# COMMUNICATING WITH MICROCOMPUTERS

## An introduction to the technology of man-computer communication

IAN H. WITTEN

# COMMUNICATING WITH MICROCOMPUTERS

## An introduction to the technology of man-computer communication


**IAN H. WITTEN**

*Man-Machine Systems Laboratory*
*Department of Electrical Engineering Science*
*University of Essex, Colchester, U.K.*



ACADEMIC PRESS
*A Subsidiary of Harcourt Brace Jovanovich, Publishers*
London New York Toronto Sydney San Francisco

ACADEMIC PRESS INC. (LONDON) LTD.
24/28 Oval Road,
London NW1

*United States Edition published by*
ACADEMIC PRESS INC.
111 Fifth Avenue
New York, New York 10003



*British Library Cataloguing in Publication Data*
Witten, I H
Communicating with microcomputers. – (Computers and people).
1. Microcomputers
2. Computer input-output equipment
I. Title II. Series
001.6'44 QA76.5 80-40650
ISBN 0-12-760750-1
ISBN 0-12-760752-8 Pbk

Printed in Great Britain by
Whitstable Litho Ltd., Whitstable, Kent

# Preface

The microprocessor revolution has given us new power to handle information in almost every walk of life, and effective techniques of man-computer interaction are essential if this power is to be exploited for the benefit of ordinary people. Information processing is now virtually free. It is communication — both within a microcomputer system and between it and the world outside — that constrains applications. If machines cannot interact with ordinary people in ways with which they feel at ease, computers will remain the province of a technological priesthood and continue to be treated with suspicion by everyone else. If they can, the enormous potential of computers to aid and enhance our lives will be made available to all.

This book introduces the non-specialist to the technology of communicating with microcomputers. By placing particular emphasis on low-cost techniques associated with small systems and personal computers, the reader's attention is focussed on the positive nature of the microprocessor revolution — how machines can help people — rather than the negative aspects which have received much publicity in the non-technical press.

With computers, as with all complex and extensive subjects, it is difficult to know where to start. The book begins by considering communication within the computer system itself — the representation of information in electrical form and the data paths which carry it from one subsystem to another. The concepts of synchronization, of contention, of central and distributed control, and of addressing range emerge quite naturally from this approach.

We carefully steer away from the issue of programming, preferring not to treat it at all than to treat it cursorily and inadequately — there are many texts on the subject. Anyway, programming is not central to the study of man-computer communication channels, although it is certainly necessary for their implementation. The processor is treated as just another subsystem like the store or the interfaces to external devices: its role is played down quite deliberately in accordance with its new diminished status as a single, cheap, integrated circuit. The central component of a computer system from our viewpoint is the communication protocol, which appropriately enough is a logical construct rather than a piece of software or hardware.

Proceeding from communication within the computer system to communicating with the world outside, we find that the most important media for both input and output are light and electrical signals (which can be directly converted to sound waves or other mechanical effects). The principles of a variety of low-level interfaces are described: switches, keyboards, lights, various kinds of display, optical detection, analogue-to-digital and digital-to-analogue conversion, serial line interfaces. It is of great advantage to have already studied the means of communication within the computer system in the previous chapter, for the most mystifying part of commercial interface devices is the inward-looking rather than the outward-looking part.

The communication media of light and sound are treated extensively from a higher-level point of view, as graphics and speech. It is easy to justify concentration on graphics as a man-computer communication channel, for we are all familiar with its applications in amusement arcades and television games, and are beginning to see more serious uses in teletext and viewdata (the Post Office's Prestel). Computer speech is less common, more esoteric — despite the predominance of speech in everyday human communication. It is my belief, however, that speech systems are poised for take-off, and the next year or two will see a terrific explosion of them. We already have speaking toys, speaking language translators, reading machines for the blind — although they are rare. One frequently sees announcements in the technical press of devices such as typewriter attachments that let blind people check their work using speech, voice-operated domestic television controllers, and new continuous speech recognition systems on the market. Cheap speech synthesizers for computer hobbyists are manufactured in both Britain and the USA. Computer speech is losing its mystique — and this book will help.

The level of the book is suitable for the layman with some acquaintance with electronics. A tutorial glossary at the end of Chapter 1 serves to refresh the reader's memory of simple technical terms and electronic devices.

Ian H Witten
May 1980

# Acknowledgements

I would like to thank the many friends and colleagues who have helped me not just in writing this book but in developing an appreciation of the importance and range of man-computer studies. Brian Gaines and David Hill have been especially influential in promoting awareness of the various communication channels, and Chris Corbett, Pete Madams and Steve Matheson have educated me in the hardware of computer buses and interfaces. Dorinda Bath, Angie Corbett, and Paul Griffith have worked with me on aspects of speech communication, and many of my ideas on this subject derive from fascinating conversations with Walter Lawrence. Phil McCrea helped to design the structure of the book, and constructive suggestions on improving the text have come from Bruce Anderson, Rod Cuff, and John Foster. Kel Fidler, Graham Long, and Peter Noakes have helped me in debugging my ideas.

Margaret Swanson ably typed much of the book, and it was set on a PDP-11/45 computer and printed using a Diablo printer. Mike Sansom drew the pictures, and the Essex Electronics Centre kindly loaned equipment for most of the photographs in Chapter 1. I would like to thank City Computer Systems for Figure 4.1, Dr Ian Benest for Figure 4.3, Monotype International for Figure 4.5, and Quest Automation for Figure 4.28.

Permission to reproduce the following material is gratefully acknowledged: Figure 4.20 from Texas Instruments, "The Memory and Microprocessor Data Book", p 92; Appendix to Chapter 3 from Motorola, "M6800 Microcomputer System Design Data", pp 39-48; and Figures 5.10 and 5.11 from the Joint Speech Research Unit (Crown Copyright reserved).

# Contents

# Computers and Communication

Computers are moving out of the private and impersonal security of the air-conditioned computer room into the field.

> Day and night a small perspex box sits in a field in a remote part of Suffolk. Every fifteen minutes it wakes up, checks the air temperature and the light level falling on nearby plants, starts a cassette tape recorder and records the data, and goes back to sleep. Ecologists are using it to measure climatic variations to investigate the rate of photosynthesis by plants.

They are coming down-market, from the banks and oil companies to the toybox.

> Five-year-old Anna has a new toy. In a clearly-recognizable voice it asks her to spell a word and she types it on the keyboard. When she gets it wrong, it tells her and gives her one more try. Sometimes she doesn't understand the word, so she presses a "repeat" key to hear it again. After ten words, she is told her score and starts again with a different list of words. The toy can do other things too, like play word guessing games. Of course, it isn't perfect. Educationally, it may even be undesirable, for when the words come out on the lighted display the letters are clumsy and all in capitals. Sometimes she still can't understand a word after several repeats. But she's fascinated, and will play for hours more.

Figure 1.1 Microcomputer in a field

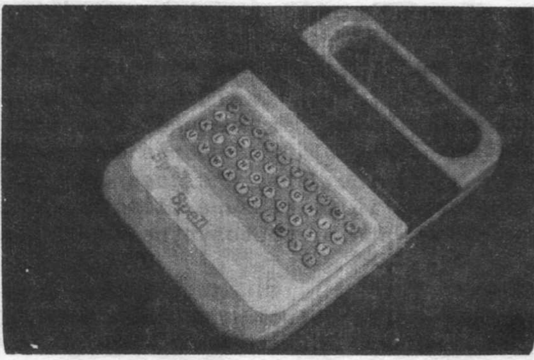

Figure 1.2 Speak 'n Spell toy

Now machines help people to communicate with other people —

Young Barry is autistic — he refuses to communicate with people. He too has just been given a new toy — a little grey box. He picks it up, chews at it, and tries to balance it on a joystick which sticks out the top. By chance he touches two shiny buttons on the side and it makes a noise like a whisper. He's intrigued. Eventually he discovers that if you move the joystick while pressing another pair of buttons it makes speech-like noises — inarticulate, drunken-sounding, but definitely human. He looks at his teacher sitting beside him and smiles — an exciting gift for her because he usually avoids eye-contact. Later on he finds a lever on the side of the box, which makes the voice high or low. But he can't operate it together with the joystick, for that needs three hands. It is rare for him to solicit assistance from someone else; but he gestures to the teacher to help. Now they have something to share.

— and machines to communicate with machines.

When trucks drive off the car ferry at Harwich, they used to stop to be weighed and measured, for road taxes depend in a complicated way on the size of the vehicle. Now they drive straight across the weighbridge, and two electronic eyes a couple of metres apart watch their profile as they go past and calculate from it their length. Easy? — no, for truck drivers are canny and accelerate and decelerate past the eye to try to fool it. But it isn't fooled, for the profiles seen by the two watchers differ slightly when speed changes occur and the device knows how to compensate for them.

What is happening? Everyone has heard of the "microprocessor revolution", the "computer on a chip". But the change is not just one of size, or even of cost. The kind of uses to which computers are put are changing too. To compute means to calculate, reckon, count. That is what computers were originally designed for — to solve mathematical problems by number-crunching at super-human speed. In the late 1940's the Government set up a committee to decide how many computers Britain needed. It came up with an answer — five! They hadn't reckoned on the field, the toybox, the hospital, shop, factory, office, the home.

Nowadays, computers are better thought of as information-processing machines rather than as number-crunchers. And as the whole point of information is that it should be communicated, the purpose of the processing is to communicate it in a different form. This way we see computers in their modern role of facilitating the communication of information. This is not to say that number-crunching has disappeared: mathematical problems still have to be solved, but they assume less importance now because of the far greater volume of communication-enhancing applications. The new role has dwarfed, not ousted, the old. In the four vignettes above, which incidentally are all taken from real life, only two of the machines — the speaking toy and the truck measurer — do a significant amount of calculation. And, curiously enough, for the speech toy the vast bulk of the calculations are for the purpose of decoding stored speech, not keeping score.

Let's not over-glorify this new role of machines. The result of more effective communication of information can be good or bad.

> Cindy's job is to pick substandard potatoes off a conveyer belt. She used to work in a big shed, standing by the belt with potatoes moving slowly past and picking up the mouldy and misshapen ones, throwing them into a bin. Now she has a desk in a nearby room with a TV set which shows the potatoes on the conveyer. Whenever she sees a bad one, she identifies it by pointing on the screen with a pen and the machine rejects it into the bin for her. Actually she prefers this change, for it was noisy and dirty in the shed compared with the peace of the office. But she can't talk with the other girls as she used to — for there aren't any. The new system makes her so efficient that they are redundant.

And the new potential for information-gathering can be intolerably intrusive, even sinister.

> At a checkout desk in a supermarket sits Dawn. They have just replaced her old till with a new, electronic one. She prefers it because it's quieter and easier to operate; it's better for the management because the accounts and a tally of goods that have been sold are kept on a central computer, which all the tills communicate with. What Dawn doesn't know — yet — is that each entry she makes is monitored, timed, and recorded: at the end of the day her manager will be able to compare her speed and the number of mistakes she made with all the other checkout girls.

As usual, technology is neutral in the struggle between good and evil. What it does is enhance the possibilities of both. Barry's life has been changed for the better, Dawn's for the worse. Cindy's is cleaner and quieter but her friends are out of work; Anna's is more interesting — at least for the next hour or so. This book is about the technology which makes it all possible. It aims to spread understanding so that people can appraise these new developments intelligently, see the problems for themselves, and participate in their solution.

Without recent advances in integrated circuit electronics like the microprocessor, none of these devices would exist, for they would all be impossibly expensive. However, the techniques which facilitate communication between man and machine, or between one machine and another, are all independent of the microprocessor. Modern electronics is the delivery van which brings them to the doorstep, not the goods themselves. People who design the devices obviously need to know about microprocessors and about programming. But the principles and technology behind communication are really much more important, for these, and not raw processing, are the limiting factor in most applications. The major creative step is the conception of the device rather than its design and implementation, and for this an understanding of the potential of the various communication methods is vital.

To begin, however, I must give you some idea of what microprocessors are and why they are having such a great impact upon society and upon industry.

## MICROPROCESSORS

A microprocessor is a device that follows a plan. The plan can be anything you like, provided it can be specified exactly as a sequence of steps. People often think of microprocessors as performing arithmetic, like a pocket calculator. So they can, of course, since arithmetic operations can be represented as plans — think of the procedure for addition, or long division. But there are plenty of other kinds of plan. Simply counting events can be expressed as a plan. Or triggering an alarm when certain conditions are met (the conditions must be stated precisely, like "temperature greater than 80 C, rather than "dangerously hot"). Or dialling telephone digits. Or associating a list of telephone numbers with names. Or grading potatoes, knitting, timing heart-beats.

A microprocessor can do any information-processing task that can be expressed, precisely, as a plan. It is totally uncommitted as to what its plan will be. It is a truly general-purpose information-processing device. The plan which it is to execute — which will, in other words, control its operation — is stored electronically. This is the principle of "stored program control". Without a program the microprocessor can do nothing. With one, it can do anything. (Anything, that is, that can be specified as a plan. Some things, like recognizing places or faces from pictures, cannot . . . yet — although Cruise missiles do recognize target terrain from a stored map.)

The way you have to formulate a plan for a microprocessor is quite different from how you would do it for a person. If my wife asks me to turn up the central heating, she is specifying the intended result of my action rather than the way I should go about it — the communication is goal-directed. A microprocessor's plan must, in contrast, be procedure-directed; taking the form of a program of instructions to be executed to accomplish the result. It is no good saying "go into the kitchen and turn the knob on the wall 3 degrees clockwise" — which route should be taken? — what if the door is closed? — which wall? — where on the wall? The program must be expressed in miniscule steps of detail. To get a microprocessor to do something, we must ourselves know how to do it, in detail. This is why it can't recognize faces — although we can, we don't know how.

Furthermore, microprocessors can only perform <u>information-processing</u> tasks. To take action on the outside world, or to receive signals from it, a connection must be provided between the microprocessor's representation of information (as digital electronic signals) and the real-world representation — like dots of light on a display screen, or a musical note, or the motion of knitting needles. Such a connection between information representations is called an "interface". We will have more to say about interfaces later.

## THE ECONOMICS OF INFORMATION PROCESSING

The important laws that govern microprocessors and related integrated circuit devices are laws of economics rather than laws of electronics. Let us summarize them.

1. Microprocessors are cheap because they are general-purpose and their development costs are shared between many users.
2. Microprocessor-based products are expensive to specify and develop because a microprocessor is a general-purpose, uncommitted device, and gives no clue to how it should be used.
3. While hardware always costs money, software — the program — is (almost) free to copy, once it has been developed.

Consider the first law — a microprocessor is cheap because it is general-purpose. Microprocessors cost only a few pounds. The raw material is insignificant — a chip of silicon, a plastic pack, some metal legs. You pay for organisation, for testing, for quality control, for distribution. Yet the development of a microprocessor — including design, chip development, generation of test procedures, setting up a production line — is enormously expensive. It is precisely because the final product is totally uncommitted as to how it is to be used that it is cheap — sales are high and the development cost is shared amongst millions of users. Microprocessors are the perfect example of mass-market economics.

For example, one popular microprocessor, the Motorola 6800, which was introduced in 1974, is currently selling around 200,000 a year and is priced at 5 - 10 pounds. The development cost was extremely high and difficult to estimate because of the novelty of large-scale integrated circuit technology in 1974. To make a device of similar complexity nowadays, provided that its function was completely specified in advance, would cost in the region of 200,000 pounds. It is only through sharing this development cost amongst a large number of purchasers that the individual chip price is so low. What makes sharing possible is the uncommitted nature of the device.

The net result is that information-processing hardware is virtually free.

The drawback comes with the second law. Microprocessor-based product design is costly because microprocessors are general-purpose. I have stressed that a microprocessor gives no indication of how it is to be used. It offers the peculiar feature of <u>deferred design</u>, in that design of a microprocessor-based application is deferred until after the processor itself has been manufactured. The design is, to a very large extent, the program or plan. Because the microprocessor is so general-purpose, the designer has a new problem. Instead of building up the product design from elementary components, he must consider how to strip down the limitless possibilities afforded by the microprocessor, inhibiting its general-purpose, uncommitted nature to achieve the product he requires. This is a new task to the engineer, one of restricting possibilities rather than generating them. It's more like sculpture than Meccano! Coupled with this are the inevitable problems of a new, rapidly-developing, and complex technology.

Of course, in the ideal case when the product characteristics are completely, unambiguously, and firmly specified from the outset, the job is not difficult. Usually, however, product specification and design tend to proceed hand in hand — at least to some extent. With weak project management this can lead to a microprocessor-based disaster. Although it may seem easy to avoid, in fact managers are often unaware of the virtually limitless information-processing possibilities of microprocessors, and as the project develops and their understanding increases, they cannot resist the temptation to enhance the final product.

The third law really accounts for the explosive growth in the use of microprocessors. While the first two alter the balance between hardware and development costs, the cost of a prototype product is broadly comparable whether a microprocessor is used or not — say to within an order of magnitude. But once the prototype exists, laws 1 and 3 combine to make subsequent copies very cheap indeed.

Software is (almost) free to copy. There is a loose analogy between the development of a microcomputer-based system and the production of a recording of orchestral or choral music, with a costly overhead of musicians, rehearsals, and so on. It is expensive to make, but (almost) free to copy. In fact copying software is cheaper than copying audio tapes because it can be done at electronic speeds while tape recorders are electromechanical devices which cannot work quickly. Of course, easy copying of software brings its own problem, namely, pirating.

The fact that software is free to copy leads to the maxim, "let the microprocessor do the work". Software implementation of a task may or may not incur a higher development cost than hardware, but it will certainly be cheaper in production.

## HARDWARE BUILDING BLOCKS

At the centre of a microcomputer system is the microprocessor chip itself. Usually the choice of processor is not critical to the final unit — there are several, similarly-priced, that could be used. What it does affect seriously, however, is the product development process, and there are strong reasons for conservatism in processor choice, because of the investment in experience and development tools.

A microprocessor cannot operate without a program, and