Microcomputers in Engineering and Science

J. Ffynlo Craine
Graham R. Martin

Microcomputers in Engineering and Science

J. Ffynlo Craine
Graham R. Martin

ADDISON-WESLEY PUBLISHING COMPANY

Wokingham, England · Reading, Massachusetts · Menlo Park, California
Don Mills, Ontario · Amsterdam · Sydney · Singapore · Tokyo · Mexico City · Bogota
Santiago · San Juan

The programs presented in this book have been included for their instructional value. They have been tested with care but are not guaranteed for any particular purpose. The publisher does not offer any warranties or representations, nor does it accept any liabilities with respect to the programs.

Cover design by John Gibbs.
Typeset by H. Charlesworth & Co. Ltd. directly from magnetic tape supplied by the authors using **troff**, the UNIX text-processing system.
Printed in Finland by Werner Söderström Osakeyhtiö. Member of Finnprint.

British Library Cataloguing in Publication Data
Craine, J.F.
Microcomputers in engineering and science.—
(International computer science series)
1. Science—Data processing. 2. microcomputers
3. Engineering—Data processing
I. Title. II. Martin, G.R. II. Series
502'.8'5404 Q183.9
ISBN 0-201-14217-1

Library of Congress Cataloging-in-Publication Data
Craine, J.F.
Microcomputers in engineering and science.
(International computer science series)
Bibliography: p.
Includes index.
1. Microcomputers. 2. Microcomputers—Programming.
3. Computer interfaces. 4. Engineering—Data processing.
5. Science—Data processing. I. Martin, G.R.
II. Title. III. Series.
TK7888.3.C73 1985 502.8'5416 85-18490
ISBN 0-201-14217-1

ABCDEF 898765

Microcomputers in Engineering and Science

INTERNATIONAL COMPUTER SCIENCE SERIES

PREFACE

The material in this book is based upon many years of experience, both designing microcomputer-based systems and lecturing on microcomputers and their applications. These lectures have been directed at practising engineers from many disciplines as well as undergraduates reading engineering and computer science at the University of Warwick. Microcomputers are small, robust, and cheap enough to be used as components in a wide range of systems; industrial controllers and laboratory instrumentation are just two applications where mechanical, electromechanical, and non-programmable electronic control systems are being replaced by microcomputers.

As the use of mainframe computers became widespread, engineers and scientists learnt how to program computers to solve purely numerical problems. Today, however, the use of microcomputers as integral parts of larger real-time systems represents a wide departure from this more 'conventional' use of computers. To design and apply microcomputer-based systems in control and instrumentation requires an understanding not only of the microcomputer and its programming, but also of the remainder of the system. Too many books on microcomputers and microprocessors gloss over the problems of interfacing and the real world signals with which they are involved. Likewise, books on engineering control and instrumentation usually neglect the microcomputer itself. In this book we have attempted to redress the balance and include coverage of all the topics essential to the microcomputer applications engineer or the scientific laboratory user. Where our objective of covering all this material in a single volume has meant that it has been impossible fully to do justice to a topic, the text provides references to other books for those readers wanting a deeper insight.

After a brief history of the use and development of microcomputers, the book proceeds to explain how information is represented and processed within a microcomputer. Despite its many shortcomings, BASIC is introduced as a simple programming language because it has become virtually the *lingua franca* of the microcomputing fraternity. A more modern language, Pascal, is then described and its advantages over BASIC, such as the way in which it encourages a structured programming approach, are explained.

The third chapter explains the internal architecture of the microcomputer, and as each section is introduced, the instructions associated with its use are described. Programming examples illustrate how the constructs available in high level programming languages can be implemented in assembly language. The explanation of microprocessor operation has been made as independent of microprocessor type as possible, but where examples are used, 6809 assembly language has been employed because its particularly regular instruction set makes it attractive for this purpose. Peculiarities of the instruction sets of some other microprocessors have been explained where appropriate, and Appendix I provides a comparison of five commonly used types.

Chapter 4 introduces input/output devices and various programming techniques for their operation. The interfacing of microcomputers is of central importance when they are embedded within larger systems. For this reason two further chapters are devoted to digital and analog interfacing, with examples covering commonly encountered problems. The importance of transducers in the control and instrumentation of physical systems is often overlooked, and in this book a whole chapter has been devoted to actuators and sensors for commonly encountered quantities such as position and temperature. As systems become larger, the communication of information between microcomputers becomes progressively more important, and Chapter 8 is dedicated to the subject of data communication. The final chapter deals with the problems encountered in the development and maintenance of microcomputer-based systems. This is another area which is often given scant cover, yet topics such as estimating the amount of effort which a system will require for its development, and design for testability, are of extreme importance in microcomputer-based design.

This book assumes that the reader has only a very basic knowledge of electrical theory such as Ohm's law and simple networks. It is aimed not only at students reading more obviously 'microcomputer related' courses such as computer science and electronic engineering, but at students reading any of the physical sciences or branches of engineering at university or polytechnic, practising engineers, and even the computer hobbyist who is interested in the serious application of the microcomputer.

The publishers wish to thank the following for permission to reproduce material: Digital Equipment Co. Ltd for Figure A.3 and examples from the LSI11 Instruction Set (pp. 425-9); INMOS Ltd for Figure 8.13; Motorola Inc. for programming models of the 6809 and 68000 microprocessors, and examples from their Instruction Sets (pp. 425-9); Rockwell International for Figures 4.19 and A.1 and examples from the 6502 Instruction Set (pp. 425-9); and Zilog (UK) Ltd for Figure A.2 and examples from the Z80 Instruction Set (pp. 425-9).

Finally, the authors wish to record their thanks to Derek Chetwynd for his help in the writing of Chapter 7, and to Andrew McGettrick and Jan van Leeuwen for their valuable comments which caused the book to expand considerably from its originally planned size.

Warwick — J.F. Craine
May 1985 — G.R. Martin

CONTENTS

Chapter 1 INTRODUCING THE MICROCOMPUTER

1.1 Microcomputers

The development of the microcomputer during the 1970s brought about a revolution in engineering design. The industrial revolution at the turn of the nineteenth century heralded the development of machines which could replace physical drudgery by mechanical means. Apart from a few exceptions, however, these machines required manual supervision because the problem of controlling this mechanical power was not at all straightforward.

Many types of automatic control system have appeared during the twentieth century, based upon electronic, mechanical, hydraulic and fluidic principles. In each case the design techniques have been similar because each component of the system usually contributes a single well-defined function to the system's behaviour.

The microcomputer represents a fundamentally different approach to the design of a system. Its physical form is quite simple and reliable, consisting of a few general purpose elements which can be programmed to make the system function as required. It is the controlling program which must be designed to give the system the required behaviour, and which will contain 'components' and 'subassemblies' just like any other kind of engineering. The program, or **software**, is just as much a part of the engineered system as the physical **hardware** but it is much less susceptible to failure, provided that it is designed properly.

The idea of programmed systems is not new; electronic computers have been in existence for many decades. However, it has taken the development of the large scale integrated circuit — the **silicon chip** — to produce computers which are cheap, rugged, and reliable enough to be incorporated into engineering designs as components. The techniques of software design are well known to computer scientists and it is not surprising that the principles of good software design and 'software engineering' are essentially those of good engineering design. We shall see in later chapters of this book that engineering design using software allows systems to be designed more easily than using more conventional techniques.

It is the combination of developments in electronic device technology with those in computer technology which has enabled the microcomputer to be produced, and the next two sections will show how these technologies have 'converged' to produce the microelectronic industry which we see today.

1.2 The silicon chip

The basic 'building brick' of modern electronic circuits is the transistor in which an input signal is used to control a larger output signal. The earliest transistors were made in the late 1940s from crystals of a material called germanium, but by the end of the 1950s silicon, the material which is still used for the great majority of electronic devices, was beginning to be used in transistor manufacture. One advantage of silicon is that it allows many of the production steps to be carried out using photographic techniques, and it was soon realised that these techniques could be developed to allow more than one transistor to be fabricated on the same crystal chip of pure silicon. Methods were then discovered for producing other types of electronic component such as resistors, capacitors and diodes on the same chip as the transistors and, importantly, the connections between these components could also be included on the same chip to produce an **integrated circuit**. The fact that the patterns which make up the components of the circuit and the connections between them are all defined photographically means that integrated circuits can be mass produced with very little human labour; there is very little assembly work involved. Conventional circuit assembly techniques are expensive and can be prone to wiring errors.

Integrated circuits were relatively expensive at first and their use was limited to applications where their small size and weight were essential, such as in aircraft and spacecraft. Here another advantage of integrated circuits, that of improved reliability, was important. Many of the failures in conventional electronic circuits occur in the connections between components rather than in the components themselves, and by reducing the number of components the reliability of the system can be substantially increased.

Much of the manufacturing cost of an integrated circuit lies in the packaging of the chip to make a usable electronic component, so that by including as many elements as possible in each integrated circuit the number of packages and connections can be minimised and the cost and reliability optimised. These considerations have led to a steady increase in attainable complexity over the years, with the number of elements which could be incorporated in a single chip almost doubling annually throughout the 1960s and 1970s as manufacturing techniques improved. At the same time, these manufacturing improvements have enabled price reductions to be made, and the price of a given type of integrated circuit often falls by about 30% per annum for the first few years after its introduction.

The main disadvantages of increasing complexity lie in the associated increase in design cost, and the danger that increased specialisation of the circuit will result in a limited range of applications and hence sales. This was already becoming a problem by 1970 and manufacturers became very active in trying to discover large-volume markets for integrated circuits and to define integrated circuit functions which would satisfy as many applications as possible. For example, a counting circuit could be used in an electronic clock or a digital voltmeter, but to fabricate all the electronics for each system on a single chip would require two new designs, each more expensive to produce than that of the simpler but more versatile counter circuit.

Modern integrated circuits (i.c.s) are designed using computer-aided techniques to perform specified functions rather than to have a specified circuit. The economics of i.c. manufacture are such that their design and development are expensive, while production is relatively cheap once the high cost of the manufacturing plant has been recovered. Ideally, an integrated circuit should have a very large number of potential applications so that the development costs can be spread over a large production run. This can be achieved by designing i.c.s which are **programmable** to a greater or lesser degree. The integrated circuits used in mains-driven electronic clocks, for example, can often be programmed to operate on 50 Hz (European) or 60 Hz (American) mains frequencies, and to display the time according to either the 12-hour or 24-hour clock system. In this case four different modes of operation can be obtained from a single design of circuit without incurring the development and inventory costs of four different designs.

1.3 From computer to microcomputer

The development of electronic technology has been the basis of the development of the computer, although the earliest computer antedates electronics by many decades. In fact the choice of candidate for 'first computer' depends very much upon what we choose to regard as a computer; for example one calculating aid, the abacus, has been known since antiquity.

Blaise Pascal invented the first real calculating machine in 1642; this was a purely mechanical device consisting of a set of geared wheels arranged so that a complete revolution of any wheel rotated the wheel immediately to its left through one-tenth of a revolution. In the first half of the nineteenth century Babbage designed programmable calculating 'engines' using mechanical techniques, although computers as we know them today were impractical because of the precision engineering required to construct them. The twentieth century saw the arrival of electromechanical computers, which were developed for specialised military tasks such as code-breaking and gun-aiming. However, these lacked speed and they were soon superseded by fully electronic designs using thermionic valves.

After the Second World War the computer was developed for civilian use as a large and expensive arithmetic-performing machine for business and scientific purposes. The power consumption and unreliability of valves limited the usefulness of computers until valves were ousted by the introduction of the transistor, and new information storage techniques appeared which allowed smaller and more powerful computers to be produced.

The 1960s saw the introduction of the **minicomputer** which was small and cheap enough at a few thousands of pounds — or dollars — to be produced in relatively large numbers for industrial control and laboratory instrumentation purposes. The minicomputer became yet cheaper when integrated circuits were introduced. As these circuits became more complex the number of integrated circuits required to construct a functioning computer fell, until simple minicomputers using only one or two printed circuit boards became possible. These minicomputers enabled significant reductions to be made in the time needed to design, develop, and commission many projects requiring large amounts of electronic logic.

Unlike fixed electronic logic circuits, a computer is a 'do-anything' machine which can be programmed to perform any of a wide range of tasks. The same **physical hardware** can be used in a number of different applications with **software (programs)** defining the behaviour of the system in each case.

More recent developments in integrated circuit technology have led to the introduction of **microcomputers;** small computers fabricated using relatively few integrated circuit components. In fact an entire microcomputer can be made as a single chip. At the heart of any computer is a **Central Processing Unit** or **CPU**, and the corresponding heart of the microcomputer is the **microprocessor** or **MPU** (MicroProcessor Unit), which is simply a CPU implemented on a silicon chip. Its processing power is greater than that of its giant predecessors and yet it is cheap and robust enough to be treated as simply another engineering component.

The microcomputer was conceived as a device which could be programmed in a very flexible fashion to give almost any desired behaviour by means of a list of electronic instructions. Using a microcomputer involves programming skill in producing these lists of instructions as well as more conventional electronic and mechanical design techniques. As its name suggests, the microcomputer is organised in much the same way as a conventional computer; indeed, it may be regarded as the 'natural' outcome of the 'evolution' of the computer from its earliest days.

1.4 Systems using microprocessors

Electronic systems are used for handling information in the most general sense; this information might be a telephone conversation, instrument readings or a company's accounts, but in each case the same main types of operation are involved: the processing, storage and transmission of information. In conventional electronic design these operations are combined at the functional level: for example a counter, whether electronic or mechanical, stores the current count and increments it by one as required. A system such as an electronic clock which employs counters has its storage and processing capabilities spread throughout the system because each counter is able to store and process numbers.

Present day microprocessor based systems depart from this conventional approach by separating the three functions of processing, storage, and transmission into different sections of the system. This partitioning into three main functions was devised by von Neumann during the 1940s, and was not conceived especially for microcomputers. Almost every computer ever made has been designed with this structure, and despite the enormous range in their physical forms, they have all been of essentially the same basic design.

In a microprocessor based system the processing will be performed in the **microprocessor** itself, the storage will be by means of **memory** circuits and the communication of information into and out of the system will be by means of special **Input/Output (I/O)** circuits. It would be impossible to identify a particular piece of hardware which performed the counting in a microprocessor based clock because the time would be stored in the memory and incremented at regular intervals by the microprocessor. However, the software which defined the system's behaviour would contain sections that performed as counters. This apparently rather abstract approach to the **architecture** of the microprocessor and its associated circuits allows it to be very flexible in use, since the system is defined almost entirely in software. The design process is largely one of **software engineering**, and the similar problems of construction and maintenance which occur in conventional engineering are encountered when producing software.

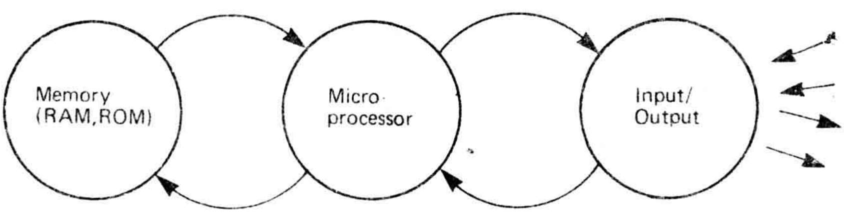

Figure 1.1 Three sections of a typical microcomputer

Figure 1.1 illustrates how these three sections within a microcomputer are connected in terms of the communication of information within the machine. The system is controlled by the microprocessor which supervises the transfer of information between itself and the memory and input/output sections. The external connections relate to the rest (that is, the non-computer part) of the engineering system.

Although only one storage section has been shown in the diagram, in practice two distinct types of memory ('ROM' and 'RAM') are used, as will be seen in section 1.10.5. In each case the word 'memory' is rather inappropriate since a computer memory is more like a filing cabinet in concept; information is stored in a set of numbered 'boxes' and it is referenced by the serial number of the 'box' in question.

The microprocessor processes data under the control of the program, controlling the flow of information to and from memory and input/output devices. Some input/output devices are general purpose types while others are designed for controlling special hardware such as disc drives or controlling information transmission to other computers. Most types of I/O device are programmable to some extent, allowing different modes of operation, while some actually contain special purpose microprocessors to permit quite complex operations to be carried out without directly involving the main microprocessor.

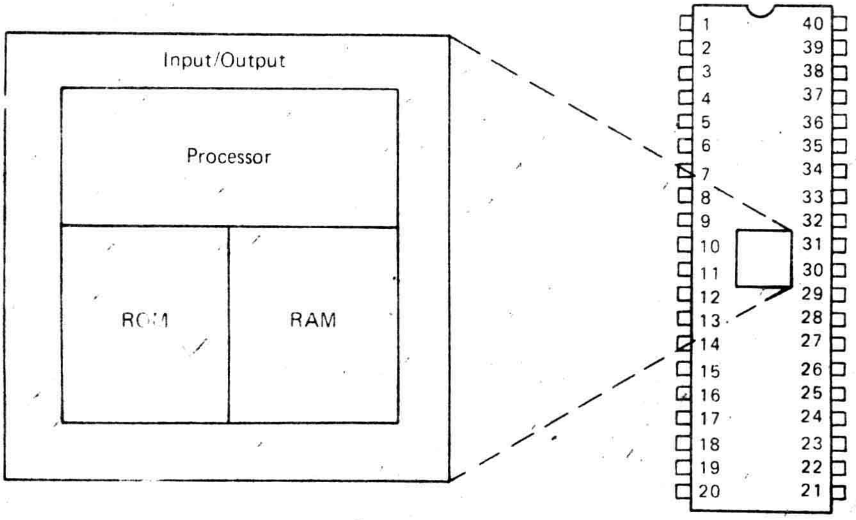

Figure 1.2 Sections of a single-chip microcomputer

The microprocessor, memory, and input/output circuit may all be contained on the same integrated circuit provided that the application does not require too much program or data storage. This is usually the case in low-cost applications such as the **controllers** used in microwave ovens and automatic washing machines. The use of a single package allows considerable cost savings to be made when articles are manufactured in large quantities. As technology develops, more and more powerful processors and larger and larger amounts of memory are being incorporated into **single chip microcomputers** with resulting savings in assembly costs in the final product. For the foreseeable future, however, it will continue to be necessary to interconnect a number of integrated circuits to make a microcomputer whenever larger amounts of storage or input/output are required.

Another major engineering application of microcomputers is in process control. Here the presence of the microcomputer is usually more apparent to the user because provision is normally made for programming the microcomputer for the particular application. In process control applications the benefits of fitting the entire system on to a single chip are usually outweighed by the high design cost involved, because this sort of equipment is produced in smaller quantities. Moreover, process controllers are usually more complicated so that it is more difficult to make them as single integrated circuits. Two approaches are possible; the controller can be implemented as a general purpose microcomputer rather like a more robust version of a hobby computer, or as a 'packaged' system, like that shown in Figure 1.3, designed for replacing controllers based upon older technologies such as electromagnetic relays. In the former case the system would probably be programmed in a conventional programming language such as the ones to be introduced in the next chapter, while in the other case a special-purpose language might be used, for example one which allowed the function of the controller to be described in terms of relay interconnections. In either case programs can be stored in RAM, which allows them to be altered to suit changes in application, but this makes the overall system vulnerable to loss of power unless batteries are used to ensure continuity of supply. Alternatively programs can be stored in ROM, in which case they virtually become part of the electronic 'hardware' and are often referred to as **firmware**.

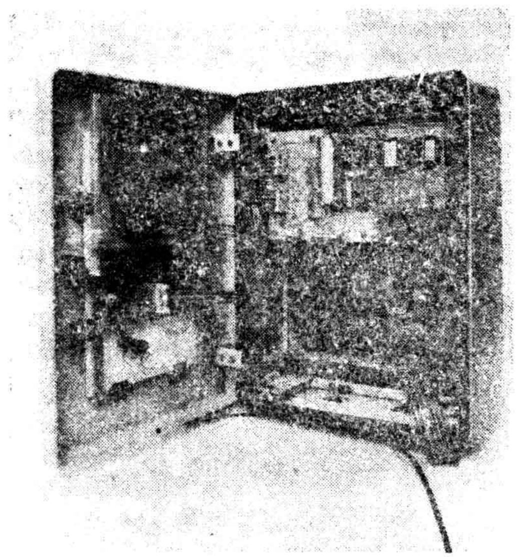

Figure 1.3 Purpose-designed process controller

More sophisticated process controllers require minicomputers for their implementation, although the use of large scale integrated circuits 'blurs' the distinction between mini- and microcomputers. Products and process controllers of various kinds represent the majority of present-day microcomputer applications, the exact figures depending upon one's interpretation of the word 'product'. Virtually all engineering and scientific uses of microcomputers can be assigned to one or other of these categories.

1.5 Using a microcomputer

Having discussed the range of applications and principles of operation of the microcomputer, it is pertinent to ask what advantages and disadvantages result from using microcomputer technology in preference to more