehigh University and is an Associate Professor of Industrial Engineering. His academic background includes a B.S. and M.S. in Aerospace Engineering. After considerable experience in the aerospace industry, he returned for his Ph.D. in Industrial Engineering with specialization in Manufacturing Systems at the Pennsylvania State University. Prior to joining Lehigh, he was associated with the University of Rhode Island, West Virginia University, the National Bureau of Standards, and was a faculty fellow with the U.S. Air Force ICAM program.

PART ONE

FUNDAMENTALS OF ROBOTICS

This first part of our book is intended to introduce the subject of industrial robotics, both its social significance and its technological importance. Robotics is a prominent component of manufacturing automation which will affect human labor at all levels, from unskilled workers to professional engineers and managers of production. Future robots may find applications outside of the factory in banks, restaurants, and even homes. It is possible, perhaps likely, that robotics will become a field, like today's computer technology, which is pervasive throughout our society. Our book undertakes the ambitious objective of providing technical literacy in this fascinating field. Part One introduces the fundamentals.

Part One contains two chapters. Chapter One defines the term robot in the context of industrial automation. It also provides a short history of the development of the technology, including a section on how robotics has been perceived by the public in science fiction stories.

Chapter Two entitled Fundamentals of Robot Technology, Programming, and Applications can almost be considered a summary of the entire book. We survey each of these three areas. Each is interrelated to the others: robotics technology is controlled by means of programming, and the ability to program a robot is dependent on its level of technology. Successful implementation of robotics in useful applications is obviously a function of the technology and programming. It is important for the reader to have an appreciation of all three areas in order to understand the technical details of each separate area. Chapter Two attempts to generate this appreciation of the three areas.

CHAPTER
ONE
INTRODUCTION

The field of robotics has its origins in science fiction. The term robot was derived from the English translation of a fantasy play written in Czechoslovakia around 1920. It took another 40 years before the modern technology of industrial robotics began. Today, robots are highly automated mechanical manipulators controlled by computers.

In this chapter, we survey some of the science fiction stories about robots, and we trace the historical development of robotics technology. Let us begin our chapter by defining the term robotics and establishing its place in relation to other types of industrial automation.

1-1 AUTOMATION AND ROBOTICS

Automation and robotics are two closely related technologies. In an industrial context, we can define automation as a technology that is concerned with the use of mechanical, electronic, and computer-based systems in the operation and control of production. Examples of this technology include transfer lines, mechanized assembly machines, feedback control systems (applied to industrial processes), numerically controlled machine tools, and robots. Accordingly, robotics is a form of industrial automation.

There are three broad classes of industrial automation: fixed automation, programmable automation, and flexible automation. Fixed automation is used when the volume of production is very high and it is therefore appropriate to design specialized equipment to process the product (or a component of a product) very efficiently and at high production rates. A good example of fixed automation can be found in the automobile industry, where highly integrated transfer lines consisting of several dozen workstations are used to perform machining operations on engine and transmission components. The economics of fixed automation are such that the cost of the special equipment can be divided over a large number of units, and the resulting unit costs are low relative to alternative methods of production. The risk encountered with fixed automation is this; since the initial investment cost is high, if the volume of production turns out to be lower than anticipated, then the unit costs become greater than anticipated. Another problem with fixed automation is that the equipment is specially designed to produce the one product, and after that product's life cycle is finished, the equipment is likely to become obsolete. For products with short life cycles, the use of fixed automation represents a big gamble.

Programmable automation is used when the volume of production is relatively low and there are a variety of products to be made. In this case, the production equipment is designed to be adaptable to variations in product configuration. This adaptability feature is accomplished by operating the equipment under the control of a "program" of instructions which has been prepared especially for the given product. The program is read into the production equipment, and the equipment performs the particular sequence of processing (or assembly) operations to make that product. In terms of economics, the cost of the programmable equipment can be spread over a large number of products even though the products are different. Because of the programming feature, and the resulting adaptability of the equipment, many different and unique products can be made economically in small batches.

The relationship of the first two types of automation, as a function of product variety and production volume, is illustrated in Fig. 1-1. There is a third category between fixed automation and programmable automation, which is called "flexible automation." Other terms used for flexible automation include "flexible manufacturing systems," (or FMS) and "computer-integrated manufacturing systems." The concept of flexible automation has only developed into practice within the past 15 or 20 years. Experience thus far with this type of automation suggests that it is most suitable for the midvolume production range, as shown in Fig. 1-1. As indicated by its position relative to the other two types, flexible systems possess some of the features of both fixed automation and programmable automation. It must be programmed for different product configurations, but the variety of configurations is usually more limited than for programmable automation, which allows a certain amount of integration to occur in the system. Flexible automated systems typically consist of a series of workstations that are interconnected by a materials-handling and storage system. A central computer is used to control the various activities that occur in the system, routing the various parts to the appropriate stations and controlling the programmed operations at the different stations.

One of the features that distinguishes programmable automation from flexible automation is that with programmable automation, the products are made in batches. When one batch is completed, the equipment is reprogram-

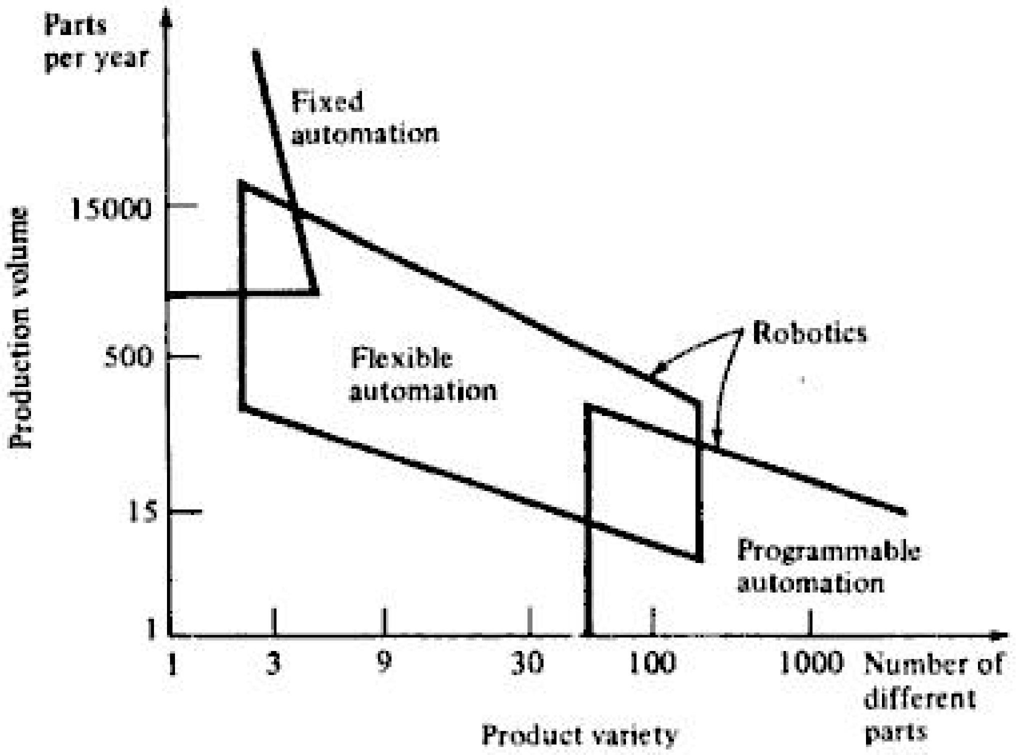


Figure 1-1 Relationship of fixed automation, programmable automation, and flexible automation as a function of production volume and product variety.

med to process the next batch. With flexible automation, different products can be made at the same time on the same manufacturing system. This feature allows a level of versatility that is not available in pure programmable automation, as we have defined it. This means that products can be produced on a flexible system in batches if that is desirable, or several different product styles can be mixed on the system. The computational power of the control computer is what makes this versatility possible.

Of the three types of automation, robotics coincides most closely with programmable automation. An industrial robot is a general-purpose, programmable machine which possesses certain anthropomorphic, or humanlike, characteristics. The most typical humanlike characteristic of present-day robots is their movable arms. The robot can be programmed to move its arm through a sequence of motions in order to perform some useful task. It will repeat that motion pattern over and over until reprogrammed to perform some other task. Hence, the programming feature allows robots to be used for a variety of different industrial operations, many of which involve the robot working together with other pieces of automated or semiautomated equipment. These operations include machine loading and unloading, spot welding, and spray painting.

The "official" definition of an industrial robot is provided by the Robotics Industries Association (RIA), formerly the Robotics Institute of America (RIA):

> An industrial robot is a reprogrammable, multifunctional manipulator designed to move materials, parts, tools, or special devices through variable programmed motions for the performance of a variety of tasks.

This definition reinforces our conclusion that industrial robots should be classified as a form of programmable automation.

While the robots themselves are examples of programmable automation, they are sometimes used in flexible automation and even fixed automation systems. These systems consist of several machines and/or robots working together, and are typically controlled by a computer or a programmable controller. A production line that performs spot welds on automobile bodies is an example of this kind of system. The welding line might consist of two dozen robots or more, and is capable of accomplishing hundreds of separate spot welds on two or three different body styles (e.g., sedans, coupes, and station wagons). The robot programs are contained in the computer or programmable controller and are downloaded to each robot for the particular automobile body that is to be welded at each station. Owing to this feature, such a line might appropriately be considered a high-production flexible automation system.

Today the human analogy of an industrial robot is very limited. Robots do not look like humans, and they do not behave like humans. Instead, they are one-armed machines which almost always operate from a fixed location on the factory floor. Future robots are likely to have a greater number of attributes similar to the attributes of humans. They are likely to have greater sensor capabilities, more intelligence, a higher level of manual dexterity, and a limited degree of mobility. There is no denying that the technology of robotics is moving in a direction to provide these machines with more and more capabilities like those of humans.

1-2 ROBOTICS IN SCIENCE FICTION

Notwithstanding the limitations of current-day robotic machines, the popular concept of a robot is that it looks and acts like a human being. This humanoid concept has been inspired and encouraged by a number of science fiction stories.

Certainly one of the first works of relevance to our discussion of robotics in science fiction was a novel by Mary Shelley, published in England in 1817. Titled *Frankenstein*, the story deals with the efforts of a scientist, Dr. Frankenstein, to create a humanoid monster which then proceeds to raise havoc in the local community. The story has been popularized in several versions over the years through the medium of motion pictures. The movie screen image of the Frankenstein monster gone astray from the plans of its well-intentioned creator has made a lasting impression on the minds of millions of people. This impression has carried over to robots where the word conjures up similar images of science and technology in danger of running amuck.

A Czechoslovakian play in the early 1920s by Karel Capek, called *Rossum's Universal Robots*, gave rise to the term robot. The Czech word "robota" means servitude or forced worker, and when translated into English, the translated word became robot. The story concerns a brilliant scientist named Rossum and his son who develop a chemical substance that is similar to protoplasm. They use the substance to manufacture robots. Their plan is that the robots will serve humankind obediently and do all physical labor. Rossum continues to make improvements in the design of the robots, eliminating unnecessary organs and other parts, and finally develops a "perfect" being. The plot takes a sour turn when the perfect robots begin to dislike their subservient role and proceed to rebel against their masters, killing all human life.

Among science fiction writers, Isaac Asimov has contributed a number of stories about robots, starting in 1939, and indeed is credited with coining the term "robotics." The picture of a robot that appears in his work is that of a well-designed, fail-safe machine that performs according to three principles. These principles were called the Three Laws of Robotics by Asimov, and they are:

1. A robot may not injure a human being or, through inaction, allow a human to be harmed.
2. A robot must obey orders given by humans except when that conflicts with the First Law.

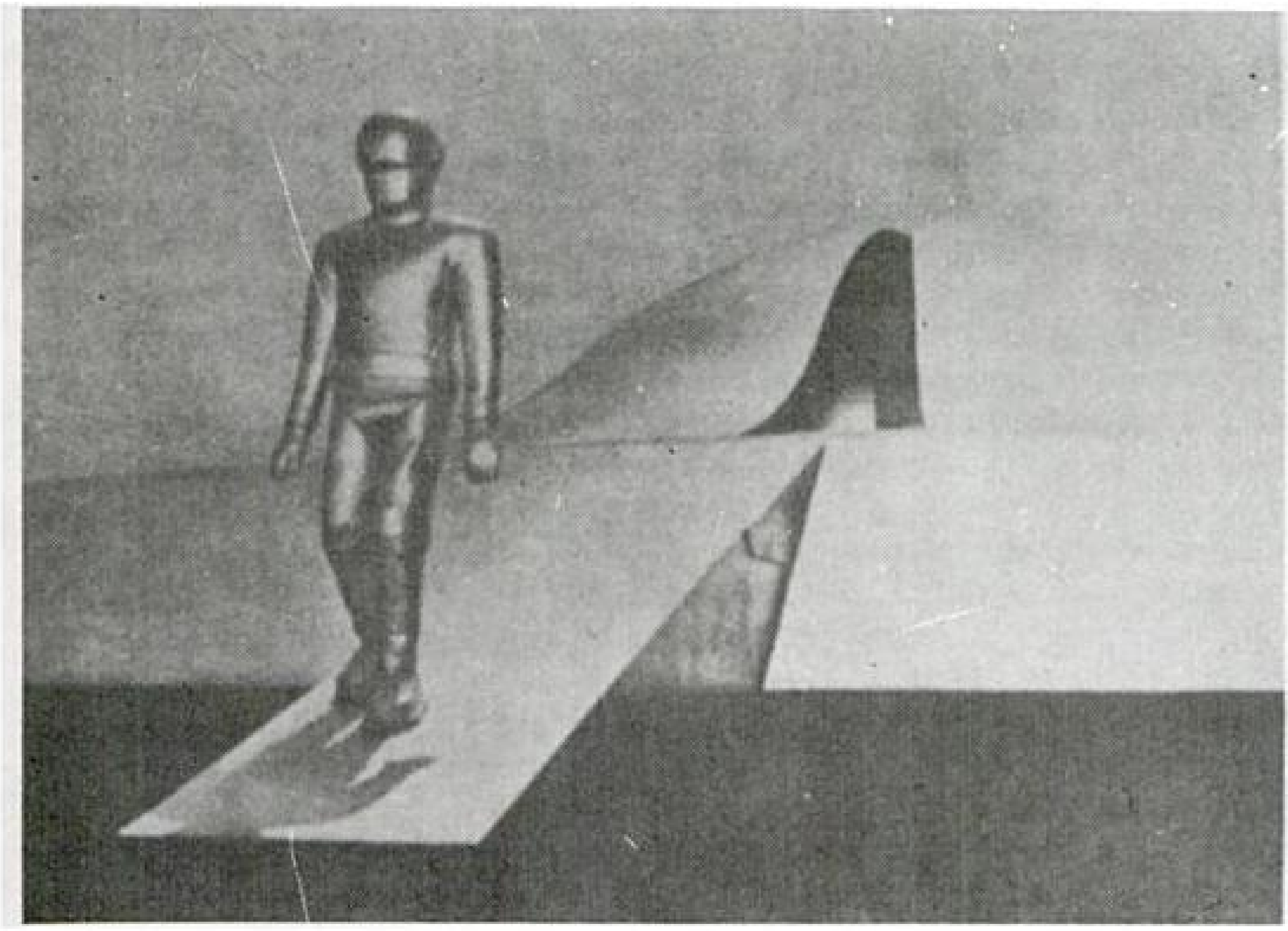


Figure 1-2 The robot Gort in the film *The Day the Earth Stood Still*. (Courtesy of Twentieth-Century-Fox.)

3. A robot must protect its own existence unless that conflicts with the First or Second Laws.

A number of movies and television shows have added to the lore of robotics, some picturing robots as friendly servants and companions, others showing them in different ways. The 1951 movie *The Day the Earth Stood Still* was about a mission from a distant planet sent to earth in a flying saucer to try to establish the basis for peace among the world's nations. The crew of the flying saucer consisted of only two members, a humanlike being and an omniscient, omnipotent, indestructible robot named Gort. The robot was a "universal" (excuse the pun) peacekeeper, and when a planet got out of line, punishment was swift and final. The mission to earth was not a complete success (obviously), but it demonstrated the terrible destructive power of future weapons. Figure 1-2 shows a photo of Gort from the film.

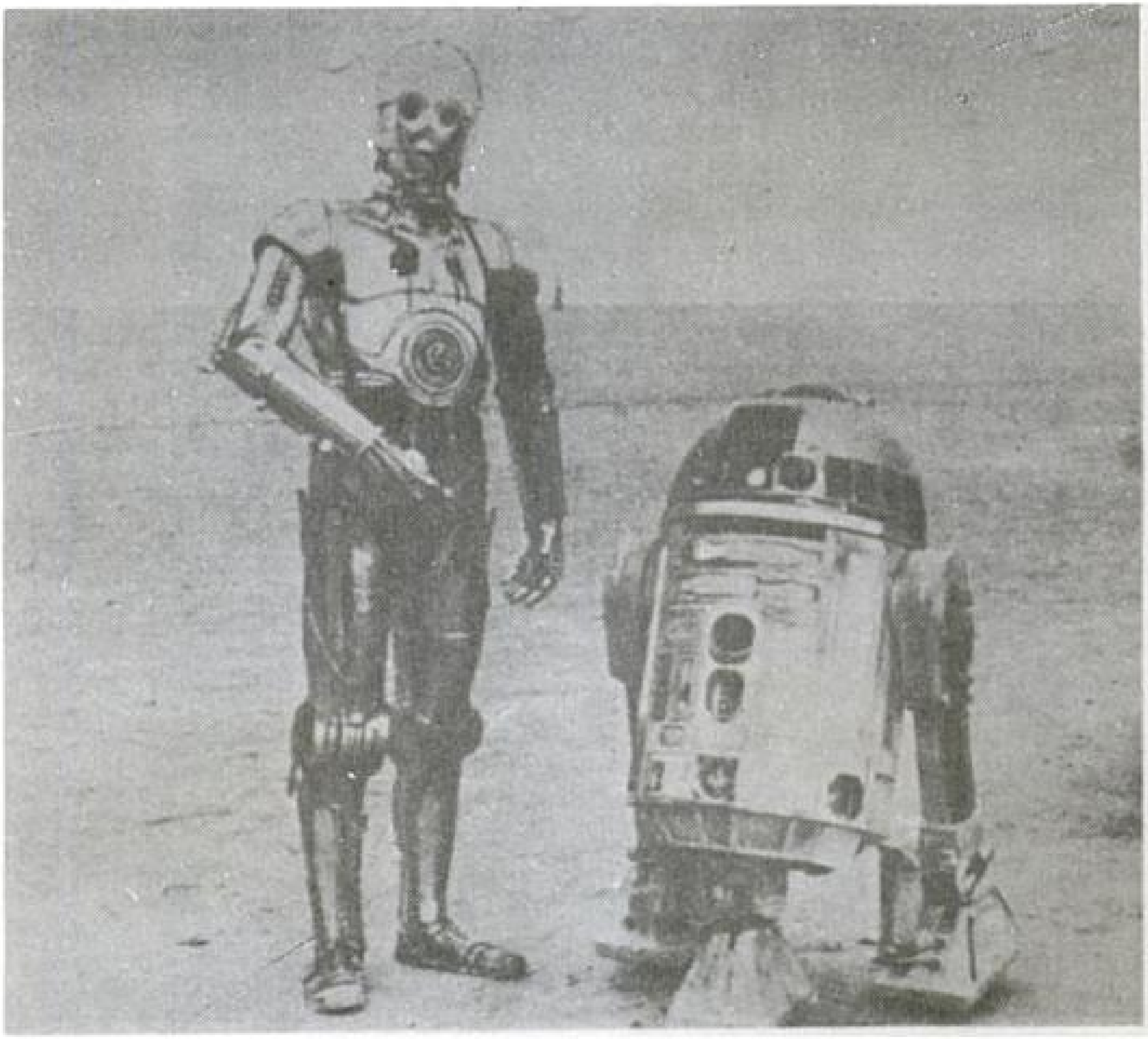


Figure 1-3 The friendly robots R2D2 and C3PO from *Star Wars*. (Courtesy of LUCASFILM LTD., © Lucasfilm Ltd. (LFL) 1977. All rights reserved.)

The 1968 movie *2001: A Space Odyssey* contained not a mechanical robot but a highly intelligent, talking computer named HAL. The job of the computer was to monitor and control the systems on-board the spaceship on its way to the planet Jupiter, and to be a friend and companion to the spaceship crew. During the voyage, one of HAL's circuits fails and its personality goes bad. It begins killing off the members of the crew in order to protect itself and is only stopped in a final contest between itself and the one remaining crew member.

The *Star Wars* series (*Star Wars* in 1977, *The Empire Strikes Back* in 1980, and *The Return of the Jedi* in 1983) pictured robots as friendly, harmless machines. The robots, R2D2 and C3PO (Figure 1-3), are able to move around, they are intelligent, and they can communicate with their human masters. They do not play major roles in these movies except mostly as comic relief. However, to movie audiences, they stand as significant characters because they are so benevolent and because they represent the opportunities offered by robotics and other advanced technologies to be helpful and unthreatening to humans.

1-3 A BRIEF HISTORY OF ROBOTICS

Science fiction has no doubt contributed to the development of robotics, by planting ideas in the minds of young people who might embark on careers in robotics, and by creating awareness among the public about this technology. We should also identify certain technological developments over the years that have contributed to the substance of robotics. Table 1-1 presents a chronological listing which summarizes the historical developments in the technology of robotics.

Some of the early developments in the field of automata deserve mention although not all of them deal directly with robotics. In the seventeenth and eighteenth centuries, there were a number of ingenious mechanical devices that had some of the features of robots. Jacques de Vaucanson built several human-sized musicians in the mid-1700s. Essentially these were mechanical robots designed for a specific purpose: entertainment. In 1805, Henri Maillardet constructed a mechanical doll which was capable of drawing pictures. A series of cams were used as the "program" to guide the device in the process of writing and drawing. Maillardet's writing doll is on display in the Franklin Institute in Philadelphia, Pennsylvania. These mechanical creations of human form must be regarded as isolated inventions reflecting the genius of men who were well ahead of their time. There were other mechanical inventions during the industrial revolution, created by minds of equal genius, many of which were directed at the business of textile production. These included Hargreaves' spinning jenny (1770), Crompton's mule spinner (1779), Cartwright's power loom (1785), the Jacquard loom (1801), and others.

In more recent times, numerical control and telecherics are two important

Table 1-1 Chronology of developments related to robotics technology, including significant robot applications

Date	Development
mid-1700s	J. de Vaucanson built several human-sized mechanical dolls that played music.
1801	J. Jacquard invented the Jacquard loom, a programmable machine for weaving threads or yarn into cloth.
1805	H. Maillardet constructed a mechanical doll capable of drawing pictures.
1946	American inventor G. C. Devol developed a controller device that could record electrical signals magnetically and play them back to operate a mechanical machine. U.S. patent issued in 1952.
1951	Development work on teleoperators (remote-control manipulators) for handling radioactive materials. Related U.S. patents issued to Goertz (1954) and Bergsland (1958).
1952	Prototype Numerical Control machine demonstrated at the Massachusetts Institute of Technology after several years of development. Part programming language called APT (Automatically Programmed Tooling) subsequently developed and released in 1961.
1954	British inventor C. W. Kenward applied for patent for robot design. British patent issued in 1957.
1954	G. C. Devol develops designs for "programmed article transfer." U.S. patent issued for design in 1961.
1959	First commercial robot introduced by Planet Corporation. It was controlled by limit switches and cams.
1960	First "Unimate" robot introduced, based on Devol's "programmed article transfer." It used numerical control principles for manipulator control and was a hydraulic drive robot.
1961	Unimate robot installed at Ford Motor Company for tending a die casting machine.
1966	Trallfa, a Norwegian firm, built and installed a spray painting robot.
1968	A mobile robot named "Shakey" developed at SRI (Stanford Research Institute). It was equipped with a variety of sensors, including a vision camera and touch sensors, and it can move about the floor.
1971	The "Stanford Arm," a small electrically powered robot arm, developed at Stanford University.
1973	First computer-type robot programming language developed at SRI for research called WAVE. Followed by the language AL in 1974. The two languages were subsequently developed into the commercial VAL language for Unimation by Victor Scheinman and Bruce Simano.
1974	ASEA introduced the all-electric drive IRb6 robot.
1974	Kawasaki, under Unimation license, installed arc-welding operation for motorcycle frames.
1974	Cincinnati Milacron introduced the T^3 robot with computer control.
1975	Olivetti "Sigma" robot used in assembly operation—one of the very first assembly applications of robotics.

Table 1-1 (*cont.*)

Date	Development
1976	Remote Center Compliance (RCC) device for part insertion in assembly developed at Charles Stark Draper Labs in United States.
1978	PUMA (Programmable Universal Machine for Assembly) robot introduced for assembly by Unimation, based on designs from a General Motors study.
1978	Cincinnati Milacron T^3 robot adapted and programmed to perform drilling and routing operations on aircraft components, under Air Force ICAM (Integrated Computer-Aided Manufacturing) sponsorship.
1979	Development of SCARA type robot (Selective Compliance Arm for Robotic Assembly) at Yamanashi University in Japan for assembly. Several commercial SCARA robots introduced around 1981.
1980	Bin-picking robotic system demonstrated at University of Rhode Island. Using machine vision, the system was capable of picking parts in random orientations and positions out of a bin.
1981	A "direct-drive robot" developed at Carnegie-Mellon University. It used electric motors located at the manipulator joints without the usual mechanical transmission linkages used on most robots.
1982	IBM introduces the RS-1 robot for assembly, based on several years of in-house development. It is a box-frame robot, using an arm consisting of three orthogonal slides. The robot language AML, developed by IBM, also introduced to program the RS-1.
1983	Report issued on research at Westinghouse Corp. under National Science Foundation sponsorship on "adaptable-programmable assembly system" (APAS), a pilot project for a flexible automated assembly line using robots.
1984	Several off-line programming systems demonstrated at the Robots 8 show. Typical operation of these systems allowed the robot program to be developed using interactive graphics on a personal computer and then downloaded to the robot.

technologies in the development of robotics. Numerical control (NC) was developed for machine tools in the late 1940s and early 1950s. As its name suggests, numerical control involves the control of the actions of a machine tool by means of numbers. It is based on the original work of John Parsons who conceived of using punched cards containing position data to control the axes of a machine tool. He demonstrated his concept to the United States Air Force, which proceeded to support a research and development project at the Massachusetts Institute of Technology. The MIT project used a three-axis milling machine to demonstrate the prototype for NC in 1952. Subsequent work at MIT led to the development of APT (Automatically Programmed Tooling), a part programming language to accomplish the programming of the NC machine tool. It is interesting to note that the Jacquard loom and the player piano, developed around 1876, can be considered to be precursors of the modern NC machine tool. Both operated using a form of punched paper tape as a program to control the actions of the respective machines.

The field of telecherics deals with the use of a remote manipulator controlled by a human being. Sometimes called a teleoperator, the remote manipulator is a mechanical device which translates the motions of the human operator into corresponding motions at a remote location. A common use of a teleoperator is in the handling of dangerous substances, such as radioactive materials. The human can remain in a safe location; yet by peering through a leaded glass window or by viewing on closed-circuit television, the operator can guide the movements of the remote arm. Early telecheric devices were entirely mechanical, but more modern systems use a combination of mechanical systems and electronic feedback control. Work on teleoperator designs for handling radioactive materials dates back to the 1940s. Telecheric devices were used by the Atomic Energy Commission starting around the same time.

It is the combination of numerical control and telecherics that forms the basis for the modern robot. The robot is a mechanical manipulator whose motions are controlled by programming techniques very similar to those used in numerical control. There are two individuals who must be credited with recognizing the confluence of these two technologies and the potential it might offer in industrial applications. The first was a British inventor named Cyril Walter Kenward who applied for a British patent for a robotic device in March 1954. This patent was issued in 1957. The sketch for this device is shown in Fig. 1-4.

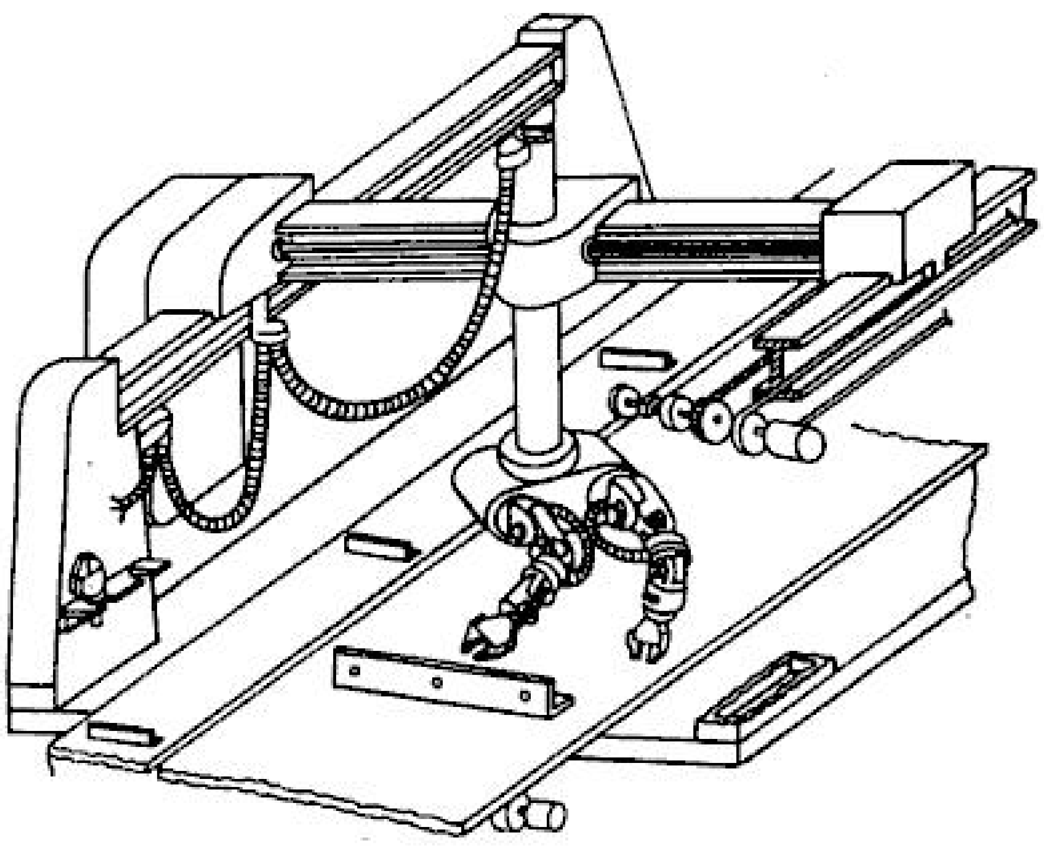


Figure 1-4 Sketch of the robotic device for which Cyril Walter Kenward was issued a British patent in 1957.

The second person who must be mentioned in this context is George C. Devol, the American inventor, who must be credited with two inventions that led to the development of modern day robots. The first was a device for recording electrical signals magnetically and playing them back to control a machine. The device is dated around 1946 and the U.S. patent for it was issued in 1952. The second invention was titled "Programmed Article Transfer" and the U.S. patent for this device was issued in 1961. The description of the device in the June 13, 1961 edition of the *U.S. Patent Record* is presented in Fig. 1-5, and the sketch based on the diagram which accompanied the description is shown in Fig. 1-6. Although Devol's patent followed Kenward's by several years, it was Devol's work that established the foundation for the modern industrial robot. What made Devol's invention into an industry in the United States rather than in the United Kingdom was the presence of a catalyst in the person of Joseph Engelberger.

Joseph F. Engelberger graduated from Columbia University with a graduate degree in physics in 1949. As a student, he had read with fascination several of Asimov's novels. By the mid-1950s he was the chief engineer for an aerospace division of a company located in Stamford, Connecticut. The division was in the business of making controls for jet engines. Hence, by the

2,988,237
PROGRAMMED ARTICLE TRANSFER
George C. Devol, Jr., Brookside Drive, Greenwich, Conn.
Filed Dec. 10, 1954, Ser. No. 474,574
28 Claims. (Cl. 214—11)

1. Apparatus having automatic control means, including a mechanical output device and power operating means therefor, position representing means coupled to said mechanical output device for conjoint operation therewith, said position representing means including an assembly of separate sensing units each having its own individual output and combinational code means sensed by and relatively movable with respect to said sensing units through a series of positions corresponding to positions of said device, said combinational code means including a uniquely identifying combination of control portions of different kinds opposite the respective sensing units in each said position such that the combination of control portions at each said position is different from the combinations of control portions in all the others of said series of positions, a program-controller having a series of recorded combinational code position symbols duplicating selected combinations in said series of positions, said program-controller having as many position-symbol sensing elements for sensing said combinational code position symbols as there are sensing units in said position representing means, a series of indidivual coincidence detectors each having its respective direct signal coupling to a corresponding one of said sensing units and to a corresponding one of said sensing elements, said power operating means having control means responsive to said coincidence detectors jointly.

Figure 1-5 Description of George C. Devol's "Programmed Article Transfer" as it appeared in the *U.S. Patent Record* dated June 13, 1961.

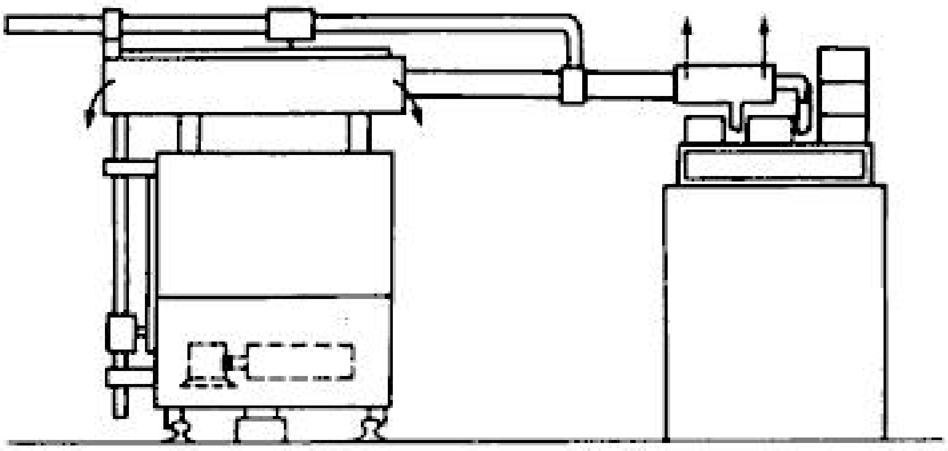


Figure 1-6 Sketch of Devol's "Programmed Article Transfer" similar to the diagram that accompanied the patent description.

time a chance meeting took place in 1956, Engelberger was predisposed by education, avocation, and occupation toward the notion of robotics. As fate would have it, Joseph Engelberger met George Devol at a cocktail party held in Fairfield, Connecticut. During the conversation, Devol told Engelberger about his invention of the programmed article transfer device, and the two subsequently began discussing the possibility of commercializing the invention. Through the financial backing of the Consolidated Diesel Electric Company (now Condec Corp.), Engelberger and Devol started to develop plans and prototypes for the universal helper, or "Unimate." In 1962, the Unimation Company was founded as a joint venture between Consolidated Diesel Electric and the Pullman Corporation. Engelberger became president of the company and has promoted the development and the application of robotics ever since. Figure 1-7 shows Engelberger and Devol being served by a Unimate.

The first recorded installation of a Unimate robot was at the Ford Motor Company for unloading a die-casting machine. (It is ironic to note that although Ford was one of the very first companies to use a robot, it refused for many years to recognize the word robot, preferring instead to use the term "universal transfer device" or UTD.) More applications followed, slowly at first, using robots not only from Unimation, but also from a number of other companies in the United States, Europe, and Japan. Some of the more significant robot installations are included in Table 1-1.

There were many other worthwhile contributions to the field of robotics, although space limits our including all of them. It is appropriate to note some of the pioneering work at Stanford University and Stanford Research Institute on computer-oriented robot languages. In 1973, the experimental language called WAVE was developed. This was followed by the development of the AL language in 1974, another language designed for research. The first commercial robot language was VAL, developed by Victor Scheinman and Bruce Simano for Unimation, Inc. The language was first used to program Unimation's PUMA robot, a relatively small jointed-arm robot whose design

Figure 1-7 Joseph Engelberger (left) and George Devol being served drinks by a Unimate robot. (Photo courtesy of Joseph F. Engelberger.)

was based on studies of assembly automation that had been done by General Motors. PUMA stands for Programmable Universal Machine for Assembly.

The Stanford work on robot languages, and much of the subsequent work that has been done in robotics, is largely based on developments in computer technology. Although computers were certainly available at the birth of the robotics industry, it was not until the mid to late 1970s that the economics were right for the use of a small computer as the robot controller. Today, nearly all robots introduced into the market use computer controls. Indeed, the field of robotics is often considered to be a combination of machine tool technology and computer science.

1-4 THE ROBOTICS MARKET AND THE FUTURE PROSPECTS

Annual sales for industrial robots have been growing in the United States at the rate of about 25 percent per year. Figure 1-8 presents the current statistics and projections for annual sales of industrial robots and the resulting number

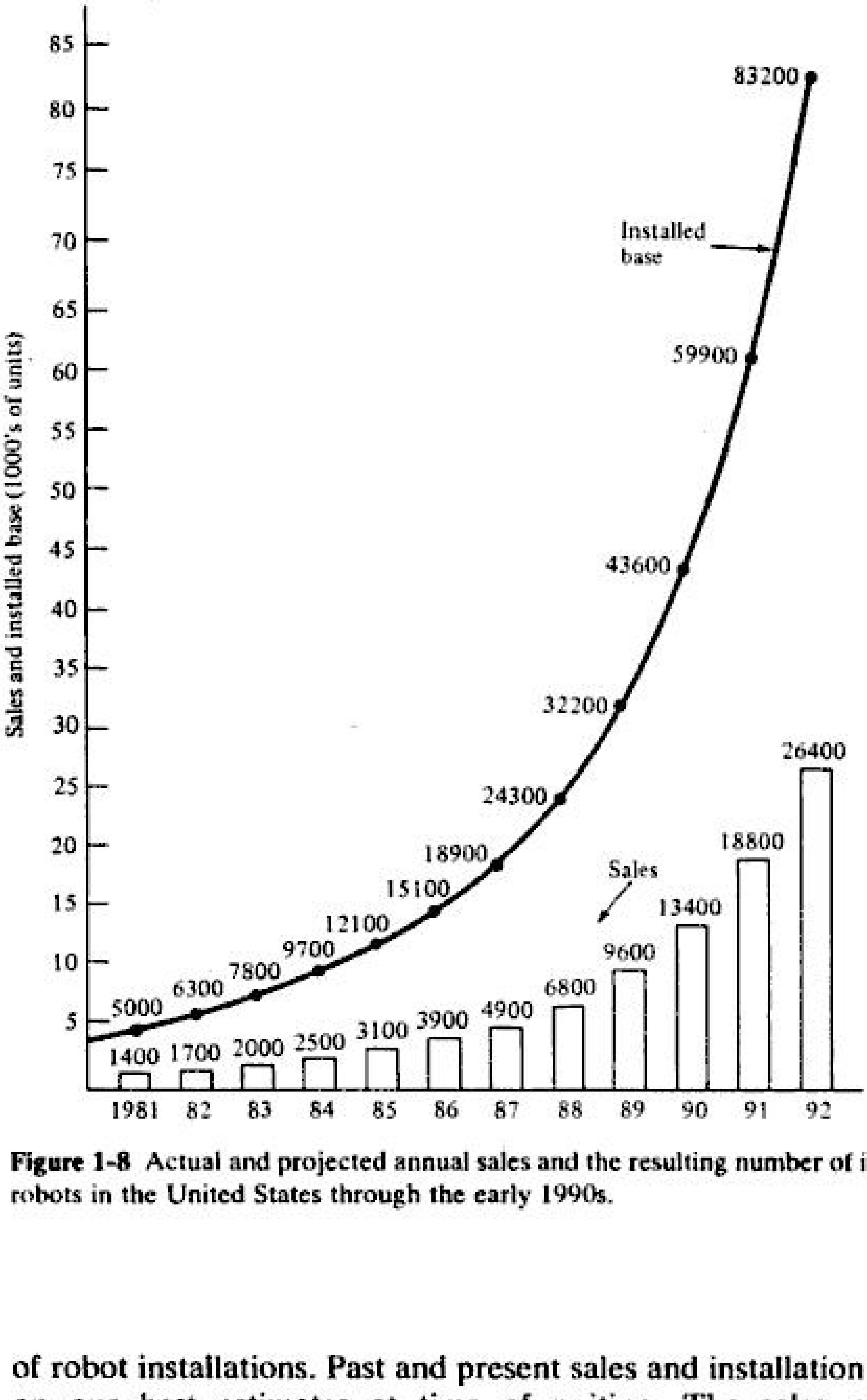


Figure 1-8 Actual and projected annual sales and the resulting number of installations of industrial robots in the United States through the early 1990s.

of robot installations. Past and present sales and installation statistics are based on our best estimates at time of writing. The sales projections show a continued average annual growth rate of 25 percent through 1987. Around 1987, we expect the growth rate to increase in the United States due to several factors. First, more people in industry are becoming aware of the technology and aware of its potential for useful applications. Second, the technology of robotics will improve over the next few years in such a way as to make robots more user friendly, easier to interface to other hardware and software, and easier to install. Third, as the market grows, we expect economies of scale in the production of robots to effect a reduction in the unit price, making robot application projects easier to justify. Fourth, the robotics market is expected to expand beyond the large corporation, which has been the traditional customer for this technology, and into the medium-sized and smaller companies. This will result in a substantial increase in the customer base for industrial robots. It cannot be determined whether these factors will all converge in the year 1988 to create a sudden surge in demand. However, for the purposes of the projection, we are using 1988 as the year when the sales growth rate will increase, and we are using 40 percent as our estimate of that new growth rate.

The projected installed base in Fig. 1-8 represents the accumulation of these annual sales, adjusted for obsolete robots that have been discarded. We believe it is reasonable to assume that installed robots will become worn out and/or technologically obsolete after an average seven-year service life. Advances in the technology and reductions in pricing will make new units relatively attractive compared to old units in service.

Robotics is a technology with a future, and it is a technology for the future. If present trends continue, and if some of the laboratory research currently underway is ultimately converted into practicable technology, robots of the future will be mobile units with one or more arms, multiple sensor capabilities, and the computational and data processing power of today's mainframe computers. They will be able to respond to human voice command. They will be able to receive general instructions and will translate those instructions using artificial intelligence into a specific set of actions required to carry them out. They will be able to see, hear, feel, apply a precisely measured force to an object, and move under their own power. In short, future robots will have many of the attributes of human beings. It is hard to imagine that robots will ever replace humans in the sense of Karel Capek's play, "Rossum's Universal Robots." On the contrary, robotics is a technology that can be harnessed solely for the benefit of humankind. However, like other technologies, there are potential dangers involved, and safeguards must be instituted to prevent its harmful use.

Getting from the present to the future will require much work in mechanical engineering, electrical engineering, computer science, industrial engineering, materials technology, manufacturing systems engineering, and the social sciences. The purpose of this book is to explore and examine these areas which constitute the technology, programming, and application of industrial robotics.

1-5 ORGANIZATION OF THIS BOOK

The text for this book is organized into seven parts. Part One is introductory. This first chapter provides the motivation and rationale for learning about robotics. Chapter Two presents an overview of robot technology and programming. This is important because in the flow of the book, we will be discussing technical topics which must be placed into context relative to other topics. Chapter Two provides the survey of the entire field necessary to establish those relationships.

Part Two examines the technical topics that relate to the robot and the peripheral hardware used with the robot. Chapter Three discusses the mechanical components of the robot and the control systems used to control the joints of the arm and wrist. Chapter Four presents some of the mathematical analysis that is used in robotics to study the motion of the manipulator. Chapter Five covers end effectors, the mechanical hands and other devices that are attached to the robot arm to perform useful work. Chapters Six and Seven are concerned with the sensors that are used in robotics, including machine vision, an important technology which is likely to find many applications in robotics work in the future.

Part Three deals with robot programming. Chapter Eight is concerned with the fundamentals of how to program robots and what the requirements for robot programming are. Chapter Nine and its appendixes cover some of the robot textual languages that are in common use. Chapter Ten presents a survey of artificial intelligence and its relationship to robotics. We anticipate that robots of the future will possess far greater intelligence and reasoning power than current-day robots, and the field of artificial intelligence will provide the methodologies for these capabilities.

Part Four is concerned with applications engineering. What are the engineering and economic problems that must be addressed in installing robots, and what are some of the applications of robots today? Chapter 11 describes work cell design and control—how to use the technology and programming of robotics in industrial applications. Chapter Twelve presents the methods that should be used to justify a robot investment. Part Five presents a survey of how robotics is used in industry. Chapters Thirteen through Fifteen discuss the various types of robot applications in manufacturing today.

Closely related to applications engineering are the implementation issues associated with the introduction of robotics into the factory. Part Six surveys some of these issues. Chapter Sixteen proposes a seven-step approach for the implementation of robotics in the firm, from initial familiarization with the technology to installation of the robot work cell. Chapter Seventeen discusses some of the additional problem areas that must be confronted during implementation. These areas include safety, training, maintenance, and quality control.

Part Seven deals with social issues and the future of robotics. Chapter Eighteen explores the possible social impact of robotics, giving particular attention to the problems confronting labor. In chapters Nineteen and Twenty, we speculate about the following questions. What will the technology of robotics be like in the future? And what kinds of applications will robots be performing in the future?

REFERENCES

1. J. S. Albus, *Brains, Behavior, and Robotics*, BYTE Books (subsidiary of McGraw-Hill), Peterborough, NH, 1981, chap. 8.

2. R. U. Ayres and S. M. Miller, "Industrial Robots on the Line," *Technology Review*, 34–47 (May/June 1982).
3. R. U. Ayres and S. M. Miller, *Robotics—Applications and Social Implications*, Ballinger, Cambridge, MA, 1983, chaps. 1 and 2.
4. R. C. Dorf, *Robotics and Automated Manufacturing*, Reston, Reston, VA, 1983, chap. 3.
5. J. F. Engelberger, *Robotics in Practice*, AMACOM (Division of American Management Association), 1980, author's preface.
6. R. M. Glorioso and F. C. C. Osorio, *Engineering Intelligent Systems*, Digital Press, Digital Equipment Corp., 1980, chap. 15.
7. M. P. Groover, *Automation, Production Systems, and Computer-Aided Manufacturing*, Prentice-Hall, Englewood Cliffs, NJ, 1980, chap. 1.
8. "Joseph F. Engelberger—The American Machinest Award," *American Machinist*, 68–75 (December 1982).
9. B. Rooks, "The Cocktail Party That Gave Birth to the Robot," *Decade of Robotics*, IFS Publications, Bedford, England, 1983.

CHAPTER
TWO

FUNDAMENTALS OF ROBOT TECHNOLOGY, PROGRAMMING, AND APPLICATIONS

Robotics is an applied engineering science that has been referred to as a combination of machine tool technology and computer science. It includes such seemingly diverse fields as machine design, control theory, microelectronics, computer programming, artificial intelligence, human factors, and production theory. Research and development are proceeding in all of these areas to improve the way robots work and think. It is likely that the research efforts will result in future robots that will make today's machines seem quite primitive. Advancements in technology will enlarge the scope of the industrial applications of robots.

Our problem in this chapter is to define the basic technology, programming, and applications of current day industrial robotics. The technical fields listed above are highly interdependent in the manner in which they are used in robotics. In order to appreciate robotics technology and programming, one must be aware of the way robots are applied in industry. In order to understand the use of sensors in robotics, one must be familiar with the way robots are programmed. To comprehend the use of an end effector, one must know that a fundamental function of a robot is to handle parts and tools. In this chapter, therefore, we provide that survey of the entire field of robotics to establish the necessary framework for the reader to relate the various topics in the chapters that follow.

To describe the technology of a robot, we must define a variety of technical features about the way the robot is constructed and the way it operates. Robots work with sensors, tools, and grippers, and these terms must be defined. The programming of robots is accomplished in several ways.

Although we discuss this subject in considerable detail later in the book, a concise description is presented in this chapter. Finally, robots are used to perform work in industry, and we provide a survey of these industrial applications. To survey these various topics, this chapter is organized into the following sections:

Robot anatomy
Work volume
Drive systems
Control systems and dynamic performance
Precision of movement
End effectors
Sensors
Robot programming and work cell control
Applications

For many of these topics, it is appropriate to delve much deeper into the subject, well beyond the basic introduction intended by this chapter. We discuss these topics in greater depth in subsequent chapters of the book.

2-1 ROBOT ANATOMY

Robot anatomy is concerned with the physical construction of the body, arm, and wrist of the machine. Most robots used in plants today are mounted on a base which is fastened to the floor. The body is attached to the base and the arm assembly is attached to the body. At the end of the arm is the wrist. The wrist consists of a number of components that allow it to be oriented in a variety of positions. Relative movements between the various components of the body, arm, and wrist are provided by a series of joints. These joint movements usually involve either rotating or sliding motions, which we will describe later in this section. The body, arm, and wrist assembly is sometimes called the manipulator.

Attached to the robot's wrist is a hand. The technical name for the hand is "end effector" and we will discuss end effectors later in this chapter and in much greater detail in a later chapter. The end effector is not considered as part of the robot's anatomy. The arm and body joints of the manipulator are used to position the end effector, and the wrist joints of the manipulator are used to orient the end effector.

Four Common Robot Configurations

Industrial robots are available in a wide variety of sizes, shapes, and physical configurations. The vast majority of today's commercially available robots possess one of four basic configurations:

1. Polar configuration
2. Cylindrical configuration
3. Cartesian coordinate configuration
4. Jointed-arm configuration

The four basic configurations are illustrated in the schematic diagrams of Fig. 2-1.

The polar configuration is pictured in part (*a*) of Fig. 2-1. It uses a telescoping arm that can be raised or lowered about a horizontal pivot. The pivot is mounted on a rotating base. These various joints provide the robot

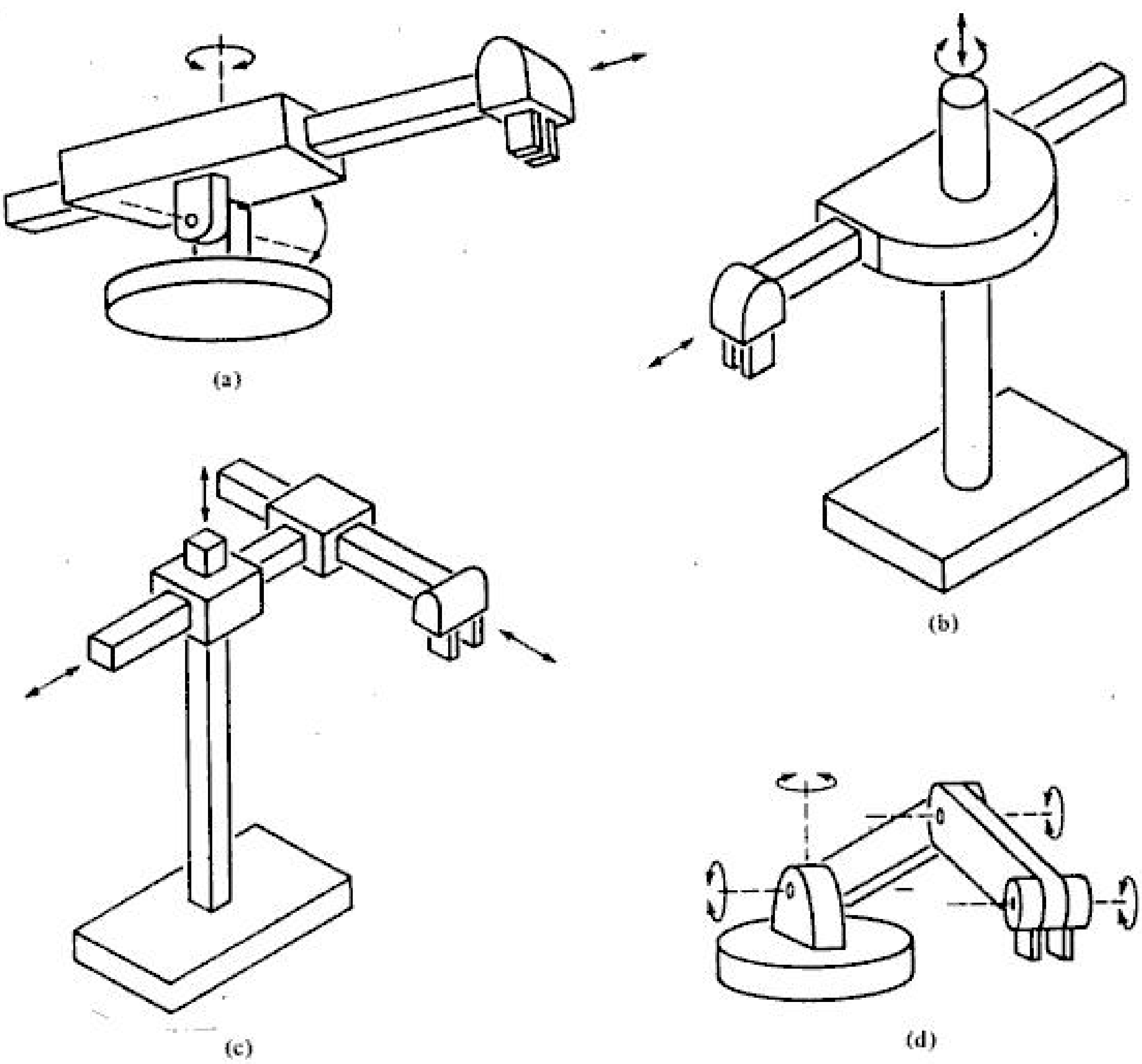


Figure 2-1 The four basic robot anatomies: (a) polar, (b) cylindrical, (c) cartesian, and (d) jointed-arm. (Reprinted from Reference [7].)

with the capability to move its arm within a spherical space, and hence the name "spherical coordinate" robot is sometimes applied to this type. A number of commercial robots possess the polar configuration. These include the familiar Unimate 2000 series, pictured in Fig. 2-2. Another robot which is much smaller than the Unimate is the MAKER 110, made by United States Robots, and illustrated in Fig. 2-3.

The cylindrical configuration, as shown in Fig. 2-1(*b*), uses a vertical column and a slide that can be moved up or down along the column. The robot arm is attached to the slide so that it can be moved radially with respect to the column. By rotating the column, the robot is capable of achieving a work space that approximates a cylinder. An example of the cylindrical configuration is pictured in Fig. 2-4.

The cartesian coordinate robot, illustrated in part (*c*) of Fig. 2-1, uses three perpendicular slides to construct the *x*, *y*, and *z* axes. Other names are sometimes applied to this configuration, including *xyz* robot and rectilinear robot. By moving the three slides relative to one another, the robot is capable

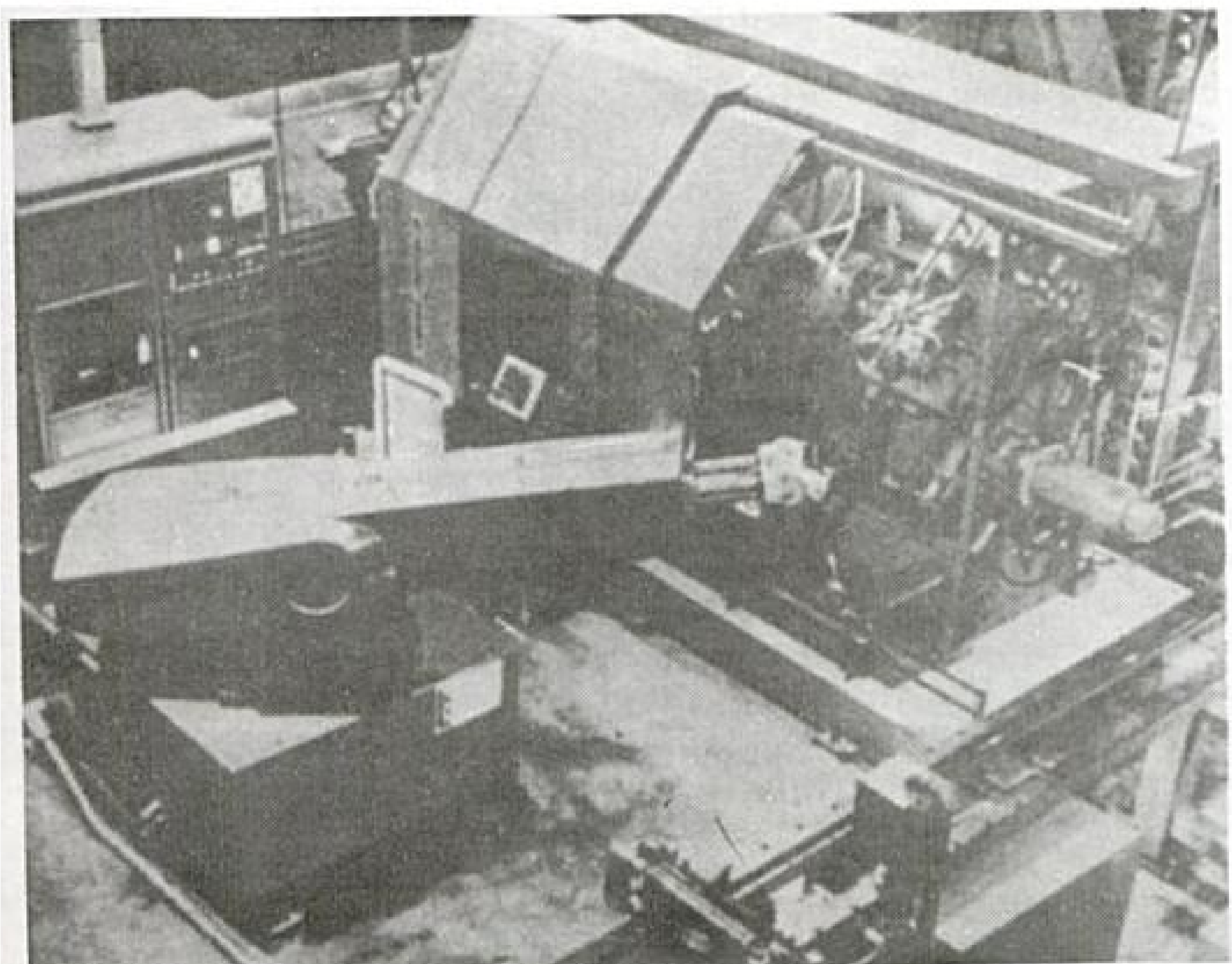


Figure 2-2 Unimate 2000—polar configuration. Here, the Unimate performs a machine loading and unloading operation. The 2000 series robots have provided many years of service. (Photo courtesy of Unimation, Inc.)

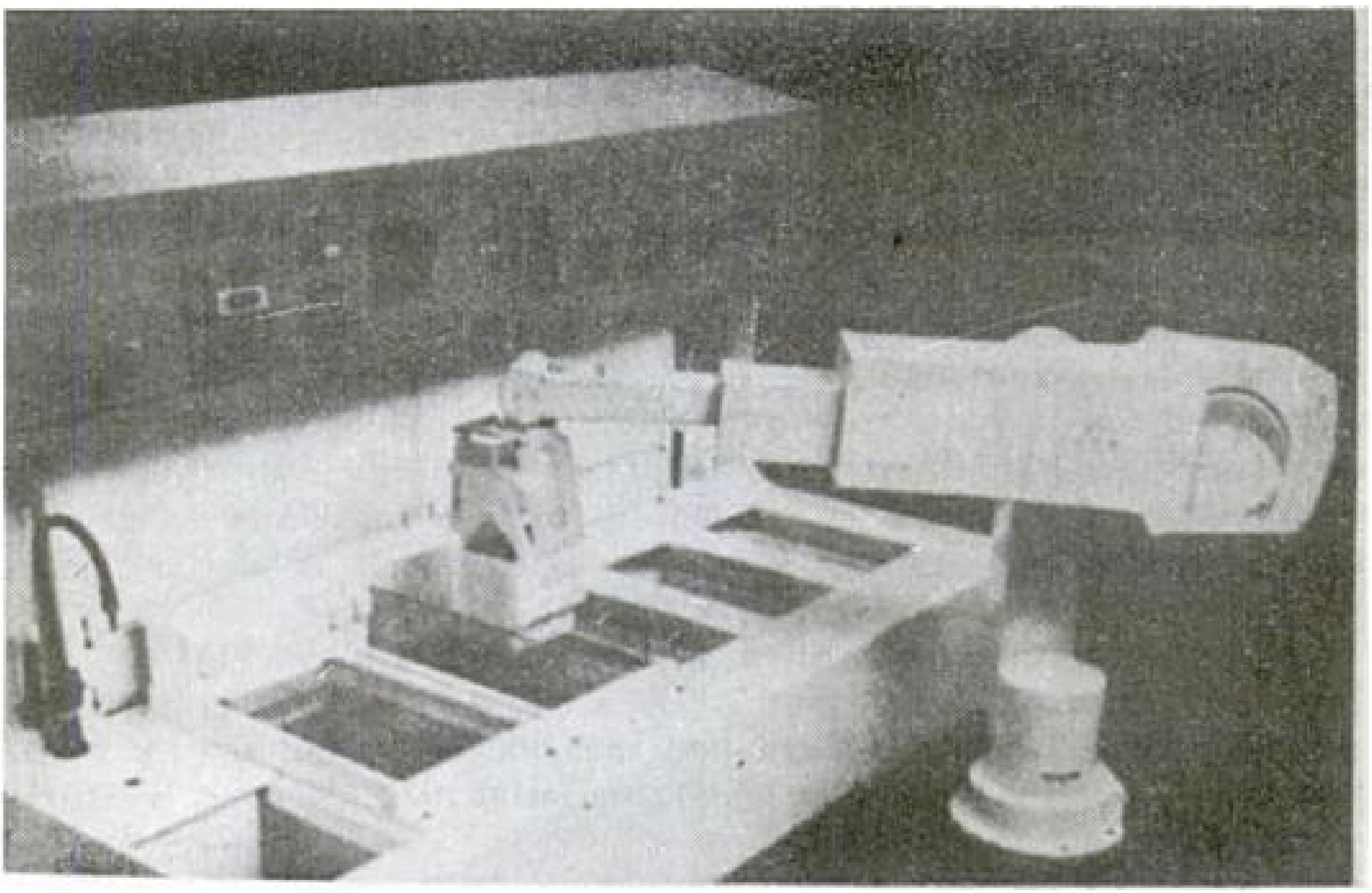


Figure 2-3 The MAKER 110—polar configuration. The MAKER performs a semiconductor wafer-etching application in the electronics industry. (Photo courtesy of United States Robots.)

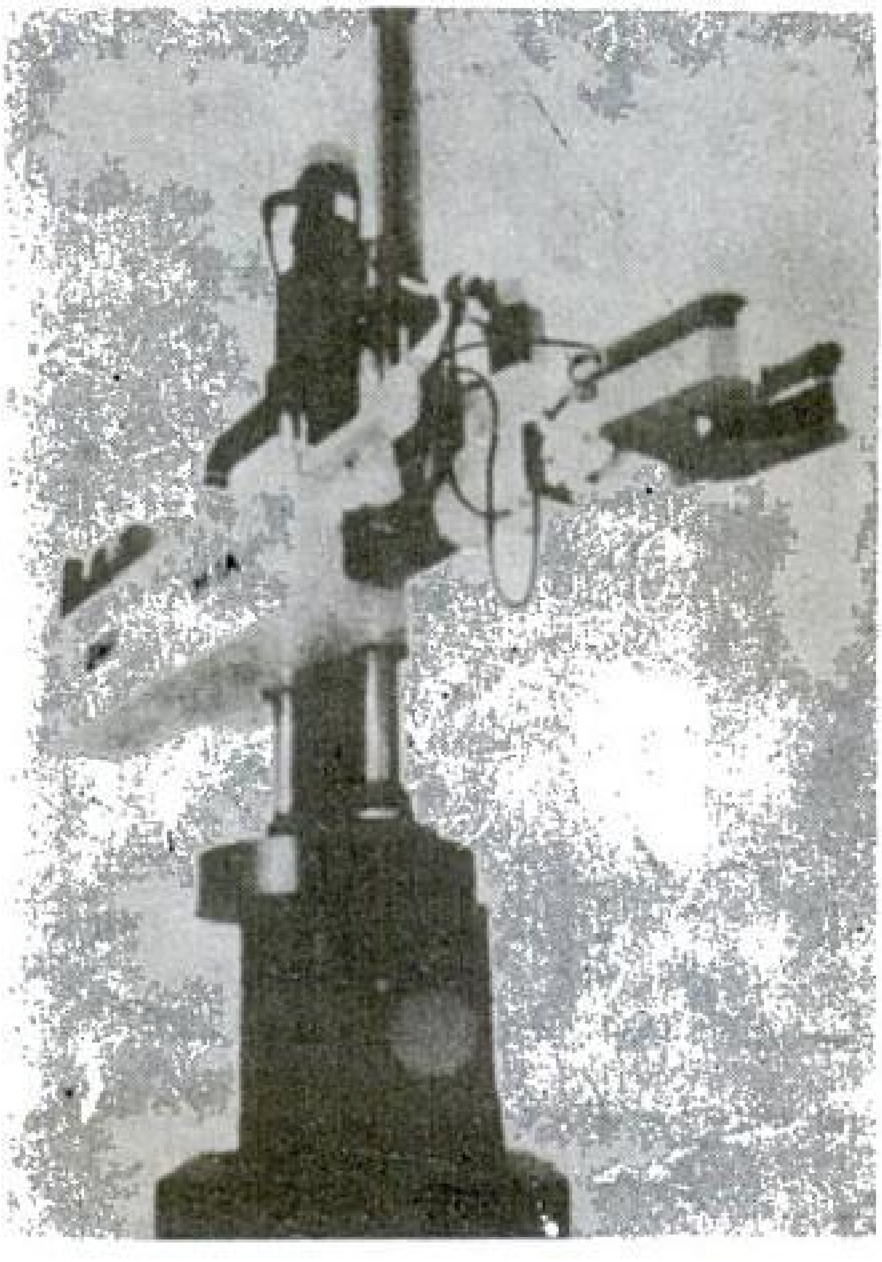


Figure 2-4 GMF Model M-1A—cylindrical configuration. (Photo courtesy of GMF Robotics.)

of operating within a rectangular work envelope. An example of this configuration is the IBM RS-1 robot (currently called the Model 7565), pictured in Fig. 2-5. The RS-1, because of its appearance and construction, is occasionally referred to as a "box" configuration. "Gantry" robot is another name used for cartesian robots that are generally large and possess the appearance of a gantry-type crane. An example is shown in Fig. 2-6.

The jointed-arm robot is pictured in Fig. 2-1(*d*). Its configuration is similar to that of the human arm. It consists of two straight components, corresponding to the human forearm and upper arm, mounted on a vertical pedestal. These components are connected by two rotary joints corresponding to the shoulder and elbow. A wrist is attached to the end of the forearm, thus providing several additional joints. Several commercially available robots possess the jointed-arm configuration, including the Cincinnati Milacron T3 (Model 776) robot, illustrated in Fig. 2-7. A special version of the jointed arm robot is the SCARA, whose shoulder and elbow joints rotate about vertical axes. SCARA stands for Selective Compliance Assembly Robot Arm, and this configuration provides substantial rigidity for the robot in the vertical direction, but compliance in the horizontal plane. This makes it ideal for many assembly tasks. A SCARA robot is pictured in Fig. 2-8.

There are relative advantages and disadvantages to the four basic robot anatomies simply because of their geometries. In terms of repeatability of

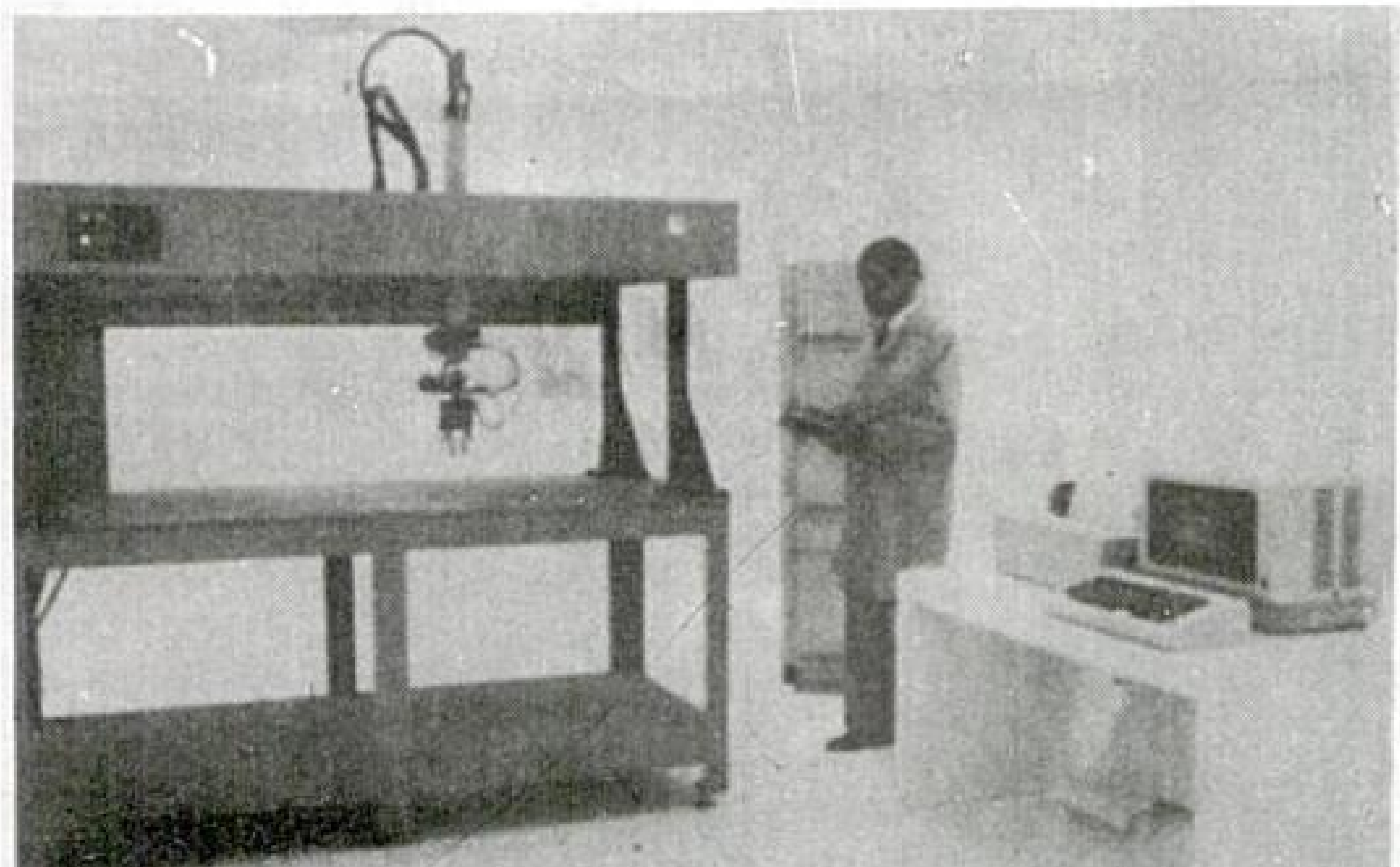


Figure 2-5 The RS-1 (Model 7565)—cartesian coordinate robot. (Photo courtesy of IBM Corporation.)

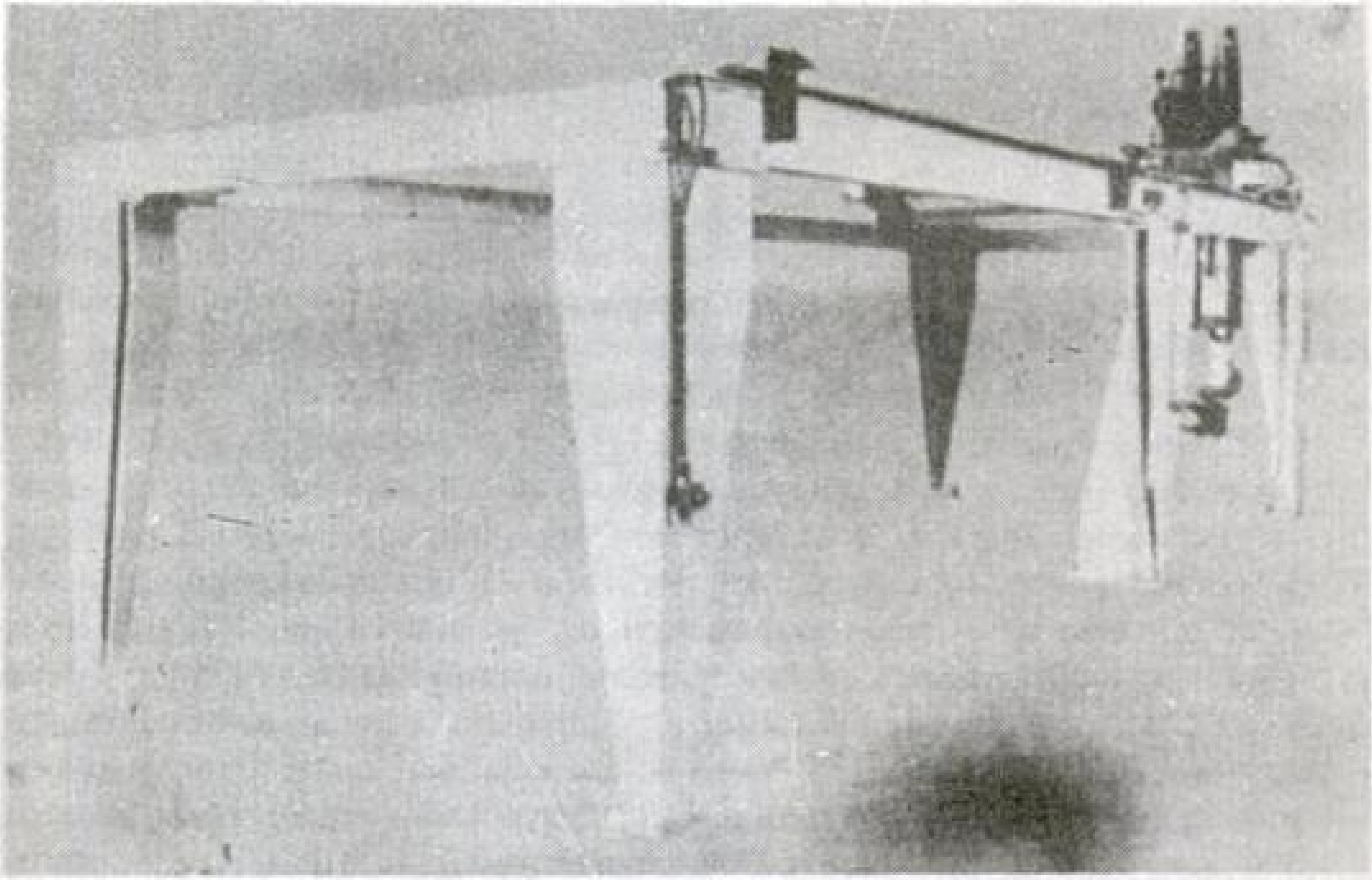


Figure 2-6 Cincinnati Milacron T3-800 series robot—gantry type robot, a large cartesian coordinate configuration. (Photo courtesy of Cincinnati Milacron.)

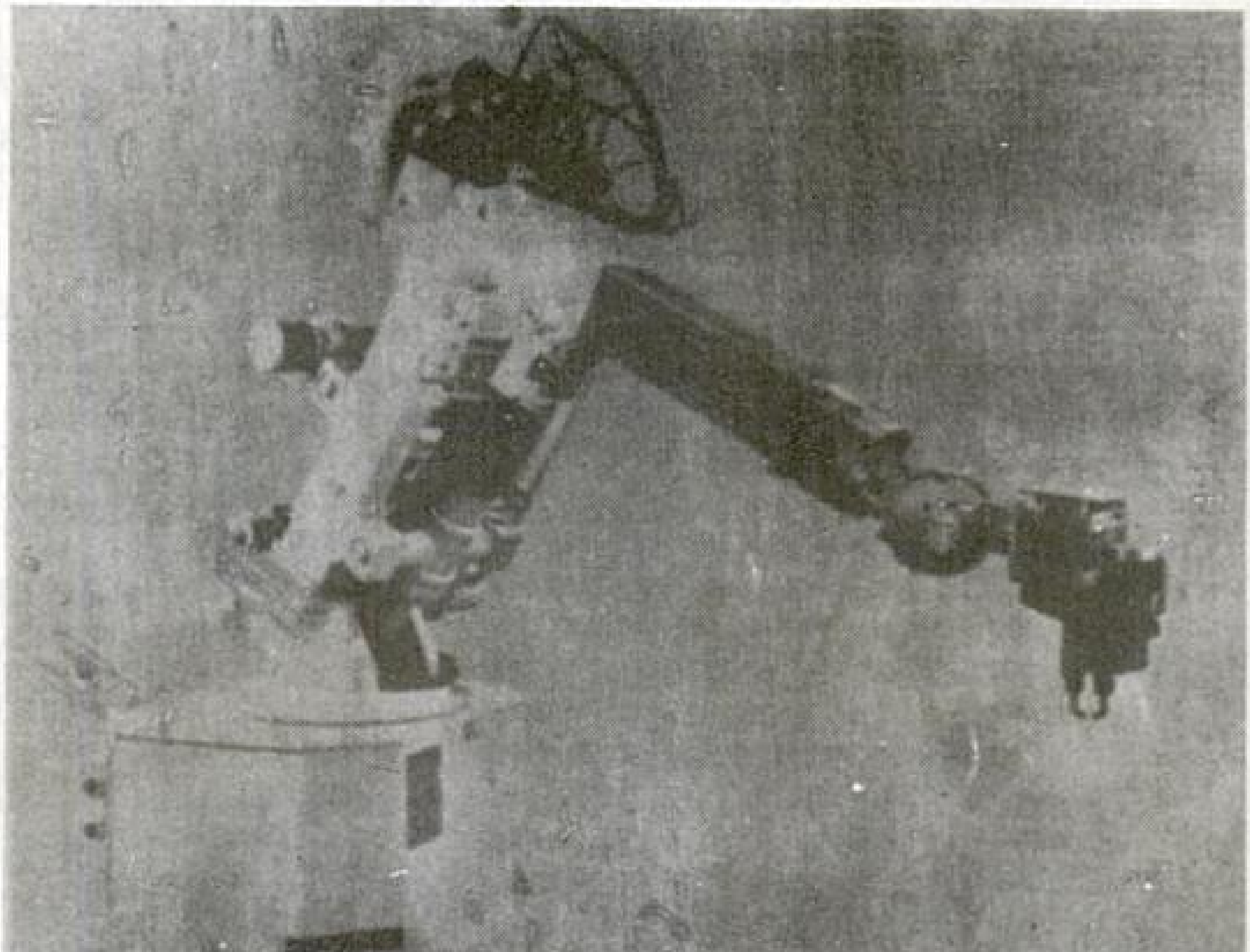


Figure 2-7 The T3-776 robot jointed-arm configuration. (Photo courtesy of Cincinnati Milacron.)

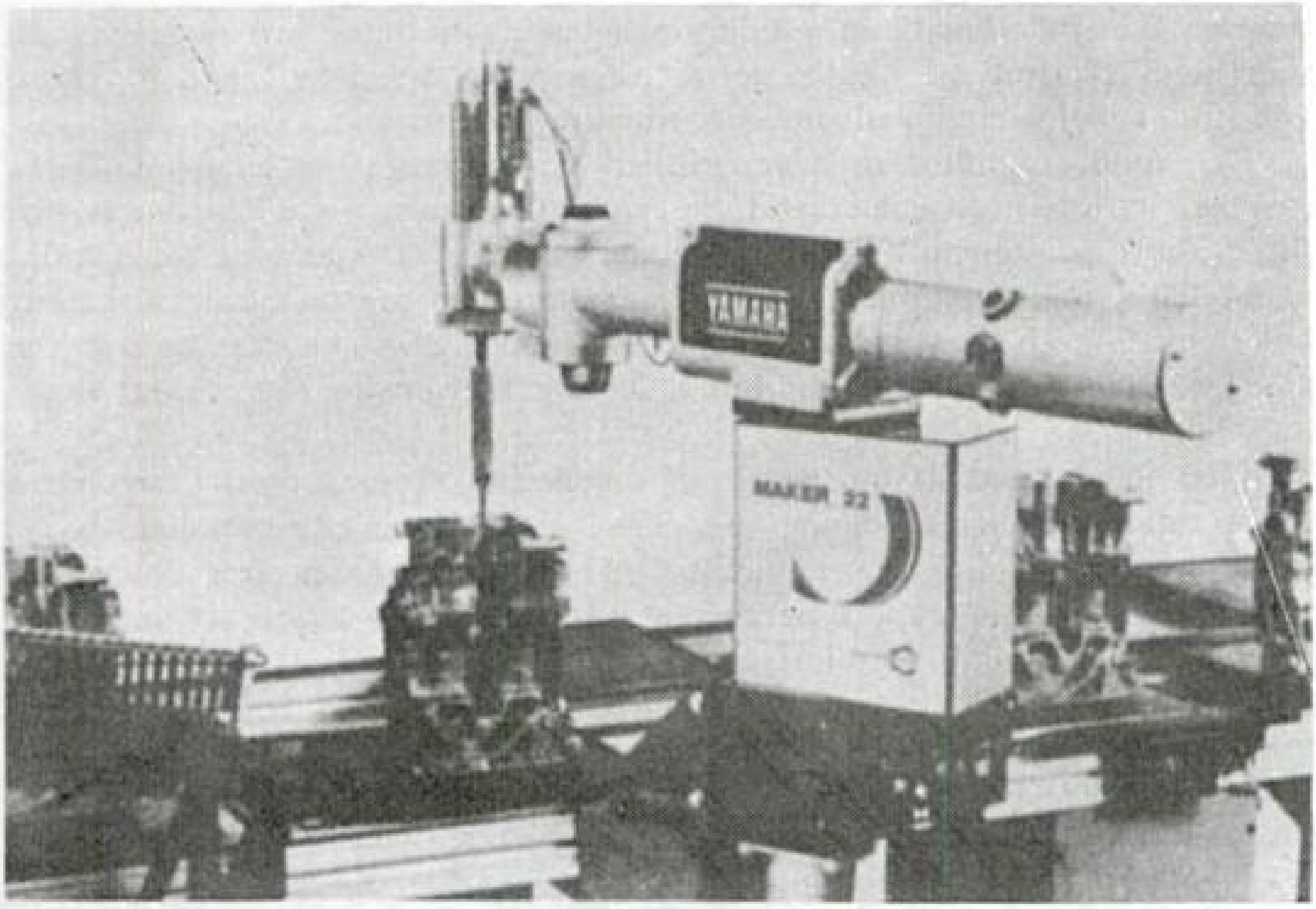


Figure 2-8 SCARA (Selective Compliance Assembly Robotic Arm) robot. (Photo courtesy of United States Robots.)

motion (the capability to move to a taught point in space with minimum error), the box-frame cartesian robot probably possesses the advantage because of its inherently rigid structure. (We will define repeatability and other related terms in Sec. 2-5.) In terms of reach (the ability of the robot to extend its arm significantly beyond its base), the polar and jointed arm configurations have the advantage. The lift capacity of the robot is important in many applications. The cylindrical configuration and the gantry *xyz* robot can be designed for high-rigidity and load-carrying capacity. For machine-loading applications, the ability of the robot to reach into a small opening without interference with the sides of the opening is important. The polar configuration and the cylindrical configuration possess a natural geometric advantage in terms of this capability.

Robot Motions

Industrial robots are designed to perform productive work. The work is accomplished by enabling the robot to move its body, arm, and wrist through a series of motions and positions. Attached to the wrist is the end effector which is used by the robot to perform a specific work task. The robot's movements can be divided into two general categories: arm and body motions, and wrist motions. The individual joint motions associated with these two categories are sometimes referred to by the term "degrees of freedom," and a typical industrial robot is equipped with 4 to 6 degrees of freedom.

The robot's motions are accomplished by means of powered joints. Three joints are normally associated with the action of the arm and body, and two or three joints are generally used to actuate the wrist. Connecting the various manipulator joints together are rigid members that are called links. In any link–joint–link chain, we shall call the link that is closest to the base in the chain the input link. The output link is the one that moves with respect to the input link.

The joints used in the design of industrial robots typically involve a relative motion of the adjoining links that is either linear or rotational. Linear joints involve a sliding or translational motion of the connecting links. This motion can be achieved in a number of ways (e.g., by a piston, a telescoping mechanism, and relative motion along a linear track or rail). Our concern here is not with the mechanical details of the joint, but rather with the relative motion of the adjacent links. We shall refer to the linear joint as a type *L* joint (*L* for *L*inear). The table of Fig. 2-9 illustrates the linear joint. The term prismatic joint is sometimes used in the literature in place of linear joint.

There are at least three types of rotating joint that can be distinguished in robot manipulators. The three types are illustrated in Fig. 2-9. We shall refer to the first as a type *R* joint (*R* for *R*otational). In the type *R* joint the axis of rotation is perpendicular to the axes of the two connecting links. The second type of rotating joint involves a twisting motion between the input and output links. The axis of rotation of the twisting joint is parallel to the axes of both links. We shall call this a type *T* joint (*T* for *T*wisting). The third type of rotating joint is a revolving joint in which the input link is parallel to the axis of rotation and the output link is perpendicular to the axis of rotation. In essence, the output link revolves about the input link, as if it were in orbit. This joint will be designated as a type *V* joint (*V* for re*V*olving).

The arm and body joints are designed to enable the robot to move its end effector to a desired position within the limits of the robot's size and joint movements. For robots of polar, cylindrical, or jointed-arm configuration, the 3 degrees of freedom associated with the arm and body motions are:

1. Vertical traverse—This is the capability to move the wrist up or down to provide the desired vertical attitude.
2. Radial traverse—This involves the extension or retraction (in or out movement) of the arm from the vertical center of the robot.
3. Rotational traverse—This is the rotation of the arm about the vertical axis.

The degrees of freedom associated with the arm and body of the robot are shown in Fig. 2-10 for a polar configuration robot. Similar degrees of freedom are associated with the cylindrical configuration and jointed-arm robot. For a cartesian coordinate robot, the three degrees of freedom are vertical move-

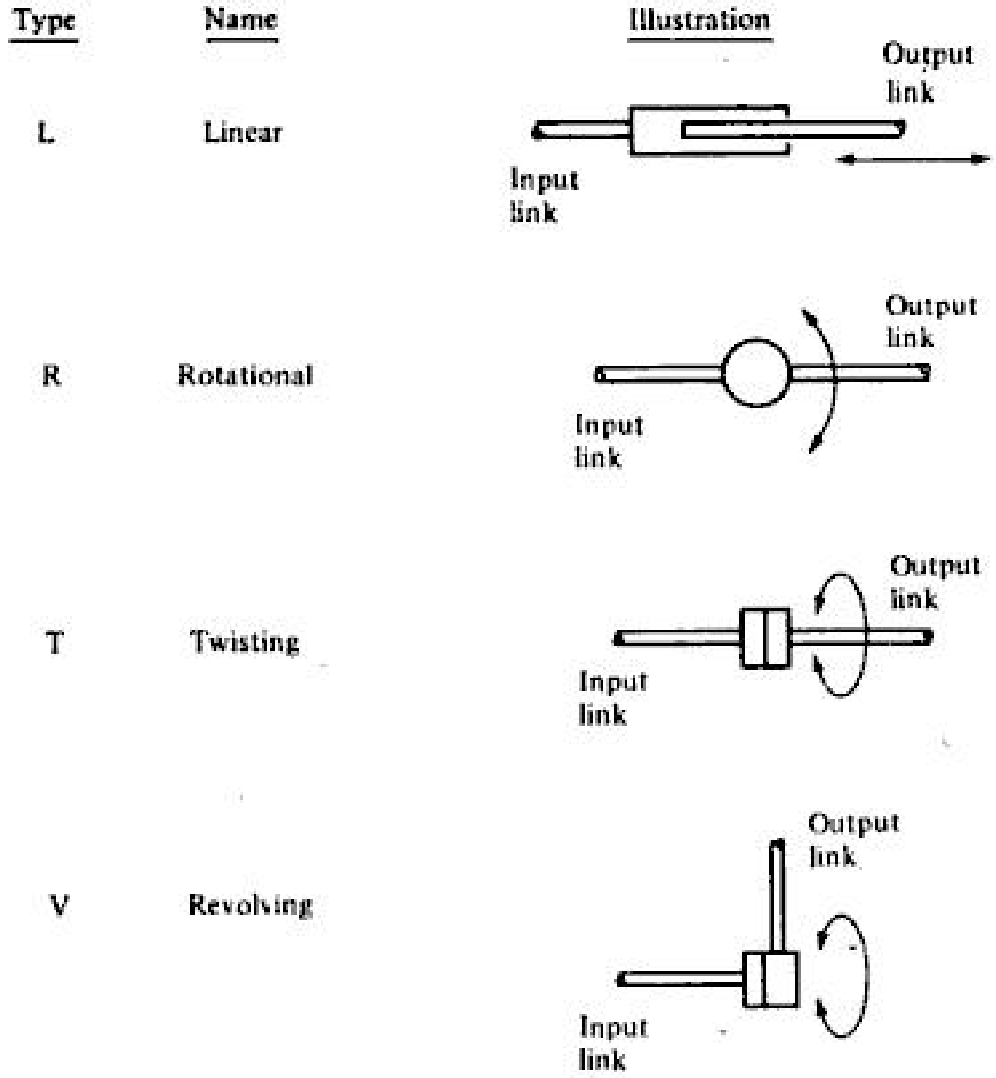


Figure 2-9 Several types of joints used in robots: (a) rotational joint with rotation along an axis perpendicular to arm member axes, (b) rotational joint with twisting action, (c) linear motion joint, usually achieved by a sliding action.

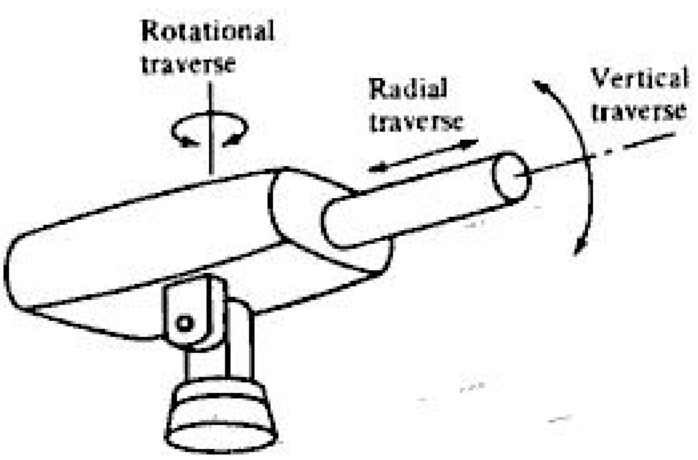


Figure 2-10 Three degrees of freedom associated with arm and body of a polar coordinate robot.

ment (z-axis motion), in-and-out movement (y-axis motion), and right-or-left movement (x-axis motion). These are achieved by corresponding movements of the three orthogonal slides of the robot arm.

The wrist movement is designed to enable the robot to orient the end effector properly with respect to the task to be performed. For example, the hand must be oriented at the appropriate angle with respect to the workpiece

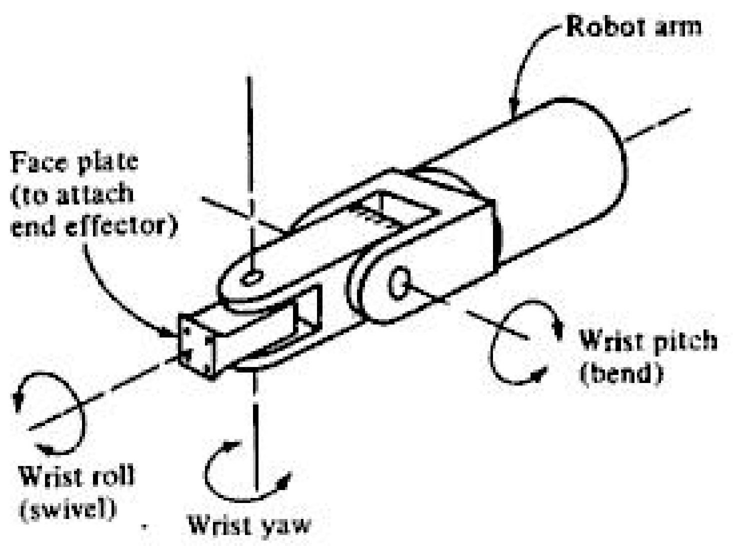


Figure 2-11 Three degrees of freedom associated with the robot wrist.

in order to grasp it. To solve this orientation problem, the wrist is normally provided with up to 3 degrees of freedom (the following is a typical configuration):

1. Wrist roll—Also called wrist swivel, this involves rotation of the wrist mechanism about the arm axis.
2. Wrist pitch—Given that the wrist roll is in its center position, the pitch would involve the up or down rotation of the wrist. Wrist pitch is also sometimes called wrist bend.
3. Wrist yaw—Again, given that the wrist swivel is in the center position of its range, wrist yaw would involve the right or left rotation of the wrist.

These degrees of freedom for the wrist are illustrated in Fig. 2-11. The reason for specifying that the wrist roll be in its center position in the definitions of pitch and yaw is because rotation of the wrist about the arm axis will alter the orientation of the pitch and yaw movements.

Joint Notation Scheme

The physical configuration of the robot manipulator can be described by means of a joint notation scheme, using the joint types defined earlier in this section (*L*, *R*, *T*, and *V*). Considering the arm and body joints first, the letters can be used to designate the particular robot configuration starting with the joint closest to the base and proceeding to the joint that connects to the wrist. Accordingly, a jointed-arm robot (excluding the wrist assembly) would have three rotational joints and would be designated as either TRR or VVR. Typical notations for the four basic configurations are summarized in Table 2-1.

The joint notation scheme permits the designation of more or less than the three joints typical of the basic configurations indicated in the table. It can also be used to explore other possibilities for configuring robots, beyond the four basic types.

Table 2-1 Notation scheme for designating robot configurations

Robot configuration (arm and body)	Symbol
Polar configuration	*TRL*
Cylindrical configuration	*TLL, LTL, LVL*
Cartesian coordinate robot	*LLL*
Jointed arm configuration	*TRR, VVR*
Robot configuration (wrist)	**Symbol**
Two-axis wrist (typical)	:*RT*
Three-axis wrist (typical)	:*TRT*

The notation system can be expanded to include wrist motions by designating the two or three (or more) types of wrist joint. The notation starts with the joint closest to the arm interface, and proceeds to the mounting plate for the end effector. Wrist joints are predominantly rotating joints of type *R* and *T*. Hence, a typical wrist mechanism with three rotational joints would be indicated by *TRR* (Fig. 2-11). This notation is simply added to the notation for the arm and body configuration. For example, a polar coordinate robot with a three-axis wrist might be designated as *TRL*:*TRT*.

The scheme can also provide for the possibility of robots that move on a track in the floor or along an overhead rail system in the factory. As an illustration, a *TRL*:*TRT* robot fastened to a platform on wheels that can be driven along a track between several machine tools would be designated by the following notation: *L*-*TRL*:*TRT*. In this case, even though the wheels of the platform rotate, the motion of the robot is linear.

2-2 WORK VOLUME

Work volume is the term that refers to the space within which the robot can manipulate its wrist end. The convention of using the wrist end to define the robot's work volume is adopted to avoid the complication of different sizes of end effectors that might be attached to the robot's wrist. The end effector is an addition to the basic robot and should not be counted as part of the robot's working space. A long end effector mounted on the wrist would add significantly to the extension of the robot compared to a smaller end effector. Also, the end effector attached to the wrist might not be capable of reaching certain points within the robot's normal work volume because of the particular combination of joint limits of the arm.

The work volume is determined by the following physical characteristics

Types of Drive Systems

Commercially available industrial robots are powered by one of three types of drive systems. These three systems are:

1. Hydraulic drive
2. Electric drive
3. Pneumatic drive

Hydraulic drive and electric drive are the two main types of drives used on more sophisticated robots.

Hydraulic drive is generally associated with larger robots, such as the Unimate 2000 series (Fig. 2-2). The usual advantages of the hydraulic drive system are that it provides the robot with greater speed and strength. The disadvantages of the hydraulic drive system are that it typically adds to the floor space required by the robot, and that a hydraulic system is inclined to leak oil which is a nuisance. Hydraulic drive systems can be designed to actuate either rotational joints or linear joints. Rotary vane actuators can be utilized to provide rotary motion, and hydraulic pistons can be used to accomplish linear motion.

Electric drive systems do not generally provide as much speed or power as hydraulic systems. However, the accuracy and repeatability of electric drive robots are usually better. Consequently, electric robots tend to be smaller, requiring less floor space, and their applications tend toward more precise work such as assembly. The MAKER 110 (Fig. 2-3) is an example of an electric drive robot that is consistent with these tendencies. Electric drive robots are actuated by dc stepping motors or dc servomotors. These motors are ideally suited to the actuation of rotational joints through appropriate drive train and gear systems. Electric motors can also be used to actuate linear joints (e.g., telescoping arms) by means of pulley systems or other translational mechanisms.

The economics of the two types of drive systems are also a factor in the decision to utilize hydraulic drive on large robots and electric drive on smaller robots. It turns out that the cost of an electric motor is much more proportional to its size, whereas the cost of a hydraulic drive system is somewhat less dependent on its size. These relationships are displayed conceptually in Fig. 2-13. As the illustration suggests, there is a hypothetical break-even point, below which it is advantageous to use electric drive and above which it is appropriate to use hydraulic drive. Having explained these factors, it should be noted that there is a trend in the design of industrial robots toward all electric drives, and away from hydraulic robots because of the disadvantages discussed above.

Pneumatic drive is generally reserved for smaller robots that possess fewer degrees of freedom (two- to four-joint motions). These robots are often limited to simple "pick-and-place" operations with fast cycles. Pneumatic power can

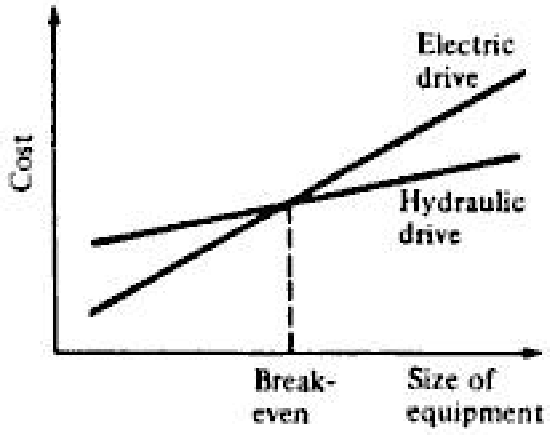


Figure 2-13 Cost vs. size for electric drive and hydraulic drive.

be readily adapted to the actuation of piston devices to provide translational movement of sliding joints. It can also be used to operate rotary actuators for rotational joints.

Speed of Motion

The speed capabilities of current industrial robots range up to a maximum of about 1.7 m/s (about 5 ft/sec). This speed would be measured at the wrist. Accordingly, the highest speeds can be obtained by large robots with the arm extended to its maximum distance from the vertical axis of the robot. As mentioned previously, hydraulic robots tend to be faster than electric drive robots.

The speed, of course, determines how quickly the robot can accomplish a given work cycle. It is generally desirable in production to minimize the cycle time of a given task. Nearly all robots have some means by which adjustments in the speed can be made. Determination of the most desirable speed, in addition to merely attempting to minimize the production cycle time, would also depend on other factors, such as:

The accuracy with which the wrist (end effector) must be positioned
The weight of the object being manipulated
The distances to be moved.

There is generally an inverse relationship between the accuracy and the speed of robot motions. As the required accuracy is increased, the robot needs more time to reduce the location errors in its various joints to achieve the desired final position. The weight of the object moved also influences the operational speed. Heavier objects mean greater inertia and momentum, and the robot must be operated more slowly to safely deal with these factors. The influence of the distance to be moved by the robot manipulator is illustrated in Fig. 2-14. Because of acceleration and deceleration problems, a robot is capable of traveling one long distance in less time than a sequence of short distances whose sum is equal to the long distance. The short distances may not permit the robot to ever reach the programmed operating speed.

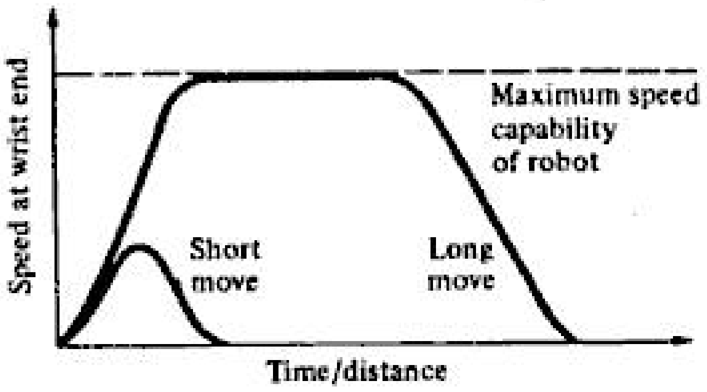


Figure 2-14 Influence of distance versus speed.

Load-Carrying Capacity

The size, configuration, construction, and drive system determine the load-carrying capacity of the robot. This load capacity should be specified under the condition that the robot's arm is in its weakest position. In the case of a polar, cylindrical, or jointed-arm configuration, this would mean that the robot arm is at maximum extension. Just as in the case of a human, it is more difficult to lift a heavy load with arms fully extended than when the arms are held in close to the body.

The rated weight-carrying capacities of industrial robots ranges from less than a pound for some of the small robots up to several thousand pounds for very large robots. An example is the Prab Versatran Model FC which has a rated load capacity of 2000 lb. The small assembly robots, such as the MAKER 110, have weight-carrying capabilities in the vicinity of 5 lb. The manufacturer's specification of this feature is the gross weight capacity. To use this specification, the user must consider the weight of the end effector. For example, if the rated load capacity of a given robot were 5 lb, and the end effector weighed 2 lb, then the net weight-carrying capacity of the robot would be only 3 lb.

2-4 CONTROL SYSTEMS AND DYNAMIC PERFORMANCE

In order to operate, a robot must have a means of controlling its drive system to properly regulate its motions. In this section we briefly describe the various types of control systems and the associated performance characteristics which are determined by the control system. A more thorough treatment of these topics is provided in Chaps. 3 and 4.

Four Types of Robot Controls

Commercially available industrial robots can be classified into four categories according to their control systems. The four categories are:

1. Limited-sequence robots
2. Playback robots with point-to-point control

3. Playback robots with continuous path control
4. Intelligent robots

Of the four categories, the limited-sequence robots represent the lowest level of control and the intelligent robots are the most sophisticated.

Limited-sequence robots do not use servo-control to indicate relative positions of the joints. Instead, they are controlled by setting limit switches and/or mechanical stops to establish the endpoints of travel for each of their joints. Establishing the positions and sequence of these stops involves a mechanical setup of the manipulator rather than robot programming in the usual sense of the term. With this method of control, the individual joints can only be moved to their extreme limits of travel. This has the effect of severely limiting the number of distinct points that can be specified in a program for these robots. The sequence in which the motion cycle is played out is defined by a pegboard or stepping switch or other sequencing device. This device, which constitutes the robot controller, signals each of the particular actuators to operate in the proper succession. There is generally no feedback associated with a limited sequence robot to indicate that the desired position has been achieved. Any of the three drive systems can be used with this type of control system; however, pneumatic drive seems to be the type most commonly employed. Applications for this type of robot generally involve simple motions, such as pick-and-place operations.

Playback robots use a more sophisticated control unit in which a series of positions or motions are "taught" to the robot, recorded into memory, and then repeated by the robot under its own control. The term "playback" is descriptive of this general mode of operation. The procedure of teaching and recording into memory is referred to as programming the robot. Playback robots usually have some form of servo-control (e.g., closed loop feedback system) to ensure that the positions achieved by the robot are the positions that have been taught.

Playback robots can be classified into two categories: point-to-point (PTP) robots and continuous-path (CP) robots. Point-to-point robots are capable of performing motion cycles that consist of a series of desired point locations and related actions. The robot is taught each point, and these points are recorded into the robot's control unit. During playback, the robot is controlled to move from one point to another in the proper sequence. Point-to-point robots do not control the path taken by the robot to get from one point to the next. If the programmer wants to exercise a limited amount of control over the path followed, this must be done by programming a series of points along the desired path. Control of the sequence of positions is quite adequate for many kinds of applications, including loading and unloading machines and spot welding.

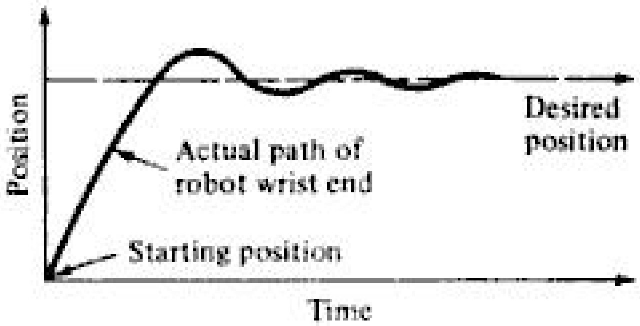
Continuous-path robots are capable of performing motion cycles in which the path followed by the robot is controlled. This is usually accomplished by making the robot move through a series of closely spaced points which describe the desired path. The individual points are defined by the control unit rather than the programmer. Straight line motion is a common form of continuous-path control for industrial robots. The programmer specifies the starting point and the end point of the path, and the control unit calculates the sequence of individual points that permit the robot to follow a straight line trajectory. Some robots have the capability to follow a smooth, curved path that has been defined by a programmer who manually moves the arm through the desired motion cycle. To achieve continuous-path control to more than a limited extent requires that the controller unit be capable of storing a large number of individual point locations that define the compound curved path. Today this usually involves the use of a digital computer (a microprocessor is typically used as the central processing unit for the computer) as the robot controller. CP control is required for certain types of industrial applications such as spray coating and arc welding.

Intelligent robots constitute a growing class of industrial robot that possesses the capability not only to play back a programmed motion cycle but to also interact with its environment in a way that seems intelligent. Invariably, the controller unit consists of a digital computer or similar device (e.g., programmable controller). Intelligent robots can alter their programmed cycle in response to conditions that occur in the workplace. They can make logical decisions based on sensor data received from the operation. The robots in this class have the capacity to communicate during the work cycle with humans or computer-based systems. Intelligent robots are usually programmed using an English-like and symbolic language not unlike a computer programming language. Indeed, the kinds of applications that are performed by intelligent robots rely on the use of a high-level language to accomplish the complex and sophisticated activities that can be accomplished by these robots. Typical applications for intelligent robots are assembly tasks and arc-welding operations.

Speed of Response and Stability

Speed of response and stability are two important characteristics of dynamic performance related to control systems design. The speed of response refers to the capability of the robot to move to the next position in a short amount of time. This response time is obviously related to the robot's motion speed, a feature that we have already discussed. It is also a function of the control system. In robotics, stability is generally defined as a measure of the oscillations which occur in the arm during movement from one position to the next. A robot with good stability will exhibit little or no oscillations either during or at the termination of the arm movement. Poor stability would be indicated by a large amount of oscillation. It is generally desirable in control systems design for the system to have good stability and a fast response time. Unfortunately, these are competing objectives.

The stability of a robot can be controlled to a certain extent by incorporating damping elements into the robot's design. A high level of damping will increase the robot's stability (reduce its tendency toward oscillation).



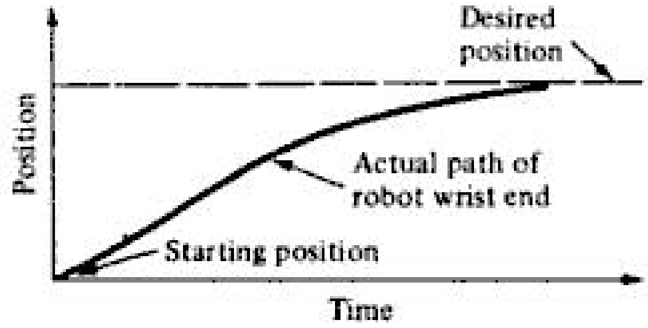


Figure 2-15 Concept of speed of response and stability in robotics: (a) low damping—fast response, (b) high damping—slow response.

The problem with high damping is that it reduces the speed of response. Accordingly, there is a compromise that must be struck between the stability of the robot and its ability to operate at high speeds.

The concept of stability and its relation to damping is illustrated in Fig. 2-15. In the two diagrams of the figure, the position of the robot's wrist is shown as a function of time for two cases: small damping and large damping. With low damping, the robot arm moves to the programmed position quickly, but exhibits considerable oscillation about the position. With a large amount of damping built into the system, the arm movement to the desired position is very sluggish but there is no oscillatory motion about the final position.

2-5 PRECISION OF MOVEMENT

The preceding discussion of response speed and stability is concerned with the dynamic performance of the robot. Another measure of performance is precision of the robot's movement. We will define precision as a function of three features:

1. Spatial resolution
2. Accuracy
3. Repeatability

These terms will be defined with the following assumptions. First, the definitions will apply at the robot's wrist end with no hand attached to the wrist. Second, the terms apply to the worst case conditions, the conditions under which the robot's precision will be at its worst. This generally means that the robot's arm is fully extended in the case of a jointed arm or polar configuration. Third, our definitions will be developed in the context of a point-to-point robot. That is, we will be concerned with the robot's capability to achieve a given position within its work volume. It is easier to define the various precision features in a static context rather than a dynamic context. It is considerably more difficult to define, and measure, the robot's capacity to achieve a defined motion path in space because it would be complicated by speed and other factors.

Spatial Resolution

The spatial resolution of a robot is the smallest increment of movement into which the robot can divide its work volume. Spatial resolution depends on two factors: the system's control resolution and the robot's mechanical inaccuracies. It is easiest to conceptualize these factors in terms of a robot with 1 degree of freedom.

The control resolution is determined by the robot's position control system and its feedback measurement system. It is the controller's ability to divide the total range of movement for the particular joint into individual increments that can be addressed in the controller. The increments are sometimes referred to as "addressable points." The ability to divide the joint range into increments depends on the bit storage capacity in the control memory. The number of separate, identifiable increments (addressable points) for a particular axis is given by

$$\text{Number of increments} = 2^n$$

where n = the number of bits in the control memory.

For example, a robot with 8 bits of storage can divide the range into 256 discrete positions. The control resolution would be defined as the total motion range divided by the number of increments. We assume that the system designer will make all of the increments equal.

> **Example 2-1** Using our robot with 1 degree of freedom as an illustration, we will assume it has one sliding joint with a full range of 1.0 m (39.37 in.). The robot's control memory has a 12-bit storage capacity. The problem is to determine the control resolution for this axis of motion.
>
> The number of control increments can be determined as follows:
>
> $$\text{Number of increments} = 2^{12} = 4096$$
>
> The total range of 1 m is divided into 4096 increments. Each position will be separated by
>
> $$1\ \text{m}/4096 = 0.000244\ \text{m} \quad \text{or} \quad 0.244\ \text{mm}$$
>
> The control resolution is 0.244 mm (0.0096 in.).

This example deals with only one joint. A robot with several degrees of freedom would have a control resolution for each joint of motion. To obtain the control resolution for the entire robot, component resolutions for each joint would have to be summed vectorially. The total control resolution would depend on the wrist motions as well as the arm and body motions. Since some

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

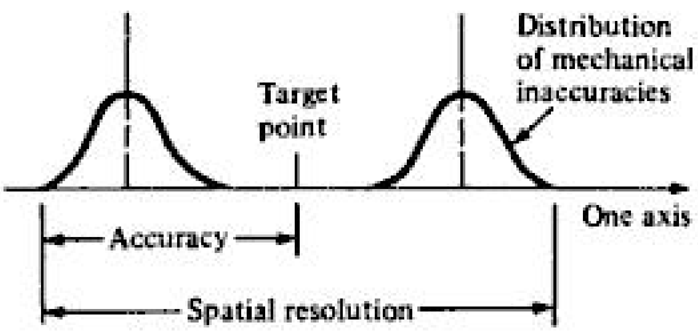


Figure 2-17 Illustration of accuracy and spatial resolution in which mechanical inaccuracies are represented by a statistical distribution.

accuracy of a robot is affected by several factors. First, the accuracy varies within the work volume, tending to be worse when the arm is in the outer range of its work volume and better when the arm is closer to its base. The reason for this is that the mechanical inaccuracies are magnified with the robot's arm fully extended. The term error map is used to characterize the level of accuracy possessed by the robot as a function of location in the work volume. Second, the accuracy is improved if the motion cycle is restricted to a limited work range. The mechanical errors will tend to be reduced when the robot is exercised through a restricted range of motions. The robot's ability to reach a particular reference point within the limited work space is sometimes called its local accuracy. When the accuracy is assessed within the robot's full work volume, the term global accuracy is used. A third factor influencing accuracy is the load being carried by the robot. Heavier workloads cause greater deflection of the mechanical links of the robot, resulting in lower accuracy.

Repeatability

Repeatability is concerned with the robot's ability to position its wrist or an end effector attached to its wrist at a point in space that had previously been taught to the robot. Repeatability and accuracy refer to two different aspects of the robot's precision. Accuracy relates to the robot's capacity to be programmed to achieve a given target point. The actual programmed point will probably be different from the target point due to limitations of control resolution. Repeatability refers to the robot's ability to return to the programmed point when commanded to do so.

These concepts are illustrated in Figure 2-18. The desired target point is denoted by the letter *T*. During the teach procedure, the robot is commanded to move to point *T*, but because of the limitations on its accuracy, the programmed position becomes point *P*. The distance between points *T* and *P* is a manifestation of the robot's accuracy in this case. Subsequently, the robot is instructed to return to the programmed point *P*; however, it does not return to the exact same position. Instead, it returns to position *R*. The difference between *P* and *R* is a result of limitations on the robot's repeatability. The robot will not always return to the same position *R* on subsequent repetitions of the motion cycle. Instead, it will form a cluster of points on both sides of the position *P* in Fig. 2-18.

move along the straight line path. For more complex motions (e.g., those encountered in spray-painting operations), it is usually more convenient for the programmer to physically move the robot arm and end effector through the desired motion path and record the positions at closely spaced sampling intervals. Certain parameters of the motion cycle, such as the robot's speed, would be controlled independently when the job is set up to operate. Accordingly, the programmer does not need to be concerned with these aspects of the program. The programmer's principal concern is to make sure that the motion sequence is correct.

Textual programming methods use an English-like language to establish the logic and sequence of the work cycle. A computer terminal is used to input the program instructions into the controller but a teach pendant is also used to define the locations of the various points in the workspace. The robot programming language names the points as symbols in the program and these symbols are subsequently defined by showing the robot their locations. In addition to identifying points in the workspace, the robot languages permit the use of calculations, more detailed logic flow, and subroutines in the programs, and greater use of sensors and communications. Accordingly, the use of the textual languages corresponds largely to the so-called intelligent robots.

Some examples of the kinds of programming statements that would be found in the textual robot languages include the following sequence:

```
SPEED 35 IPS
MOVE P1
CLOSE 40 MM
WAIT 1 SEC
DEPART 60 MM
```

The series of commands tells the robot that its velocity at the wrist should be 35 in./sec in the motions which follow. The MOVE statement indicates that the robot is to move its gripper to point P1 and close to an opening of 40 mm. It is directed to wait 1.0 s before departing from P1 by a distance of 60 mm above the point.

A future enhancement of textual language programming will be to enter the program completely off-line, without the need for a teach pendant to define point locations in the program. The potential advantage of this method is that the programming can be accomplished without taking the robot out of production. All of the current methods of programming require the participation of the robot in order to perform the programming function. With off-line programming, the entire program can be entered into a computer for later downloading to the robot. Off-line programming would hasten the changeover from one robot work cycle to a new work cycle without a major time delay for reprogramming. Unfortunately, there are certain technical problems associated with off-line programming. These problems are mainly concerned with defining the spatial locations of the positions to be used in the work cycle, and that is why the teach pendant is required in today's textual robot languages.

PART TWO

ROBOT TECHNOLOGY: THE ROBOT AND ITS PERIPHERALS

Part Two of the book is focused on the mechanical and electronic technology of industrial robots. This includes the technology of both the manipulator itself and the peripheral devices and systems that work with the robot.

The design of the robot manipulator represents a significant challenge to mechanical and electrical engineers. As suggested by our discussion in Chap. Two, the problem is to configure a physical system that is capable of positioning its end effector to within several thousandths of an inch of a desired target location, is relatively lightweight, possesses high-lift capacity, and moves at high speeds between positions in the workspace. Chapters Three and Four consider this design problem. Chapter Three examines the various components of the individual joints that make up the manipulator. Chapter Four treats the mathematical analysis of the arm position and motion. These mathematical methods are utilized to aid engineers in the design of robots, and by the robot control computer to calculate motion trajectories and frame transformations during the work cycle.

Chapter Five considers one of the important peripheral devices used with robots in industrial applications—their end effectors. The different types of end effector are discussed, along with the engineering considerations for their design. We concentrate in this chapter on the grippers used to handle

The force due to the spring is

$$K_s y - K_s x$$

Summing all of the forces we get

$$M\frac{d^2y}{dt^2} + K_d\frac{dy}{dt} + K_s y = K_s x \tag{3-2}$$

In this system the input is x, representing the displacement of the end of the spring, and the system output is y, representing the displacement of the block. The system has been described by a second-order linear differential equation which relates the input and the output. This mathematical description of the system allows us to analyze its behavior. Before beginning the analysis, however, we will develop other useful tools for building the models.

Transfer Functions

Linear differential equations can be rewritten using the differential operator, s. The variable, s, is used to represent the mathematical operation of taking the derivative of a time-dependent variable with respect to time. Thus, functions which are variables of time [e.g., $x(t)$ and $y(t)$] become functions of the variable s [e.g., $X(s)$ and $Y(s)$]. By using s with Laplace transforms, linear differential equations can be converted to equivalent expressions which are functions of s. (It is assumed that the reader is familiar with the Laplace transform and its s operator for linear systems analysis.) Using s, Eq. (3-2) can be written as

$$Ms^2 Y(s) + K_d s Y(s) + K_s Y(s) = K_s X(s) \tag{3-3}$$

The transfer function relates the output of the system to an input. The spring–mass–damper transfer function can be derived by rewriting Eq. (3-3) as

$$\frac{Y(s)}{X(s)} = \frac{K_s}{Ms^2 + K_d s + K_s} \tag{3-4}$$

Block Diagrams

It is often useful to provide a schematic representation of the system in addition to the mathematical model. A common means of graphically representing the relationships among the components of the system is the block diagram. Block diagrams are constructed from four basic elements:

Function blocks
Signal arrows
Summing junctions
Takeoff points

The four components are illustrated in Fig. 3-2. A function block, shown in

and the roots of the characteristic equation, Eq. (3-5), are given by

$$s_{1,2} = -\frac{K_d}{2M} \pm \frac{\sqrt{K_d^2 - 4MK_s}}{2M} \tag{3-6}$$

The performance of the system is dependent on the values of M, K_d, and K_s. One aspect of the system performance that can be determined by analyzing the roots of the characteristic equation is the "damping" of the system. Depending on the values of the parameters in the characteristic equation, the system may respond in one of four ways. The four responses classify the system into one of the following types:

1. Undamped system
2. Underdamped system
3. Critically damped system
4. Overdamped system

We will now briefly describe these four types of system response.

Undamped In order for the system to be undamped, the damping coefficient, K_d, must be equal to zero. In this case the roots of the characteristic equation are given by

$$s_{1,2} = \pm j\sqrt{\frac{K_s}{M}} \tag{3-7}$$

These are imaginary roots. Assuming a step input X to the system, the response can be described as

$$y = C_1 \sin(\omega_n t) + C_2 \cos(\omega_n t) + X \tag{3-8}$$

where $\omega_n = \sqrt{K_s/M}$ and is called the natural frequency of the system. The response represented by Eq. (3-8) is shown in Fig. 3-7(a) where it can be seen that the undamped response is oscillatory.

Underdamped When there is a small amount of damping in the system, that is, where

$$K_d^2 < 4MK_s$$

the roots may be rewritten as

$$s_{1,2} = -\frac{K_d}{2M} \pm j\frac{\sqrt{4MK_s - K_d^2}}{2M} \tag{3-9}$$

Substituting $a = Kd/2M$ and $\omega_d = \sqrt{(4MK_s - K_d^2)}/2M$, Eq. (3-9) may be written as

$$a_{1,2} = -a \pm j\omega_d \tag{3-10}$$

In most on–off controllers either M_1 or M_2 is zero. The practical use of an on–off controller usually requires that the error must move through some range before switching actually takes place. This prevents the controller from oscillating at too high a frequency. This range is referred to as the differential gap.

Proportional Control

In cases where a smoother control action is required a proportional controller may be used. Proportional control provides a control signal that is proportional to the error. Essentially it acts as an amplifier with a gain K_p. Its action is represented by

$$m(t) = K_p e(t) \tag{3-16}$$

Using the differential operator notation introduced earlier the transfer function would be

$$\frac{M(s)}{E(s)} = K_p \tag{3-17}$$

Integral Control

In a controller employing an integral control action the control signal is changed at a rate proportional to the error signal. That is, if the error signal is large, the control signal increases rapidly; if it is small, the control signal increases slowly. This may be represented by

$$m(t) = K_i \int e(t)\, dt \tag{3-18}$$

where K_i is the integrator gain. The corresponding transfer function is

$$\frac{M(s)}{E(s)} = K_i/s \tag{3-19}$$

using $1/s$ as the operator for integration. If the error were to go to zero, the output of the controller would remain constant. This feature allows integral controllers to be used when there is some type of constant load on the system. Even if there is no error the controller would still maintain an output signal to counteract the load.

Proportional-plus-Integral Control

Sometimes it is necessary to combine control actions. A proportional controller is incapable of counteracting a load on the system without an error. An integral controller can provide zero error but usually provides slow response.

SOLUTION

$$V_{s1} = (24\text{ V})(\sin 90°) = 24\text{ V}$$

$$V_{s2} = (24\text{ V})(\cos 90°) = 0\text{ V}$$

Example 3-7 At time t the excitation voltage to a resolver is 24 V and $V_{s1} = 17$ V and $V_{s2} = -17$ V. What is the angle?

SOLUTION

$$\arcsin\left(\frac{17}{24}\right) = 45° \quad \text{or} \quad 135°$$

$$\arccos\left(-\frac{17}{24}\right) = 135° \quad \text{or} \quad 225°$$

The shaft angle must be 135°.

Encoders

As more systems become controlled by computers and related devices the use of digital position encoders is increasing. Encoders are available as two basic types: incremental and absolute. This refers to the type of data available from the encoder. There are various categories of encoding devices, but we will limit our discussion to those that are most commonly used in robots. These are optical encoders.

A simple incremental encoder is illustrated in Fig. 3-13. It consists of a

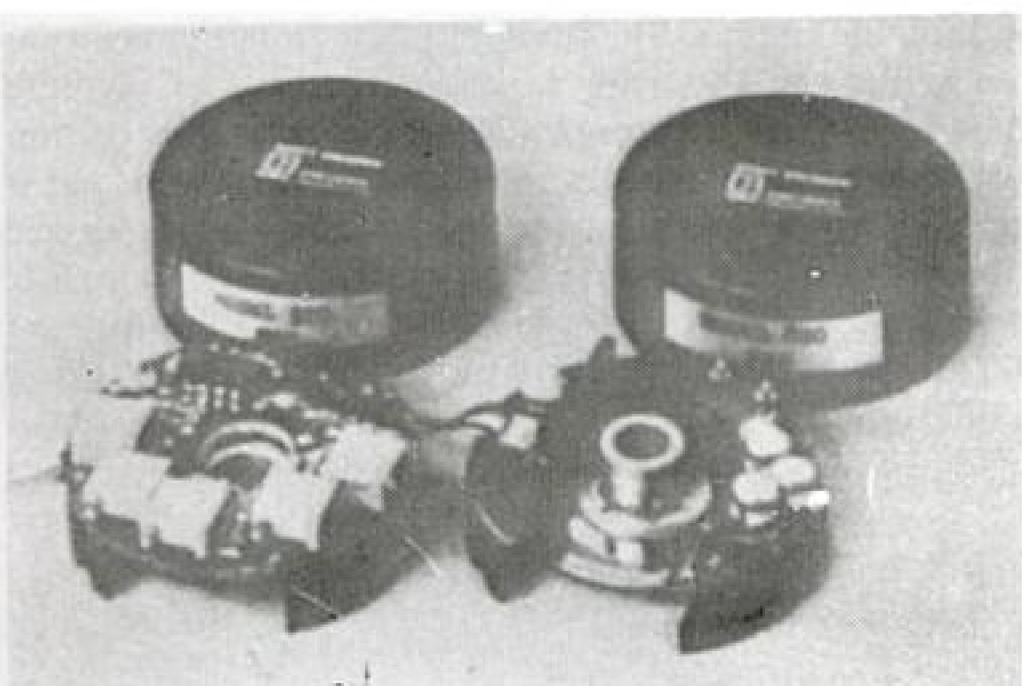


Figure 13-13 Incremental optical encoder (Courtesy of Litton Systems, Incorporated, Encoder Division).

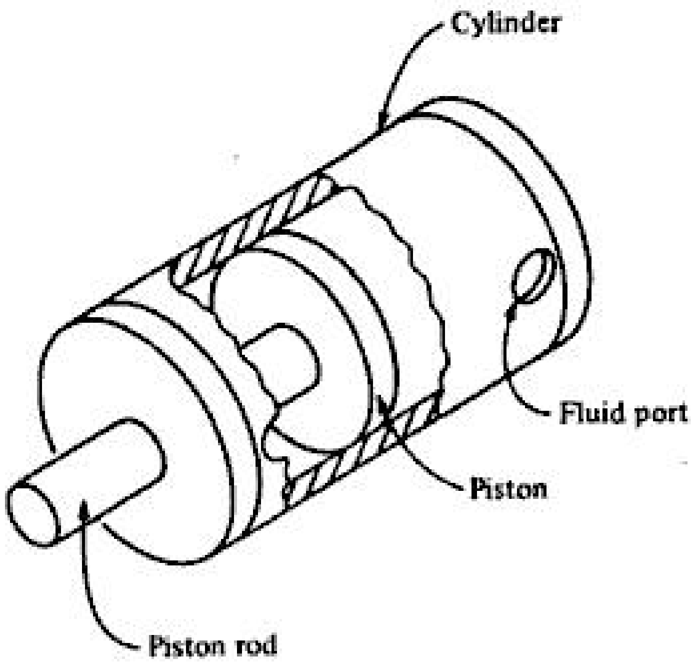


Figure 3-16 Cylinder and piston.

where $V(t)$ is the velocity of the piston, $f(t)$ is the fluid flow rate (volumetric), $F(t)$ is the force, $P(t)$ is the pressure of the fluid, and A is the area of the piston. Since the requirements of a robot are to carry a payload at a given speed we can use the relations described for choosing the appropriate actuator.

Example 3-10 What is the velocity of the piston and the force generated by the piston if the fluid pressure is 1500 lb/in.^{2} inside the cylinder, the piston is 2.0 in. in diameter, and the flow rate is 10 in.^{3}/min?

The piston area is 3.14 in.^{2}.

$$F = (1500 \text{ lb/in.}^2)(3.14 \text{ in.}^2) = 4712 \text{ lb}$$

$$V = \frac{10 \text{ in.}^3\text{/min}}{3.14 \text{ in.}^2} = 3.18 \text{ in./min}$$

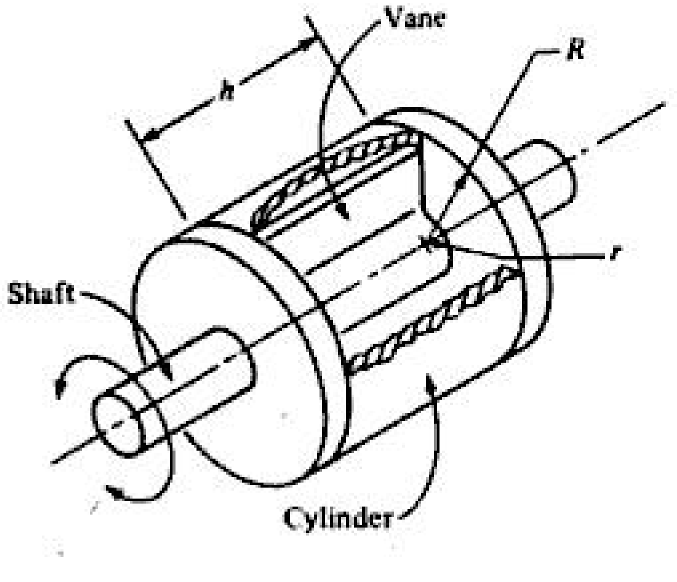


Figure 3-17 Vane actuator.

where μ = the coefficient of friction between the screw threads
β = the thread angle on an Acme or Unified thread
d_m = the mean diameter of the screw

This equation applies for Acme and Unified threads, in which there is a thread angle, β. For square threads, the value of β is 0, and the secant terms = 1.0 in Eq. (3-46).

Example 3-12 A power screw mechanism is used to actuate a linear (type *L*) joint in a new robot design. Determine the maximum force that can be transmitted to the nut moving along the power screw if the torque available to turn the screw is 2.0 in.-lb. The screw has square threads ($\beta = 0$) whose pitch is 0.1 in., the diameter of the screw is 0.50 in., and the coefficient of friction between threads is 0.25.

SOLUTION
$$F = \frac{2(2.0)}{0.5}\,\frac{\pi(0.5) - 0.25(0.1)}{0.1 + 0.25\pi(0.5)} = 25.1 \text{ lb}$$

Because of the relatively high friction in a typical screw thread, ball bearing screws are often used to actuate the linear joints of a robot manipulator. In a ball bearing screw, the nut rides on ball bearings as the screw rotates, rather than directly on the screw itself. This significantly reduces the friction of the device. The conversion from screw torque T to force F resulting at the nut is given by

$$F = \frac{2\pi TE}{P} \tag{3-47}$$

where E = efficiency factor (typically around 90%) resulting from friction losses.

Example 3-13 Let us compare the force resulting from a ball bearing screw with the force in the conventional screw mechanism of Example 3-12. The same values apply: Torque = 2.0 in-lb, pitch = 0.1 in., and we will assume an efficiency factor of 0.90.

SOLUTION:
$$F = \frac{2\pi(2.0)(0.9)}{0.1} = 113.1 \text{ lb}$$

It is clear that the ball bearing screw has a significant mechanical advantage because of the lower friction.

Other Transmission Systems

Other power transmission devices include pulley systems, chain drives, and harmonic drives. Pulley systems are usually used to transmit power from actuators located in the robot's base. In some cases the rope or cord may be

CHAPTER
FOUR
ROBOT MOTION ANALYSIS AND CONTROL

In the preceding chapter we discussed the control of a single robot joint. In order for a robot to perform useful work it is necessary for the arm to consist of a number of joints. The typical commercial industrial robot has five or six joints. It is also necessary to control the path which the end of the arm follows, as opposed to merely controlling the final positions of the joints. In this chapter we will discuss the problems of robot motion control and the mathematical techniques used to analyze manipulator positions and motions. Later in the chapter we explore the dynamics of robot manipulators. Highly motivated readers may wish to pursue the subject of this chapter in more detail, and we recommend Paul's book [9] which has become the standard reference in this area.

4-1 INTRODUCTION TO MANIPULATOR KINEMATICS

In order to develop a scheme for controlling the motion of a manipulator it is necessary to develop techniques for representing the position of the arm at points in time. We will define the robot manipulator using the two basic elements, joints and links. Each joint represents 1 degree of freedom. As discussed in Chap. Two, the joints may involve either linear motion (joint-type L) or rotational motion (joint-types R, T, and V) between the adjacent links. According to our definitions in Chap. Two, the links are assumed to be the rigid structures that connect the joints.

Joints are labeled J_n where n begins with 1 at the base of the manipulator, and links are labeled L_n, again with 1 being the link closest to the base. Figure 4-1 illustrates the labeling system for two different robot arms, each possessing 2 degrees of freedom. By the joint notation scheme described in Chap. Two,

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

the cartesian axis system is often located in the robot's base. The end-of-arm position would be defined in world space as

$$P_w = (x, y)$$

This concept of a point definition in world space can readily be extended to three dimensions, that is, $P_w = (x, y, z)$. Representing an arm's position in world space is useful when the robot must communicate with other machines. These other machines may not have a detailed understanding of the robot's kinematics and so a "neutral" representation such as the world space must be used. In order to use both representations we must be able to transform from one to the other. Going from joint space to world space is called the forward transformation and going from world space to joint space is called the reverse transformation.

Forward Transformation of a 2-Degree of Freedom Arm

We can determine the position of the end of the arm in world space by defining a vector for link 1 and another for link 2.

$$\mathbf{r}_1 = [L_1 \cos \theta_1, L_1 \sin \theta_1] \tag{4-1}$$

$$\mathbf{r}_2 = [L_2 \cos(\theta_1 + \theta_2), L_2 \sin(\theta_1 + \theta_2)] \tag{4-2}$$

Vector addition of (4-1) and (4-2) yields the coordinates x and y of the end of the arm (point P_w) in world space

$$x = L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \tag{4-3}$$

$$y = L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2) \tag{4-4}$$

Reverse Transformation of the 2-Degree of Freedom Arm

In many cases it is more important to be able to derive the joint angles given the end-of-arm position in world space. The typical situation is where the robot's controller must compute the joint angles required to move its end-of-arm to a point in space defined by the point's coordinates. For the two-link manipulator we have developed, there are two possible configurations for reaching the point (x, y), as shown in Fig. 4-3. Some strategy must be developed to select the appropriate configuration. One approach is that employed in the control system of the Unimate PUMA robot. In the PUMA's control language, VAL, there is a set of commands called ABOVE and BELOW that determines whether the elbow is to make an angle θ_2 that is greater than or less than zero, as illustrated in Fig. 4-3. For our example, let us assume the θ_2 is positive as shown in Fig. 4-2. Using the trigonometric identities,

$$\cos(A + B) = \cos A \cos B - \sin A \sin B$$

$$\sin(A + B) = \sin A \cos B + \sin B \cos A$$

of the joint positions relative to the world coordinate system. Using P_4 (x_4, y_4, z_4), which is the position of joint 4, as an example,

$$x_4 = x - \cos\theta(L_4 \cos\psi) \tag{4-12}$$

$$y_4 = y - \sin\theta(L_4 \cos\psi) \tag{4-13}$$

$$z_4 = z - L_4 \sin\psi \tag{4-14}$$

The values of L, ϕ, and θ can next be computed:

$$L = [x_4^2 + y_4^2 + (z_4 - L_1)^2]^{-1} \tag{4-15}$$

$$\sin\phi = \frac{z_4 - L_1}{L} \tag{4-16}$$

$$\cos\theta = \frac{y_4}{L} \tag{4-17}$$

The example we have just done is simple but not unrealistic. In order for a robot controller to be able to perform the calculations necessary quickly enough to maintain good performance they must be kept as simple as possible. The manipulator kinematics described in this example are very similar to those of the MAKER robot, by U.S. Robots. The only real difference is that the MAKER's wrist mechanism has more than a single joint.

One facet of our approach in the preceding analysis which should be noted by the reader is that we separated the orientation problem from the positioning problem. This approach of separating the two problems greatly simplifies the task of arriving at a solution.

4-2 HOMOGENEOUS TRANSFORMATIONS AND ROBOT KINEMATICS

The approach used in the previous section becomes quite cumbersome when a manipulator with many joints must be analyzed. Another, more general method for solving the kinematic equations of a robot arm makes use of homogeneous transformations. We describe this technique in this section, assuming the reader has at least some familiarity with the mathematics of vectors and matrices. Let us begin by defining the notation to be used.

A point vector, $\mathbf{v} = a\mathbf{i} + b\mathbf{j} + c\mathbf{k}$ can be represented in three-dimensional space by the column matrix

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

where $a = x/w$, $b = y/w$, $c = z/w$, and w is a scaling factor. For example, any of the following matrices can be used to represent the vector $\mathbf{v} = 25\mathbf{i} + 10\mathbf{j} + 20\mathbf{k}$.

The position and orientation of the end of a manipulator can be described as the product of n homogeneous transformations, one for each of the n joints of the manipulator. Similarly, the position of each and any joint in the robot arm can be considered as the product of transformations describing the rotations and translations of the preceding joint–link combinations.

Figure 4-8 illustrates the four variables affected by the joint–link combination. The length of the link, a_n, iş defined as the distance along the line that is mutually perpendicular to the axes of two adjacent joints. The twist between the axes of the joints in a plane perpendicular to a_n is the angle t_n. The distance d_n is the distance between the normal a_n and a_{n-1} of the two links. This is sometimes referred to as the joint offset. In the case of a linear joint, this is the joint variable. The angle θ_n is the angle between the links measured as the angle between the normals a_n and a_{n-1} in the plane normal to the axis of the joint. The position and orientation of joint n is completely defined with respect to joint $n-1$ by the four parameters a_n, t_n, θ_n, and d_n.

We can assign a coordinate frame to each link and establish the relationships between succeeding links using transformations for each of the four variables. The origin of the coordinate frame for link n is imbedded at the intersection of the axis, z_n, for the joint $n-1$ and the common normal, a_n from joint n to joint $n+1$. That is, the frame is imbedded at the end of the link in the successive joint. This is accomplished using:

A rotation of angle θ_n about the z_{n-1} axis
A translation of distance d_n along z_{n-1}
A translation along x_n by a length a_n
A rotation about x_n by a twist angle t_n

Using the transformations defined previously for rotation and translation, we can develop the transformation which relates the coordinate frame of link n

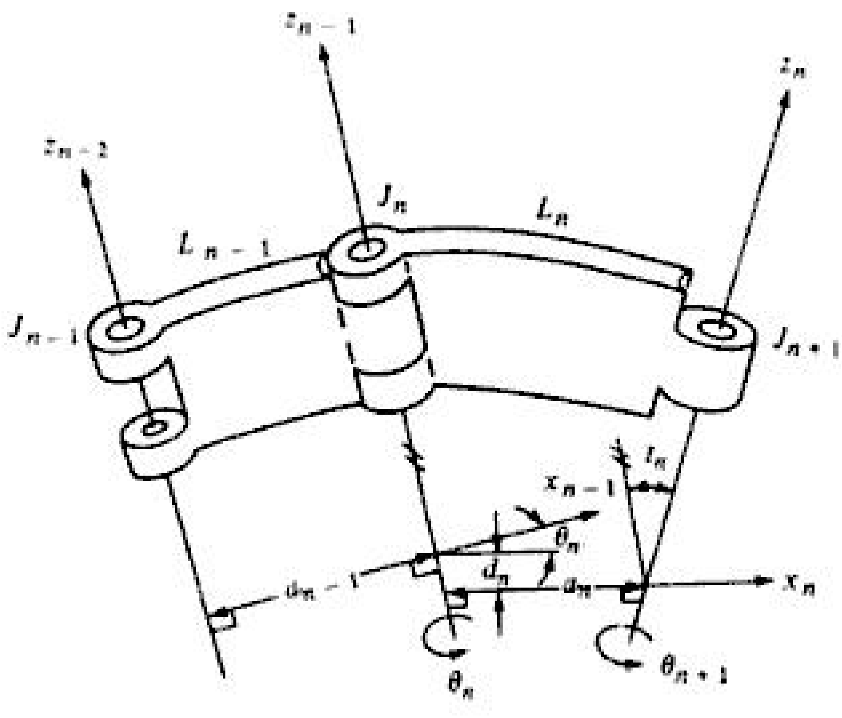


Figure 4-8 The variables in a link using the notation of Paul [9]. The rules used to define the notation are: (1) Axis z_{n-1} defines the position of the axis of rotation for joint J_n, z_n for joint J_{n+1}, and so forth. (2) Axis x_{n-1} is selected to be an extension for the common perpendicular line of length a_{n-1} between consecutive joints z_{n-2} and z_{n-1}. (3) The axis y_{n-1} is selected to provide a right-hand coordinate system with the other axes. (4) Axis x_n is an extension of the common perpendicular line of length a_n.

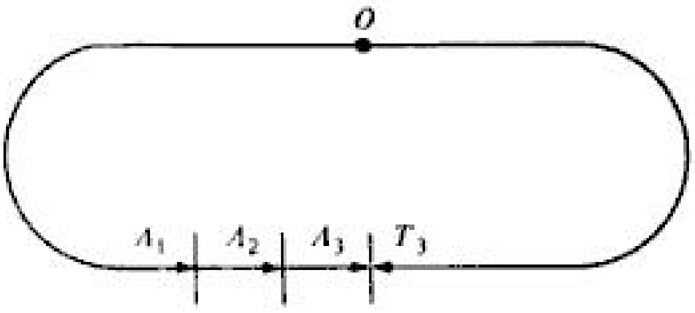


Figure 4-11 Transform graph for manipulator of Figure 4-9.

or

$$A_2^{-1}A_1^{-1}T_3 = {}^2T_3 \tag{4-37}$$

and so forth.

We can make use of these equivalents to solve for the joint positions given the end-of-arm position in the form of **T** in Eq. (4-23). Substituting Eq. (4-27) for $\mathbf{A}_1$ the inverse is

$$\mathbf{A}_1^{-1} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4-38}$$

Substituting Eqs. (4-38), (4-23), and (4-33) into (4-36) we get

$$\begin{bmatrix} q1 & q2 & q3 & q4 \\ -n_z & -o_z & -a_z & -p_z \\ q5 & q6 & q7 & q8 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & d_3\sin\theta_2 \\ \sin\theta_2 & 0 & -\cos\theta_2 & -d_3\cos\theta_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4-39}$$

where $q1 = n_x\cos\theta_1 + n_y\sin\theta_1$
$q2 = o_x\cos\theta_1 + o_y\sin\theta_1$
$q3 = a_x\cos\theta_1 + a_y\sin\theta_1$
$q4 = p_x\cos\theta_1 + p_y\sin\theta_1$
$q5 = -n_x\sin\theta_1 + n_y\cos\theta_1$
$q6 = -o_x\sin\theta_1 + o_y\cos\theta_1$
$q7 = -a_x\sin\theta_1 + a_y\cos\theta_1$
$q8 = -p_x\sin\theta_1 + p_y\cos\theta_1$

We can use this to solve for θ_1, θ_2, and d_3. From Eq. (4-39) we see that

$$q2 = o_x\cos\theta_1 + o_y\sin\theta_1 = 0 \tag{4-40}$$

so that

$$\tan\theta_1 = -\frac{o_x}{o_y} \tag{4-41}$$

We can also see that

$$\sin\theta_2 = -n_z \tag{4-42}$$

4-4 ROBOT DYNAMICS

Accurate control of the manipulator requires precise control of each joint. The control of the joint depends on knowledge of the forces that will be acting on the joint and the inertias reflected at the joint (the masses of the joints and links of the manipulator). While these forces and masses are relatively easy to determine for a single joint, it becomes more difficult to determine them as the complexity of the manipulator increases. We will explore these issues using the two-axis manipulator developed earlier. Our purpose is to introduce this problem area to the reader rather than to analyze its complexities in detail.

Static Analysis

Let us begin by considering the torques required by the joints to produce a force **F** at the tip of the robot arm as shown in Fig. 4-13. By balancing the forces on each link we get

$$\mathbf{F}_1 - \mathbf{F}_2 = 0 \tag{4-49}$$

and

$$\mathbf{F}_2 - \mathbf{F} = 0 \tag{4-50}$$

That is,

$$\mathbf{F}_1 = \mathbf{F}_2 = \mathbf{F} \tag{4-51}$$

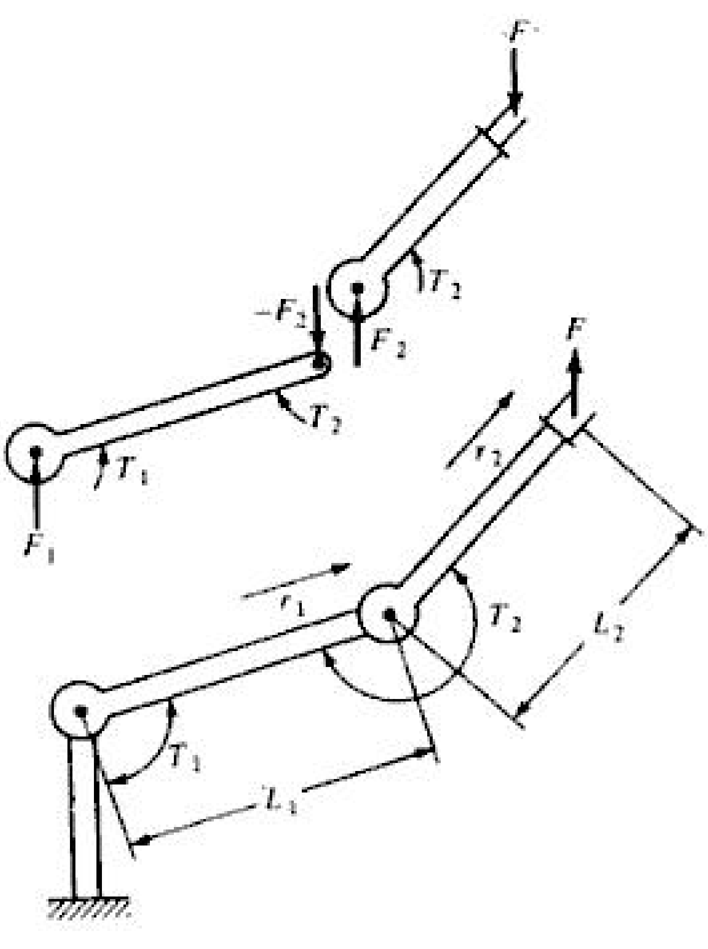


Figure 4-13 Two-link arm forces and torques.

power amplifiers, mathematical processor, executive processor, program memory, and input device. The number of joint servocontrollers and joint power amplifiers would correspond to the number of joints in the manipulator. These elements might be organized in the robot controller as shown in Fig. 4-17.

Motion commands are executed by the controller from two possible sources: operator input or program memory. Either an operator inputs commands to the system using an input device such as a teach pendant or a CRT terminal, or the commands are downloaded to the system from program memory under control of the executive processor. In the second case, the set of commands have been previously programmed into memory using the operator input device(s). For each motion command, the executive processor informs the mathematical processor of the coordinate transformation calculations that must be made. When the transformation computations are completed, the executive processor downloads the results to the joint controllers as position commands. Each joint controller then drives its corresponding joint actuator by means of the power amplifier.

Microprocessors are typically utilized in several of the components of a modern robot controller. These components include the mathematical proces-

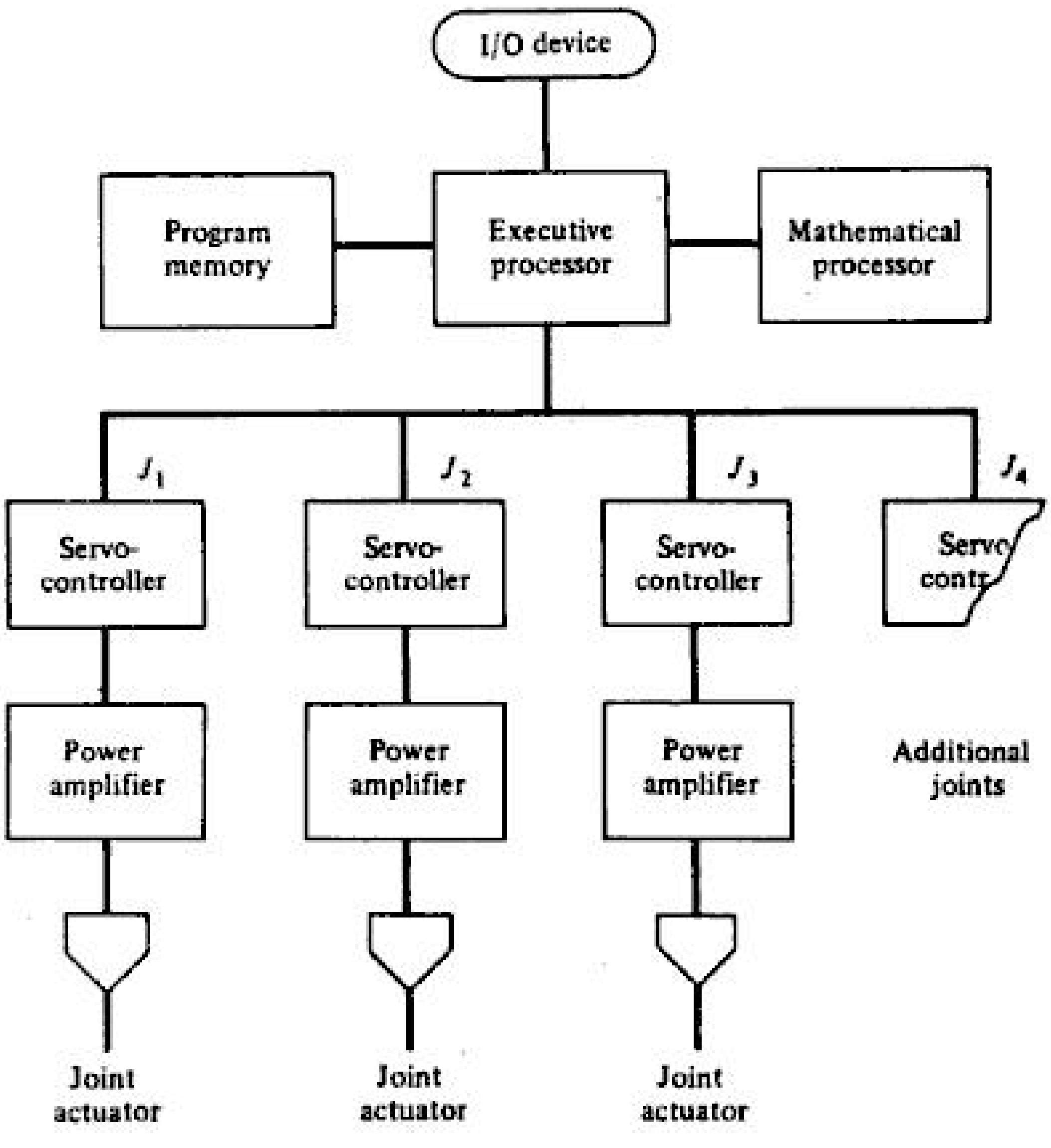


Figure 4-17 General robot controller elements.

4-20 Find the inverses of the **A** matrices in Prob. 4-19.

4-21 Find the following for the manipulator in Fig. P4-19:

(*a*) $^{5}\mathbf{T}_6$ (*b*) $^{4}\mathbf{T}_6$ (*c*) $^{3}\mathbf{T}_6$
(*d*) $^{2}\mathbf{T}_6$ (*e*) $^{1}\mathbf{T}_6$ (*f*) $^{0}\mathbf{T}_6$

4-22 Consider the coordinate frames **O**, **A**, **B**, and **C** shown in Fig. P4-22, where **O** is the reference frame. Determine, either by inspection or by means of Eq. (4-25) in the text, the homogeneous transforms corresponding to frames **A**, **B**, and **C** with respect to **O**.

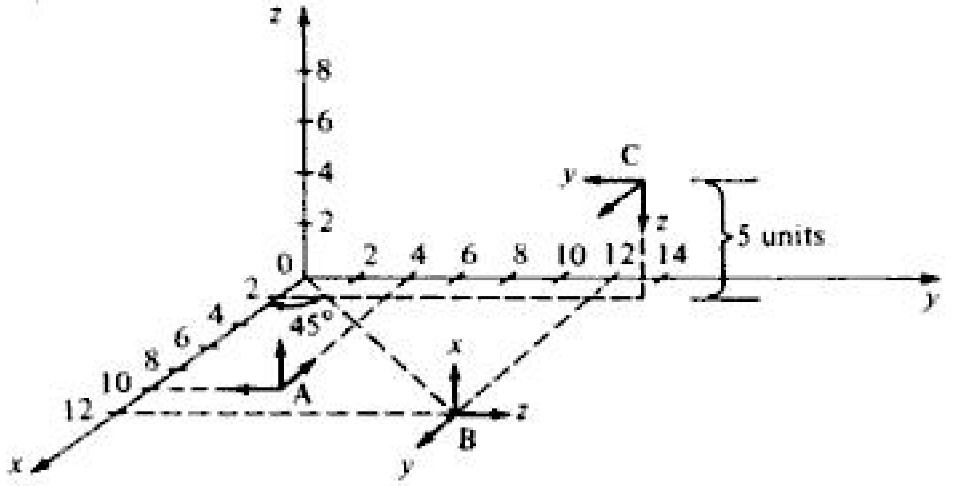


Figure P4-22

4-23 Consider a jointed-arm robot manipulator with its x, y, and z axes aligned with a reference cartesian coordinate frame but located at $(x, y) = (10\,\text{ft}, -5\,\text{ft})$. The end-of-arm of the robot is currently at $(x, y, z) = (12\,\text{ft}, 2\,\text{ft}, 2.5\,\text{ft})$ relative to the reference coordinate frame. An end effector of 10 in. in length is attached to the end-of-arm and is pointing vertically down. Relative to the tip of the end effector is a cube, with 6 in. on a side, and with its nearest corner positioned +1 ft in the x direction, +2 ft in the y direction, and 0 ft in the z direction from the tip of the end effector.

(*a*) Make a sketch of the workcell.

(*b*) Identify all transforms numerically.

(*c*) Show by means of the transform graph how you would solve for the transform for the cube relative to the end effector. That is, determine all transforms needed to find the transform of the cube relative to the end effector.

4-24 Consider the prism shown in Fig. P4-24. The positions of the prism vertices have been indicated relative to the reference axis system. Positions are given in meters. From its current position, the prism is rotated 90° about the z axis, followed by a 90° rotation about the y axis, followed by a translation of −2 m in the x direction.

(*a*) Define the transformation which describes the change in position of the prism. That is, determine the 4×4 homogeneous transform for the move.

(*b*) What are the new coordinates of the vertices of the prism after the move?

(*c*) What is the inverse transform and how should it be interpreted?

The following problems are concerned with motion of multiple joints in a robot manipulator (Sec. 4-3).

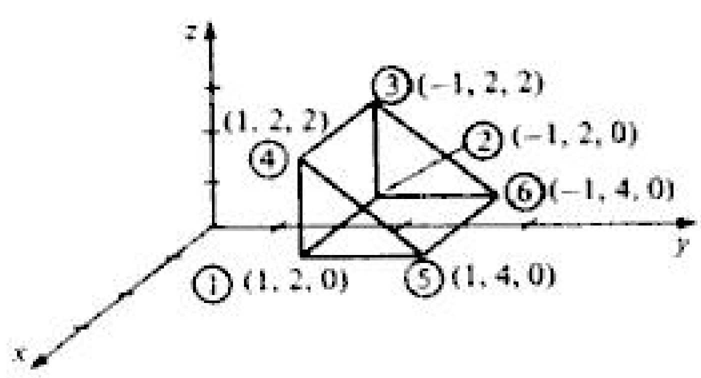


Figure P4-24

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

rather than workparts. The reason for using a gripper instead of attaching the tool directly to the robot's wrist is typically because the job requires several tools to be manipulated by the robot during the work cycle. An example of this kind of application would be a deburring operation in which several different sizes and geometries of deburring tool must be held in order to reach all surfaces of the workpart. The gripper serves as a quick change device to provide the capability for a rapid changeover from one tool to the next.

In addition to end effectors, other types of fixturing and tooling are required in many industrial robot applications. These include holding fixtures, welding fixtures, and other forms of holding devices to position the workpart or tooling during the work cycle.

5-2 MECHANICAL GRIPPERS

Basic Definitions and Operation

A mechanical gripper is an end effector that uses mechanical fingers actuated by a mechanism to grasp an object. The fingers, sometimes called jaws, are the appendages of the gripper that actually make contact with the object. The fingers are either attached to the mechanism or are an integral part of the mechanism. If the fingers are of the attachable type, then they can be detached and replaced. The use of replaceable fingers allows for wear and interchangeability. Different sets of fingers for use with the same gripper mechanism can be designed to accommodate different part models. An example of this interchangeability feature is illustrated in Fig. 5-1, in which the gripper is designed to accommodate fingers of varying sizes. In most applications, two fingers are sufficient to hold the workpart or other object. Grippers with three or more fingers are less common.

The function of the gripper mechanism is to translate some form of power input into the grasping action of the fingers against the part. The power input is supplied from the robot and can be pneumatic, electric, mechanical, or hydraulic. We will discuss the alternatives in Sec. 5-5. The mechanism must be able to open and close the fingers and to exert sufficient force against the part when closed to hold it securely.

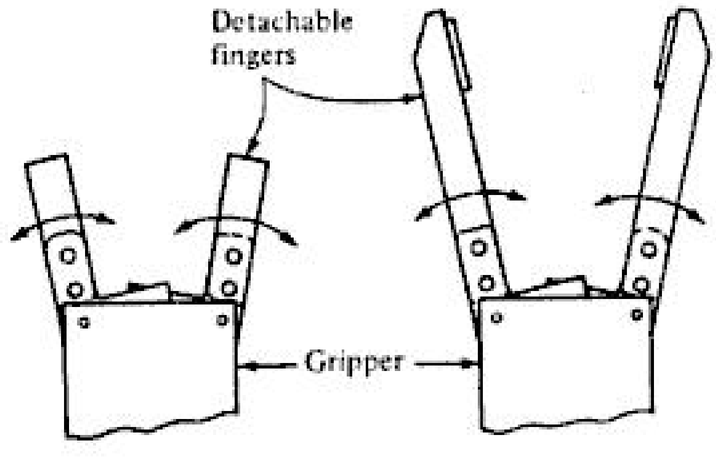


Figure 5-1 Interchangeable fingers can be used with the same gripper mechanism.

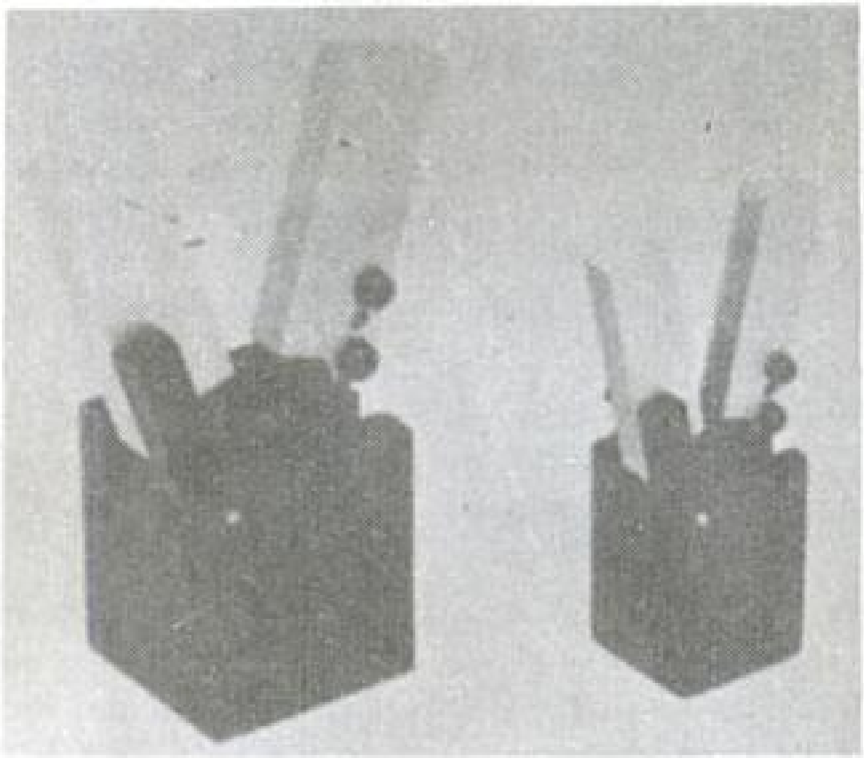


Figure 5-4 Mechanical gripper finger with pivoting movement (Photo courtesy of Phd, Inc.)

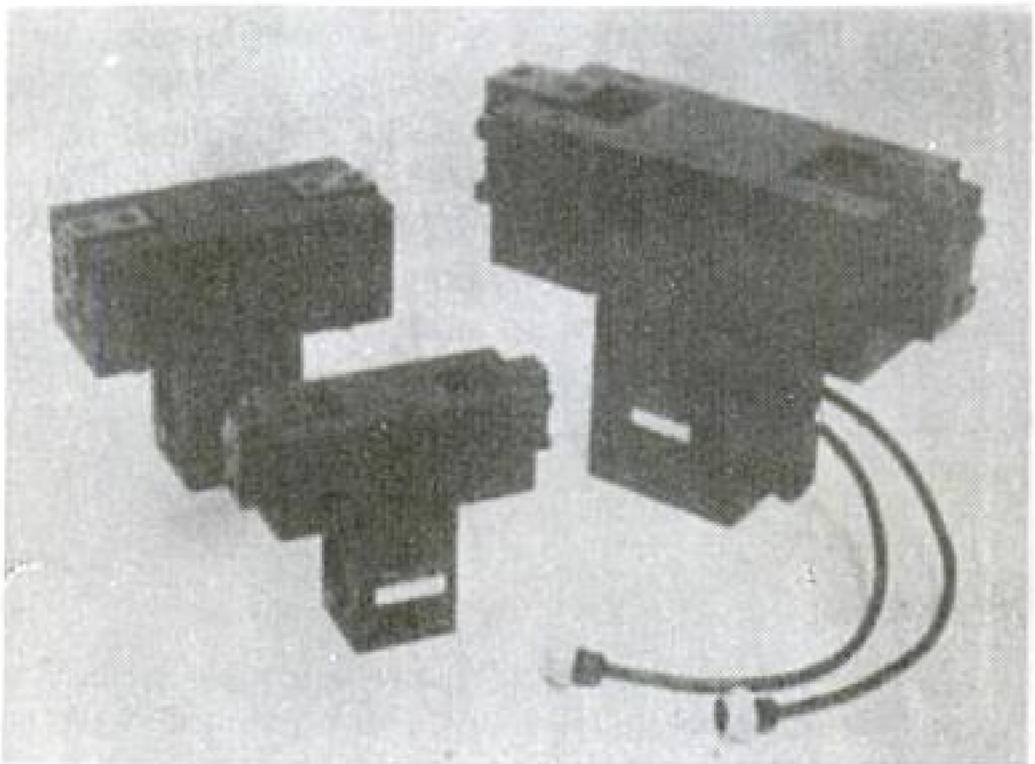


Figure 5-5 Mechanical gripper finger with linear movement using guide rails (Photo courtesy of Phd, Inc.)

linkage which would maintain the fingers in a parallel orientation to each other during actuation.

Mechanical grippers can also be classed according to the type of kinematic device used to actuate the finger movement [2]. In this classification we have the following types:

1. Linkage actuation
2. Gear-and-rack actuation
3. Cam actuation

The piston device would have to provide an actuating force of 240 lb to close the gripper with a force against the carton of 60 lb.

Example 5-4 The diagram in Fig. 5-11 shows the linkage mechanism and dimensions of a gripper used to handle a workpart for a machining operation. Suppose it has been determined that the gripper force is to be 25 lb. What is required is to compute the actuating force to deliver this force of 25 lb.

Figure 5-12(*a*) shows how the symmetry of the gripper can be used to

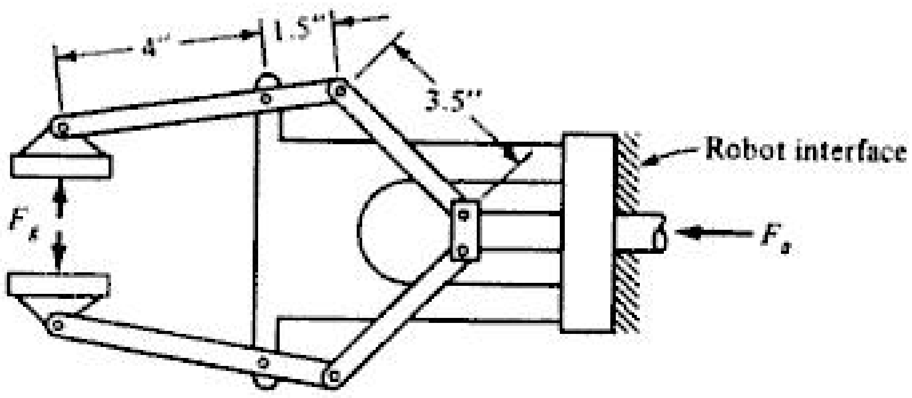


Figure 5-11 Gripper considered in Example 5-4.

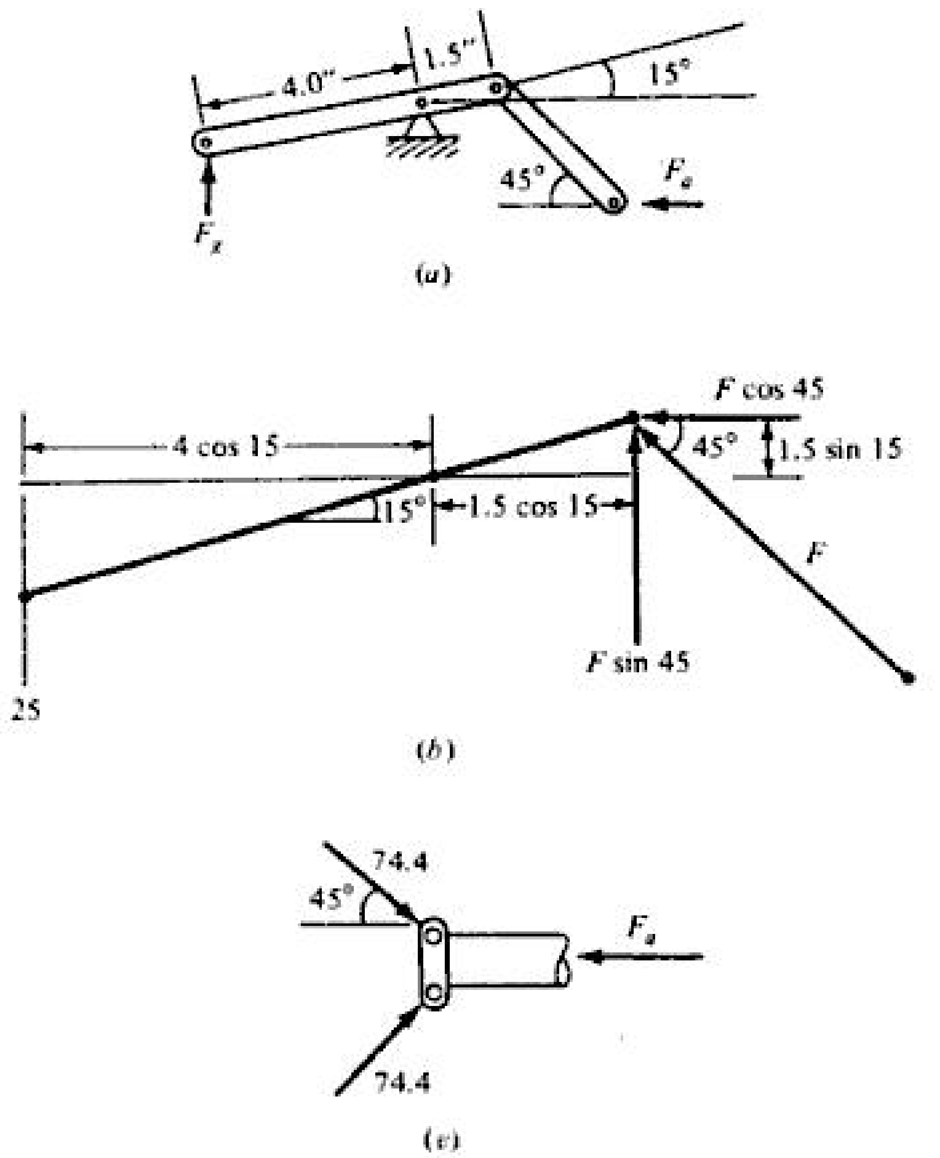


Figure 5-12 Linkage analysis of Example 5-4.

suction cup grippers used in robotics applications are:

Requires only one surface of the part for grasping
Applies a uniform pressure distribution on the surface of the part
Relatively light-weight gripper
Applicable to a variety of different materials

Magnetic Grippers

Magnetic grippers can be a very feasible means of handling ferrous materials. The stainless steel plate in Example 5-3 would not be an appropriate application for a magnetic gripper because 18-8 stainless steel is not attracted by a magnet. Other steels, however, including certain types of stainless steel, would be suitable candidates for this means of handling, especially when the materials are handled in sheet or plate form.

In general, magnetic grippers offer the following advantages in robotic-handling applications:

Pickup times are very fast.
Variations in part size can be tolerated. The gripper does not have to be designed for one particular workpart.
They have the ability to handle metal parts with holes (not possible with vacuum grippers).
They require only one surface for gripping.

Disadvantages with magnetic grippers include the residual magnetism remaining in the workpiece which may cause a problem in subsequent handling, and the possible side slippage and other errors which limit the precision of this means of handling. Another potential disadvantage of a magnetic gripper is the problem of picking up only one sheet from a stack. The magnetic attraction tends to penetrate beyond the top sheet in the stack, resulting in the possibility that more than a single sheet will be lifted by the magnet. This problem can be confronted in several ways. First, magnetic grippers can be designed to limit the effective penetration to the desired depth, which would correspond to the thickness of the top sheet. Second, the stacking device used to hold the sheets can be designed to separate the sheets for pickup by the robot. One such type of stacking device is called a "fanner," and it makes use of a magnetic field to induce a charge in the ferrous sheets in the stack. Each sheet toward the top of the stack is given a magnetic charge, causing them to possess the same polarity and repel each other. The sheet most affected is the one at the top of the stack. It tends to rise above the remainder of the stack, thus facilitating pickup by the robot gripper.

Magnetic grippers can be divided into two categories, those using electromagnets, and those using permanent magnets. Electromagnetic grippers are easier to control, but require a source of dc power and an appropriate controller unit. As with any other robotic-gripping device, the part must be

In each case, the robot must control the actuation of the tool. For example, the robot must coordinate the actuation of the spot-welding operation as part of its work cycle. This is controlled much in the same manner as the opening and closing of a mechanical gripper. We will discuss the interface between the robot and its end effector in the following section. Design and application considerations of most of the robot tools listed above will be considered in Chaps. Fourteen and Fifteen of the book.

5-5 THE ROBOT/END EFFECTOR INTERFACE

An important aspect of the end effector applications engineering involves the interfacing of the end effector with the robot. This interface must accomplish at least some of the following functions:

Physical support of the end effector during the work cycle must be provided.
Power to actuate the end effector must be supplied through the interface.
Control signals to actuate the end effector must be provided. This is often accomplished by controlling the actuating power.
Feedback signals must sometimes be transmitted back through the interface to the robot controller.

In addition, certain other general-design objectives should be met. These include high reliability of the interface, protection against the environment, and overload protection in case of disturbances and unexpected events during the work cycle.

Physical Support of the End Effector

The physical support of the end effector is achieved by the mechanical connection between the end effector and the robot wrist. This mechanical connection often consists of a faceplate at the end of the wrist to which the end effector is bolted. In other cases, a more complicated wrist socket is used. Ideally, there should be three characteristics taken into consideration in the design of the mechanical connection [7]; strength, compliance, and overload protection. The strength of the mechanical connection refers to its ability to withstand the forces associated with the operation of the end effector. These forces include the weight of the end effector, the weight of the objects being held by the end effector if it is a gripper, acceleration and deceleration forces, and any applied forces during the work cycle (e.g., thrust forces during a drilling operation). The wrist socket must provide sufficient strength and rigidity to support the end effector against these various forces.

The second consideration in the design of the mechanical connection is compliance. Compliance refers to the wrist socket's ability to yield elastically when subjected to a force. In effect, it is the opposite of rigidity. In some applications, it is desirable to design the mechanical interface so that it will

grippers. When a venturi device is used to provide the vacuum, the device can be actuated by shop air pressure. Otherwise, some means of developing and controlling the vacuum must be provided to operate the suction cup gripping device.

A second method of power transmission to the end effector is electrical. Pneumatic actuation of the gripper is generally limited to two positions, open and closed. The use of an electric motor can allow the designer to exercise a greater degree of control over the actuation of the gripper and of the holding force applied. Instead of merely two positions, the gripper can be controlled to any number of partially closed positions. This feature allows the gripper to be used to handle a variety of objects of different sizes, a likely requirement in assembly operations. By incorporating force sensors into the gripper fingers, a feedback control system can be built into the gripper to regulate the holding force applied by the fingers rather than their position. This would be useful, for example, if the objects being grasped are delicate or if the objects vary in size and the proper finger positions for gripping are not known.

Other uses of electric power for end effectors include electromagnet grippers, spot-welding and arc-welding tools, and powered spindle tools used as robot end effectors.

Hydraulic and mechanical power transmission are less common means of actuating the end effector in current practice. Hydraulic actuation of the gripper has the potential to provide very high holding forces, but its disadvantage is the risk of oil leaks. Mechanical power transmission would involve an arrangement in which a motor (e.g., pneumatic, electric, hydraulic, etc.) is mounted on the robot arm and connected mechanically to the gripper, perhaps by means of a flexible cable or the use of pulleys. The possible advantage of this arrangement is a reduction of the weight and mass at the robot's wrist.

5-6 CONSIDERATIONS IN GRIPPER SELECTION AND DESIGN

Most of this chapter has been concerned with grippers rather than tools as end effectors. As indicated in Sec. 5-4, tools are used for spot welding, arc welding, rotating spindle operations, and other processing applications. We will examine the tooling used with these operations when we discuss the corresponding applications in Chaps. Fourteen and Fifteen. In this section, let us summarize our discussion of grippers by enumerating some of the considerations in their selection and design.

Certainly one of the considerations deals with determining the grasping requirements for the gripper. Engelberger [3] defines many of the factors that should be considered in assessing gripping requirements. The following list is based on Engelberger's discussion of these factors:

1. The part surface to be grasped must be reachable. For example, it must not be enclosed within a chuck or other holding fixture.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

5-6 For the information given in the mechanical gripper design of Fig. P5-6, calculate the required actuating force if the gripper force is to be 20 lb.

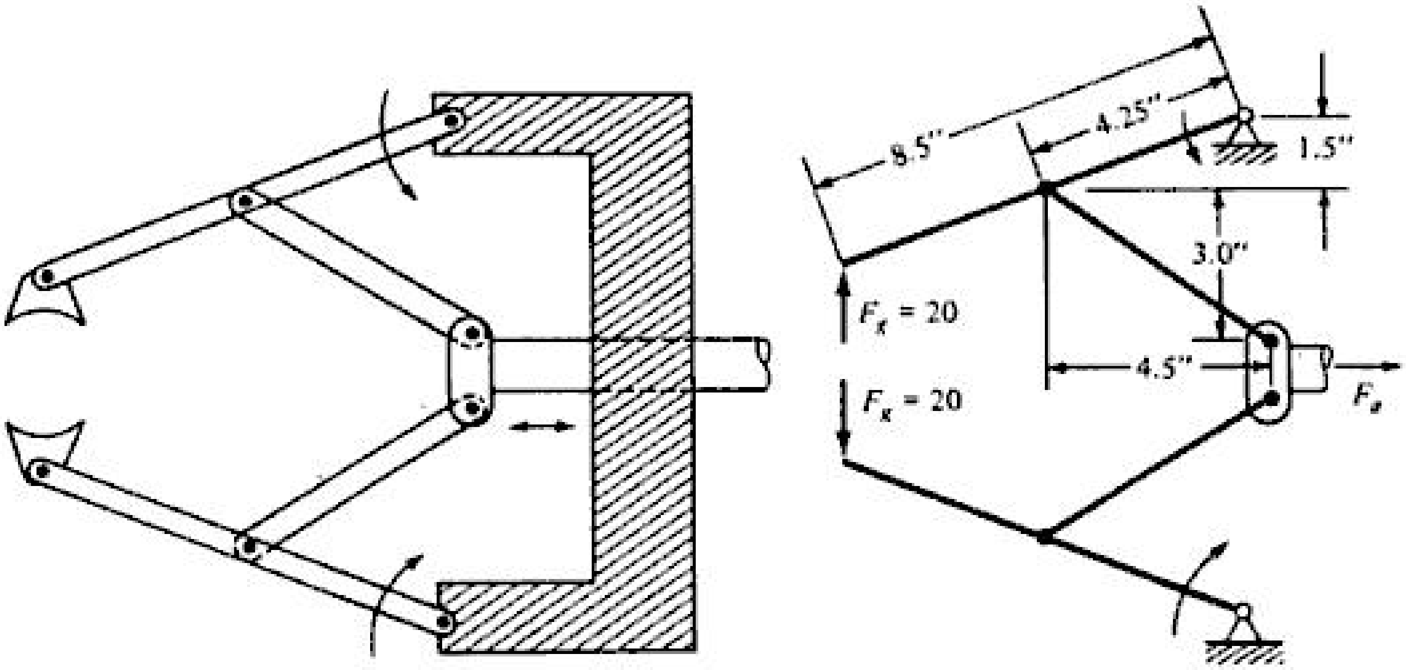


Figure P5-6

5-7 A vacuum gripper is to be designed to handle flat plate glass in an automobile windshield plant. Each plate weighs 28 lb. A single suction cup will be used and the diameter of the suction cup is 6.0 in. Determine the negative pressure required (compared to atmospheric pressure of 14.7 lb/in.^{2}) to lift each plate. Use a safety factor of 1.5 in your calculations.

5-8 A vacuum pump to be used in a robot vacuum gripper application is capable of drawing a negative pressure of 4.0 lb/in.^{2} compared to atmospheric pressure. The gripper is to be used for lifting stainless steel plates, each plate having dimensions of 15 by 35 in. and weighing 52 lb. Determine the diameter of the suctions cups to be used for the robot gripper if it has been decided that two suction cups will be used for the gripper for greater stability. A factor of safety of 1.5 should be used in the design computations.

5-9 A piston is to be designed to exert an actuation force of 120 lb on its extension stroke. The inside diameter of the piston is 2.0 in. and the ram diameter is 0.375 in. What shop air pressure will be required to provide this actuation force? Use a safety factor of 1.3 in your computations.

5-10 The mechanical gripper in Fig. P5-10 uses friction to grasp a part weighing 10 lb. The

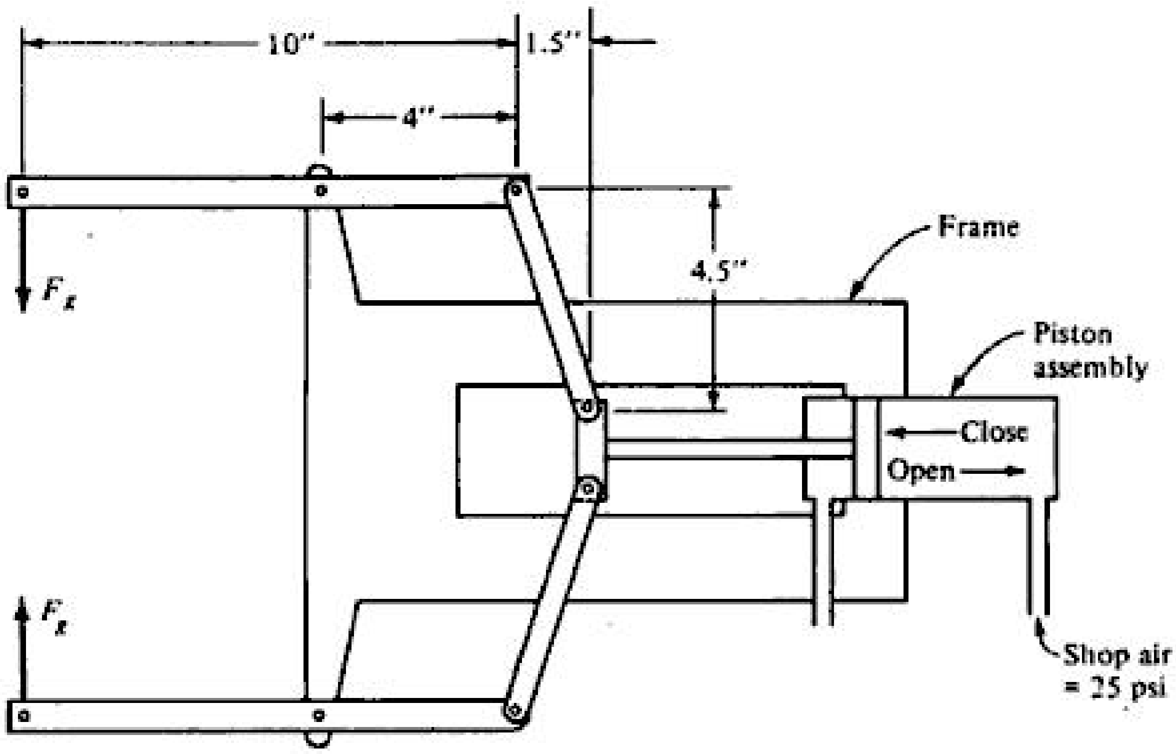


Figure P5-10

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Table 6-2 Sensor devices used in robot Workcells†

Ammeter—(miscellaneous). Electrical meter used to measure electrical current.

Eddy current detectors—(proximity sensor). Device that emits an alternating magnetic field at the tip of a probe, which induces eddy currents in any conductive object in the range of the device. Can be used to indicate presence or absence of a conductive object.

Electrical contact switch—(touch sensor). Device in which an electrical potential is established between two objects, and when the potential becomes zero, this indicates contact between the two objects. Not a commercial device. Can be used to indicate presence or absence of a conductive object.

Infrared sensor—(proximity sensor). Transducer which measures temperatures by the infrared light emitted from the surface of an object. Can be used to indicate presence or absence of a hot object.

Limit switch—(touch sensor). Electrical on–off switch actuated by depressing a mechanical lever or button on the device. Can be used to measure presence or absence of an object.

Linear variable differential transformer—(miscellaneous). Electromechanical transducer used to measure linear or angular displacement.

Microswitch—(touch sensor). Small electrical limit switch (see limit switch). Can be used to indicate presence or absence of an object.

Ohmmeter—(miscellaneous). Meter used to measure electrical resistance.

Optical pyrometer—(proximity sensor, miscellaneous). Device used to measure high temperatures by sensing the brightness of an object's surface. Can be used to indicate presence or absence of a hot object.

Photometric sensors—(proximity sensor, miscellaneous). Various transducers used to sense light. Category includes photocells, photoelectric transducers, phototubes, photodiodes, phototransistors, and photoconductors. Can be used to indicate presence or absence of an object.

Piezoelectric accelerometer—(miscellaneous). Sensor used to indicate or measure vibration.

Potentiometer—(miscellaneous). Electrical meter used to measure voltage.

Pressure transducers—(miscellaneous). Various transducers used to indicate air pressure and other fluid pressures.

Radiation pyrometer—(proximity sensor, miscellaneous). Device used to measure high temperatures by sensing the thermal radiation emitting from the surface of an object. Can be used to indicate presence or absence of a hot object.

Strain gauge—(force sensor). Common transducer used to measure force, torque, pressure, and other related variables. Can be used to indicate force applied to grasp an object.

Thermistor—(miscellaneous). Device based on electrical resistance used to measure temperatures.

Thermocouple—(miscellaneous). Commonly used device used to measure temperatures. Based on the physical principle that a junction of two dissimilar metals will emit an emf which can be related to temperature.

Vacuum switches—(proximity sensor, miscellaneous). Device used to indicate negative air pressures. Can be used with a vacuum gripper to indicate presence or absence of an object.

Vision sensors—(vision system). Advanced sensor system used in conjunction with pattern recognition and other techniques to view and interpret events occurring in the robot workplace.

Voice sensors—(voice and speech recognition). Advanced sensor system used to communicate commands or information orally to the robot.

† Adapted from refs. 1, 2, and 6.

the device must be sensitive enough to detect small forces. This design problem is usually solved by using overtravel limits. An overtravel limit is a physical stop designed to prevent the force sensor from deflecting so far that it would be damaged.

The calculations required to utilize a force-sensing wrist are complex and require considerable computation time. Also, for an arm traveling at moderate-to-high speeds, the level of control over the arm just as it makes contact with an object is limited by the dynamic performance of the arm. The momentum of the arm makes it difficult to stop its forward motion quickly enough to prevent a crash.

Joint sensing If the robot uses dc servomotors then the torque being exerted by the motors is proportional to the current flowing through the armature. A simple way to measure this current is to measure the voltage drop across a small precision resistor in series with the motor and power amplifier. This simplicity makes this technique attractive; however, measuring the joint torque has several disadvantages. First, measurements are made in joint space, while the forces of interest are applied by the tool and would be more useful if made in tool space. The measurements therefore not only reflect the forces being applied at the tool, but also the forces and torques required to accelerate the links of the arm and to overcome the friction and transmission losses of the joints. In fact, if the joint friction is relatively high (and it usually is), it will mask out the small forces being applied at the tool tip. One area where joint torque sensing shows promise of working well is with direct-drive robots. Direct-drive robots are a relatively new innovation in which the drive motors are located at the joints of the manipulator. In torque sensing, this configuration reduces the friction and transmission losses, and the problems of torque measurement which accompany these losses are thereby reduced. We will discuss direct-drive robots in more detail in Chap. Nineteen.

Tactile array sensors A tactile array sensor is a special type of force sensor composed of a matrix of force-sensing elements. The force data provided by this type of device may be combined with pattern recognition techniques to describe a number of characteristics about the impression contacting the array sensor surface. Among these characteristics are (1) the presence of an object, (2) the object's contact area, shape, location, and orientation, (3) the pressure and pressure distribution, and (4) force magnitude and location. Tactile array sensors can be mounted in the fingers of the robot gripper or attached to a work table as a flat touch surface. Figures 6-3 and 6-4 illustrate these two possible mountings for the sensor device.

The device is typically composed of an array of conductive elastomer pads. As each pad is squeezed its electrical resistance changes in response to the amount of deflection in the pad, which is proportional to the applied force. By measuring the resistance of each pad, information about the shape of the object against the array of sensing elements can be determined. The operation of a tactile array sensor (with an 8×8 matrix of pressure-sensitive pads) is

several millimeters and several feet. Some of these sensors can also be used to measure the distance between the object and the sensor, and these devices are called range sensors. Proximity and range sensors would typically be located on the wrist or end effector since these are the moving parts of the robot. One practical use of a proximity sensor in robotics would be to detect the presence or absence of a workpart or other object. Another important application is for sensing human beings in the robot workcell. Range sensors would be useful for determining the location of an object (e.g., the workpart) in relation to the robot.

A variety of technologies are available for designing proximity and range sensors. These technologies include optical devices, acoustics, electrical field techniques (e.g. eddy currents and magnetic fields), and others. We will survey only a few of the possibilities in the following paragraphs.

Optical proximity sensors can be designed using either visible or invisible (infrared) light sources. Infrared sensors may be active or passive. The active sensors send out an infrared beam and respond to the reflection of the beam against a target. The infrared-reflectance sensor using an incandescent light source is a common device that is commercially available. The active infrared sensor can be used to indicate not only whether or not a part is present, but also the position of the part. By timing the interval from when the signal is sent and the echo is received, a measurement of the distance between the object and the sensor can be made. This feature is especially useful for locomotion and guidance systems. Passive infrared sensors are simply devices which detect the presence of infrared radiation in the environment. They are often utilized in security systems to detect the presence of bodies giving off heat within the range of the sensor. These sensor systems are effective at covering large areas in building interiors.

Another optical approach for proximity sensing involves the use of a collimated light beam and a linear array of light sensors. By reflecting the light beam off the surface of the object, the location of the object can be determined from the position of its reflected beam on the sensor array. This scheme is illustrated in Fig. 6-7. The formula for the distance between the object and

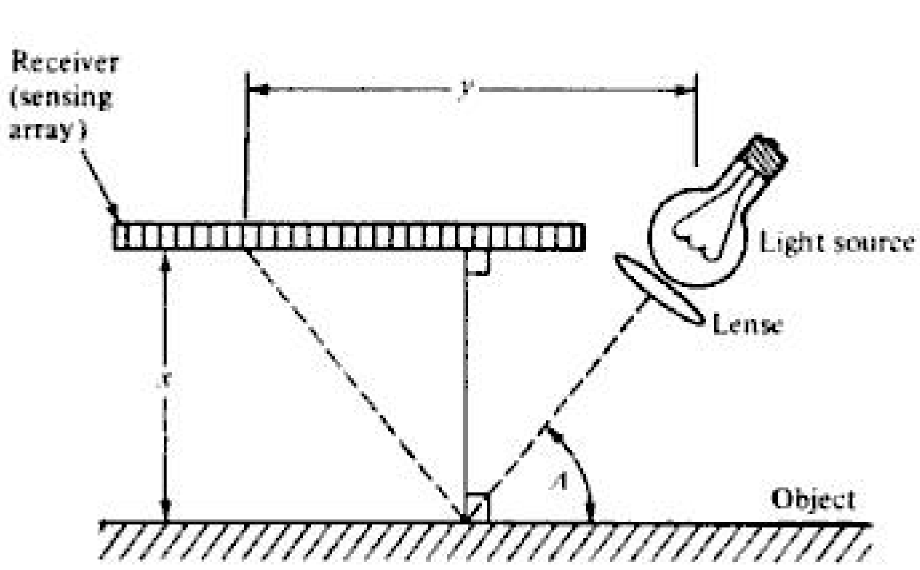


Figure 6-7 Scheme for a proximity sensor using reflected light against a sensor array.

remote center compliance (RCC) device. The use of the RCC device in assembly will be described in Chap. Fifteen.

All four categories of sensor applications (safety monitoring, interlocks, inspection, and positional data) are instances where the sensor constitutes a component of a control system used in the robot work cell to accomplish some specific control function. That control system, in turn, is a component of a larger control system which we are calling the workcell control system. All of the control functions which takes place in the workcell are coordinated and regulated by this larger system. Our discussion of robot workcell control will resume in Chap. Eleven, after we have examined robot programming, an obvious prerequisite for workcell control.

REFERENCES

1. J. F. Engelberger, *Robotics in Practice*, AMACOM (American Management Association), New York, 1980, chap. 4.
2. M. P. Groover, *Automation, Production Systems, and Computer-Aided Manufacturing*, Prentice-Hall, Englewood Cliffs, NJ, 1980, chap. 11.
3. E. Kafrissen and M. Stephans, *Industrial Robots and Robotics*, Reston, Reston, VA, 1984, chap. 5.
4. Lord Corporation, *Lord Tactile Sensors*, Marketing/Technical Brochure, Cary, NC.
5. R. N. Stauffer, "Progress in Tactile Sensor Development," *Robotics Today*, June 1983, pp. 43–49.
6. L. L. Toepperwein, M. T. Blackman, et al., "ICAM Robotics Application Guide," *Technical Report AFWAL-TR-80-4042*, vol. II, Materials Laboratory, Air Force Wright Aeronautical Laboratories, Ohio, April 1980.

PROBLEMS

6-1 It is desired to design a safety monitoring system for a robot cell in which a robot loads an automatic production machine with parts arriving on a conveyor. The robot is large and it is considered dangerous for workers to wander into the work volume of the robot. Aside from the other safety precautions that might be taken to ensure the safety of the workers, the safety monitoring system will use one or more sensors to detect the presence of humans in the cell.

(*a*) Write the detailed "functional specifications" for the sensor system. That is, make up a list of the things the sensor system must do, and the ways in which it will have to operate.

(*b*) From the list of sensors in Table 6-2, select several alternative sensors that will satisfy the functional specifications, and compare its features against the specifications. For each sensor, explain how it will be configured in the safety monitoring system. Use sketches if necessary to illustrate the configuration.

(*c*) Select the best sensor alternative, and justify your selection.

6-2 Using the list of sensors in Table 6-2 in the text, describe several methods for determining the presence or absence of a metallic part in a fixture. Assume the dimensions of the part are: length = 127.0 mm (5 in.), width = 36.0 mm (1.4 in.), and thickness = 20.0 mm (0.8 in.). The fixture is a mechanical vise with two jaws for holding the part. For each alternative, make a sketch of how the sensor would be positioned relative to the part.

6-3 Using the list of sensors in Table 6-2 in the text, describe several methods for determining the presence or absence of a nonmetallic part in a fixture. Assume the dimensions of the part are: length = 127.0 mm (5 in.), width = 36.0 mm (1.4 in.), and thickness = 20.0 mm (0.8 in.). The fixture

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

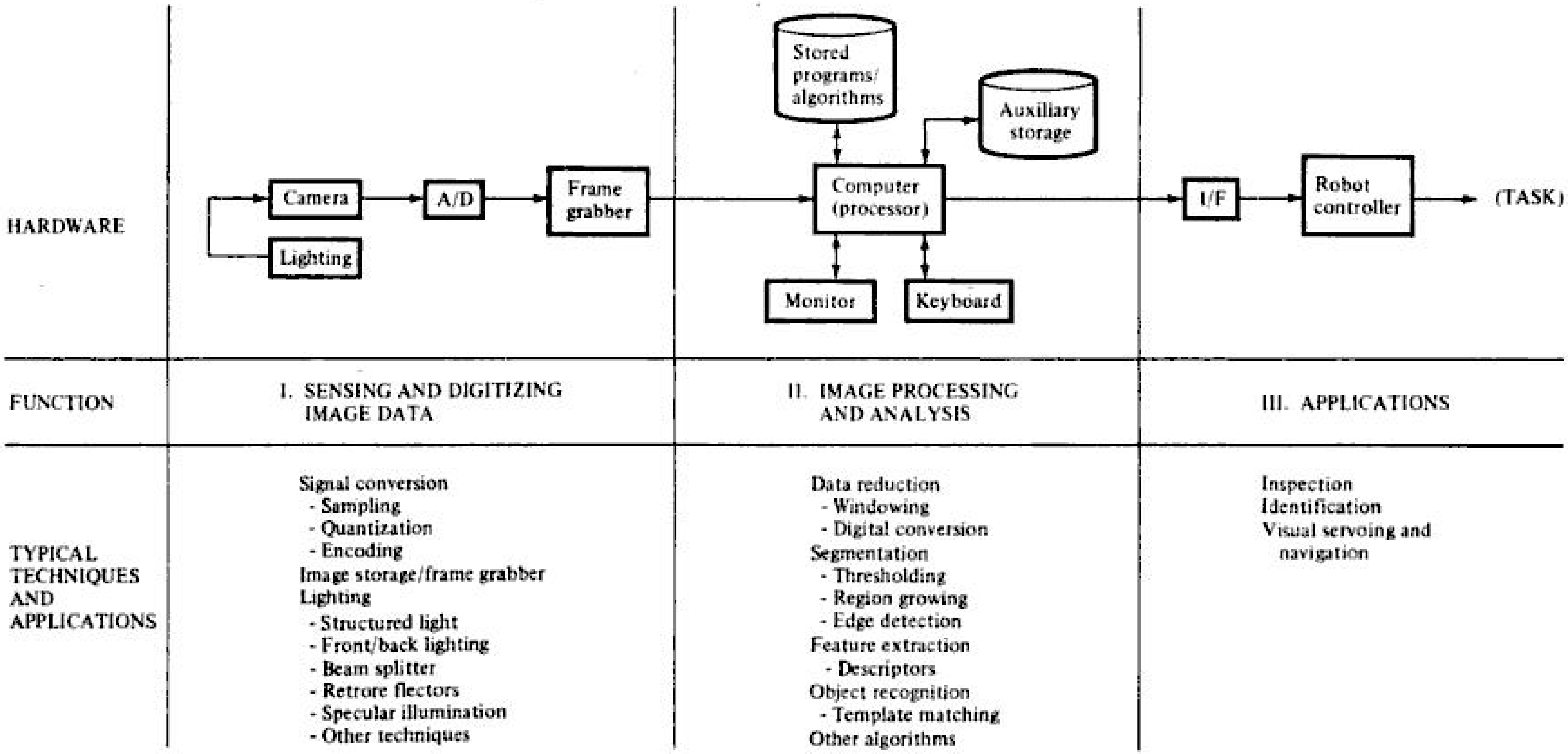


Figure 7-1 Functions of a machine vision system.

wells due to voltages applied to the central electrodes. Each isolated well represents one pixel and can be transferred to output storage registers by varying the voltages on the metal control electrodes. This is illustrated in Figs. 7-3(*a*) and (*b*).

Figure 7-4 indicates one type of CCD imager. Charges are accumulated for the time it takes to complete a single image after which they are transferred line by line into a storage register. For example, register A in Fig. 7-4

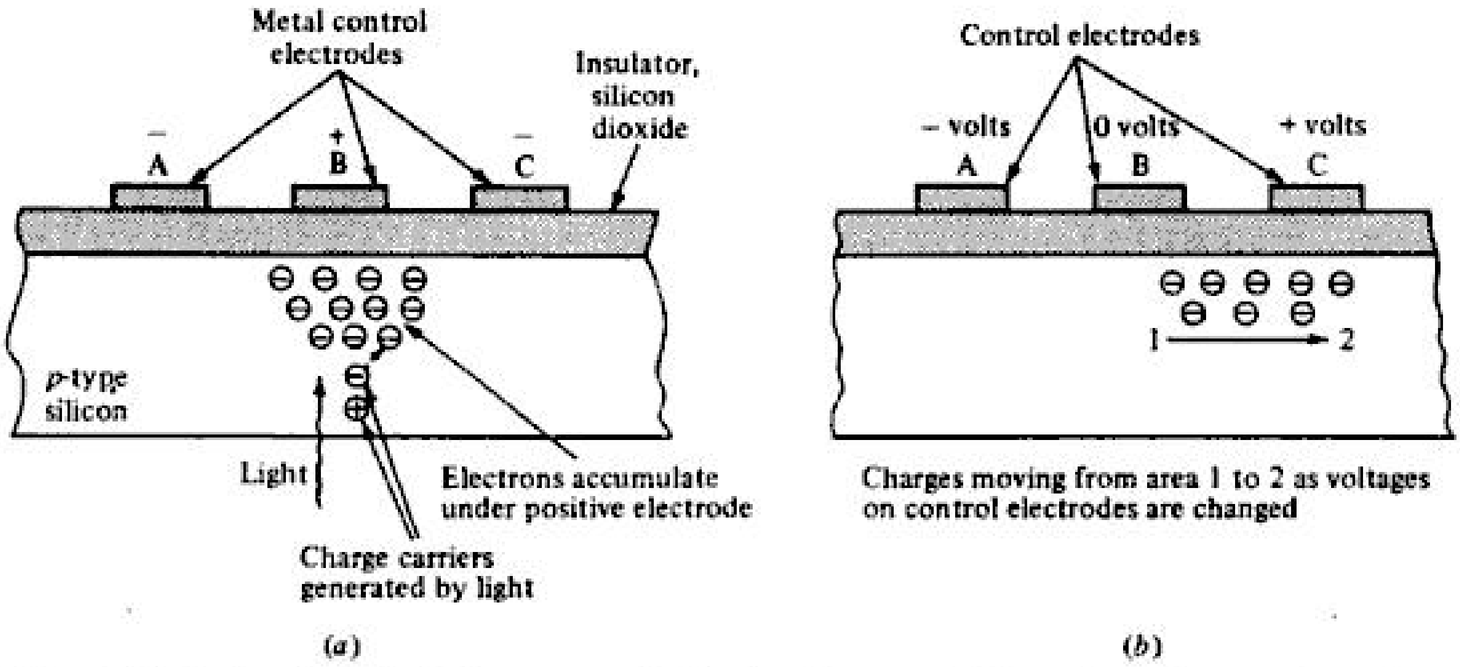


Figure 7-3 Basic principle of charge-coupled device. (a) accumulation of an electron charge in a pixel element (b) movement of accumulated charge through the silicon by changing the voltages on the electrodes A, B, and C. (Reprinted with permission of McGraw-Hill, Inc. [10].)

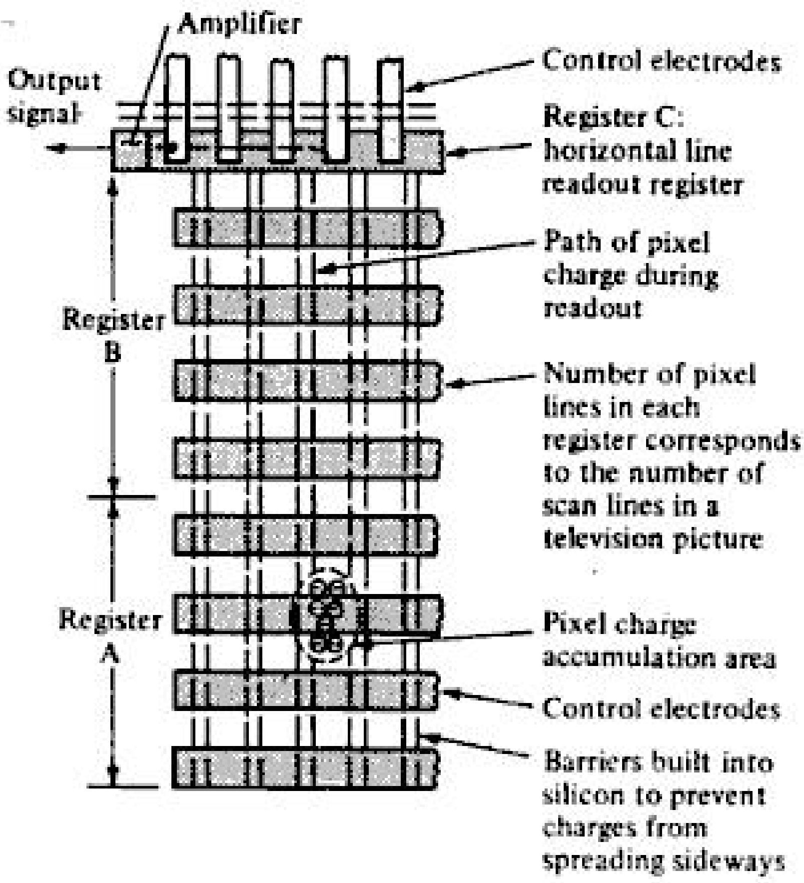


Figure 7-4 One type of charge-coupled-device imager. Register A accumulates the pixel charges produced by photoconductivity generated by the light image. The B register stores the lines of pixel charges and transfers each line in turn into register C. Register C reads out the charges laterally as shown into the amplifier. (Reprinted with permission of McGraw-Hill, Inc. [10].)

Accordingly, the scanning rate for each line is

$$= (33.33 \times 10^{-3}\ \text{s})/512\ \text{lines}$$
$$= 65.1 \times 10^{-6}\ \text{s/line}$$

The number of pixels that can be processed per line is therefore

$$= \frac{65.1 \times 10^{-6}\ \text{s/line}}{0.1 \times 10^{-6}\ \text{s/pixel}}$$
$$= 651\ \text{pixels/line}$$

In practice, an allowance would have to be made for the time required when the electron beam is shut off during its raster from one line to the next. This dead time would decrease the number of pixels used in the vidicon system. For example, the number of pixels that the system could handle might be reduced from 651 pixels per line to 512 pixels per line, which is similar to the number of pixels per line used in home television. For a 512×512 pixel image, there are a total of 262,144 pixels to consider during the faceplate scanning period. Each pixel must be processed and some type of image-processing function must be carried out during the sampling period. This results in substantial demand on the digital computer performing the function. The requirement on the capability of the vision system computer has been one of the limiting factors in the development of machine vision. Systems with a smaller number of pixels ($128 \times 128 = 16{,}384$ pixels or $256 \times 256 = 65{,}536$ pixels) impose lower computational requirements; however, the image resolution of these systems is much lower than for systems possessing a greater pixel density. For a given video signal representing one line, the number of samples taken determines the horizontal resolution of the imaging system. The total number of lines determines the vertical resolution.

Quantization Each sampled discrete-time voltage level is assigned to a finite number of defined amplitude levels. These amplitude levels correspond to the gray scale used in the system. The predefined amplitude levels are characteristic to a particular A/D converter and consist of a set of discrete values of voltage levels. The number of quantization levels is defined by

$$\text{number of quantization levels} = 2^n$$

where n is the number of bits of the A/D converter. A large number of bits enables a signal to be represented more precisely. For example, an 8-bit converter would allow us to quantize at $2^8 = 256$ different values whereas 4 bits would allow only $2^4 = 16$ different quantization levels.

Encoding The amplitude levels that are quantized must be changed into digital code. This process, termed encoding, involves representing an amplitude level by a binary digit sequence. The ability of the encoding process to distinguish between various amplitude levels is a function of the spacing of

Three important techniques that we will discuss are:

1. Thresholding
2. Region growing
3. Edge detection

In its simplest form, *thresholding* is a binary conversion technique in which each pixel is converted into a binary value, either black or white. This is accomplished by utilizing a frequency histogram of the image and establishing what intensity (gray level) is to be the border between black and white. This is illustrated for an image of an object as shown in Fig. 7-6. Figure 7-6(*a*) shows a regular image with each pixel having a specific gray tone out of 256 possible gray levels. The histogram of Fig. 7-6(*b*) plots the frequency (number of pixels) versus the gray level for the image. For histograms that are bimodal in shape, each peak of the histogram represents either the object itself or the background upon which the object rests. Since we are trying to differentiate between the object and background, the procedure is to establish a threshold

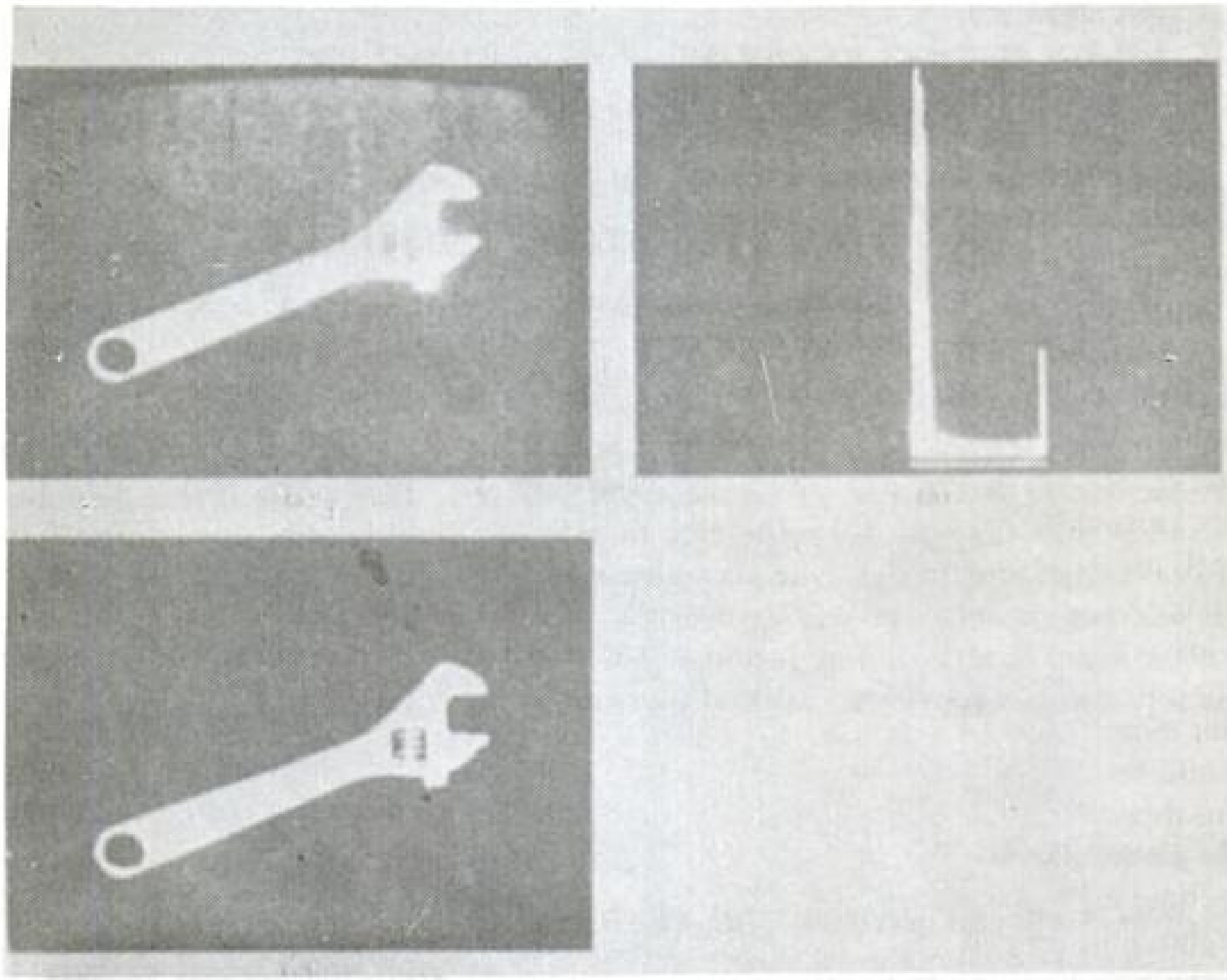


Figure 7-6 Obtaining a binary image by thresholding. (a) image of object with all gray-levels present, (b) histogram of image (c) binary image of object after thresholding. (Photos courtesy of Robotics Laboratory, Lehigh University.)

center of gravity.) The centroid is indicated in Fig. 7-9. Of the two thinness measures defined, the calculations would result in

$$\text{Compactness} = \frac{(\text{perimeter})^2}{\text{area}} = \frac{26^2}{24} = 28.17$$

$$\text{Thinness} = \frac{\text{diameter}}{\text{area}} = \frac{9}{24} = \frac{3}{8}$$

Object Recognition

The next step in image data processing is to identify the object the image represents. This identification problem is accomplished using the extracted feature information described in the previous subsection. The recognition algorithm must be powerful enough to uniquely identify the object. Object recognition techniques used in industry today may be classified into two major categories:

1. Template-matching techniques
2. Structural techniques

Template-matching techniques are a subset of the more general statistical pattern recognition techniques that serve to classify objects in an image into predetermined categories. The basic problem in template matching is to match the object with a stored pattern feature set defined as a model template. The model template is obtained during the training procedure in which the vision system is programmed for known prototype objects. These techniques are applicable if there is not a requirement for a large number of model templates. The procedure is based on the use of a sufficient number of features to minimize the frequency of errors in the classification process. The features of the object in the image (e.g., its area, diameter, aspect ratio, etc.) are compared to the corresponding stored values. These values constitute the stored template. When a match is found, allowing for certain statistical variations in the comparison process, then the object has been properly classified.

Structural techniques of pattern recognition consider relationships between features or edges of an object. For example, if the image of an object can be subdivided into four straight lines (the lines are called primitives) connected at their end points, and the connected lines are at right angles, then the object is a rectangle. This kind of technique, known as syntactic pattern recognition, is the most widely used structural technique. Structural techniques differ from decision-theoretic techniques in that the latter deals with a pattern on a quantitative basis and ignores for the most part interrelationships among object primitives. A detailed discussion of pattern recognition techniques is the subject of complete books and is beyond the scope of this text.

It can be computationally time consuming for complete pattern recognition. Accordingly, it is often more appropriate to search for simpler regions

commands—higher, more general commands than we have in today's commercially available languages. The robots will have to interpret these high-level commands and act upon them. To do this, robots of the future must possess more intelligence than today's machines. In Chap. Ten, we survey the field of artificial intelligence to see what promise this technology holds for robotics.

To overcome this disadvantage, many robots can be controlled during the teach mode to move in x-y-z coordinate motions. This method, called the world coordinate system, allows the wrist location to be defined using the conventional cartesian coordinate system with origin at some location in the body of the robot. In the case of the cartesian coordinate robot, this method is virtually equivalent to the joint mode of programming. For polar, cylindrical, and jointed-arm robots, the controller must solve a set of mathematical equations to convert the rotational joint motions of the robot into the cartesian coordinate system. These conversions are carried out in such a way that the programmer does not have to be concerned with the substantial computations that are being performed by the controller. To the programmer, the wrist (or end effector) is being moved in motions that are parallel to the x, y, and z axes. The two or three additional joints which constitute the wrist assembly are almost always rotational, and while programming is being done in the x-y-z system to move the arm and body joints, the wrist is usually being maintained by the controller in a constant orientation. The x-y-z method of defining points in space is illustrated in Fig. 8-4 for a jointed-arm robot.

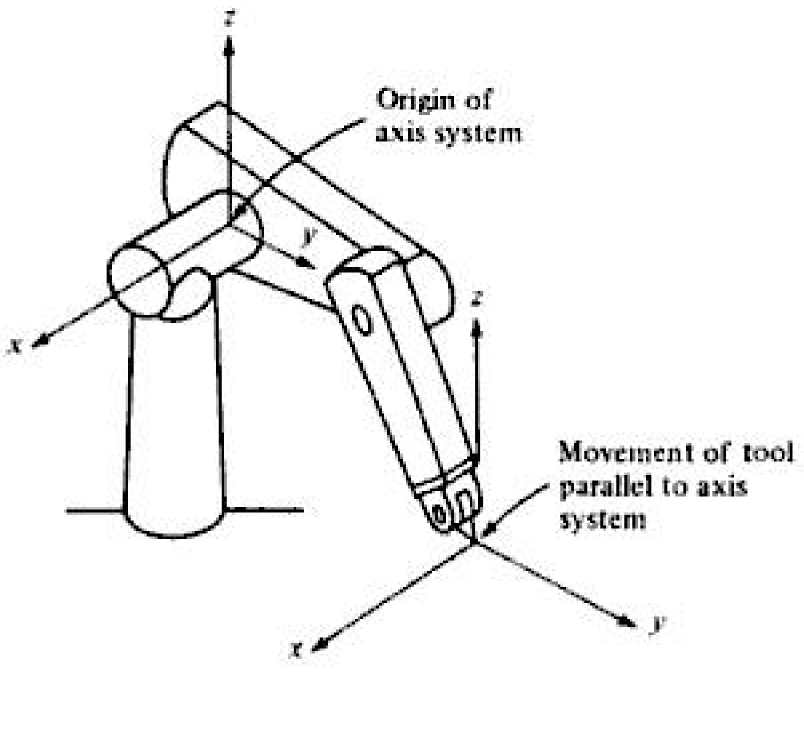


Figure 8-4 World mode or X-Y-Z method of defining points in space.

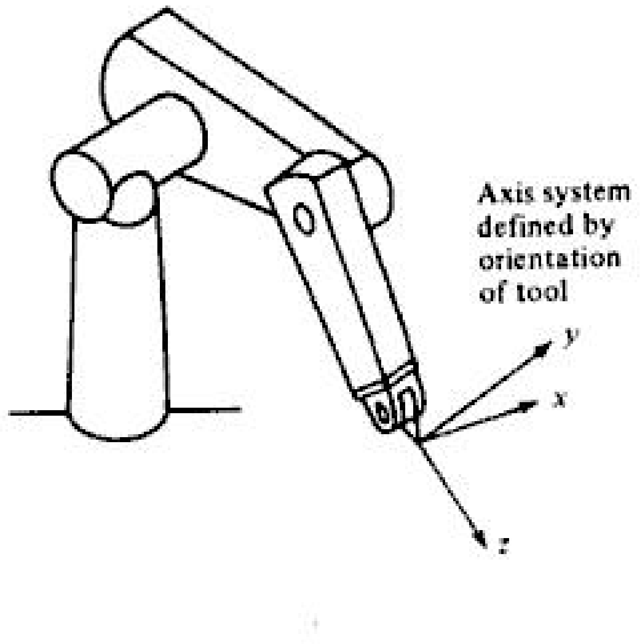


Figure 8-5 Tool mode of defining points in space.

was accomplished using the WAVE language. The research demonstrated the feasibility of robot hand–eye coordination. Development of a subsequent language began in 1974 at Stanford. The language was called AL and it could be used to control multiple arms in tasks requiring arm coordination.

Many of the concepts of WAVE and AL went into the development of the first commercially available robot textual language, VAL (for Victor's Assembly Language, after Victor Scheinman). VAL was introduced in 1979 by Unimation, Inc., for its PUMA robot series. This language was upgraded to VAL II and released in 1984.

Work in robot language development was also taking place at the T. J. Watson Research Labs of the IBM Corporation, starting around 1976. Two of the IBM languages were AUTOPASS and AML (A Manufacturing Language), the second of which has been commercially available since 1982 with IBM's robotic products. Both of these languages are directed at assembly and related tasks.

Some of the other textual languages for robots that should be mentioned include RAIL, introduced in 1981 by Automatix for robotic assembly and arc welding, as well as machine vision; MCL (Manufacturing Control Language), developed under U.S. Air Force sponsorship by McDonnell-Douglas as an enhancement of the APT (Automatically Programmed Tooling) numerical control part programming language; and HELP, available from the General Electric Company under license from the Italian firm DEA.

9-2 GENERATIONS OF ROBOT PROGRAMMING LANGUAGES

The textual robot languages possess a variety of structures and capabilities. These languages are still evolving. In this section we identify two generations of textual languages and speculate about what a future generation might be like.

First Generation Languages

The "first generation" languages use a combination of command statements and teach pendant procedures for developing robot programs. They were developed largely to implement motion control with a textual programming language, and are therefore sometimes referred to as "motion level" languages.[11] Typical features include the ability to define manipulator motions (using the statements to define the sequence of the motions and the teach pendant to define the point locations), straight line interpolation, branching, and elementary sensor commands involving binary (on–off) signals. In other words, the first generation languages possess capabilities similar to the advanced teach pendant methods used to accomplish the robot programming instructions described in Chap. Eight. They can be used to define the motion

In world modeling, the robot possesses a geometric model of its workspace by which it knows the desired locations without being taught each point. The obvious disadvantage of the first and second generation languages is the interruption of regular production work while the teach pendant is used to define the locations named in the program. This disadvantage would be avoided with off-line programming. In Chap. Eleven, we return to the topic of off-line programming when we discuss work cell simulation by computer graphics. Future applications of off-line programming are likely to require some form of simulation to verify the correctness of the program in advance of its downloading to the robot.

There are problems to be solved before future generation languages using off-line programming become a reality. Certainly one of these problems deals with the accuracy of the world model contained in the robot's memory. The model is not the same as the real world. There will always be some positional error between the actual physical objects in the work environment and the computer model used by the robot. For intricate tasks (e.g., assembly operations), the positional errors may mean the difference between success and failure in performing the task.

A second problem with off-line programming is concerned with the technology of artificial intelligence, hierarchical control, and similar approaches that would permit the robot to accept a high-level objective-oriented command (e.g., ASSEMBLE TYPEWRITER) and translate that instruction into a series of actions required to accomplish it. World modeling for robot programming is an area of significant interest in a number of research and development laboratories, both in academia and industry.

9-3 ROBOT LANGUAGE STRUCTURE

The second generation languages represent the current state of the art in textual languages. In this and the following sections of this chapter we describe the features that characterize the second generation languages in use today for robotics. Some of these features, of course, are applicable to first generation languages as well, but our discussion will include the more advanced capabilities that usually go beyond the first generation.

The language must be designed to operate with a robot system as illustrated in Fig. 9-1. It must be able to support the programming of the robot, control of the robot manipulator, and interfacing with peripherals in the work cell (e.g., sensors, and equipment). It should also support data communications with other computer systems in the factory.

Operating Systems

In using the textual languages, the programmer has available a CRT monitor, an alphanumeric keyboard, and a teach pendant. There should also be some means of storing the programs, either on magnetic tape or disk. Using the

tion languages were generally limited to the specification of integer variables that might be used, for example, for counting routines in the program logic.

Aggregates and Location Variables

An aggregate is an ordered set of constants or variables. AML, for example, permits the specification of an aggregate by enclosing it with the bracketing symbols ⟨ and ⟩, and by separating the elements in the aggregate by commas. Two examples of aggregates in the AML language are

⟨50.526, 236.003, 14.581, 25.090, 125.750⟩
⟨'we', 'they'⟩

The first example is an aggregate consisting of five real numbers. The second example consists of two strings. It is not necessary that the aggregate contain elements that are all of the same type. Any combination of integers, real numbers, and strings can be contained in the same aggregate.

An aggregate can be used to specify the joint coordinate values of a robot's joints. The first example above could be used to define a five-axis robot's joint coordinate values for a point in space. The method of specification might be done as follows:

DEFINE A1 = POINT ⟨50.526, 236.003, 14.581, 25.090, 125.750⟩

The usual interpretation of the aggregate in this statement is that the first three values (50.526, 236.003, 14.581) define the position of the wrist, or the tool attached to the wrist, in world space (x-y-z coordinates). The remaining values (25.090, 125.750) refer to the rotations of the wrist joints in degrees relative to some neutral reference position.

9-5 MOTION COMMANDS

Among the most important functions in a robot language are those which control the movement of the manipulator arm. This section describes how the textual languages accomplish these functions.

MOVE and Related Statements

One of the most important functions of the language, and the principal feature that distinguishes robot languages from computer programming languages, is manipulator motion control. In Chap. Eight, we defined the basic motion command, the MOVE statement

MOVE A1

This causes the end of the arm (end effector) to move from its present position to the point (previously defined), named A1. Recall from the previous chapter that a point is defined in terms of the robot's joint positions, and so A1 defines

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

9B-2 MONITOR COMMANDS

The VAL II monitor mode is used for functions such as defining point locations in the work volume, editing programs, executing programs, calibrating the robot arm, and similar purposes. We will sample some of the commands in this section.

Defining Locations

There are several commands in VAL II for defining locations and determining the current location of the robot. In the following commands all distances and coordinate values are in millimeters. One of the methods of defining point locations is the HERE command. The operator uses the teach pendant to move the robot manipulator to the position to be defined and uses the command as follows

```
HERE P1
```

This command, given in the monitor mode, defines the variable P1 to be the current robot arm location. A related command is

```
WHERE
```

This command queries the system to display the current location of the robot in cartesian world coordinates and wrist joint variables. It also displays the current gripper opening if the robot is equipped with a position-servoed hand.

The TEACH command is used to record a series of location values under the control of the record button on the teach pendant. Each time the record button is pressed, a location variable is defined and given the value corresponding to the location of the robot at the instant the record button is pressed. Each successive location variable is automatically assigned a new name. The assigned name is derived from the name specified in the command. For example, if the command

```
TEACH P1
```

is typed into the monitor, the first recorded location variable is P1, the next is P2, the third is P3, and so forth. It is possible to teach a complete motion path by successively positioning the robot using powered leadthrough with the teach pendant and pressing the record button.

A third command for defining positions is the POINT command. For example, the command

```
POINT PA = P1
```

sets the value of PA (a location variable) equal to the value of P1.

Program Editing and Control

Commands for entering and exiting the program editing mode are the follow-

branches to the statement 120 in the program. If the statement number is not specified, for example,

GRASP 12.7

then an error message is displayed at the operator's console. The GRASP statement provides a convenient method of grasping an object and testing to make sure that contact has been made.

The control of the gripper and simultaneous movement of the robot arm can be accomplished with a single statement rather than two separate commands as described above. The required statement is

MOVET P1, 75

This command generates a joint-interpolated motion from the previous position to the point P1, and during the motion, the hand opening is changed to 75 mm. The corresponding straight line motion command is

MOVEST P1, 75

These statements assume a servocontrolled gripper capable of responding to the 75-mm opening specification. For a pneumatically operated hand, the command is interpreted to mean "open" if the value specified is greater than zero, and "close" if the value is otherwise.

9B-5 CONFIGURATION CONTROL

For a jointed-arm robot with six joints, most points in the workspace can be reached by assuming one of eight possible spatial configurations. Normally, the robot would remain in the same configuration throughout program execution. The need for using one or more of the commands in this section is to permit the robot to achieve a position in one configuration which it would be unable to achieve in the alternative configuration. The statements

RIGHTY or LEFTY

provide for a change in the jointed-arm robot configuration so that the first three joints (base rotation, shoulder, and elbow) resemble a human's right or left arm, respectively. The statements

ABOVE or BELOW

request a change in robot configuration so that the elbow of the robot is pointed up (ABOVE) or down (BELOW). Finally, the statements

FLIP or NOFLIP

provide for a change in the range of operation of joint 5 (on a six-jointed robot) to positive (NOFLIP) or negative (FLIP) angles. The last two statements would be invoked to keep joints 4 and 6 within stop limits.

specifies that the end effector should be rotated 180° in the clockwise direction from its current position.

Open and Close Gripper

These statements command the opening and closing of a gripper-type end effector. The RAIL statements are

OPEN and CLOSE

Welding

This is a RAIL command that permits a welding operation to be carried out at a point along a path. The motion is executed by the special WELD command, explained in Sec. 9C-6.

9C-5 LEARN STATEMENT

The LEARN statement is used to define location data in conjunction with the robot teach pendant. The statement can be used to program points in world space or relative to other predefined locations. It can also be used to teach paths and frames. In the statement

LEARN P1, P2, P3, P4

the four points are learned by moving the arm to the desired locations using the motion buttons on the teach pendant, and pressing the record button to store each point location in the proper sequence.

Paths are learned by depressing the path button on the teach pendant to indicate the start of a path, and then moving the robot through the desired path by means of the teach pendant, depressing the record button at each successive point in the path. Depressing the path button a second time signals the end of the learn routine.

Frames are learned by means of the statement

LEARN FRAME COORD1

The LEARN FRAME statement prompts the user to teach three locations. These locations correspond to the origin of the frame, a point along the x axis, and a point in the xy plane of the frame coordinate system.

9C-6 WELDING

RAIL provides a number of features for controlling a welding robot. These features include the capability to do the following: move the robot along a path while controlling the welding process parameters; stepping the robot

representing the data collected from the human experts, and improving the "knowledge base" with experience.

4. *Learning*. One of the attributes of intelligence is the ability to learn from experience. If machines were capable of learning then the task of endowing them with knowledge, as in the case of expert systems, would be greatly simplified. Some systems have been developed which have shown the ability to learn from experience, but to this date limited progress has been made.

5. *Vision*. Most of the basic concepts employed in commercial vision systems are the result of AI research. One of the more interesting goals of AI vision research is to permit the systems to perform scene analysis. That is, present the vision system with a scene and allow the system to identify objects within the scene.

Some of the other areas of AI research include: automatic programming, hardware development, and deductive reasoning.

10-3 AI TECHNIQUES

AI is concerned with the use of data or knowledge. Therefore techniques must be developed for two basic tasks: data representation and data manipulation. In this section we will look at some of the approaches for representing data and using that data in some specific manner. We will not discuss actual programs for doing this, only the general techniques which might be employed by AI programs.

Knowledge Representation

When we discuss knowledge representation we are not concerned with the physical operation of the computer, that is, we are not discussing the storage of words as a series of 1s and 0s. Rather we are discussing the relationships of facts with respect to each other, for example, the statement, "Some birds have wings."

The material in this section is summarized to a large extent from Ref. 1. For a more detailed account the reader is directed to Ref. 1.

Before discussing the various representations of knowledge we must first describe the various types of knowledge which may require representation.

1. Objects. More specifically facts about objects, such as "robotics students drink heavily" or "birds have wings."
2. Events. Not only the event itself, such as "The robotics student broke his arm," but perhaps the time or cause-effect relation of the event. "The robotics student broke his arm yesterday and the nasty instructor made him pay for it."
3. Performance. If the AI system is one which is designed to control a robot

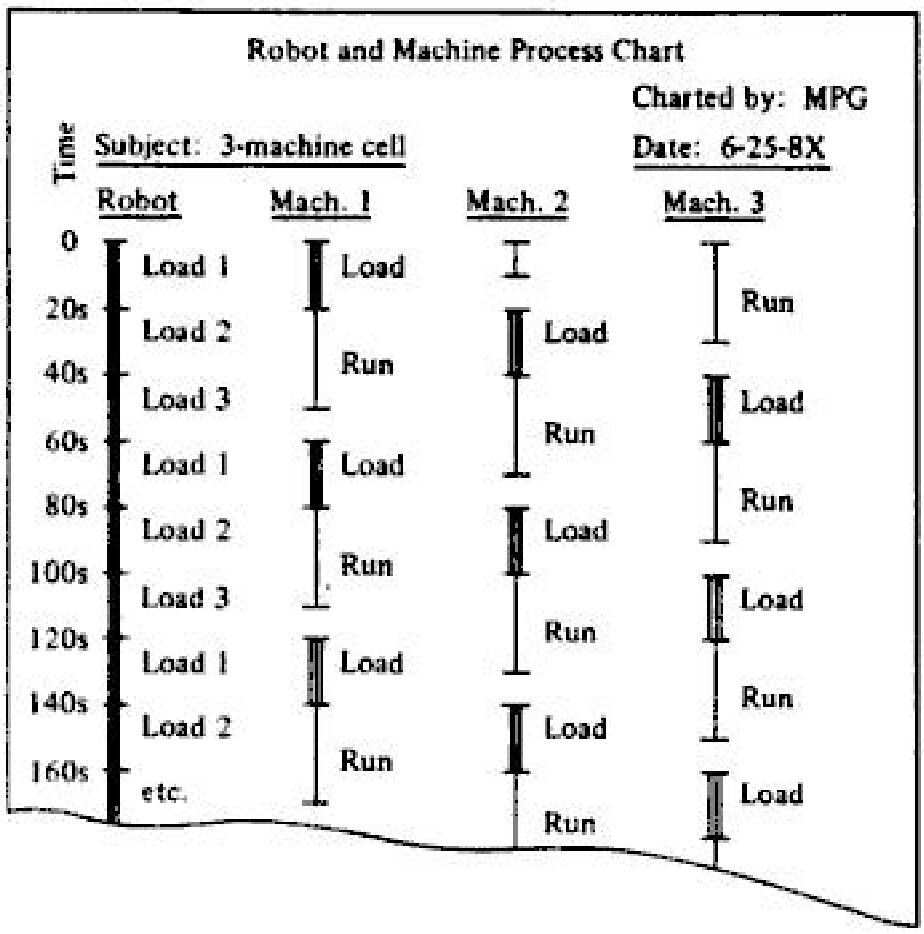


Figure 11-6 Robot and machine process chart for Example 11-1.

In this example the cycles of the three machines are the same. In this case, the question of whether or not machine interference will occur is determined by the relative values of machine cycle time and robot cycle time. The machine cycle time is the sum of service time and run time. The robot cycle time is equal to the number of machines multiplied by the service time. If the robot cycle time is greater than the machine cycle time, there will be resulting machine interference. If the machine cycle time is greater than the robot cycle time, there will be no machine interference, but the robot will be idle for part of the cycle.

In the case where the service and run times of the machines are different, the above relationships become complicated by the problem of determining the best sequence of servicing times for the machines into the robot cycle time. We will explore this problem in the exercises at the end of the chapter.

11-3 OTHER CONSIDERATIONS IN WORKCELL DESIGN

There are several other issues that must be considered in the design of the workcell. Among these considerations are the following:

1. *Changes to other equipment in the cell.* To implement the workcell and interface the robot to the other equipment in the cell, alterations will often have to be made to the equipment. Special fixtures and control devices must be devised to permit the cell to operate as a single, integrated mechanism. Examples of these fixtures and controls include work-holding nests and conveyor stops to position and orient the parts for the robot, changes in the

This represents a limited input/output capacity for a workcell with any degree of complexity. Also, today's robot controllers are generally limited to sequence control and, as the above list indicates, often do not possess the capability to incorporate any significant safety monitoring or operator interfacing into the workcell control system. With the growing use of computer controls and the need by competing robot manufacturers to increase the control capabilities of their products, it is expected that future robot controllers will be equipped with enhanced input/output capacity and the capability to control intelligent robots.

Electromechanical Relays

An electromechanical relay is a control device used to actuate electrical circuits in response to changes in incoming signals. They are commonly used in industrial applications to provide sequence control of electrically operated equipment although they are gradually being displaced by more modern devices such as programmable controllers.

Relays can be used to augment the capabilities of a robot controller in the design of a workcell control system. Their use would typically be reserved for simple robot cells, such as pick-and-place applications, and where the robot has very limited input/output capacity. With relays, it would be relatively easy to include a simple safety monitoring scheme in the workcell. Such a scheme might consist of a fence surrounding the work place with a safety gate to gain access to the cell. Using the appropriate sensors (e.g., a limit switch to indicate closure of the safety gate) the relays could be set up to stop the robot, perhaps by interrupting its power source, as soon as a hazardous condition was sensed.

The limitations of relay control include the difficulty in interfacing with plant computer systems, their hard-wired configuration which makes it difficult to change over to a new workstation control task, and the fact that they are susceptible to mechanical wear and are less reliable than computer-type controls. The functions of a relay panel can be accomplished by a programmable controller, which avoids the above problems.

Programmable Controllers

Programmable controllers were introduced in the late 1960s as a replacement for systems of electromechanical relays. Up until that time, relay panels constituted the standard technique for accomplishing sequence control in industrial operations. The programmable controller was smaller in size, more reliable, more flexible, and its use could be readily learned by shop personnel who were familiar with the logic diagrams used for relay control panels.

A programmable controller (PC) can be defined as a digitally operating device with programmable memory that is capable of generating output signals according to logic operations and other functions performed on input signals. The program for a PC determines the sequence of operations and the generation of input and output signals. A PC is programmed by specifying the same kinds of logic diagrams, called ladder diagrams, used for years to set up relay control panels. Other programming methods are also possible on many

with the conveyor feed rate. If the robot motion takes longer than 15 s, then the conveyor might have to be slowed down, or an alternative robot cycle developed. For example, the distances moved could be reduced to shorten the move element times. If the robot motion cycle takes less than 15 s, then the possibility of speeding up the parts delivery to the conveyor could be investigated.

Table 11-4 reduces the data contained in Table 11-3 to the RTM symbol notation, and presents the calculated element times as determined from the hypothetical robot values in Table 11-2. The total cycle time is 15 s. It turns out that the conveyor is the limiting factor in the cycle, requiring 2.8 s more time than the robot motion cycle. It might be possible to reduce the feed rate on the conveyor down to one part every 12.2 s. This would provide a perfect match between the feed rate and the robot cycle.

11-9 GRAPHICAL SIMULATION OF ROBOTIC WORKCELLS

RTM can be considered a method of simulating, in terms of time, the activities in the robot workcell. Another method of simulation involves graphical modeling on a CAD/CAM system. Simulation based on computer graphics can be used not only to analyze cycle times, but to design the cell itself. It turns out that a substantial amount of time is spent in designing and laying out the cell, designing or selecting the equipment, and similar activities. One industry estimate[10] is that 60 to 80 percent of the total cell implementation time is spent on these design-related problems and cell fabrication. (The remaining 20 to 40 percent of the time is spent in programming and refining the cell.) With so much effort expended on the design of the robot cell, it is reasonable to utilize labor saving tools to make the process as efficient as possible.

This section will discuss the use of computer graphics to simulate the design and operation of the robot and the workcell. We will provide an example of university research in this area and an example of a commercial package for designing and simulating the robot cell.

Research in Graphics Modeling for Robotics

Research in Lehigh University's Computer-Aided Design Laboratory in conjunction with our Institute for Robotics has led to the development of a graphics simulator of the PUMA 600 robot and the VAL language used to program the PUMA.[1] The simulator makes use of a FORTRAN callable graphics language to display the kinematic behavior of the PUMA in response to VAL motion statements. Algorithms for computing the positioning of the manipulator along segmented paths are used to simulate joint coordinate and straight line motions. Sequences of VAL commands can be entered interactively and their resulting motions shown on the graphics monitor.

CHAPTER
TWELVE
ECONOMIC ANALYSIS FOR ROBOTICS

In addition to the technological considerations involved in applications engineering for a robotics project, there is also the economic issue. Will the robot justify itself economically? The economic analysis for any proposed engineering project is of considerable importance in most companies because management usually decides whether to install the project on the basis of this analysis. In the present chapter, we consider the economic analysis of a robot project. We discuss the various costs and potential benefits associated with the robot installation, and we describe several methods for analyzing these factors to determine the economic merits of the project.

12-1 ECONOMIC ANALYSIS: BASIC DATA REQUIRED

To perform the economic analysis of a proposed robot project, certain basic information is needed about the project. This information includes the type of project being considered, the cost of the robot installation, the production cycle time, and the savings and benefits resulting from the project.

Type of Robot Installation

There are two basic categories of robot installations that are commonly encountered. The first involves a new application. This is where there is no existing facility. Instead, there is a need for a new facility, and a robot installation represents one of the possible approaches that might be used to satisfy that need. In this case, the various alternatives are compared and the best alternative is selected, assuming it meets the company's investment criteria. The second situation is the robot installation to replace a current method of operation. The present method typically involves a production operation that is performed manually, and the robot would be used somehow

cash flows is equal to the initial investment cost. In the special case when the net annual cash flows are equal, Eq. (12-2) can be recast as

$$0 = -(\text{IC}) + n(\text{NACF})$$

This is equivalent to Eq. (12-1).

The reader should note that we have adopted the logical convention that costs are treated as negative values and revenues or savings (as well as profits) are treated as positive values in these equations. The NACF is assumed to be a positive cash flow since revenues derived from the robot project would be greater than the operating costs (we hope). We have also assumed that all cash flows occur either at the beginning of the year or at the end of the year. Any investments are assumed to be transactions that occur at the beginning of the year, while the net annual cash flows are assumed to be end-of-year transactions.

Most companies today require paybacks of no more than two or three years. An investment whose cash flow pays back the investment in less than one year is considered excellent. Let us illustrate the payback method by means of the following example.

Example 12-1 Suppose that the total investment cost is estimated to be \$100,000 for a particular robot project. The total operating costs (labor, maintenance, and other annual expenses) are expected to be \$20,000 per year, and the anticipated revenues from the robot installation are \$65,000 annually. It is expected that the robot project will have a service life of 5 years. Determine the payback period that is expected of the investment.

The net annual cash flow for the robot project is \$65,000 − \$20,000 = \$45,000. Using Eq. (12-1),

$$n = \frac{100{,}000}{45{,}000} = 2.22 \text{ years}$$

One of the disadvantages of the payback period method is that it ignores the time value of money. It does not consider the objective of the company to derive a certain minimum rate of return from its investments. The other two methods to be discussed do include this consideration.

Equivalent Uniform Annual Cost Method

The equivalent uniform annual cost (EUAC) method converts all of the present and future investments and cash flows into their equivalent uniform cash flows over the anticipated life of the project. It does this by making use of the various interest factors associated with engineering economy calculations. We present a tabulation of these interest factors in an appendix to this chapter, and we save a considerable amount of explanation by assuming that the reader is familiar with their use.

To begin with, the company must select a minimum attractive rate-of-return (MARR) which is used as a criterion to decide whether a potential investment

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

previous chapters of the book, and we itemize them below as a reference checklist.

1. *Part positioning and orientation.* In most parts-handling applications the parts must be presented to the robot in a known position and orientation. Robots used in these applications do not generally possess highly sophisticated sensors (e.g., machine vision) that would enable them to seek out a part and identify its orientation before picking it up.
2. *Gripper design.* Special end effectors must be designed for the robot to grasp and hold the workpart during the handling operation. Design considerations for these grippers were discussed in Chap. Five.
3. *Minimum distances moved.* The material-handling application should be planned so as to minimize the distances that the parts must be moved. This can be accomplished by proper design of the workcell layout (e.g., keeping the equipment in the cell close together), by proper gripper design (e.g., using a double gripper in a machine loading/unloading operation), and by careful study of the robot motion cycle.
4. *Robot work volume.* The cell layout must be designed with proper consideration given to the robot's capability to reach the required extreme locations in the cell and still allow room to maneuver the gripper.
5. *Robot weight capacity.* There is an obvious limitation on the material-handling operation that the load capacity of the robot must not be exceeded. A robot with sufficient weight-carrying capacity must be specified for the application.
6. *Accuracy and repeatability.* Some applications require the materials to be handled with very high precision. Other applications are less demanding in this respect. The robot must be specified accordingly.
7. *Robot configuration, degrees of freedom, and control.* Many parts transfer operations are simple enough that they can be accomplished by a robot with two to four joints of motion. Machine-loading applications often require more degrees of freedom. Robot control requirements are unsophisticated for most material-handling operations. Palletizing operations, and picking parts from a moving conveyor are examples where the control requirements are more demanding.
8. *Machine utilization problems.* It is important for the application to effectively utilize all pieces of equipment in the cell. In a machine loading/unloading operation, it is common for the robot to be idle while the machine is working, and the machine to be idle while the robot is working. In cases where a long machine cycle is involved, the robot is idle a high proportion of the time. To increase the utilization of the robot, consideration should be given to the possibility for the robot to service more than a single machine. One of the problems arising in the multimachine cell is machine interference, discussed in Chap. Eleven.

We now proceed to deal with the specific cases of material transfer and machine loading/unloading applications in the following two sections.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

robot loading procedure. In the future, robots may be equipped with sufficient intelligence to figure out how to load the different cartons onto the pallet. At the time of this writing, it is a systems problem of significant proportions.

13-3 MACHINE LOADING AND UNLOADING

These applications are material-handling operations in which the robot is used to service a production machine by transferring parts to and/or from the machine. There are three cases that fit into this application category:

Machine load/unload. The robot loads a raw workpart into the process and unloads a finished part. A machining operation is an example of this case.

Machine loading. The robot must load the raw workpart or materials into the machine but the part is ejected from the machine by some other means. In a pressworking operation, the robot may be programmed to load sheet metal blanks into the press, but the finished parts are allowed to drop out of the press by gravity.

Machine unloading. The machine produces finished parts from raw materials that are loaded directly into the machine without robot assistance. The robot unloads the part from the machine. Examples in this category include die casting and plastic modeling applications.

The application is best typified by a robot-centered workcell which consists of the production machine, the robot, and some form of parts delivery system. To increase the productivity of the cell and the utilization of the robot, the cell may include more than a single production machine. This is desirable when the automatic machine cycle is relatively long, hence causing the robot to be idle a high proportion of the time. Some cells are designed so that each machine performs the same identical operation. Other cells are designed as flexible automated systems in which different parts follow a different sequence of operations at different machines in the cell. In either case, the robot is used to perform the parts handling function for the machines in the cell.

Robots have been successfully applied to accomplish the loading and/or unloading function in the following production operations:

Die casting
Plastic molding
Forging and related operations
Machining operations
Stamping press operations

We will discuss these applications in the following subsections. For each application, a brief description of the manufacturing process will be given. More detailed descriptions of the processes are to be found in other references.[1,2,7,8]

Machining Operations

Machining is a metalworking process in which the shape of the part is changed by removing excess material with a cutting tool. It is considered to be a secondary process in which the final form and dimensions are given to the part after a process such as casting or forging has provided the basic shape of the part. There are a number of different categories of machining operations. The principal types include turning, drilling, milling, shaping, planing, and grinding. Commercially, machining is an important metalworking process and is widely used in many different products, ranging from those that are made in low quantities to those produced in very high numbers. In mid-volume and high-volume production, the operation is very repetitive with the same machining sequence being repeated on part after part.

The machine tools that perform machining operations have achieved a relatively high level of automation after many years of development. In particular, the use of computer control (e.g., computer numerical control and direct numerical control) permits this type of equipment to be interfaced with relative ease to similarly controlled equipment such as robots.

Robots have been successfully utilized to perform the loading and unloading functions in machining operations. The robot is typically used to load a raw workpart (a casting, forging, or other basic form) into the machine tool and to unload the finished part at the completion of the machining cycle. Figure 13-5 illustrates a machine tool loading and unloading operation in which the finished parts are palletized (lower left corner of the figure) after the machining cycle.

The following robot features generally contribute to the success of the machine tool load/unload application[2]:

Dual gripper. The use of a dual gripper permits the robot to handle the raw workpart and the finished part at the same time. This permits the production cycle time to be reduced.

Up to six joint motions. A large number of degrees of freedom of the arm and wrist are required to manipulate and position the part in the machine tool.

Good repeatability. A relatively high level of precision is required to properly position the part into the chuck or other workholding fixture in the machine tool.

Palletizing and depalletizing capability. In midvolume production, the raw parts are sometimes most conveniently presented to the workcell and delivered away from the workcell on pallets. The robot's controller and programming capabilities must be sufficient to accommodate this requirement.

Programming features. There are several desirable programming features that facilitate the use of robots in machining applications. In machine cells used for batch production of different parts, there is the need to perform some sort of changeover of the setup between batches. Part of this changeover procedure involves replacing the robot program for the previous batch

the cell controller to return the empty pallet fixture to the first station to be reloaded.

Workcell controller. A programmable controller was used as the workcell controller for this application. The cell control functions included:

Interlocking to detect part presence at each station
Controlling the movement of the pallet fixtures through the system
Confirming the correctness of the assemblies
Controlling part escapement at station 2
Synchronizing robot operations
Selecting robot functions
Providing system status display
Providing an operator input station
Providing for manual operation of the system if that becomes necessary
Controlling tests and inspections
Informing the operator when to refill the magazines and feeders

Parallel Assembly Systems

The concept of a parallel arrangement in a robotic workcell is pictured in Fig. 15-20. In essence the work can take either of two (or more) routes to have the same operations performed. There are two conditions under which parallel workstations would normally be considered. The first is the situation where production cycle times at a particular workstation or group of workstations are too long to keep up with the other sections of the line. The other stations are forced to wait for the slow workstations. In this case, the use of two parallel stations effectively halves the cycle time (doubles the production rate) for the stations. This permits the workload to be more evently distributed among the workstations. The second reason for considering a parallel configuration is when reliability of a certain station (or group of stations) is a problem. Production requirements may be such that a shutdown of the line cannot be tolerated. For example, in the motor assembly line it is likely that certain stations will have greater chance of breakdown, and those are the stations that would be provided with a duplicate station in parallel. This would improve the

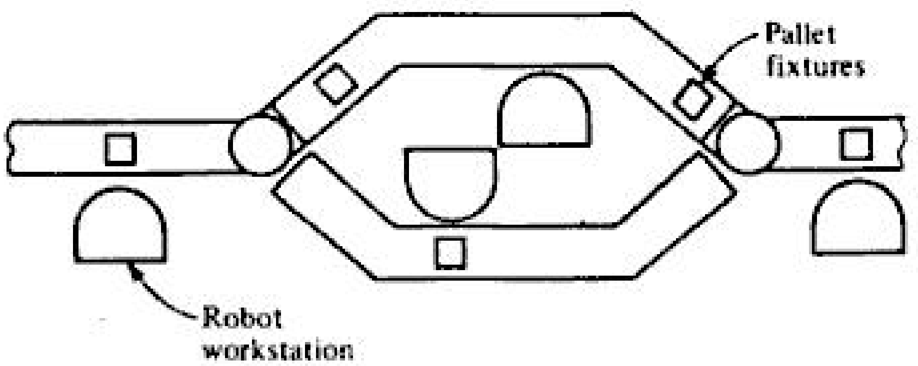


Figure 15-20 Parallel workstations on an assembly system.

operation without the use of an automatic screwdriver is difficult. Even with a powered device to perform the operation, the process of turning the screw into the part requires time. If the objective of using a threaded fastener is to allow for subsequent disassembly (e.g., for service of the product), then the use of screws may be an appropriate design decision. However, if the particular assembly or subassembly is designed to be permanent, then perhaps a better choice than screw fastening would be a press-fit or adhesive bonding of the parts.

Another consideration in the design of an assembly is the direction in which the parts are to be added in the assembly operation. If the parts can all be added without reorienting the partially completed subassembly, then time and money can be saved. On the other hand, if the subassembly requires many reorientations, then handling time is being spent without adding any real value. Similarly, if all the components can be added along the same axis direction, a robot with fewer degrees of freedom can perform the assembly tasks. This suggests that stacking of parts during the construction of an assembly is advantageous.

Today's robots are typically one-armed machines. Coordination of more than one arm at a time is difficult with current control technology. Interpreting this limitation in terms of limitations on the assembly process, an automated mating or joining operation should require the robot to handle no more than one part at a time. Assemblies that require the robot to manipulate two parts simultaneously or to maintain the relationship between two parts while adding a third may require a significant amount of fixturing. The solution to these problems is to design the parts so that they maintain their relationships with each other by designing such features into the components as locating bosses, grooves, and other mating elements.

In order to facilitate automatic assembly, it is often appropriate to add certain features to the components. For example, breaking edges and corners on parts, and adding chamfers to the holes will make it easier to accomplish part insertion tasks. These design features added to the components will minimize the robot's accuracy requirements and should allow faster operating cycles. Also, the design of distinct alignment features into the parts, or purposely making an otherwise symmetric part into an asymmetric part, makes it easier to feed and mate the parts in the proper orientation. These added part features will probably necessitate extra processing operations which may increase manufacturing costs of the components. The increases in part costs must be justified by corresponding reductions in assembly costs.

There are many other considerations in designing for automated assembly, and this section is not intended to be a thorough treatment of the subject. We leave it to the interested reader to explore the literature, in particular Refs. 1, 3, and 11.

15-8 INSPECTION AUTOMATION

Inspection is a quality control operation that involves the checking of parts, assemblies, or products for conformance to certain criteria generally specified

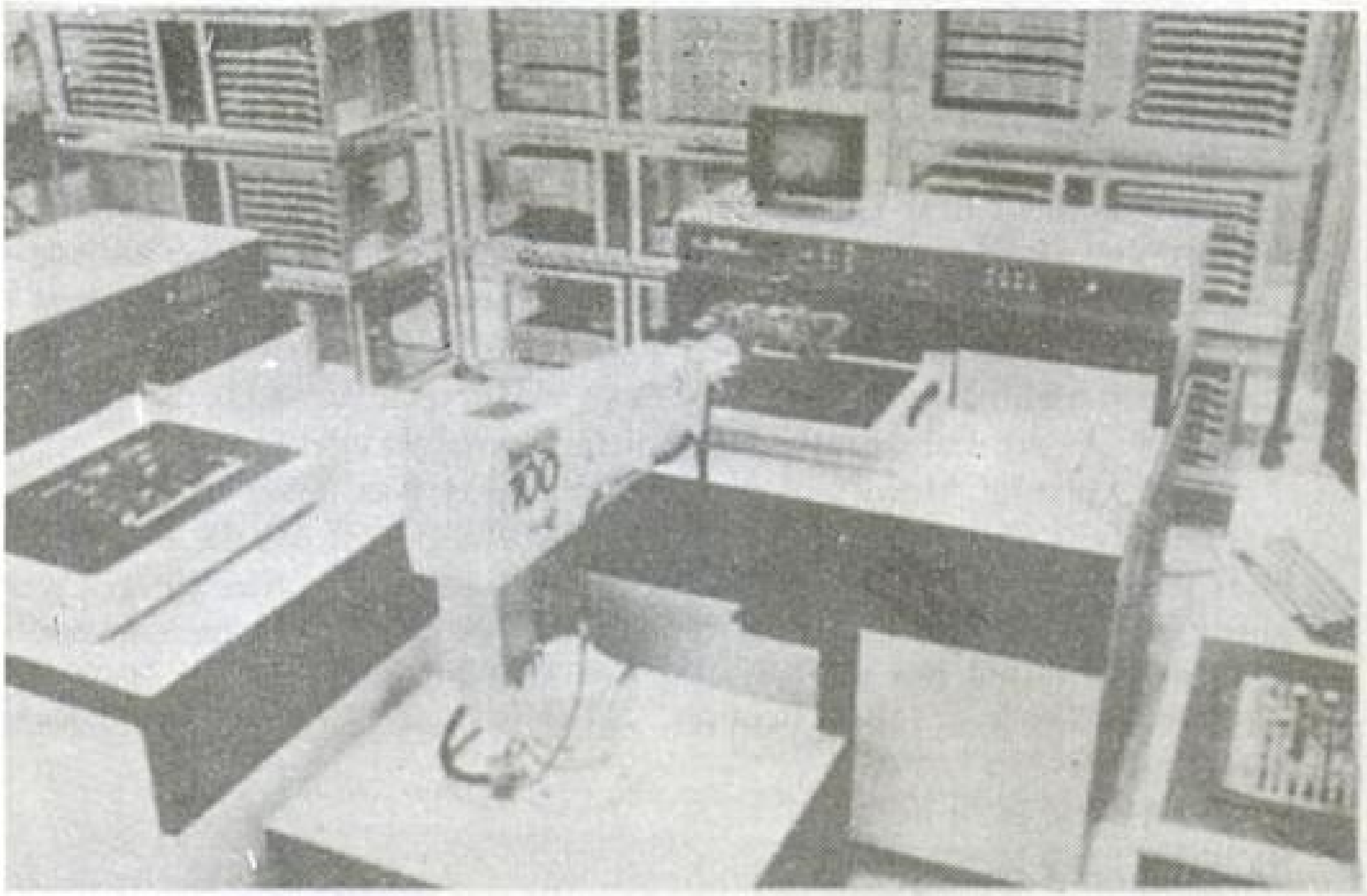


Figure 15-24 Automatic test cell of Example 15-6. (Photo courtesy of United States Robots, subsidiary of Square D Company)

tester can be designed to test the response of all the circuits in a printed circuit board.

Example 15-6 Figure 15-24 shows a robot cell for testing printed circuit boards. The PC boards are presented to the robot in a tote bin. The robot unloads the board from the tote bin and places it in one of the available testers. It then signals the tester to perform the test. When the tester completes the functional testing, it informs the robot and indicates whether the board has passed the testing procedure. The robot then sorts the PCB accordingly into the appropriate output tote bins.

Integrating Inspection into the Manufacturing Process

As we have discussed earlier in this chapter, inspection is a vital component of the automated assembly process. This is true not only in assembly but also in other automated manufacturing methods as well. As the human operator is removed from the workstation, the function of checking the work must be taken over by other means. One of the features of a robotic workcell is that the inspection can usually be added for a nominal capital cost. The inspection process can often be accomplished on the finished part at the same time as the production process is working on the next part. Therefore, the added time to inspect can be minimized. It is likely that the automated factory of the future will be characterized by a very high level of integration between the manufac-

16-1 INITIAL FAMILIARIZATION WITH ROBOTICS TECHNOLOGY

Many companies are in the following situation: None of their personnel have any expertise in robotics, but they believe that there are potential applications for robots in their plants. In order to become involved with the technology and be capable of making rational decisions on robot projects, these personnel are faced with the problem of quickly becoming knowledgeable with the field.

The sources of information on robotics include books, technical magazines and trade journals, robot manufacturing companies, consulting firms, technical seminars, conferences, and trade shows. At the time of this writing, there are several dozen books on robotics available. There are more than half a dozen magazines, trade journals, and newsletters devoted specifically to the field of robotics. In addition, many of the other trade publications include feature articles on topics in robotics. There are well over 250 organizations that provide products and services related to robotics. These organizations include manufacturers, suppliers, consultants, schools, and research institutions working in robotics or fields related to robotics.

In addition to these materials, there are also seminars, conferences, and trade shows devoted to the technology and application of robotics. At Lehigh University we offer a workshop seminar on "robotics in manufacturing." Similar seminars are offered by organizations like the Institute of Industrial Engineers (IIE) and Robotics International of the Society of Manufacturing Engineers (SME). More specialized seminars are also offered by a number of organizations on topics such as end effector design, developing robot workcells, and arc-welding applications. The annual Robot Conference and Trade Show, cosponsored by the Robotic Industries Association and Robotics International of SME, provides an opportunity for the robot industry to show its products and exchange ideas at the various technical paper presentations included in the conference.

In the process of introducing robotics into a firm, the importance of management support should not be underestimated. Many of the reports of successful robot installations point to this as a critical factor. The implementation of robotics in a firm is usually a long process, perhaps spanning several years before the first application project is completed. It is important that management provide continuing support and encouragement during this startup period. Some companies have lost valuable time by turning on and off their support to their manufacturing staff functions as attempts were made to implement robotics.

Another management issue involves the approval process required to install a robot project. One might assume that the authorization to implement a robot project would constitute management support for the project. However, the approval process in the firm might be such that the manager who decided to invest in the robot system is not the same as the manager who will use it in the plant. For the project to be successful, it is important that the manager responsible for using the system also be committed to its success.

To make the final selection of the robot, the following decision procedure is suggested. The procedure consists of preparing a detailed listing of the technical features for the particular application and then systematically comparing these features against the specifications of the alternative models under consideration. It is advantageous to divide the list of technical features into two categories: "must" and "desirable." The "must" features are ones that must be satisfied by the robot in order to perform the application. If any of the candidates do not satisfy the "must" features, then that robot model is excluded from further consideration.

The "desirable" features are ones that are not necessarily required to accomplish the application but would be highly beneficial during installation and/or operation. The specifications of each robot candidate would be compared to each of the desirable features, and a rating score would be assigned to the candidate to indicate how well the robot satisfies the particular feature. There may be differences in relative importance among the various features, and this would be taken into account by giving each feature a maximum possible point score. Determination of the rating score for the different robot models in each feature category would be a judgment call that the applications engineer would have to make based on the relative merits of each candidate.

Example 16-1 Figure 16-2 illustrates a possible format for making the comparison of the application features against the available robot specifications. This is based on an actual decision table developed by a company for selecting a robot for an arc-welding application. The organization of the form has been changed and some of the features have been stated differently here for clarity. The company, of course, will remain anonymous. The robots are identified in Fig. 16-2 as A, B, C and D, rather than using the actual company and model number.

The features are divided into the two categories: "must" and "desirable," according to our suggested procedure. The must features were considered essential for the welding application. It turned out that one of the models being considered has only five axes, whereas six axes were considered a requirement. Therefore, that model (Model C) was eliminated from further consideration.

The desirable features were each evaluated as to its relative priority by giving it a possible range of point score values (the numbers in parentheses in Fig. 16-2). The applications engineer made judgments to determine how each of the remaining three models rated for the given feature. It should be noted that some of the desirable features listed in the form included nontechnical considerations as well as technical considerations. Price, delivery, and vendor evaluation were considered important by the company in selecting the robot model. Based on the sums of the point scores, Model D was selected as the best robot for this application.

the cycle, rather than using the gate closure for restarting. Other possible physical barriers include safety rails and chains, although these are not as effective as a full fence.

Another approach that has been used in industry as a physical barrier is a steel post in the floor at the limits of the programmed motion cycle, so that an out-of-control robot arm crashes into the post rather than go beyond its allowed space. However, there are certain undesirable aspects to this type of barrier that make it unsuitable. First, if the robot arm crashes into the steel post at high speed, it will no doubt be damaged or destroyed. This kind of mishap could occur during programming of the robot rather than as a result of a malfunction of the robot itself. Thus, although the steel post is intended to protect against a robot control failure, it is possible that the robot arm could be ruined due to human error instead. A second feature which makes the post unsatisfactory is that it does not prevent intruders from entering the robot cell. A third undesirable aspect of the steel post is that a human could be pinned between the robot arm and the post. This would probably result in greater injury to the human than by simply being struck in open space by the robot arm.

In a robot cell designed to operate with humans as coworkers in the production process, certain features must be designed into the cell layout to protect the workers. This is typically encountered when human workers are employed to load and unload workparts in the cell and the robot is used to perform a processing operation such as arc welding or grinding. In these cases, some form of two-position parts manipulator can be used to exchange parts between the robot and the worker. One possible workcell configuration for accomplishing this exchange is illustrated in Fig. 17-1. It uses a rotary indexing table to move raw workparts from the human operator's position to the robot position for processing, and simultaneously moves completed parts from the robot position to the operator's position for unloading. This arrangement prevents inadvertent collision between the worker and the robot. Some form of operator interface is required to permit the worker to index the

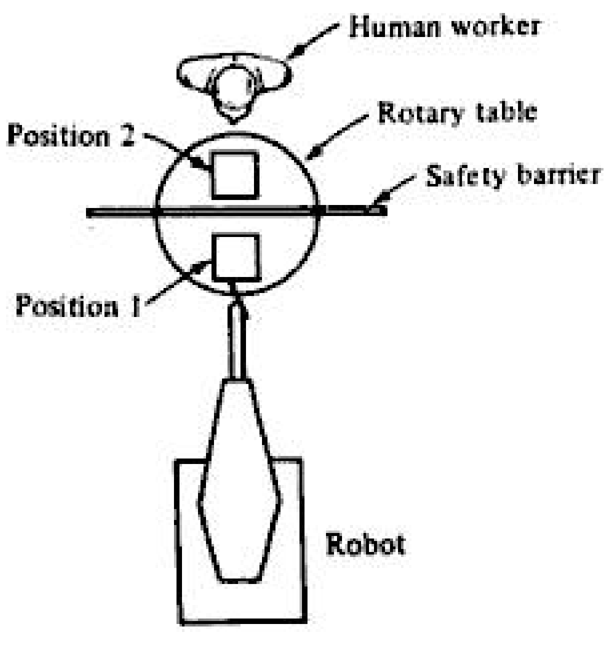


Figure 17-1 Cell layout using part manipulator to separate human worker from robot for safety and production efficiency.

company into five categories:

Awareness
Justification
Application
Operations and maintenance
Safety

Awareness training provides a survey of robotics, including technology, applications, economics, and social implications. It also explores the future trends and research developments taking place in robotics. This category of training is typically presented to managers and engineers to encourage the implementation of robotics and to examine the opportunities for applying this technology. Training programs designed for production and maintenance personnel are designed to dispel the mystery surrounding robots by explaining how the machines operate and by presenting examples of successful applications.

Training in the justification of robots is intended for engineers and managers who are responsible for implementing robot projects in the company. Justification training deals with economic issues and the unique problems that arise in the justification of robots as compared to other investment projects. It is often the case that conventional company justification criteria do not include some of the benefits that accrue from the use of robots. The purpose of the justification training is to examine these benefits and to incorporate consideration of them in the justification procedures.

Training in applications is designed principally for technical people (engineers, engineering managers, production managers, and foremen) who must select the applications and plan the installations. Coverage in this category would include technical areas such as basic technology (Chap. Two), robot programming (Chaps. Eight and Nine), and applications engineering issues (Chaps. Eleven through Fifteen).

Training in robot operation and maintenance must be provided for production and maintenance personnel. It is designed to give these persons the detailed technical skills and knowledge required to use and service the equipment. Most robot manufacturers provide training programs as part of the sales contract with the customer. These programs are held either at the robot company's facilities or, in some cases, at the customer's plant where the robot is to be installed. A typical training course would cover areas such as programming, operation, maintenance, and repair of the robot. A typical training course offered by the robot manufacturer may last 1 or 2 days for a relatively simple robot and from 1 to 2 weeks for a more complex robot. This type of training should be timed to coincide with the installation of the robot. Because of employee turnover and the need to periodically upgrade operating personnel, the user company must plan on regular training programs for its people.

The construction and operation of robots are generally similar to other

maintenance program is to be initiated in the plant which is expected to increase the MTBF to 300 hours and reduce the MTTR to 6 hours. Determine the effect of the PM program on the availability of the robot.

Before the PM program is introduced the availability is

$$\text{Availability} = \frac{200 - 8}{200} = 0.96 \text{ or } 96\%$$

As a result of preventive maintenance, the expected availability of the robot will become

$$\text{Availability} = \frac{300 - 6}{300} = 0.98 \text{ or } 98\%$$

In order to operate a successful PM program, it is important to maintain a record of the maintenance performance of each piece of equipment in the plant. This record should include data on the times between failures of the equipment, the times to repair, the nature of the maintenance problem, and the remedial action taken including the components repaired or replaced. By keeping this kind of maintenance log for each robot, statistics such as MTBF and MTTR can be calculated so that the most appropriate PM can be planned and scheduled for each robot. Another reason for keeping a record of the maintenance performance for each robot is to help develop an effective spare parts policy.

Spare Parts Policy

A robot might consist of several hundred to several thousand components. Certain of these components are subject to gradual wear or sudden failure which could disable the entire robot. It is important that the user company consider the problem of keeping an inventory of parts on hand to replace those on the robot that wear or fail. Ottinger[5] suggest a budgetary estimate for spare parts of 10 percent of the robot base price. Many of the robot manufacturers provide a list of recommended spare components that should be stocked by the user.

The user's spare parts policy can cover the full range of inventory levels. At the one end is the policy of keeping no spare parts on hand, except miscellaneous items such as fuses that the company would probably have anyway. At the opposite end of the range, the company might elect to keep a complete duplicate robot available to replace a robot that has failed. The decision depends on finding an appropriate balance between the cost of robot downtime and the cost of spare parts inventory. For an automobile manufacturer with several dozen spot-welding robots on its car body assembly line, it might be worthwhile to have one or more entire robots on hand as spares to replace robots on the line which periodically fail. The broken robot would be quickly removed from its position in the line and replaced by the spare robot, and the program for that station would be entered into the memory of the

17-2 Four robots of the same manufacturer and model are used in a robot-automated manufacturing line. The line is fully integrated and if one robot fails, the entire line must be stopped until repairs are completed. Data have been collected for each robot on times between failures and repair times. The data were taken after the normal break-in problems were solved and cover the first 1200 hours of operation of the line. The data are as follows:

Robot	Time between failures, hours	Repair times, hours
1	196	4.7
	115	2.1
	280	10.8
	304	5.2
	237	6.6
2	76	1.1
	404	7.4
	282	3.9
	126	6.2
	205	9.3
3	165	3.4
	358	8.2
	260	5.5
	329	7.6
4	45	3.3
	124	5.5
	236	9.4
	288	6.1
	301	7.0
	201	3.2

(*a*) Compute the mean time between failure (MTBF) and the mean time to repair (MTTR) for this robot model according to the four samples given.

(*b*) Determine the availability for this robot model from your computed values of MTBF and MTTR in part (*a*).

(*c*) What is the effective availability of the manufacturing line as a result of the fact that there are four robots?

17-3 The cost of a certain robot is $90,000. It is desired to establish the appropriate level of spare parts inventory for this machine. Based on previous experience, it is known which components of the robot are most likely to fail. The spare parts inventory level will be established so that the total cost of inventory plus lost production time is minimized. The cost of inventory is 35 percent per year of the value of the spare parts in stock.

The robot operates one shift per day (8 hours per day) for a total of 2000 hours per year. The mean time between failure (MTBF) of the robot is 250 hours. When a breakdown occurs, the repair time averages 4 hours if the required spare part is available. If the part is not available, the company loses the 4 hours repair time plus the time for the needed part to be delivered. The delivery time results in the loss of 2 working days. For an 8-hour shift, an average of 20 hours of production time is lost while the part is being delivered and repairs are being made. In addition, there is a delivery charge of $25 per order for the needed parts. The company figures that the cost of downtime is $75 per hour. The following table indicates the various alternative spare parts levels being considered and the corresponding probability that the robot can be repaired using

The purpose of this chapter is to consider the effects that robotics has had and will have in these areas. We will use the above list as an outline for the chapter, exploring each of the five areas (although not all at the same level of detail). In our coverage of the labor issue, we will consider the effects of robotics and automation on direct labor in manufacturing, and we will also consider the likely effects on professional and semiprofessional staffs (engineering, production planning, and the clerical support for these functions). We will not attempt to draw any grand conclusions about these social issues, nor will we make any proposals for national policy to deal with them. Our objective is simply to describe the issues and their impact.

18-1 PRODUCTIVITY AND CAPITAL FORMATION

In our previous discussions of applications and applications engineering, we have described how robots can reduce cost and increase productivity in manufacturing. Productivity improvement in the United States is an important social issue and a major national concern. The conventional definition of productivity is the following:

$$\text{Productivity} = \frac{\text{units of output}}{\text{units of input}}$$

The units of output can be reduced to monetary terms for purposes of comparing the products of different industries. Labor hours have traditionally been used as the units of input in productivity measurement, and the resulting ratio gives an indication of labor productivity only. However, capital (equipment), technical knowledge, and various other inputs are also considered to be ingredients contributing to productivity. When these other inputs are combined, the ratio is called the total factor productivity. Technical knowledge is a factor which should be interpreted to include several facets. One facet is certainly the technological improvements that are incorporated into successive generations of capital equipment. For example, a robot purchased today can be expected to have certain improvements in technology (e.g., intelligence, programming ease, accuracy, input/output interface capabilities, etc.) as compared to a robot purchased 10 years ago for perhaps the same price. Another facet of technical knowledge is the managerial expertise in operating the business. Manufacturing management practice has been improved, for example, by computerized methods such as materials requirements planning (MRP), shop floor control, and other computer-integrated manufacturing techniques.

Robotics is an input to the productivity ratio which represents both capital and technical knowledge. As an input, it is a substitute for human labor in determining productivity. Presumably by making this substitution, productivity is improved. In the bargain, human workers are denied work opportunities with potentially serious financial and emotional consequences to themselves

route sheets), estimating, purchasing, and many of the tasks associated with production planning and control. The use of integrated systems of computer-aided design and computer-aided manufacturing will allow the repetitive and routine portions of these planning functions to be accomplished automatically. Some of the automated planning software packages which are currently available but not all widely used, include CAPP (computer-aided process planning), MRP (material requirements planning), and automated numerical control part programming.

As direct manual labor is gradually replaced by automated systems, the need for work measurement (direct time study) and the use of piece rate incentive systems will be reduced. It is unclear whether work measurement, as it is traditionally accomplished today, will even be done in the automated factories of tomorrow. Industrial engineers who have typically been responsible for work measurement in production plants, will be faced with the problem of measuring the work and designing the appropriate incentive systems for indirect labor and other personnel who will operate the equipment in the automated factory. New types of work standards related to product quality, production yield, throughput, lead time, and other measures will have to be devised.

Robotics is a technology which is at the same time highly specialized and highly interdisciplinary in nature. Engineering staffs who develop robot systems and applications must reflect this interdisciplinary nature. Robotics is a combination of computer science, machine tool technology, mechanism design, and control systems. Its application requires a mixture of electrical engineering, human factors, engineering economy, workplace layout design, and robot programming. The implementation of robot systems and other forms of automation requires not only engineers who represent the individual disciplines but engineers who are also capable of integrating their own disciplines and those of others.

18-3 EDUCATION AND TRAINING

Several issues related to education and training are raised by the mass introduction of robotics into society. We are not referring to the specific training issues discussed in Chap. Seventeen, but rather the more general social issues that must be confronted. Many of these issues are identified in the two reports published by the Office of Technology Assessment[9,10] listed among the references. They include:

Shortage of trained technical staffs in robotics and other programmable automation technologies

Need for a more technologically literate work force

Shortage of technical instructors and state-of-the-art laboratory equipment in the schools

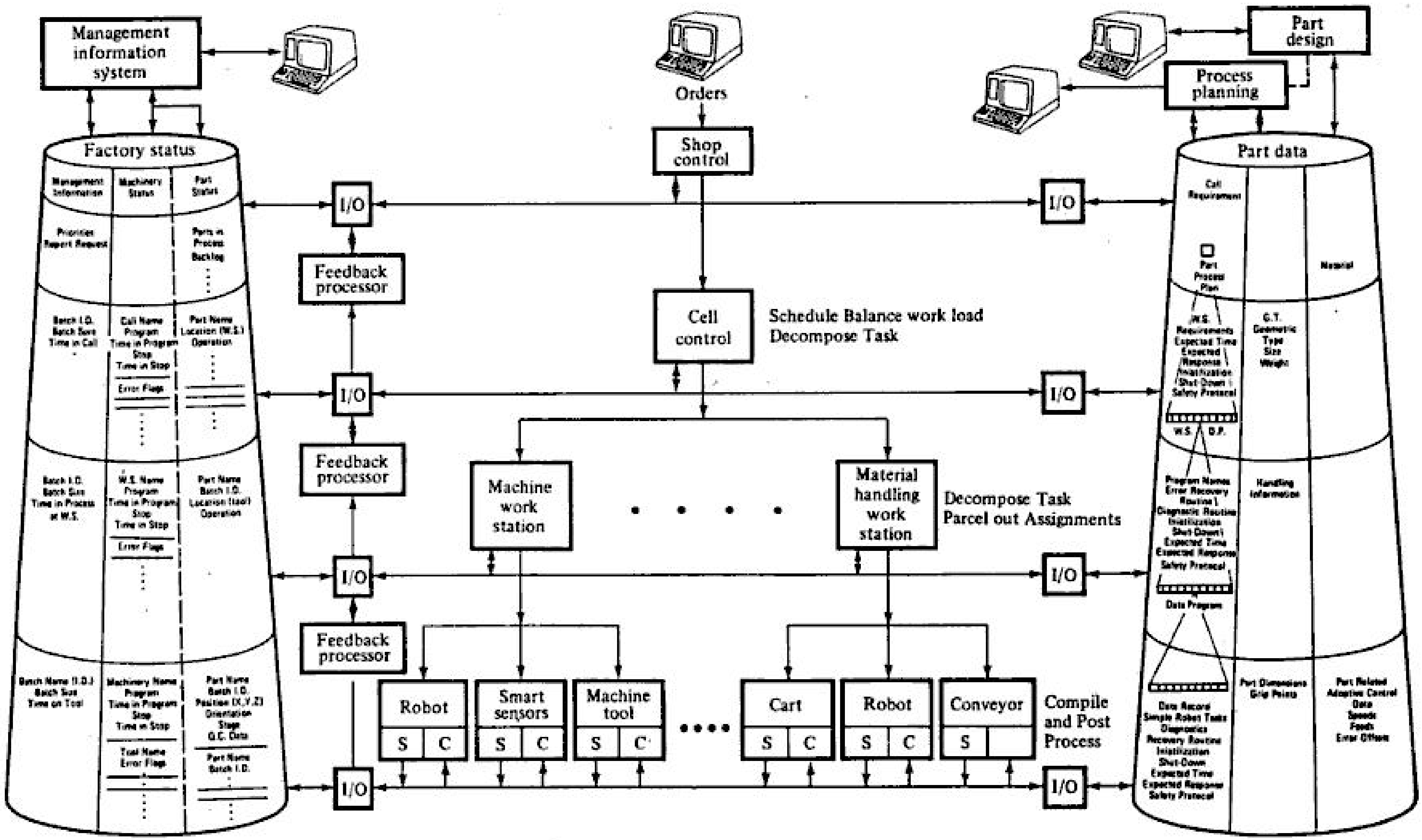


Figure 19-1 Hierarchical control structure used in the Real-Time Control System at the National Bureau of Standards. (*Reprinted by permission from Albus*[1])

vehicles required impractically long times to move a few feet. The navigation problem is an active research area.[9,19]

Walking Machines

Wheeled vehicles have limitations; they can only travel over relatively smooth surfaces. Vehicles with tank tracks would be an improvement for rough terrain. Walking machines[12,14,21] offer the greatest versatility for dealing with a variety of surfaces and obstacles. However, walking machines must overcome all of the same technological hurdles as autonomous locomotive wheeled vehicles, with the additional problem of coordinating the motions of the legs. In addition, since it is assumed that such vehicles will be used over rough terrains, they must be highly adaptive to the irregularities of the terrain.

There are a number of factors that must be considered in the design and control of walking machines. These factors include the number of legs, gait selection, balance, and coordination of the legs. Research has been done on one-legged, two-legged, four-legged, and six-legged machines. A one-legged

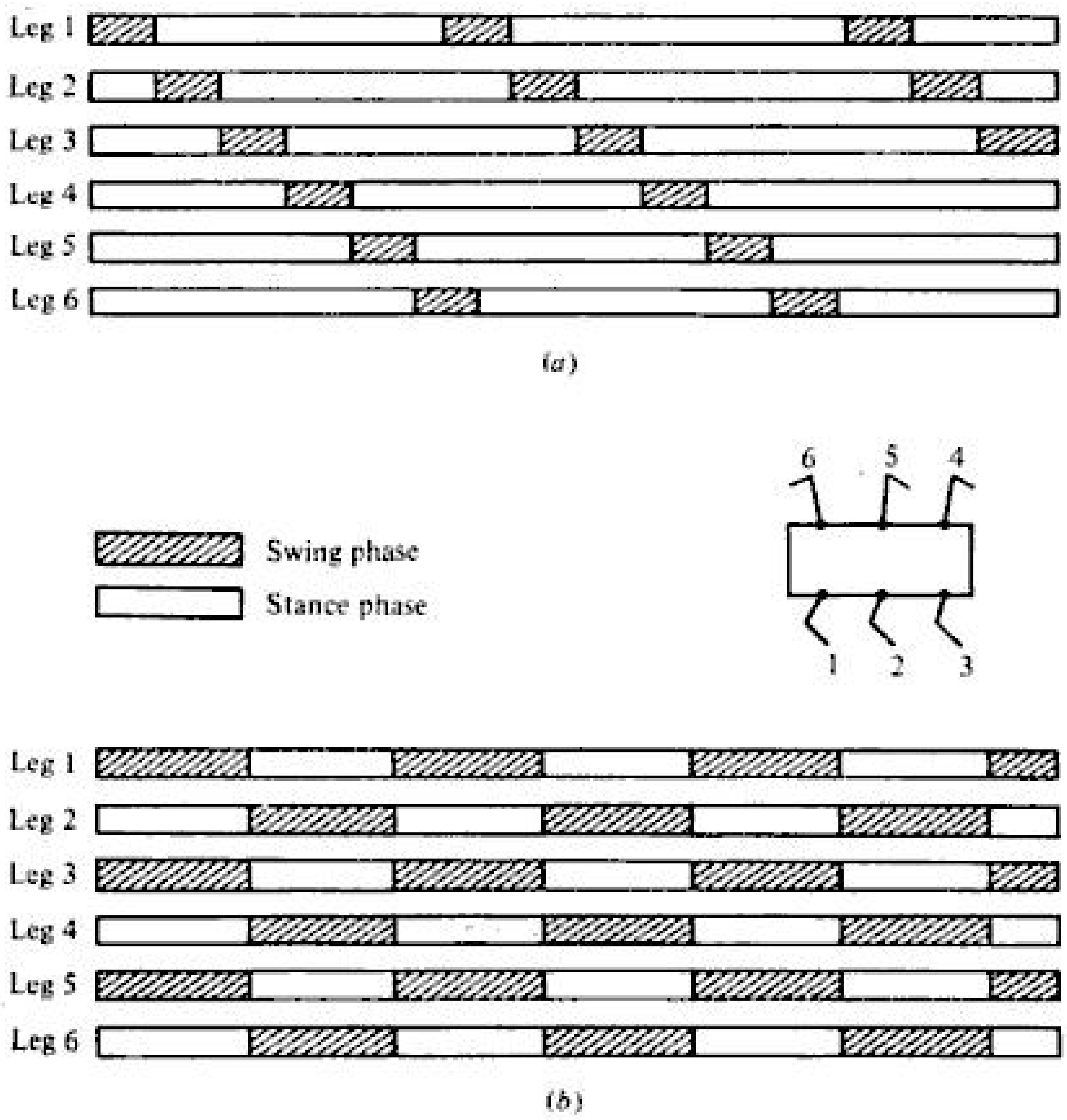


Figure 19-4 Some of the possible gaits that might be used by a walking machine.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

OTHER McGRAW-HILL INTERNATIONAL EDITIONS IN RELATED FIELDS

Boothroyd: FUNDAMENTALS OF METAL MACHINING AND MACHINE TOOLS
Dieter: ENGINEERING DESIGN, SI Metric Edition
Dieter: ENGINEERING DESIGN
Hicks: STANDARD HANDBOOK OF ENGINEERING CALCULATIONS, 2/e
Jelen: COST AND OPTIMIZATION ENGINEERING, 2/e
Koren: COMPUTER CONTROL OF MANUFACTURING SYSTEMS
Krar: MACHINE TOOL OPERATIONS
Krar: TECHNOLOGY OF MACHINE TOOLS, 3/e
McGraw-Hill: DICTIONARY OF ENGINEERING
Raven: AUTOMATIC CONTROL ENGINEERING, 3/e
Rothbart: MECHANICAL DESIGN AND SYSTEMS HANDBOOK, 2/e
Shigley: MECHANICAL ENGINEERING DESIGN, First Metric Edition
Sumanth: PRODUCTIVITY ENGINEERING AND MANAGEMENT

The McGraw-Hill Companies

Tata McGraw-Hill
Publishing Company Limited
7 West Patel Nagar, New Delhi 110 008

Visit our website at : www.tatamcgrawhill.com

ISBN-13: 978-0-07-026509-7
ISBN-10: 0-07-026509-7

9 780070 265097