INTRODUCTION TO THE Z80 MICROCOMPUTER

ADI J. KHAMBATA

INTRODUCTION TO THE Z80 MICROCOMPUTER

ADI J. KHAMBATA
St. Paul Technical-Vocational Institute

JOHN WILEY & SONS
New York Chichester Brisbane Toronto Singapore

ISBN 0-471-86167-7
Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

To my brother Navroze (Pinky)

PREFACE

This book is intended as a supplement to the textbook *MICROPROCESSORS/MICROCOMPUTERS: Architecture, Software, and Systems* (written by this author and published by JOHN WILEY & SONS, INC., New York, January 1982). It may also be used as a supplement for other basic texts, or as a brief stand-alone introduction to the Z80.

During the planning and writing of *MICROPROCESSORS/MICROCOMPUTERS*, we encountered the following problem. In addition to offering theoretical descriptions and discussions, we wanted to describe at least one real, existing microcomputer system. However, including a description of any one particular product in that text would immediately have dated the publication and also made it unacceptable to other users who might be committed to another product. This dilemma was resolved by adopting a unique approach. A series of separate, softcover supplements, each covering a popular microcomputer system, will accompany the basic textbook. Four such supplements are presently planned. They will cover the ZILOG Z80, the INTEL 8080/8085, the MOTOROLA MC6800, and the 6502 by MOS TECHNOLOGY. This approach will enable us to introduce additional supplements on future products that find acceptance in educational institutions as well as update existing supplements as needed. It also allows users of the main textbook to purchase only the supplements that are of direct use for their specific needs. In addition to the paperback supplements, separate laboratory manuals—on the most popular microcomputers—will also be published.

This publication is the first product supplement. It is based primarily on the Z80 courses that I have designed and taught at St. Paul Technical–Vocational Institute, St. Paul, Minnesota, and in local industries over the past several years. As a result of this experience, I have introduced a number of features in this book. First, each chapter starts with a brief listing of chapter objectives, in order to give students a quick overview of the specific items that will be covered in each chapter. Upon completing a study of the chapter students will be able to review the objectives, thereby (hopefully) gaining further understanding of the material covered in that chapter.

Second, since this book may be used as a companion to my hardcover text, references to that book are included for the principal topics covered in each chapter. The references are made by both chapter and sections within the chapters. I feel that this simple convenience will save students a lot of time and effort when they wish to refer to the main textbook for a quick review of any particular topic.

Third, at the end of each chapter, numerous review questions are included. To aid both students and instructors, the questions are structured in formats such as true/false, multiple choice, and fill-in-the-blanks. For most questions, several possible answers are provided in parentheses at the end of the question. The student has only to select the appropriate answer. However, selection of the correct answer requires an understanding of the subject matter of the chapter. Furthermore, the earlier questions in any chapter are simple, but they become progressively more complex and more challenging.

Finally, two distinct styles of presentation are introduced in each chapter, namely, a list style and a descriptive prose style. The descriptive prose style is certainly easier to read and, in many cases, easier to understand since more explanation is provided. However, classroom experience has indicated to me that the prose style does very little to aid students in remembering specific technical points or events. For instance, if a certain operation involves six tasks to be performed in a specified sequence, it is much easier for a student to remember an enumeration of those tasks in six discrete steps than to remember them from a prose description.

The Z80 CPU chip and its principal support chips are described in this book. In addition, chapters are included for initializing, or programming, three of the most commonly used support chips (the PIO, the CTC and the DMA chips). The hardware architecture of the Z80/CPU chip is described in Chapter 1. This includes the principal registers and the various addressing modes as well as package pin assignments, presented in convenient tabular form. Six groups of Z80 instructions are presented and described in Chapter 2. They include LOAD instructions (8 and 16 bits), STACK PUSH/POP, EXCHANGE, BLOCK TRANSFER/SEARCH, 8-bit arithmetic/logic and 16-bit arithmetic instructions. This chapter also includes detailed explanations (aided by tables) of instruction formats and various symbols used in the instructions. Chapter 3 continues with the instruction set and covers ROTATE/SHIFT, BIT MANIPULATION, JUMP and CALL/RETURN instruction groups. Chapter 4 completes the instruction set. It includes the INPUT/OUTPUT instructions involved in the interrupt system of the Z80. Chapter 5 concludes the discussion of the CPU chip. Timing charts of the various functions (both with and without WAITstates) are included in this chapter.

Chapter 6 covers the Parallel Input Output (PIO) chip. It includes the architecture of the system, the interrupt control protocol and the interrupt timing involved in the three operational modes. Chapter 7 is a continuation of the previous chapter. It introduces the concept of customizing the PIO chip for specific applications by initializing it accordingly. Two specific examples of different applications are included. The Counter/Timer Circuit (CTC) chip is described in Chapter 8. Both the counter and timing functions are covered along with the various operating modes. Chapter 9 shows how the CTC can be preprogrammed for specific applications. Examples for both counter and timer applications are included.

The Direct Memory Access (DMA) chip is covered in Chapter 10. Operations involving transfer only, search only, and combined search-transfer are described. Also, the byte-at-a-time, burst, continuous and transparent modes of operation are covered in reasonable detail. Programming the DMA chip for various applications is covered in Chapter 11 with several examples. The Serial Input Output (SIO) chip is the subject of Chapter 12. Both the byte-oriented and the bit-oriented protocols are included. Both asynchronous and synchronous modes of operation are covered. The operation and use of the Cyclic Redundancy Check (CRC) character is also described.

I would like to thank the many people who helped me during the writing and publication of this book. At Wiley, Judy Green, Engineering Technology Editor, Susan Weiss, Associate Editor, and Alice Tufel, Senior Administrative Assistant, were extremely helpful during the entire writing and publication process. Their efforts and cooperation are most appreciated. Dr. George Richter, Technical Division Manager at St. Paul Technical-Vocational Institute, constantly encouraged and supported me. Several of my students (past and present) and former engineering colleagues at Sperry Univac reviewed parts of the manuscript and offered many valuable comments and suggestions; special contributions were made by Kenneth Jarosch, Philip Gaines, Walter Knights, and Jay

Meyers. I would also like to thank the following reviewers whose comments and suggestions were invaluable in preparing the final version of the manuscript: Lou Gross, Columbus Technical Institute; Irving L. Kosow, Southern Technical Institute; J.W. Carter, Memphis State University; David Hata, Portland Community College; Peter Julian, Columbus Technical Institute; Arthur H. Seidman, Pratt Institute; David Terrell, I.T.T. Technical Institute; Charles Van Buren, DeVry Technical Institute.

As before, this book is a Khambata family project. My daughter Pixie, assisted by my wife Ruth, typed the original manuscript; corrections and proofreading were handled by my son Jim and his wife Shelly; and all diagrams and timing charts were drawn by my son Danny and his wife Renee. I thank all of them.

Adi J. Khambata

St. Paul, Minnesota
February 7, 1982

ACKNOWLEDGMENTS

The author gratefully acknowledges and thanks ZILOG, INC., Cupertino, California, for granting permission to use and reproduce in this book block diagrams and timing charts, as well as the instruction set, of the Z80 Microcomputer from ZILOG published Technical Manuals. These manuals and the material used from them are identified in the following list.

Since this product supplement is primarily intended for student use in the classroom, minor modifications and additions are made in the block diagrams and timing charts to conform to explanatory material in the text and to aid the student in learning and understanding the subject matter.

From the Z80-CPU & Z80A-CPU TECHNICAL MANUAL © 1977

Page 4	Fig. 2.0-2	Z80 CPU Register Configuration
Page 3	Fig. 2.0-1	Z80 CPU Block Diagram
Page 39		Flag Register Format
Page 41	Table 6.0-1	Summary of Flag Operation (Note: Some of the notation symbols have been changed to accommodate the types available on my typewriter.)
Pages 43-54	Tables 7.0-1 thru 7.0-11	Summary of OP codes and execution times (Note: Flag symbols have been changed to accommodate available types on my typewriter. The columns for M cycles and number of T cycles are not included. Finally word descriptions of each word are added.)
Page 11	Fig. 4.0-0	Basic CPU Timing example.
Page 12	Fig. 4.0-1	Instruction OP code Fetch
Page 13	Fig. 4.0-1A	Instruction OP code Fetch with WAIT states
Page 13	Fig. 4.0-2	Memory Read or Write Cycles
Page 14	Fig. 4.0-2A	Memory Read or Write Cycles with WAIT states
Page 15	Fig. 4.0-3	Input or Output Cycles
Page 15	Fig. 4.0-3A	Input or Output Cycles with WAIT states
Page 16	Fig. 4.0-4	Bus Request/Acknowledge Cycle
Page 16	Fig. 4.0-5	Interrupt Request/Acknowledge Cycle
Page 18	Fig. 4.0-6	Non-Maskable Interrupt Request/Operation
Page 18	Fig. 4.0-7	Halt Exit

From the Z80-PIO & Z80A-PIO TECHNICAL MANUAL © 1977

Page 3	Fig. 2.0-1	PIO Block Diagram
Page 3	Fig. 2.0-2	Port I/O Block Diagram
Page 13	Fig. 5.0-1	Mode O (Output) Timing
Page 13	Fig. 5.0-2	Mode 1 (Input) Timing
Page 14	Fig. 5.0-3	Port A, Mode 2 (Bidirectional) Timing
Page 14	Fig. 5.0-4	Mode 3, Bit Control Mode Timing
Page 15	Fig. 6.0-1	Interrupt Acknowledge Timing
Page 17	Fig. 7.0-1	A method of extending the interrupt priority daisy chain

From the Z80-CTC & Z80A-CTC TECHNICAL MANUAL © 1977

Page 2	Fig. 2.0-1	CTC Block Diagram
Page 3	Fig. 2.0-2	Channel Block Diagram
Page 16		CTC Write Cycle
Page 17		CTC Read Cycle
Page 18		CTC Counting and Timing Diagram
Page 19		Interrupt Acknowledge Timing Diagram
Page 20		Return from Interrupt Timing Diagram

From the Z80-SIO TECHNICAL MANUAL © 1977

Page 1		Z80-SIO Block Diagram
Page 6	Fig. 4	Transmit and Receive Data Path
Page 13	Fig. 7	Synchronous Formats
Page 21	Fig. 8	Transmit/Receive SDLC/HDLC Message Format
Page 41	Fig. 14d	Return from Interrupt Cycle
Page 41	Fig. 14c	Interrupt Acknowledge Cycle
Page 41	Fig. 14b	Write Cycle
Page 41	Fig. 14a	Read Cycle

From the Z80-DMA TECHNICAL MANUAL (January 1981) © 1981

Page 2-3	Fig. 5	Basic Functions of Z80 DMA
Page 8-7	Fig. 62	Variable-Cycle and Edge Timing
Page 8-7	Fig. 63	WAIT line Sampling in Variable-Cycle Timing
Page 8-1	Fig. 48	CPU-to-DMA Write Cycle Requirements
Page 8-1	Fig. 49	CPU-to-DMA Read Cycle Requirements
Page 8-2	Fig. 50	Sequential Memory-to-I-O Transfer Standard Timing
Page 8-2	Fig. 51	Sequential I/O-to-Memory Transfer Standard Timing
Page 8-4	Fig. 54	Bus Request and Acceptance Timing
Page 8-4	Fig. 56	Bus Release on End-of-Block (Burst & Continuous Modes)
Page 8-4	Fig. 58	Bus Release on Not-Ready (Burst Mode)
Page 8-4	Fig. 57	Bus Release on Match (Burst and Continuous Modes)
Page 8-4	Fig. 55	Bus Release in Byte Mode

1 THE Z-80 CPU ARCHITECTURE

CHAPTER OBJECTIVES

The objectives of this chapter are as follows:

1. To introduce students to the Z-80 μC system and its principal hardware components.
2. To describe the hardware architecture of the Z-80/CPU.
3. To describe the general-purpose (GP) and the special-purpose (SP) registers and their respective functions.
4. To present the functions performed by the arithmetic-logic-unit (ALU).
5. To discuss the program status word (PSW) and present its format.
6. To describe and discuss the 12 addressing modes used in the Z-80/CPU.
7. To discuss the functions of the CPU package pins and present them in convenient tabular form.

TEXTBOOK REFERENCES

For the convenience of students who wish to review the relevant material of this chapter in the main (hardcover) textbook, the following sections and/or pertinent chapters are suggested:

1. For general review of CPU registers and counters - Sec. 1-3.2
2. For Index Register - Secs. 5-6 and 5-7
3. For Program Status Word - Sec. 3-5.3
4. For Addressing Modes - Chapter 5

5. For Interrupt Page Address Register - Secs. 9-5 and 9-7

1-1 INTRODUCTION

The Z-80 microcomputer system is designed and manufactured by ZILOG, INC. of Cupertino, California. At the present time, MOSTEK, INC. of Carrollton, Texas is the official second source for the Z-80 products.

The Z-80 devices are designed by personnel who were formerly employed by INTEL CORPORATION, and had designed the INTEL 8080A which is an enhanced version of the INTEL 8008 μC. Thus the Z-80 naturally tends to have many similarities with the 8080A and can be considered as an enhancement of the 8080A. INTEL's enhanced version of the 8080A is the 8085A.

1-2 THE Z-80 SYSTEM

1-2.1 Principal Hardware Components

The Z-80 μC system consists of 5 principal hardware components which are briefly described below:

1. The Z-80/CPU - This is the central processing unit which contains the arithmetic logic unit (ALU) and the other logical components required for performing the usual CPU functions.

2. The Z-80/PIO - This is a programmable, 2-port, parallel I/O interface designed for transfer of 8-bit, parallel, data between the Z-80/CPU and a variety of peripherals. The interfaces are TTL comparible and both ports are bidirectional.

3. The Z-80/CTC - This is a programmable Counter Timer Circuit with four independent channels, capable of providing counting and timing control functions for real-time events. The inputs and the outputs on this chip are also TTL compatible.

4. The Z-80/S10 - This is a programmable, dual-channel, serial I/O interface designed for communication with a variety of serial data peripherals. It is primarily a serial-to-parallel and parallel-to serial converter/controller using a number of different protocols.

5. The Z-80/DMA - This is a programmable Direct Memory Access Controller which can directly transfer data between the memory and the above-mentioned PIO and S10 interfaces in a cycle-stealing mode of operation.

Note - All of the above interfaces and controllers monitor the status of the peripherals. CPU polling is not involved. They also include priority interrupt systems which can be implemented by using the daisy chain logic technique. Additional external

logic is not required for this.

1-2.2 The Development System

ZILOG, INC. also provides a hardware/software development system for the Z-80 product line. Briefly, they include the following features:

1. The program development is RAM-based, not ROM-based.

2. The system is configured around a stand-alone, floppy disk.

3. The system provides real-time debugging capability.

4. The offline support software includes:

   Assembler
   Compiler
   Simulator, and
   Test pattern generation capability.

5. The resident software includes:

   RAM-based Assembler,
   RAM-based Text Editor,
   The BASIC Compiler,
   Real-time debugging program,
   Disk Operating System (DOS) with file maintenance, and
   ROM-based Executive firmware.

6. Diagnostic capability is also available. It includes:

   Rapid fault detection by means of system diagnostic software, and
   Continuous memory diagnostics performed in a background operation mode.

7. It is possible to replace RAM chips by ROM or PROM chips.

1-3 PRINCIPAL FEATURES OF THE Z-80 CPU

The main features of the CPU chip are presented below in summarized forms:

1. The Z-80 is a third generation μP which uses the N-channel, depletion-mode, silicon-gate technology.

2. A single, +5 volt ±5% power supply is used in the system.

3. A single-phase, TTL level, clock is used.

4. The Z-80 uses a 2.5 MHz clock. This gives an instruction execution time of 1.5 μs.

5. For a higher throughput rate, the Z-80 uses a 4.0 MHz clock which gives an instruction execution time of 1.0 μs.

6. The Z-80/CPU chip is available in a standard 40-pin, dual-in-line (DIP), package.

7. The Z-80 has an instruction set of 157 instructions which is software compatible with the 78 instructions of the INTEL 8080A μP.

8. The Z-80 instruction set includes instructions for:

   Bit manipulation,
   Byte manipulation,
   Character string operations, and
   Block transfers of data.

9. Two 16-bit Index Registers are available. They enhance the processing of look-up tables and arrays.

10. The Z-80 includes a duplicate set of general-purpose (GP) registers and status registers. The CPU has a total of 16 such registers.

11. The Z-80/CPU generates all the control signals for standard memory chips. Thus,

    Using only an external address decoder, static memory chips can be readily interfaced with the CPU,
    All controls for refreshing dynamic MOS memory chips are provided, and
    The control bus signals are capatible with the commonly-used, 18-pin and 22-pin 4K RAMs.

12. Expanded 16-bit arithmetic operations, as well as BCD operations, are available in the Z-80 CPU.

13. The Z-80 chips are designed to operate in temperature environments ranging from 0°C to +55°C.

1-4 THE CPU ARCHITECTURE

In the Z-80/CPU, there is a total of 22 registers/accumulators which are divided into two groups, namely, the general-purpose (GP) group and the special-purpose (SP) group which contains registers dedicated to performing only certain specific functions. Four of these registers are 16-bit units and the rest are 8-bit registers. The CPU has two accumulators. Additionally, the CPU has a 16-bit instruction register, with its associated instruction decoder, and the timing and control. All the above-mentioned registers are programmable, i.e., they can be accessed by the programmer.

1-4.1.1 GP Registers, Accumulators and Status Flag Registers

1. There is a total of 16, 8-bit, GP registers. Two of these are accumulators and two are status flag registers, assocated with each of the two accumulators.

2. These 16 registers are divided into two sets, the main set and the alternate set as shown in Figure 1-1.

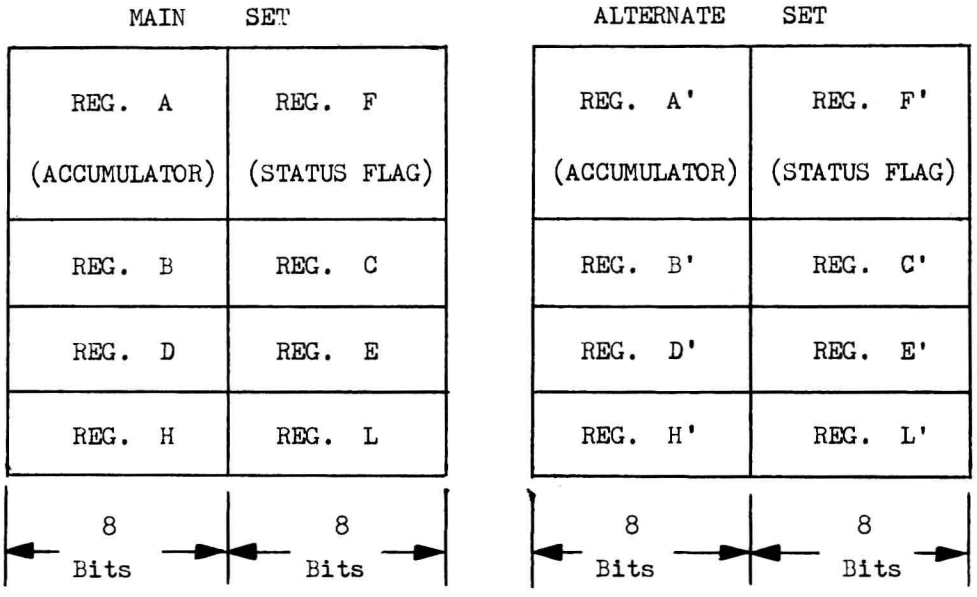

Fig. 1-1 General-purpose registers in the Z-80/CPU.

3. The Main Register Set units are labelled A (for the accumulator) and F (for its corresponding flag register). The rest are labelled B, C, D, E, H and L. Since registers A and F perform specific functions, it is more appropriate to say that the true GP registers are B, C, D, E, H and L registers only.

4. The second set is called the Alternate Register Set and is labelled in exactly the same manner but with a prime attached to each letter symbol. Once again B', C', D', H' and L' are the true GP register while A' and F' are the accumulator and the status flag registers respectively.

5. Accumulator A is the primary accumulator. Whenever a PUSH or a POP operation is performed involving the accumulator and the flag register, both registers are always transferred.

6. The remaining six registers, in both the main and the alternate set, can be used either individually as 8-bit units or they can be paired and used as 16-bit units as shown below:

Main Set	Alternate Set
B with C	B' with C'
D with E	D' with E'
H with L	H' with L'

7. The register pairs BC, DE,HL and B'C', D'E', H'L' can be used as secondary registers and/or data counters.

8. Under program control, either the main set or the alternate set can be selected and operated by means of a single instruction. Only one set can be used at any one time, not both.

1-4.1.2 Special-Purpose Registers

1. The Z-80/CPU has 6 special-purpose registers (SPR).

2. There are two 8-bit registers and four 16-bit registers as shown in Figure 1-2.

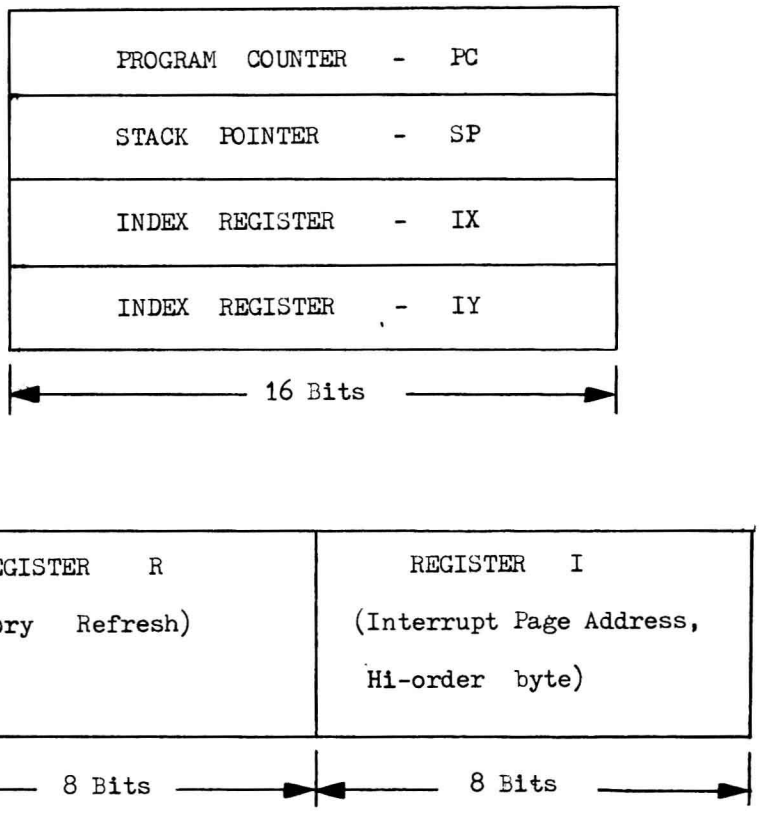

Fig. 1-2 Special-purpose registers in the Z-80/CPU.

3. A brief explanation of each SPR register follows:

   A. Program Counter (PC) - At any time, the PC holds the 16-bit address of the current instruction fetched from the program memory.

      After the contents of the PC (i.e., the address of the instruction) are placed on the address bus, the PC is automatically incremented.

      When a JUMP or a BRANCH is executed, the new address is automatically inserted in the PC.

   B. Stack Pointer (SP) - The Z-80 uses a pointer-type Stack to provide return addresses from BRANCH operations.

a previously-designated portion of the external RAM memory is used for the stack.

The stack is organized on a last-in, first-out (LIFO) basis.

By means of PUSH/POP instructions, data can be pushed on to the stack from any specified CPU register. Likewise, data can be popped from the stack into any specified CPU register.

The stack provides unlimited nesting capability.

Capability for multiple-level interrupts is also provided.

C. Index Registers (IX and IY) - These are two completely independent index registers.

Their primary function is to provide a 16-bit base address in the Indexed Addressing Mode.

They are extensively used in table look-up operations when locations in the memory are to be accessed.

For indexed addressing, a displacement byte is included in the instruction. The displacement is a signed twos complement integer and it is added to the base address in the index register to obtain the true or effective address.

D. Memory Refresh Register (R) - The Z-80 is one of the μPs which provides for the use of both static and dynamic MOS RAM chips in its system.

Dynamic memory chips are relatively inexpensive but can hold the information in them for only a very short time.

Thus, at millisecond intervals, the contents of each memory location are read out and rewritten in the same memory location.

The above process is called dynamic memory refreshing.

A counter is needed to track and update each memory address as each location is refreshed.

In the Z-80/CPU, the R register, which is a counter, performs the address incrementing function.

The memory refreshing operation is completely transparent to the programmer or the user.

Although the R register is not used for operations other than refreshing, it is programmable, i.e., the programmer can externally load the register for testing purposes.